Release History

Release History

Version 30-Apr-2021

General Information

Code

FIX	Bug fixed
ERR	Serious error
ODD	Odd behavior of software
IMP	Improvement
CHG	Change in behavior of software

Release Information

Software Release from 01-Feb-2021

Build 130863

Build 130863 IMP 02620 TriCore Debugger <trace>.List (List trace contents)	**data-cycle assignment for Aurix implemented** The traced data-cycles are automatically assigned to their refering program-cycles in the Trace.List window now.
Build 130863 CHG 02619 Cortex-A/-R Armv8 Debugger	**discontinue announcement** The usage of the TRACE32 executable named t32marm64 is now deprecated. t32marm64 was merged into t32marm, which now supports both 32-bit and 64-bit Arm CPUs. Please modify the executable name accordingly in your preferred start method(s) e.g. shortcut properties, shell scripts, ... To ease the migration of existing installations, we will provide a redirection executable or shell script in place of t32marm64 for a limited time.
Build 130863 IMP 02618	**support for the new Mixed-Signal Probe**
Build 130863 IMP 02617	**support for the new PowerTrace III trace module** This TRACE32 software release supports the PowerTrace III trace module for all pre-processors with AutoFocus technology.
Build 130863 CHG 02616	**discontinued support for target operating systems** The target operating systems Chorus, RTX Tiny, CMX Tiny, HI7000, ZEOS and Symbian are not supported anymore.
Build 130863 IMP 02615 Var.DRAW (Graphical variable display)	**Var.DRAW extended to display two-dimensional arrays** The command Var.DRAW has been extended to show contents of two-dimensional arrays.
Build 130863 IMP 02614 <trace>.EXPORT.TracePort (Export trace packets as recorded at trace port)	**new option for command CoreSightTrace.EXPORT.TracePort** New option /CoreSightByteStream for the command CoreSightTrace.EXPORT.TracePort to export single CoreSight stream.
Build 130863 IMP 02613 FLASH.SPI.CFI (Generate SPI FLASH sector declaration by CFI)	**new options /QuadPI and /OctalPI for command FLASH.SPI.CFI**
Build 130863 IMP 02612 NAME.User (Create new user channel)	**new command NAME.User for adding user-defined channels** User channels allow to combine (trace) channels via mathematical operations. The resulting new channel can then be referenced in any trace command by 'User.<name>'.

Build 130863 IMP 02611 Integrated Development Environment PRinTer.FILE (Re-route printer output to a file in specified file format) WinPrint (Print address or record range of a window)	**new PRinTer file format PCL** The PCL file format has been added to the PRinTer.FILE command.
Build 130863 IMP 02610 Integrated Development Environment ComPare (Compare files)	**the option of the command ComPare can now be combined** The options /Case, /IgnoreSpace and IgnoreCRLF of the command ComPare can now be combined. Previously only one option was allowed.
Build 130863 IMP 02609 JTAG.SWD.Init (Initialize the debug port) JTAG.SWD.ReadDapBus (Read register from DAP) JTAG.SWD.ReadScan (Read register from DAP) JTAG.SWD.SHIFT (Shift data by using the SWIO pin) JTAG.SWD.WriteDapBus (Write register to DAP) JTAG.SWD.WriteScan (Write register to DAP)	**new JTAG.SWD command group** new JTAG.SWD commands for manual Serial Wire Debug control.
Build 130863 IMP 02608 Trace-based Code Coverage COVerage.EXPORT.ListCalleEs (Export the function callees) COVerage.EXPORT.ListCalleRs (Export the function callers) COVerage.ListCalleEs (Display coverage for callees function) COVerage.ListCalleRs (Display coverage for callers function)	**new COVerage commands** New commands COVerage.ListCalleRs, COVerage.ListCalleRs, COVerage.EXPORT.ListCalleEs and COVerage.EXPORT.ListCalleRs. The new commands allow to list/export coverage information for callers and callees of functions.
Build 130863 IMP 02607 <trace>.Chart (Display trace contents graphically) <trace>.STATistic (Statistic analysis)	**new option /MACHINE for command Trace commands** New option /MACHINE for Trace.STATistic and Trace.Chart commands.
Build 130863 IMP 02606 TASK.List.TREE (Display tasks in a tree structure)	**new option /Machine for command TASK.List.TREE**

Build 130863 IMP 02605 Trace-based Code Coverage sYmbol.ECA.LOAD (Load a single ECA file) sYmbol.ECA.LOADALL (Load all ECA files)	**new options /SkipErrors and /LENient for sYmbol.ECA.LOAD** The /SkipErrors option ensures that warnings are issued instead of error messages. For scripts, error messages cause the script to stop. Warnings keep the script running. The /LENient option allows loading of ECA files with minor errors as invalid file version or checksum mismatch.
Build 130863 IMP 02604 sYmbol.DeleteMACRO (Delete macro information)	**new command sYmbol.DeleteMACRO added**
Build 130863 IMP 02603 COVerage.EXPORT.JSON (Export code coverage in JSON format)	**new command COVerage.EXPORT.JSON added** The command COVerage.EXPORT.JSON exports coverage information about functions and lines to a file in JSON format compatible to Gcov.

Build 130863 CHG 02602
Script Language PRACTICE
	obsolete PRACTICE function
	SYStem.CONFIG.PCH() replaced by SYStem.PCH()
	VERSION.LICENSE() replaced by LICENSE.SERIAL()

Build 130863 IMP 02601 Script Language PRACTICE	**new PRACTICE functions** CHIP.GTM.TIOModule(), COMPonent.PROPerty.ADDRess(), CONVert.LINEAR11TOFLOAT(), CONVert.LINEAR16TOFLOAT(), ETM.TraceCore(), LICENSE.RequiredForCPU(), PYthon.TERMinal.LINE(), SYStem.JtagClock.Whisker()
Build 130863 IMP 02600 Script Language PRACTICE	**new PRACTICE function synonyms** additional PRACTICE function synonyms introduced: SYStem.PCH() for SYStem.CONFIG.PCH()

Software Release from 01-Sep-2020

Build 125398

Build 125398 CHG 02599 Debug Modules PowerTrace	**discontinued hardware support** This is the last TRACE32 software release which supports TRACE32 Debug Interface LA-7701 TRACE32 PowerTrace 64MB LA-7706 TRACE32 PODBUS Ethernet Controller
Build 125398 CHG 02598 TRACE32 Front-End TRACE32 Software	**discontinued software and hardware support** This is the last TRACE32 software release which supports TRACE32 Frontends for Virtio (VDI) and VaST.
Build 125398 CHG 02597 JTAG Debugger Debugger for 80C196EA MMDSP Debugger Host Driver Software Integrated Development Environment x186 Monitor x196 Monitor XA51 Monitor TRACE32 Software	**restricted TRACE32 ICD software development** This is the last TRACE32 software release which contains for certain ICD/debugger software derived from the standard TRACE32 development trunk. A separate branch from the TRACE32 software will be created for supporting legacy ICD/debugger hardware only. Only bugfixes and host OS adaptations are planned for the future. effected CPU architectures: XA51, h85, i196, 186, MMDSP+
Build 125398 CHG 02596 In-Circuit Emulator FIRE Fully Integrated RISC Emulator Host Driver Software Integrated Development Environment TRACE32 Software	**restricted TRACE32 FIRE software development** This is the last TRACE32 software release which contains FIRE software derived from the standard TRACE32 development trunk. A separate branch from the TRACE32 software will be created for supporting FIRE hardware only. Only bugfixes and host OS adaptations are planned for the future.
Build 125398 CHG 02595 Host Driver Software	**discontinued host OS version** This is the last TRACE32 software release which supports Windows XP, Windows Vista and Linux 32Bit.
Build 125398 IMP 02594 Trace Analyzer <trace>.Find (Find specified entry in trace) <trace>.FindAll (Find all specified entries in trace) <trace>.List (List trace contents) <trace>.ListVar (List variable recorded to trace)	**new option /TASK for Trace commands** new option /TASK for Trace.List, Trace.Find, Trace.FindAll and Trace.ListVar to filter for tasks.
Build 125398 IMP 02593 PER.view (Display peripherals)	**new option /AlternatingBackGround for PER.view command** The new /AlternatingBackGround option allows to display an alternating background color in the PER.View window.

Build 125398 CHG 02592 JTAG Debugger FLASH.SPI.CMD (Send data to SPI FLASH device) FLASHFILE.SPI.CMD (Send data to SPI FLASH device)	**renamed commands FLASH.SPICMD and FLASHFILE.SPICMD** The commands FLASH.SPICMD and FLASHFILE.SPICMD have been renamed to FLASH.SPI.CMD and FLASHFILE.SPI.CMD.
Build 125398 CHG 02591 Trace Analyzer <trace>.EXPORT.cycles (Export trace data) LA.IMPORT.cycles (Import bus trace data)	**renamed commands Trace.EXPORT.flow and LA.IMPORT.flow** The commands Trace.EXPORT.flow and LA.IMPORT.flow have been renamed to Trace.EXPORT.cycles and LA.IMPORT.cycles.
Build 125398 IMP 02590 STOre (Store settings as PRACTICE script)	**new command option SYStemLOG for STOre <file>** New command option STOre <file> SYStemLOG to save the settings of the SYStem.LOG window.
Build 125398 IMP 02589 Trace Analyzer ETM.TraceDataPriority (Define data trace priority)	**new command ETM.TraceDataPriority** New command ETM.TraceDataPriority to define data trace priority on ETMv4.
Build 125398 IMP 02588 JTAG Debugger FLASH.SPI.GETSFDP (Read FLASH discovery parameters) FLASHFILE.SPI.GETSFDP (Read FLASH discovery parameters)	**new commands FLASH.SPI.GETSFDP and FLASHFILE.SPI.GETSFDP** New commands introduced to read Serial Flash Discovery Parameters from SPI FLASH. The several parameter values can be utilized afterwards using the FLASH.SPI.SFDP() and FLASHFILE.SPI.SFDP() functions.
Build 125398 IMP 02587 ARM Debugger CEVA-X Debugger Trace Analyzer ETM.TraceCORE (Core specific default tracing)	**new command ETM.TraceCORE** New command ETM.TraceCORE to allow core specific default tracing.
Build 125398 IMP 02586 Trace Analyzer COVerage.EXPORT.CSV (Export coverage data in CSV format) ISTATistic.EXPORT.CSV (Export instruction statistics in CSV format)	**new commands COVerage.EXPORT.CSV and ISTAT.EXPORT.CSV** New commands COVerage.EXPORT.CSV and ISTAT.EXPORT.CSV introduced to export coverage/instruction statistic data in comma separated values (CSV) format.

Build 125398 IMP 02585 JTAG Debugger FLASH.SPI.CFI (Generate SPI FLASH sector declaration by CFI) FLASHFILE.SPI.CFI (Generate SPI FLASH sector declaration by CFI)	**new commands FLASH.SPI.CFI and FLASHFILE.SPI.CFI** New commands introduced to generate the FLASH declaration by using CFI information.
Build 125398 IMP 02584 Trace Analyzer <trace>.STATistic.TASKLOCK (Analyze lock accesses from tasks)	**new command Trace.STATistic.TASKLOCK** New command introduced to analyze lock accesses from tasks in the trace.
Build 125398 IMP 02583 Script Language PRACTICE	**new PRACTICE function synonyms** additional PRACTICE function synonyms introduced: hardware.POWERTRACE2LITE() for Analyzer.CONFIG.POWERTRACE2LITE(), hardware.POWERTRACE3 for Analyzer.CONFIG.POWERTRACE3(), TASK.MACHINE.ID() for TASK.MACHINEID()
Build 125398 CHG 02581 EPROM/FLASH Simulator PowerIntegrator - Logic and Bus Analyzer PowerProbe - Logic and Protocol Analyzer TRACE32 Software Stimuli Generator	**discontinued software and hardware support** The t32pbi executables aren't supported anymore.
Build 125398 CHG 02580 JTAG Debugger PODBUS Ethernet Controller ICD In-Circuit Debugger TRACE32 Software	**discontinued software and hardware support** The TRACE32 controller based software (like t32win.exe, t32cde, t32qt, ... + mcpm*.t32) doesn't support PodbusEthernetController and ICD debug modul usage anymore. Please use in this case host based TRACE32 software executables (like t32marm.exe, t32marm, t32marm-qt, ...) instead.
Build 125398 CHG 02579 In-Circuit Emulator Host Driver Software ICE In-Circuit Emulator Integrated Development Environment TRACE32 Software	**restricted TRACE32 ICE software development** This is the last TRACE32 software release which contains ICE software derived from the standard TRACE32 development trunk. A separate branch from the TRACE32 software will be created for supporting ICE hardware only. Only bugfixes and host OS adaptations are planned for the future.

Software Release from 01-Feb-2020

Build 117056

Build 117056 ODD 02578 Logical Display of Peripherals Script Language PRACTICE	**PRACTICE and PER macro naming rules changed** Macro names must not start with a numerical digit anymore.
Build 117056 IMP 02577 Trace Analyzer <trace>.STATistic.TASKORINTRState (Task and ISR2 statistic analysis)	**new trace commands** New commands Trace.STATistic.TASKORINTRState and Trace.Chart.TASKORINTRState introduced to analyze tasks and (ORTI) interrupts.
Build 117056 IMP 02576 Integrated Development Environment List.auto (Display program listing)	**new mouse feature** New functionality on middle mouse button for List and Var windows: single click performs an add to watch window and double click a view in a new window.
Build 117056 IMP 02575 JTAG Debugger Data.VECTOR (Display memory as vectors)	**new command Data.VECTOR** The new Data.VECTOR command allows to display memory contents as vectors side by side.
Build 117056 IMP 02574 JTAG Debugger Data.ATTACH (Attach data sequence)	**new command group Data.ATTACH** The new Data.ATTACH command group allows to define a sequence of Data.Set commands that are executed when the debugger is activated with SYStem.Mode Attach.

Build 117056 IMP 02573
Script Language PRACTICE
	new PRACTICE function synonyms
	additional PRACTICE function synonyms introduced:
	OS.FILE.EXIST() for FILE.EXIST()
	long forms added: TRANSlation.* for TRANS.*()
	TRANSlation.ENABLE() for TRANS.ENABLE()
	TRANSlation.INTERMEDIATE() for TRANS.INTERMEDIATE()
	TRANSlation.INTERMEDIATE.VALID() for TRANS.INTERMEDIATE.VALID()
	TRANSlation.INTERMEDIATEEX() for TRANS.INTERMEDIATEEX()
	TRANSlation.INTERMEDIATEEX.VALID() for TRANS.INTERMEDIATEEX.VALID()
	TRANSlation.LINEAR() for TRANS.LINEAR()
	TRANSlation.LINEAR.VALID() for TRANS.LINEAR.VALID()
	TRANSlation.LINEAREX() for TRANS.LINEAREX()
	TRANSlation.LINEAREX.VALID() for TRANS.LINEAREX.VALID()
	TRANSlation.LIST.LOGRANGE() for TRANS.LIST.LOGRANGE()
	TRANSlation.LIST.LOGRANGE.ZONE() for TRANS.LIST.LOGRANGE.ZONE()
	TRANSlation.LIST.NUMBER() for TRANS.LIST.NUMBER()
	TRANSlation.LIST.NUMBER.ZONE() for TRANS.LIST.NUMBER.ZONE()
	TRANSlation.LIST.PHYSADDR() for TRANS.LIST.PHYSADDR()
	TRANSlation.LIST.PHYSADDR.ZONE() for TRANS.LIST.PHYSADDR.ZONE()
	TRANSlation.LIST.TYPE() for TRANS.LIST.TYPE()
	TRANSlation.LIST.TYPE.ZONE() for TRANS.LIST.TYPE.ZONE()
	TRANSlation.LOGICAL() for TRANS.LOGICAL()
	TRANSlation.LOGICAL.VALID() for TRANS.LOGICAL.VALID()
	TRANSlation.PHYSICAL() for TRANS.PHYSICAL()
	TRANSlation.PHYSICAL.VALID() for TRANS.PHYSICAL.VALID()
	TRANSlation.PHYSICALEX() for TRANS.PHYSICALEX()
	TRANSlation.PHYSICALEX.VALID() for TRANS.PHYSICALEX.VALID()
	TRANSlation.TABLEWALK() for TRANS.TABLEWALK()

Build 117056 IMP 02572 Script Language PRACTICE	**new PRACTICE functions** additional PRACTICE functions introduced: COVerage.LOAD.KEY(), DATE.UnixTimeUS(), IProbe.Probe(), JTAG.MIPI34(), OS.FILE.readable(), OS.FILE.EXIST(), OS.FILE.BASENAME(), OS.Window.LINE(), RTS.BUSY(), sYmbol.SECEXIST(), VPU()

Software Release from 01-Sep-2019

Build 112182

Build 112182 CHG 02571 JTAG Debugger Break.CONFIG.InexactAddress Break.CONFIG.MatchASID Break.CONFIG.MatchMachine Break.CONFIG.MatchZone Break.CONFIG.UseContextID Break.CONFIG.UseMachineID Break.CONFIG.VarConvert	**trigger onchip command group and dialog changed** The trigger onchip dialog and the command parser was changed for all architectures, because commands were renamed and moved to the global Break.CONFIG.state dialog and parser. The Break.CONFIG.state dialog was adapted to show the new settings made by the commands. New Break.CONFIG commands were introduced. The following list gives an overview of the deprecated TrOnchip commands which were replaced by new Break.CONFIG commands: TrOnchip.ContextID was renamed to Break.CONFIG.UseContextID, TrOnchip.CONVert was renamed to Break.CONFIG.InexactAddress, TrOnchip.CONVert0cycle was renamed to Break.CONFIG.InexactAddress0cycle, TrOnchip.MachineID was renamed to Break.CONFIG.UseMachineID, TrOnchip.MatchASID was renamed to Break.CONFIG.MatchASID, TrOnchip.MatchMachine was renamed to Break.CONFIG.MatchMachine, TrOnchip.MatchZone was renamed to Break.CONFIG.MatchZone, TrOnchip.VarCONVert was renamed to Break.CONFIG.VarConvert. The new commands introduced for selected architectures are Break.InexactData, Break.InexactTrigger and Break.InexactResume.
Build 112182 IMP 02570 JTAG Debugger Trace-based Code Coverage Trace Analyzer	**preset filter for coverage command group added** Preset filter introduced to improve tree views.
Build 112182 IMP 02569 JTAG Debugger JTAG.XUSEFUSE (Program Xilinx UltraScale eFUSEs)	**new command JTAG.XUSEFUSE added** new command introduced to burn eFuses of Xilinx UltraScale FPGAs
Build 112182 CHG 02568 TriCore Debugger	**Support for OCTL removed**
Build 112182 IMP 02567 Script Language PRACTICE MENU.Program (Interactive programming) MENU.ReProgram (Menu programming)	**new command PERMENU for menu files added** The new menu programming command PERMENU converts the tree structure of per files to men files.

Build 112182 IMP 02566 Script Language PRACTICE MENU.Program (Interactive programming) MENU.ReProgram (Menu programming)	**WAIT expression added to menu parser** Parsing the menu file is delayed until the WAIT condition is evaluated to TRUE. Useful for: e.g. configurable cores, menus depending on target access.
Build 112182 IMP 02565 Integrated Development Environment STOre (Store settings as PRACTICE script)	**new option AREA added** Option AREA added to command STOre to save the actual AREA settings to a PRACTICE script.
Build 112182 IMP 02564 Script Language PRACTICE OS.OPEN (Open file in default application)	**new command OS.OPEN added**
Build 112182 ODD 02563 Host Driver Software	**new TRACE32 screen driver model** Lauterbach implemented a new screen driver model to support better different GUI frameworks. The GUI specific software parts were split out into shared libraries (t32screen*). Detailed information can be found under https://www.lauterbach.com/3737.
Build 112182 IMP 02562 Script Language PRACTICE	**new PRACTICE functions** additional PRACTICE functions introduced: ADDRESS.ACCESS.CMP(), ADDRESS.ACCESS.CMPSTRICT(), CHIP.GTM.MCSModule(), CHIP.GTM.ATOMModule(), CHIP.GTM.TOMModule(), CHIP.GTM.TIMModule(), CIProbe(), CONVert.TIMERAWTOHEX(), CPU.ADDRESS(), ERROR.CMDLINE(), ERROR.MESSAGE(), ERROR.POSITION(), EVAL.TIme(), FLASH.SECTOR.OPTION(), hardware.POWERTRACESERIAL.ADAPTER.NAME(), hardware.POWERTRACESERIAL.ADAPTER.REV(), JTAG.XUSEFUSE.CNTL(), JTAG.XUSEFUSE.DNA(), JTAG.XUSEFUSE.KEY(), JTAG.XUSEFUSE.RESULT(), JTAG.XUSEFUSE.RSA(), JTAG.XUSEFUSE.SEC(), JTAG.XUSEFUSE.USER(), JTAG.XUSEFUSE.USER128(), MMU.DEFAULTPT2(), MMU.DEFAULTPT.ZONE(), MMU.DEFAULTTRANS.LOGRANGE.ZONE(), MMU.DEFAULTTRANS.PHYSADDR.ZONE(), MMU.FORMAT.DETECTED(), MMU.FORMAT.DETECTED.ZONE(), MMU.FORMAT.ZONE(), SYStem.INSTANCECOUNT(), SYStem.IRISconfig.RemoteServer(), SYStem.Option.DUALPORT(), SYStem.CONFIG.PCH(), SYStem.Option.FASTACCESS(), SYStem.Option.PARTitionconfig(), TIME.ZERO(), TRANS.LIST.LOGRANGE(), TRANS.LIST.LOGRANGE.ZONE(), TRANS.LIST.NUMBER(), TRANS.LIST.NUMBER.ZONE(), TRANS.LIST.PHYSADDR(), TRANS.LIST.PHYSADDR.ZONE(), TRANS.LIST.TYPE(), TRANS.LIST.TYPE.ZONE()

Build 105499

Build 105499 IMP 02560 XC2000/C166SV2 Debugger GTM Debugger (TriCore) PCP Debugger (TriCore) TriCore Debugger	**improved MCDS trace decoder speed**
Build 105499 IMP 02559 Hypervisor-aware Debugging OS-aware Debugging	**option /ACCESS added for TASK.CONFIG and EXTension.LOAD** The defined access class will be used always from the OS and Hypervisor awareness.
Build 105499 IMP 02558 Data.IMAGE (Display image data)	**Options to display RGB images with user defined format** new options /RGBCUSTOM and /RGBBITS added
Build 105499 IMP 02557 TriCore Debugger Data.STANDBY (Standby data-sequences)	**new command Data.STANDBY added**
Build 105499 IMP 02556 Integrated Development Environment DIALOG.Program (Interactive programming) EDIT (TRACE32 editor) MENU.Program (Interactive programming) PEDIT (Edit PRACTICE script) Analyzer.Program (Program trigger unit) Data.PROGRAM (Editor for writing assembler program) Integrator.Program (Program trigger unit) Probe.Program (Program trigger unit) PER.Program (Interactive programming)	**new internal text editor** New text editor version with syntax highlighting, nesting awareness, undo/redo and other features activated.

Build 105499 IMP 02555 ICD In-Circuit Debugger FLASH.Create (Declare FLASH) FLASH.CFI (Generate FLASH declaration by CFI) FLASH.CHANGEtype (Changes the FLASH type) STOre (Store settings as PRACTICE script)	**new option /DISableBulkErase added** For commands FLASH.Create and FLASH.CHANGEtype option /DISableBulkErase added. Now it is also possible to declare it explicitly for non-CFI flash devices to avoid unintended chip erase for truncated flash address range.
Build 105499 IMP 02554 Integrated Development Environment TargetSystem.NewInstance (Start new TRACE32 PowerView instance)	**TargetSystem.NewInstance supports new port types** TargetSystem.NewInstance supports GDB, TCF port as well as multiple API ports now. Added new options: /API<n>.PORT <port>, /GDB.PORT <port>, /GDB.PPROTocolTCP|UDP, /TCF.PORT <port>
Build 105499 IMP 02553 Trace Analyzer <trace>.Chart.RUNNABLE (Runnable activity chart)	**New commands to analyze AUTOSAR runnable trace information**
Build 105499 IMP 02551 Host Driver Software	**config.t32: new keyword DNSRETRIES=** There is a new option DNSRETRIES= for the NET related settings under PBI= and LINK= in config.t32. DNSRETRIES= sets the number of retries when trying to resolve a DNS name on startup. DNSRETRIES= has a backwards compatible default of 0. This can be helpful when the DNS entry for a TRACE32 debugger is dynamically created or changed by DHCP and needs time to propagate through the DNS hierarchy.
Build 105499 IMP 02550 Host Driver Software	**DHCP behaviour improved** The DHCP behaviour of the firmware and the controller code has been changed for more reliable and faster startup behaviour. Additionally the debugger now no longer releases the assigned address on "QUIT" and instead relies on the DHCP server to handle any timeouts. This leads to less changes in DNS and thus to a more stable user experience. As a side effect of these changes the "arp -s" method to set the IP address as described in the manual, now only works after ~4 minutes since powerup or plugging in the ethernet cable have passed.
Build 105499 CHG 02549 Script Language PRACTICE	**PRACTICE function deleted** XLEN()

Build 105499 IMP 02548 Script Language PRACTICE	**new PRACTICE functions** additional PRACTICE functions introduced: FLASH.ID(), INTERFACE.IRIS(), MMU.INTERMEDIATEEX(), MMU.INTERMEDIATEEX.VALID(), MMU.LINEAREX(), MMU.LINEAREX.VALID(), MMU.PHYSICALEX(), MMU.PHYSICALEX.VALID(), SYStem.CONFIG.JTAGTAP(), SYStem.Option.MACHINESPACES(), SYStem.MEMACCESS(), SYStem.Option.MMUSPACES(), SYStem.Option.ZoneSPACES() Trace.FILE(), Trace.LOAD(), TRACEPORT.LaneCount(), TRANS.INTERMEDIATEEX(), TRANS.INTERMEDIATEEX.VALID(), TRANS.LINEAREX(), TRANS.LINEAREX.VALID(), TRANS.PHYSICALEX(), TRANS.PHYSICALEX.VALID()

Build 100486

Build 100486 IMP 02546 Script Language PRACTICE LOG.DO (Log calls of PRACTICE scripts)	**Log file for PRACTICE scripts** The command LOG.DO allows logging all PRACTICE script calls. The command was mainly introduced for support purposes.
Build 100486 CHG 02545 Integrated Development Environment ChDir (Change directory)	**The prefix ChDir. can now be used for all commands** The ChDir. command prefix can now be used for all commands. Previously it was limited to IDE commands.
Build 100486 IMP 02544 Integrated Development Environment PATH.Delete (Delete search path) PATH.DOWN (Define search path at end of list) PATH.List (List search path) PATH.RESet (Reset search path) PATH.Set (Define search path) PATH.UP (Define search path at top of list)	**Improved usage of search paths for scripts, per files ...** Single command PATH is replaced by command group PATH.
Build 100486 IMP 02543 Integrated Development Environment SETUP.DropCoMmanD (Set command for files dropped into command line)	**Change default for file drop to command line** SETUP.DropCoMmanD allows changing the default command which is used when a file of certain type is dropped into the TRACE32 command line.
Build 100486 CHG 02542 Trace-based Profiling ISTATistic.ADD (Add trace contents to ISTAT database)	**PC snooping can be used by ISTAT commands** TRACE32 takes instruction execution information from Analyzer/Onchip trace and PC/timestamp information from the SNOOPer trace in order to improve timestamp accuracy.

Build 100486 IMP 02541 AUTOFOCUS Self Calibration CAnalyzer.SAMPLE (Set sample time offset) CAnalyzer.ShowFocus (Display data eye) CAnalyzer.ShowFocusClockEye (Show clock eye) CAnalyzer.TERMination (Configure parallel trace termination)	**Support for Whisker MIPI20T-HS CombiProbe/uTrace Arm** New commands to use new capability of Whisker MIPI20T-HS CombiProbe/uTrace Arm.
Build 100486 IMP 02540 Multicore Debugging TargetSystem.NewInstance (Start new TRACE32 PowerView instance)	**New command TargetSystem.NewInstance** The command TargetSystem.NewInstance creates the config file for the new TRACE32 instance and then starts the TRACE32 instance for the specified architecture (e.g. t32marm.exe). Each started TRACE32 instance gets a name, which simplifies the handling of multiple instances. The new instance is automatically linked to the same PowerDebug module or the same MCI Server as the instance that accepted the command.
Build 100486 IMP 02539 Multicore Debugging InterCom.ENable (User-defined InterCom name, auto-assigned port number) InterCom.NAME (Assign user-defined InterCom name)	**Assign name to TRACE32 instance** The command InterCom.Enable assigns a name and a port number to a TRACE32 instance. The command InterCom.Name assigns a name to a TRACE32 instance. The given name can then be used by the following commands to simply their use: InterCom.execute, InterCom.executeNoWait, InterCom.Evaluate, InterCom.PING, InterCom.WAIT and SYnch.Connect.
Build 100486 IMP 02538 Multicore Debugging InterCom.execute (Execute command via InterCom system) InterCom.Evaluate (Evaluate function via InterCom system) InterCom.executeNoWait (Execute command via InterCom system) InterCom.WAIT (Wait for remote InterCom system)	**Keywords and names simplify use of InterCom commands** All listed InterCom commands allow using the keywords SELF, ALL and OTHERS to address TRACE32 instance(s), as well as the name assigned to a TRACE32 instance and a name pattern.
Build 100486 IMP 02537 Multicore Debugging SYnch.Connect (Connect to other TRACE32 PowerView instances)	**Keywords and names simplify use of SYnch.Connect command** The SYnch.Connect command allow using the keywords SELF, ALL and OTHERS to address TRACE32 instance(s), as well as the name assigned to a TRACE32 instance and a name pattern.
Build 100486 02536 Snooper SNOOPer.CORE (Select cores for PC snooping)	**Core-specific PC snooping for SMP systems** Allows to specify core(s) for SNOOPer.Mode PC in SMP systems.

Build 100486 IMP 02535 Uniform Look-And-Feel for all Processors SHA1SUM (Calculate SHA1 checksum of a file)	**New command to calculate SHA-1 checksum for a file** The SHA1SUM command calculates a 160 bit checksum for the given files using the Secure Hash Algorithm.
Build 100486 IMP 02533 Script Language PRACTICE	**new PRACTICE functions** additional PRACTICE functions introduced: Analyzer.PCIE.CONFIG(), Analyzer.PCIE.ISCONFIGURED(), Analyzer.PCIE.Register(), COMPonent.NAME(), COMPonent.TYPE(), COVerage.SourceMetric(), ELA.VERSION(), InterCom.NAME(), InterCom.PODPORTNAME(), JTAG.SEQuence.RESULT(), MMU.DEFAULTTRANS.LOGRANGE(), MMU.DEFAULTTRANS.PHYSADDR(), PATH.NUMBER(), PATH.PATH(), STRing.SCANBack(), sYmbol.ISREGISTER(), sYmbol.ISSTACK(), sYmbol.ISSTATIC(), sYmbol.REGISTER(), sYmbol.SECNAME(), SYStem.ADAPTER.FW.OUTDATED(), SYStem.Option.TOPOlogy(), SYStem.Option.TOPOlogy.SOCKETS(), TERM.TRIGGERED(), Var.EXIST(), Var.ISREGISTER(), Var.ISSTACK(), Var.ISSTATIC(), Var.REGISTER(), XLEN()
Build 100486 IMP 02532 Uniform Look-And-Feel for all Processors TERM.view (Terminal display) TERM.METHOD (Select terminal protocol)	**Multiple open TERMinal windows are now possible** A channel number was introduced to allow the concurrent use of several TERMinal windows.
Build 100486 IMP 02531 ASM Debugger Data.USRACCESS (Prepare USR access)	**Set-up of Data.USRACCESS is reset on SYStem.Mode change** Set-up of Data.USRACCESS is reset on every change of the SYStem.Mode. If Data.USRACCESS is not configured, data accesses to access class USR cause an error.
Build 100486 IMP 02530 Logical Display of Peripherals AUTOINDENT (Indent content of peripheral file automatically)	**Automatic indentation for PER files** The AUTOIDENT command allows activating various indentation styles.

Build 100486 IMP 02529 Logical Display of Peripherals PER.view (Display peripherals) CSV (Enables CSV capabilities) PERCMD (Row definition in CSV-formatted *.per file)	**CSV format allowed for PER files** PER files can be written within a spreadsheet editor, since the PER.view command can now also compile CSV files.
Build 100486 CHG 02528 Integrated Development Environment DO (Start PRACTICE script)	**autostart.cmm replaces t32.cmm** autostart.cmm is located in the system directory and always executed when TRACE32 instance (e.g. t32marm.exe) is started. If no autostart.cmm exists, TRACE32 falls back to the outdated behavior: t32.cmm is executed from either the working or the system directory, but only if not start-up script was explicitly specified as command line option. For details refer to "Automatic Start-up Scripts" in practice_user.pdf
Build 100486 IMP 02527 Trace Analyzer IProbe.state (Display the IProbe configuration window)	**IProbe command group is supported by PowerTrace Serial** As soon as the IProbe is switched from the DISable state to OFF/Arm state, the trace memory is split up. 50% of the trace memory can still be used to record the trace data exported via the Aurora/PCIe trace port. The other 50% are assigned to the IProbe.
Build 100486 IMP 02526 Integrated Development Environment CmdPOS (Controls the position of TRACE32 in MWI window mode) FramePOS (Controls the position of TRACE32 in MDI window mode) WinPOS (Define window dimensions and window name)	**Improved support for multimonitor desktops under MS Windows** The maximum possible window creation size was enlarged to cover the full multimonitor desktop size under MS Windows.
Build 100486 02525 Uniform Look-And-Feel for all Processors Break.DeletePATtern (Delete breakpoints allowing wildcards) Break.PATtern (Set temporary breakpoints allowing wildcards) Break.SetPATtern (Set breakpoints allowing wildcards)	**Set/delete breakpoints on symbol pattern** Breakpoints can be set on symbol pattern.
Build 100486 IMP 02524 Debugging via XCP SYStem.CONFIG XCP (XCP specific settings)	**Support for ASAM XCP** Support for debugging over XCP as specified in ASAM MCD-1 (XCP) standard, version 1.5 by ASAM e.V for GTM, Qorivva, RH850 and Tricore.

Build 100486 IMP 02523 Hypervisor-aware Debugging ARM/Cortex Trace (parallel) Intel~ Processor Trace QorIQ PowerPC NEXUS Aurora Trace Port <trace>.List (List trace contents)	**Decoding of machine id supported** The machine ID within the Context-ID or ownership packet is now decoded.
Build 100486 CHG 02522 ARM/Cortex Trace (parallel) Intel~ Processor Trace MPC57xx NEXUS High Speed Serial Trace Port QorIQ PowerPC NEXUS Aurora Trace Port MPC55xx/MPC56xx NEXUS Debugger and Trace <trace>.List (List trace contents)	**Cycle type for Context-ID and Ownership packets changed** If a Context-ID or ownership packet is decoded and if it is assignable to a task, the "task" cycle type and the task name is displayed. The displayed data value is not longer the packet content, but a TRACE32 internal value. If a Context-ID or ownership packet is decoded and if it can not be assigned to a task or any other protocol-specific content, the cycle type "traceid" and the packet content is displayed.
Build 100486 IMP 02521 Uniform Look-And-Feel for all Processors Break.List (Display list of breakpoints)	**Break.List window was revised** Enabling/disabling breakpoints is now possible with a single click to the check mark in the new "icon" column. Change Breakpoint window is opened by pushing the Change button. Double click to a Program breakpoint opens now source list window (List command). Double click to a Read or Write breakpoint opens Data.View window.
Build 100486 CHG 02520 Uniform Look-And-Feel for all Processors	**Break.IMPLementation was renamed to Break.METHOD** Break.IMPLementation was renamed to Break.METHOD.

Build 93173

Build 93173 IMP 02519 FLASH Programming (Memory-Mapped) NAND FLASH Programming	**Allows TRACE32 to send SPI FLASH commands** Allows TRACE32 to send valid FLASH command sequences to a SPI FLASH device.
Build 93173 IMP 02518 Trace Analyzer GROUP.Create (Create a new group)	**GROUP command group supports access classes and machineIDs** GROUP command group now supports access classes (such as N: Z: H: for the ARM architecture) and MachineIDs.
Build 93173 IMP 02517 Trace-based Code Coverage COVerage.EXPORT (Export code coverage information to an XML file)	**Improved COVerage.EXPORT for C++ projects** Unique identifiers are provided as additional property for functions. They can be used to identify functions in case of overloading.
Build 93173 IMP 02516 Trace Analyzer BTrace.state (Display BTrace configuration window)	**Command group BTrace enables script controlled trace sink** New command group BTrace allows a script controlled trace sink. First use case: apply Trace.STATistic commands to function run-times measured with the benchmark counters of a RH850 debugger.
Build 93173 IMP 02515 Trace-based Code Coverage COVerage.Option.SourceMetric (Select code coverage metric)	**TRACE32 supports now MCDC coverage** TRACE32 supports now MCDC coverage without code instrumentation. Requires generation of *.eca (Extended Code Analysis) files with preprocessor t32cast.exe. t32cast.exe can be found in ~~/demo/t32cast/bin.
Build 93173 IMP 02514 Integrated Development Environment <trace>.DRAW.channel (Plot no-data values against time) <trace>.DRAW.Data (Plot data values against time) <trace>.DRAW.Var (Plot variable values against time)	**Mouse interface changed for Trace.Draw command group** Zooming into the selected area is now done via double-click with the left mouse button. Free selection is now possible in the draw area. Value range or time/record range selection is possible via the scale area. Drag and drop of the selected area to another trace window with graphical display aligns this window to the same area.

Build 93173 IMP 02513 Integrated Development Environment <trace>.Chart (Display trace contents graphically) <trace>.Chart.TASK (Task activity chart) <trace>.Chart.sYmbol (Symbol analysis) <trace>.Timing (Waveform of trace buffer)	**Mouse interface changed for Trace.Chart/Timing windows** Zooming into the selected area is now done via double-click with the left mouse button. Drag and drop of the cursor to another trace window aligns this window to the same cursor position. Drag and drop of the selected area to another trace window with graphical display aligns this window to the same area.
Build 93173 IMP 02512 Integrated Development Environment Frame.view (Display stack frame) List.auto (Display program listing) Var.View (Display variables) Var.Watch (Open Var.Watch window)	**Mouse interface changed for all text windows** Free text selection is now possible with the left mouse. Click and drag on a selection allows drag and drop.
Build 93173 ODD 02511 Script Language PRACTICE	**modified PRACTICE function behaviour** function hardware.ICD() returns FALSE() in case of PBI=MCILIB or PBI=MCISERVER now
Build 93173 IMP 02510 Script Language PRACTICE	**new PRACTICE functions** additional PRACTICE functions introduced: ACCESS.isGUEST(), ACCESS.isGUESTEX(), ACCESS.isHYPERVISOR(), ACCESS.isHYPERVISOREX(), ADDRESS.isGUEST(), ADDRESS.isGUESTEX(), ADDRESS.isHYPERVISOR(), ADDRESS.isHYPERVISOREX(), ADDRESS.MACHINEID(), BTrace.RECORDS(), BTrace.REF(), BTrace.STATE(), BTrace.SIZE(), Data.STRing.Byte(), FLASH.CFI.WIDTH(), FLASH.ProgramMODE(), FLASH.ProgramMODE.OPTION(), FLASHFILE.GETBADBLOCK.COUNT(), FLASHFILE.GETBADBLOCK.NEXT(), JTAG.SEQuence.EXIST(), JTAG.SEQuence.LOCKED(), NEXUS.RTTBUILD(), RTM(), RTMBASE(), RTPBASE(), TASK.MACHINE.ACCESS(), TASK.MACHINE.NAME(),

Build 93173 IMP 02509 Script Language PRACTICE	**new PRACTICE function synonyms** PRACTICE functions synonyms introduced: OS.PCF() ==> OS.PresentConfigurationFile(), OS.PDD() ==> OS.PresentDemoDirectory(), OS.PED() ==> OS.PresentExecutableDirectory(), OS.PEF() ==> OS.PresentExecutableFile(), OS.PHD() ==> OS.PresentHomeDirectory(), OS.PHELPD() ==> OS.PresentHELPDirectory(), OS.PLF() ==> OS.PresentLicenseFile(), OS.PPD() ==> OS.PresentPracticeDirectory(), OS.PPF() ==> OS.PresentPracticeFile(), OS.PSD() ==> OS.PresentSystemDirectory(), OS.PTD() ==> OS.PresentTemporaryDirectory(), OS.PWD() ==> OS.PresentWorkingDirectory(), PID() ==> ProcessID()
Build 93173 IMP 02508 Integrated Development Environment TAR (Pack files into an archive)	**Pack files into an archive** Packs the specified files without compression into a tape archive.
Build 93173 IMP 02507 RTOS Debugger for Windows Data.LOAD.CrashDump (Load MS Windows Crash Dump file) TASK.CrashDump (Windows crash dump analysis)	**Support for loading Windows crash dump** After loading the full Crash Dump file into the simulator or the target memory with the command Data.LOAD.CrashDump, the Windows awareness command TASK.CrashDump.LOADNT is used to auto-load the kernel debug symbols. After that the command TASK.CrashDump.LOADREG could be used to load the context of all the available cores.
Build 93173 IMP 02506 FLASH Programming (Memory-Mapped) FLASH.HOOKSCRIPT (PRACTICE script based FLASH programming prolog)	**PRACTICE script based FLASH programming prolog** The <hook_script> can perform checks, set-ups etc. to guarantee that the FLASH programming works properly afterwards, e.g. to avoid fatal problems that might occur when the FLASH programming erases or modifies FLASH sectors that contain information that is necessary to operate the debug interface or the chip. First use case: TriCore TC3xx.
Build 93173 ODD 02505 Script Language PRACTICE DO (Start PRACTICE script)	**PRACTICE error handling** PRACTICE scripts containing syntax errors are automatically popped (ENDDO) from PRACTICE stack during loadtime (DO <scriptname>) now.

Build 88288

Build 88288 CHG 02561 Host Driver Software	**discontinued host OS version** This is the last TRACE32 software release which supports Solaris 8, RedHat RHEL5, Suse SLES10/SLED10 and CentOS5.
Build 88288 IMP 02504 ARM/Cortex Trace (parallel) ETM.DataTrace (Configure data-trace) ETM.TimeStampsTrace (Specify data trace correlation method (ETMv4))	**ETMv4 data trace correlation extended to use timestamps** ETM.TimeStampTrace ON advises TRACE32 to assigns the data address/data value information of the data trace stream to the appropriate load/store instruction by using the timestamps. This requires that a timestamp unit is available.
Build 88288 CHG 02503 ASM Debugger Data.CLEARVM (Clear the TRACE32 virtual memory (VM:))	**New behavior for TRACE32 Virtual Memory** The TRACE32 Virtual Memory (VM:) is no longer initialized by 64KB blocks upon a write access. Instead, the data are just written to the specified addess/address range. The command Data.CLEARVM clears now exactly the address/address range specified.
Build 88288 CHG 02502 ASM Debugger Data.Out (Write port) Data.Set (Modify memory)	**Data.Set/Data.Out commands allow different access/data width** Data.Set and Data.Out commands allow specifying an access width additionally to the data width.
Build 88288 IMP 02501 MMU Support SYStem.LOG.List (Log the accesses made by TRACE32) SYStem.LOG.Set (Select the TRACE32 accesses to be logged)	**SYStem.LOG.List now shows address translation failures** SYStem.LOG.List window now shows address translation failures as TRANS errors. If SYStem.LOG.Set TRANSlation is ON all address translation calls are shown in the SYStem.LOG.List window (logical addresses only).
Build 88288 CHG 02500	**Demo time for missing multicore licensing reduced** A 10 minute demo time is given if multicore debugging is not licensed (hardware-based tools).
Build 88288 CHG 02499 TRACE32 Maintenance Check	**Demo time for invalid/missing maintenance reduced** A 10 minutes demo time is given when no valid maintenance exists.

Build 88288 IMP 02498 ARM Debugger Cortex-A/-R Armv8 Debugger Intel~ x86/x64 JTAG Debugger SYStem.Option MACHINESPACES (Address extension for guest OSes) SYStem.Option MACHINESPACES (Address extension for guest OSes) SYStem.Option MACHINESPACES (Address extension for guest OSes)	**Hypervisor-aware debugging feature introduced (Part 1-2)** The TRACE32 machinespaces framework was added for architectures ARM and Intel x86 to support Hypervisor environments. This framework allows the debugger to perform stop-mode debugging of targets running a hypervisor and one or multiple guest machines. In conjunction with a hypervisor awareness it is possible to debug all guest machines concurrently, whether they are currently active or currently inactive. TRACE32 allows to load an individual OS awareness for each virtual machine on the target. The machinespaces framework is enabled with the command SYStem.Option.MACHINESPACES ON. A new identifier <machineid> is introduced which allows to identify virtual machines on the target. TRACE32 addresses are augmented with the machineid to specify which machine the address belongs to. The TRACE32 debugger address translation (enabled with TRANSlation.ON) is extended to fully support the two-stage address translation which is usually applied for hardware virtualized guest machines. The first translation stage translates guest virtual addresses to intermediate physical addresses. The second translation stage translates intermediate physical addresses to absolute physical addresses. The MMU registers needed for the page table walk steps are automatically fetched from the hardware registers or, for inactive guests, from hypervisor data structures. If SYStem.Option.MACHINESPACES is enabled, - addresses are extended with a machine ID identifier, separating the machine ID with three colons from the (extended) virtual address: <accessclass>:<machineid>:::<spaceid>::<virtaddress> The machine ID clearly specifies which host or guest machine the address belongs to. The host machine always uses machineid 0. Guests have a machineid larger than 0. TRACE32 currently supports up to 30 machines.

Build 88288 IMP 02497 Data.LOAD.Elf (Load ELF file) EXTension.LOAD (Load extension) Frame.view (Display stack frame) MMU.FORMAT (Define MMU table structure) Register.view (Display registers) TASK.List.MACHINES (List machines) TASK.List.TREE (Display tasks in a tree structure) TRANSlation.List (List MMU translation table) TRANSlation.TableWalk (Automatic MMU page table walk)	**Hypervisor-aware debugging feature introduced (Part 2-2)** - the debugger address translation (MMU and TRANS command groups) can be individually configured for each virtual machine by specifying a machineid in the address parameters. Where appropriate, MMU commands accept the new option /MACHINE <machineid>. Page tables of all machines can be viewed. Access class I: is used to specify intermediate addresses. - individual symbol sets can be loaded for each virtual machine by loading them onto the appropriate extended address: Data.LOAD.<type> <file> <accessclass>:<machineid>:::0 <options> Data.LOAD.<type> <file> <accessclass>:<machineid>:::<spaceID>::0 <options> - multiple awarenesses can be loaded to individual machines using the EXTension.LOAD command with the /machine option: EXTension.LOAD <awarenessfile> /MACHINE <machineid> [/NAME <extname>] - extension commands may use an infix to specify uniquely a command of a specific extension: TASK.[<extname>.]<extcommand> EXTension.[<extname>.]<extcommand> - new commands TASK.List.MACHINES and TASK.List.TREE are available to display the target's hypervisor/guest/process/thread structure - task names may be extended by the machine name, separated by three colons, to uniquely identify them: <machinename>:::<taskname> - register sets and stack frames of machines and tasks of arbitrary guests can be viewed, e.g.: Register.view /MACHINE <machineid> Frame.view /TASK "<machinename>:::<taskname>"
Build 88288 02496 Trace Analyzer <trace>.Chart.DatasYmbol (Analyze pointer contents graphically) <trace>.Chart.GROUP (Group activity chart) <trace>.Chart.TASK (Task activity chart) <trace>.Chart.sYmbol (Symbol analysis) <trace>.Timing (Waveform of trace buffer)	**Chart and timing windows with Track option keep zoom** The zoom factor of the <trace>.Chart and <trace>.Timing windows is retained, even if the trace content changes, if the /Track option is used.
Build 88288 IMP 02494 TriCore AGBT with PowerTrace Serial TriCore AGBT with Serial Preprocessor RTS.state (Open status and control window)	**RTS implemented for TriCore** RTS implemented for TriCore architecture.

Build 88288 IMP 02493 OS-aware Debugging TASK.select (Display context of specified task) TASK.CONFIG (Configure OS Awareness)	**TASK.select is now default command for TASK command group** Now TASK.select is default command for TASK command group. It was TASK.CONFIG before.
Build 88288 IMP 02491 High-Level-Language Debugging Var.AddWatch (Add variable to Var.Watch window) Var.Watch (Open Var.Watch window)	**Var.Watch keeps formatting if opened without variable** Var.Watch window applies formatting options to all added variables, if opened with format parameters but without variable names.
Build 88288 IMP 02490 High-Level-Language Debugging Var.AddWatch (Add variable to Var.Watch window) Var.View (Display variables) Var.Watch (Open Var.Watch window)	**Extend arithmetic in Hll expressions to more the 64 bits** Integer arithmetic in HLL expressions is extended to support more than 64-bits.
Build 88288 CHG 02489 VE (Virtual execution mode)	**Command group VM.* renamed** Command group VM.* renamed to VE.* (VirtualExecution).
Build 88288 IMP 02488 Script Language PRACTICE SPRINTF (Write formatted data to a PRACTICE macro)	**Use sprintf() style formatting for PRACTICE macros** The command SPRINTF allows to assign text and formatted data to a PRACTICE macro in the style of the sprintf() function of C/C++.
Build 88288 IMP 02487 Script Language PRACTICE ECHO (Write text and data to an AREA window (with format decoration)) PRINTF (Write formatted data to an AREA window)	**Streamlined output to AREA window from PRACTICE** The ECHO command keeps type and radix information of data and PRACTICE macros. This improves the readablity of the output. The PRINTF command allows to format outputs in the style of the printf() function in C/C++.
Build 88288 CHG 02486 Host Driver Software	**Minimum requirements for Linux/Motif executables changed** After the DVD 2017/09 the minimum requirements for the Linux/Motif executables will be raised to: glibc >= 2.12, kernel >= 2.6.32 This corresponds to the same requirements as the Linux/Qt builds.

Build 88288 ODD
02485
Script Language PRACTICE
	renamed PRACTICE functions
	ADDRESS.WIDTH() replaced by ADDRESS.MAU()
	Data.MAU() replaced by ADDRESS.MAU()

Build 88288 IMP 02484 Script Language PRACTICE	**new PRACTICE functions** additional PRACTICE functions introduced: ADDRESS.EXPANDACCESSP(), ADDRESS.MAU(), Break.Program.EXIST(), Break.ReadWrite.EXIST(), CPU.DEVICEID(), Data.Long.Byte(), Data.Long.Long(), Data.Long.Word(), Data.Quad.Byte(), Data.Quad.Long(), Data.Quad.Word(), Data.Word.Byte(), FPU.RAW(), PER.ADDRESS.isNONSECURE(), PER.ADDRESS.isNONSECUREEX(), PER.ADDRESS.isSECURE(), PER.ADDRESS.isSECUREEX(), SYStem.Option.SWDCONTROLSTATUS(), TASK.MACHINEID(),

Software Release from 01-Feb-2017

Build 81148

Build 81148 CHG 02466	**discontinued host OS version** This is the last TRACE32 software release which supports Windows 2000.

Build 80996 IMP 02483 MPC57xx NEXUS High Speed Serial Trace Port QorIQ PowerPC NEXUS Aurora Trace Port MPC55xx/MPC56xx NEXUS Debugger and Trace RTS.state (Open status and control window)	**RTS implemented for PowerPC (all E200zx cores) and QorIQ** RTS implemented for PowerPC (all E200zx cores) and QorIQ.
Build 80996 IMP 02482 Integrated Development Environment UNARchive (Linux and Microsoft libraries)	**Extract Linux/Microsoft libraries to directory** A new command UNARchive is available to extract files from Linux libraries (.a) and Microsoft libraries (.lib) to a directory. The two additional subcommands UNARchive.Table and UNARchive.Show help to determine the contents of the archive and to check the result of the unarchive operation.
Build 80996 IMP 02481 Logical Display of Peripherals PER.STOre (Generate PRACTICE script from PER settings)	**Generate PRACTICE script for PER settings** New command PER.STOre and extended local pulldown menu in PER.View window to generated script/command sequence for current PER settings.
Build 80996 CHG 02480 Script Language PRACTICE SCREEN.WAIT (Update screen while waiting)	**Extended functionality of SCREEN.WAIT command** SCREEN.WAIT now waits and updates the screen while waiting. It behaves similar to WAIT command. The old behavior of waiting until all processing windows are completed remained unchanged.
Build 80996 CHG 02479 RTS.state (Open status and control window)	**Features of RTS command group were reduced** No full trace, no nesting processing, no database save. Command group is under redesign.
Build 80996 IMP 02478 Trace Analyzer <trace>.Find (Find specified entry in trace) <trace>.FindAll (Find all specified entries in trace)	**Search features for Trace.Find/Trace.FindAll extended** CHANGE <search_item>: allows searching only for changes in search item. Improved data value search (multiple values, multiple ranges and not matching data).
Build 80996 IMP 02477 ARM/Cortex Trace (parallel) RTS (Real-time profiling (RTS))	**RTS command group supports ETMv4** RTS command group supports ETMv4.
Build 80996 CHG 02476 Trigger Probe for PODBUS TrPOD.state (State display)	**Command group TrPod renamed to TrPOD** Command group TrPod renamed to TrPOD.

Build 80996 IMP 02475 Android Debugging ARM Debugger Cortex-A/-R Armv8 Debugger Data.LOAD.OAT (Load OAT file)	**Android Debugging** The TRACE32 Android debug support has been extended for Android versions based on the Android RunTime (ART). This includes Android versions L, M and N. The new Android support allows the debugging of the ahead-of-time compiled Android framework and apps. TRACE32 automatically detects ahead-of-time compiled objects and and loads the DWARF/ELF info if available. Otherwise the debugger can parse the OAT data to extract the debug info. Additionally, TRACE32 supports the hybrid compilation introduced in Android N. For interpreted code, it is possible to display the stack frame with native to Java and Java to native transitions. A double click on a Java method displays the high level code together with the Dalvik disassembly. In case the code is just-in-time compiled, TRACE32 uses the symbols of the Android libart.so library to parse the JIT cache in order to get the names and ranges of the so-called hot methods. The new command Data.LOAD.OAT allows to load files generated by Android RunTime (ART).
Build 80996 IMP 02474 AREA.Create (Create or modify message area) AREA.Delete (Delete message area) AREA.List (Display a detailed list off all message areas)	**Improvements and changes for AREA command group** The new command AREA.List displays a detailed list about all created AREA windows. The new command AREA.Delete <area_name> allows to remove an specific AREA window created by the AREA.Create command. The following characters are no longer allowed in AREA names: * \ / ' " ; , &. " are automatically removed whenever possible.
Build 80996 CHG 02473 Trace-based Code Coverage	**COVerage.ACCESS command is deprecated** COVerage.ACCESS is deprecated. Please use Trace.ACCESS instead.
Build 80996 IMP 02472 Trace-based Code Coverage COVerage.ListFunc (Display coverage for functions) COVerage.ListLine (Display coverage for HLL lines) COVerage.ListModule (Display coverage for modules) COVerage.Option.SourceMetric (Select code coverage metric)	**Statement and decision coverage now supported** TRACE32 Code Coverage now supports Statement and Decision coverage.
Build 80996 IMP 02471 ARM Debugger	**New secure/nonsecure/hypervisor virtual address classes** New access classes ZVM:, NVM:, HVM: allow zone specific MMU translations of xVM: accesses to AVM:
Build 80996 ODD 02470 Trace-based Profiling <trace>.STATistic.Func (Nesting function runtime analysis) <trace>.STATistic.TREE (Tree display of nesting function run-time analysis)	**Average time is now calculated for complete function run** Average time values are now calculated only for complete function runs. Previously the average time was calculated by total time divided by count.

Build 80996 IMP 02469 Trace-based Profiling <trace>.STATistic.FuncDURation (Statistic analysis of single function) <trace>.STATistic.FuncDURationInternal (Statistic analysis of single func.)	**New commands to analyse run-time of a single function** Detailed analysis of a single function. Trace.STATistic.FuncDURation shows time between function entry and exit without time is interrupts and other tasks. Trace.STATistic.FuncDURationInternal show net time of function.

Build 80996 CHG
02468
Script Language PRACTICE
	renamed PRACTICE functions
	ADDRESS.SPACE() replaced by ADDRESS.ACCESS()
	HEADID() replaced by ID.PREPRO()

Build 80996 IMP 02467 Script Language PRACTICE	**new PRACTICE functions** additional PRACTICE functions introduced: ADDRESS.RANGE.SIZE(), Analyzer.CONFIG.POWERTRACESERIAL(), Analyzer.TraceCONNECT(), Analyzer.TRIGGER.TIME(), AREA.COUNT(), AREA.EXIST(), AREA.LINE(), AREA.MAXCOUNT(), AREA.NAME(), AREA.SELECTed(), AREA.SIZE.COLUMNS(), AREA.SIZE.LINES(), BMC.COUNTER.BYNAME(), BMC.COUNTER.BYNAME.CORE(), BMC.COUNTER.CORE(), BMC.OVERFLOW.BYNAME(), BMC.OVERFLOW.BYNAME.CORE(), BMC.OVERFLOW.CORE(), CABLE.GalvanicISOlation(), CABLE.GalvanicISOlation.SERIAL(), CAnalyzer.FEATURE(), CAnalyzer.TraceCONNECT(), FLASH.SECTOR.EXTRAvalue(), FLASH.SECTOR.RANGE(), hardware.POWERTRACEPX(), hardware.POWERTRACESERIAL(), Onchip.TraceCONNECT(), RTS.FIFOFULL(), RTS.NOCODE(), RTS.RECORD(), SYStem.CONFIG.XCP.Connected(),SYStem.CONFIG.XCP.ConnectMode(), SYStem.DCI.BssbClock(), SYStem.DCI.Bridge(), SYStem.DCI.TimeOut(), SYStem.Up.isLOCKED(), TASK.CONFIGFILE(), Trace.TraceCONNECT(),

Build 76594

Build 76594 ODD
02462
Script Language PRACTICE
	renamed PRACTICE functions
	ADDRESS.DATA() replaced by ADDRESS.isDATA()
	ADDRESS.INTERMEDIATE() replaced by ADDRESS.isINTERMEDIATE()
	ADDRESS.ONCHIP() replaced by ADDRESS.isONCHIP()
	ADDRESS.PHYSICAL() replaced by ADDRESS.isPHYSICAL()
	ADDRESS.PROGRAM() replaced by ADDRESS.isPROGRAM()
	CPU.FAMILY() replaced by CPU.BASEFAMILY()

Build 76594 ODD 02461 Script Language PRACTICE	**changed PRACTICE function** hardware.POWERTRACE2(): function return type changed from integer to boolean
Build 76594 IMP 02460 Trace Analyzer <trace>.STATistic.Func (Nesting function runtime analysis) <trace>.STATistic.TREE (Tree display of nesting function run-time analysis)	**Option /TASK !<task_name> to exclude task from statistic** The option /TASK !<task_name> allows to exclude the specified task from the nesting statistic, if it causes nesting problems.
Build 76594 CHG 02459 Script Language PRACTICE RePeaT (Loop with check at end of loop)	**Syntax of PRACTICE command RePeaT changed** The complex syntax of the RePeaT command: RePeaT <count> <block> WHILE <condition> is no longer supported.
Build 76594 CHG 02458 Intel~ x86/x64 JTAG Debugger	**Renaming of segment register aliases** To avoid confusion with access classes ES:, ESR:, GS: and GSR: all six segment selector address prefixes have been renamed from CS:/DS:/ESR:/FS:/GS:/SS: to CSS:/DSS:/ESS:/FSS:/GSS:/SSS:.
Build 76594 IMP 02457 System Trace LA.IMPORT.STPByteStream (Import STP recording from file (byte))	**Import STP byte stream for processing in TRACE32** The command allows to import a STP byte stream from a file into TRACE32. The imported STP byte stream can then be displayed and analyzed.
Build 76594 IMP 02456 Trace Analyzer <trace>.ListVar (List variable recorded to trace)	**List variables in trace** List all variable recorded to trace.
Build 76594 IMP 02455 Debug Support for Unified EFI Bootloader	**UEFI Debugging for release builds** Debugging of UEFI drivers in release builds (no debug symbols, optimized code) is now supported.

Build 76594 IMP 02454 Integrated Development Environment PRinTer.EXPORT (Export formatted printer output to file) PRinTer.FILE (Re-route printer output to a file in specified file format)	**Behavior of PRinTer dialog changed** The file extension matches the output format now. As part of this change, the commands PRinTer.EXPORT and PRinTer.FILE have been updated. PRinTer.EXPORT additionally allows to export the output in ASCII and XHTML format. Now PRinTer.FILE allows to set the file name, the output format and the append parameter.
Build 76594 ODD 02453 High-Level-Language Debugging Go.Return (Complete HLL function)	**Behavior of Go.Return command was changed** First Go.Return stops now at function epilog, here local variables are still valid. Second Go.Return stops at the return of the function.
Build 76594 IMP 02452 Trace Analyzer	**Real-time processing for HTM** Real-time processing for HTM is now possible.
Build 76594 IMP 02451 ARM Debugger	**Enables ASID specific onchip breakpoint** Enable ASID specific onchip breakpoint, when target is using the context ID register in this way.
Build 76594 IMP 02450 Intel~ x86/x64 JTAG Debugger	**Keep protection level bits in PC segment** In protected mode (large memory model), TRACE32 now preserves the protection level bits from the segment descriptors (bits RPL = bits [1:0] of segment descriptor). This prevents that the protection level is lost if the PC is modified in debug mode, which may lead in consequence to a privilege violation in the application.

Build 76594 IMP 02449 Script Language PRACTICE	**new PRACTICE functions** additional PRACTICE functions introduced: ADDRESS.EXPANDACCESS(), ADDRESS.isNONSECURE(), ADDRESS.isNONSECUREEX(), ADDRESS.isSECURE(), ADDRESS.isSECUREEX(), Analyzer.FIRST(), ART.FIRST(), ATrace.RECORDS(), ATrace.REF(), ATrace.SIZE(), ATrace.STATE(), BMC.CLOCK(), BMC.OVERFLOW(), Break.Alpha.EXIST(), Break.Beta.EXIST(), Break.Charly.EXIST(), CAnalyzer.FIRST(), CONVert.ADDRESSTODUALPORT(), CONVert.ADDRESSTONONSECURE(), CONVert.ADDRESSTOSECURE(), CORE.ISACTIVE(), CORE.NAMES(), CPU.PINCOUNT(), Data.WSTRING.BigEndian(), Data.WSTRING.LittleEndian(), FORMAT.Time(), Integrator.FIRST(), INTERFACE.NAME(), IProbe.FIRST(), JTAG.X7EFUSE.CNTL(), JTAG.X7EFUSE.DNA(), JTAG.X7EFUSE.KEY(), JTAG.X7EFUSE.RESULT(), JTAG.X7EFUSE.USER(), LOGGER.FIRST(), MMU.INTERMEDIATE.VALID(), MMU.LINEAR.VALID(), MMU.LOGICAL.VALID(), MMU.PHYSICAL.VALID(), Onchip.FIRST(), OS.FILE.ABSPATH(), OS.FILE.REALPATH(), OS.PORTAVAILABLE.UDP(), OS.PORTAVAILABLE.TCP(), PER.Byte(), PER.HByte(), PER.Long(), PER.Long.BigEndian(), PER.Long.LittleEndian(), PER.LongLong(), PER.LongLong.BigEndian(), PER.LongLong.LittleEndian(), PER.PByte(), PER.Quad(), PER.Quad.BigEndian(), PER.Quad.LittleEndian(), PER.SByte(), PER.SLong(), PER.Short(), PER.Short.BigEndian(), PER.Short.LittleEndian(), PER.TByte(), PER.Word(), PER.Word.BigEndian(), PER.Word.LittleEndian(), PRINTER.FILENAME(), Probe.FIRST(), SNOOPer.FIRST(), sYmbol.AutoLOAD.CHECK(), sYmbol.AutoLOAD.CHECKCMD(), sYmbol.AutoLOAD.CONFIG(), sYmbol.EPILOG(), SYStem.CADIconfig.RemoteServer(), SYStem.CADIconfig.Traceconfig(), SYStem.CONFIG.XCP.INFO(), SYStem.CONFIG.XCP.INFO.STR(), SYStem.MCDconfig.LIBrary(), Trace.FIRST(), Trace.METHOD.ATrace(), TRANS.INTERMEDIATE.VALID(), TRANS.LINEAR.VALID(), TRANS.LOGICAL.VALID(), TRANS.PHYSICAL.VALID(), TrOnchip.IsAvailable(), TrOnchip.IsSet()

Build 69655

Build 69655 IMP 02448 Trace Analyzer Data.Find (Search in memory) Data.FindCODE (Execute command on specified code type) <trace>.Find (Find specified entry in trace)	**New option /ALL for Trace.Find command** With the option /ALL the TRACE32 Message Line displays how often the searched item was found. The number of occurrences can be processed in a script via the function FOUND.COUNT().
Build 69655 ODD 02447 Script Language PRACTICE	**renamed PRACTICE functions** SYStem.Option.MemoryMODEL() replaced by SYStem.Option.MEMoryMODEL()
Build 69655 IMP 02446 Trace-based Profiling <trace>.Chart.sYmbol (Symbol analysis)	**FIFOFULL is now shown in Trace.Chart.Symbol window**
Build 69655 CHG 02445 TRACE32 Instruction Set Simulators	**New TRACE32 Simulator License** The TRACE32 Instruction Set Simulator allows to perform 50 PRACTICE script commands/API operations after the first "single-step" or "Go". If you want to perform further PRACTICE script commands/API operations a TRACE32 Simulator License is needed.
Build 69655 CHG 02444 ASM Debugger Data.LOAD.Binary (Load binary file)	**Data.LOAD.Binary requires now start address or range** The command Data.LOAD.Binary now requires a start address or a range. Before a load to address zero was performed when no start address or range was provided.
Build 69655 CHG 02443 High-Level-Language Debugging	**Data.LOAD.auto generates error for unknown file format** Data.LOAD.auto throws error if file format is unknown. Before the file was loaded as binary.

Build 69655 IMP 02442 High-Level-Language Debugging Trace Features <trace>.DRAW.channel (Plot no-data values against time) <trace>.DRAW.Data (Plot data values against time) <trace>.DRAW.Var (Plot variable values against time) Data.DRAW (Graphical memory display of arrays) Data.PROfile (Graphical display of data value) Var.PROfile (Graphical display of variable)	**Multiple variables of different type can be drawn** TRACE32 allows to draw multiple variables/channels with different formats e.g. integers and floats for all listed commands.
Build 69655 IMP 02441 FLASH Programming (Memory-Mapped) FLASH.CFI (Generate FLASH declaration by CFI) FLASH.TARGET2 (Define second target controlled algorithm)	**Simultaneous programming of multiple NOR flash devices** The new command FLASH.TARGET2 allows the simultaneous programming of multiple flash devices with two different flash algorithms. This is needed for example to program processor internal flash and processor external NOR flash/HyperFlash/QSPI flash together. The new option /TARGET2 for the FLASH.CFI command allows to link a second flash algorithm to a CFI conform flash device. The new function FLASH.TARGET2.FILE() is provided to get the filename of the second flash algorithm.
Build 69655 IMP 02440 TRACE32 Integration to Eclipse TRACE32 Integration to Wind River Workbench	**TRACE32 as TCF Agent** If TCF= is added to the TRACE32 configuration file, TRACE32 is started as a TCF agent. This allows to use the WindRiver Workbench or the Eclipse debugger as IDE and TRACE32 as debugging back end.
Build 69655 IMP 02439 Script Language PRACTICE	**new PRACTICE functions** additional PRACTICE functions introduced: CERBERUS.IOINFO(), CERBERUS.IOINFO.IFLCK(), CPU.FAMILY(), CPU.SUBFAMILY(), CONVert.TIMENSTOINT(), Count.Frequency(), Count.Time(), COVerage.TreeWalk(), CTIBASE(), ERROR.ID(), ERROR.OCCURED(), FLASH.TARGET2.FILE(), FOUND.COUNT(), ID.PREPROcessor() as synonym for ID.PREPRO() added, math.FABS(), math.FMAX(), math.FMIN(), math.MAX(), math.MIN(), math.TimeMAX(), math.TimeMIN(), MCDS.Module.NAME(), MCDS.Module.NUMBER(), MCDS.Module.REVision(), MCDS.Module.TYPE(), STATE.OSLK(), SYStem.M() as synonym for SYStem.MODE() added, SYStem.BAUDRATE(), SYStem.CORECLOCK(), SYStem.OSCCLOCK(), TrOnchip.Set.C6Exit(), TrOnchip.Set.ENCLU(), WINdow.COMMAND()

Build 69655 IMP 02438 Installation Guide SETUP.PDFViewer (Context-sensitive help via your favorite PDF viewer)	**New command SETUP.PDFViewer** The new command SETUP.PDFViewer allows to configure a PDF viewer for the TRACE32 manuals. Adobe Acrobat Reader stays the default viewer.

Build 65657

Build 65657 IMP 02463	**new supported host OS version** This is the first TRACE32 software release which supports Windows10.
Build 65657 02437 MMU Support TRANSlation.COMMON (Common address ranges for kernel and tasks)	**No logical operators in TRANSlation.COMMON command** Old way of specifiying multiple COMMON address ranges with OR operation has been removed (TRANS.COMMON 0x80000000--0x8FFFFFFF||0xA0000000--0xBFFFFFFF). Please use spaces or the command TRANSlation.COMMON.ADD to specify additional address ranges.
Build 65657 02436 Trace Analyzer <trace>.Find (Find specified entry in trace) <trace>.FindAll (Find all specified entries in trace)	**Improved match for search in trace** Search for addresses finds now also cycles that are close and touch the given address.
Build 65657 02435 ARM/Cortex Trace (parallel) MPC55xx/MPC56xx NEXUS Debugger and Trace <trace>.List (List trace contents) <trace>.STATistic.PAddress (Which instructions accessed data address) <trace>.STATistic.PsYmbol (Shows which functions accessed data address)	**Prestore address information for ARM-ETM/NEXUS** The columns PAddress/PsYmbol in the trace listing show the address of the instruction that was executed before a read/write access was performed. The statistic function Trace.STATistic.PsYmbol analyses which functions performed read/write accesses. The statistic function Trace.STATistic.PAddress analyses which assembler instruction performed the read/write accesses.
Build 65657 02434 High-Level-Language Debugging sYmbol.List.SOURCE (Display source file names) sYmbol.SourcePATH.List (List source search paths) sYmbol.SourcePATH.SetBaseDir (Define directory as base for relative paths) sYmbol.SourcePATH.Translate (Replace part of the source path)	**Improved source path correction** Module/source that could not be loaded, because the source path needs to be adapted in the current debug environment are printed in red in the sYmbol.List.SOURCE window. A Resolve Path command was added to the Source context menu, which fixes the paths in the current debug environment after one correct path was provided.
Build 65657 02433 FLASH Programming (Memory-Mapped) FLASH.state (FLASH programming dialog)	**FLASH.state command** New dialog to create, check and modify the setup for NOR and onchip flash programming.

Build 65657 02432 ARM Debugger Cortex-A/-R Armv8 Debugger SMMU (Hardware system MMU (SMMU))	**New command group for ARM System MMU Analysis** Allows viewing the configuration and status of the ARM System MMU and associated page tables.
Build 65657 02431 Script Language PRACTICE	**Default precision for parsed floating points increased** Default precision for parsed floating point values increased to 18 digits.
Build 65657 IMP 02430 Script Language PRACTICE ENTRY (Parameter passing) PARAMETERS (Parameter fetching) RETURNVALUES (Take return values)	**New commands PARAMETERS and RETURNVALUES** The PARAMETER command first creates PRIVATE macros for all macro names not found in the current scope and then assigns the passed values to the macros. RETURNVALUES takes the return values of a PRACTICE script/subroutine. An error message is generated, if a macro name is used, that cannot be found in the current scope. Both commands fix known issues of the ENTRY command.
Build 65657 ODD 02429 Script Language PRACTICE WRITE (Write to data file)	**WRITE %CONTinue behavior changed** The behavior of "WRITE %CONTinue" was changed to harmonize the manner of commands PRINT and WRITE. old: %CONT surpresses the writing of CR/LF and effects the next writing new: %CONT removes the CR/LF from the last writing and concatenates the new data
Build 65657 ODD 02428 Script Language PRACTICE	**changed PRACTICE functions** I2C.PIN(): returnvalue type changed from BOOLEAN to BINARY

Build 65657 ODD 02427
Script Language PRACTICE
	renamed PRACTICE functions
	CAnalyzer.PINS() replaced by CAnalyzer.PIN()
	CLOCK.DATE() replaced by DATE.DATE()
	CLOCK.DAY() replaced by DATE.DAY()
	CLOCK.MONTH() replaced by DATE.MONTH()
	CLOCK.SECONDS() replaced by DATE.SECONDS()
	CLOCK.TIME() replaced by DATE.TIME()
	CLOCK.UNIX() replaced by DATE.UnixTime()
	CLOCK.YEAR() replaced by DATE.YEAR()
	FCOS() replaced by math.FCOS()
	FEXP() replaced by math.FEXP()
	FEXP10() replaced by math.FEXP10()
	FINF() replaced by math.FINF()
	FLOG() replaced by math.FLOG()
	FLOG10() replaced by math.FLOG10()
	FNAN() replaced by math.FNAN()
	FSIN() replaced by math.FSIN()
	FSQRT() replaced by math.FSQRT()
	LINE.CALLER() replaced by PRACTICE.LINE.CALLER()
	WIN.EXIST() replaced by WINdow.EXIST()

Build 65657 IMP 02426 Script Language PRACTICE	**new PRACTICE functions** additional PRACTICE functions introduced: ADDRESS.RANGE.BEGIN(), ADDRESS.RANGE.END(), CONVert.HEXTOINT(), CONVert.INTTOFLOAT(), CONVert.INTTOMASK(), DATE.MakeUnixTime(), DATE.TimeZone(), DATE.utcOffset(), FORMAT.DecimalUZ(), FORMAT.UnixTime(), math,ABS(), math.FPOW(), math.SIGN(), math.SIGNUM(), OS.FILE.UnixTime(), OS.NAME(), PRACTICE.CoMmanD.AVAILable(), PRACTICE.FUNCtion.AVAILable(), RANDOM.RANGE(), SMMU.BASEADDRESS(), STRing.COUNT(), Trace.STREAM.OVERFLOWS(), WINdow.POSition()

Software Release from 02-Feb-2015

Build 60219

Build 60219 02425 TriCore Debugger	**Fixed erroneous read out of MCDS onchip trace.**
Build 60219 IMP 02424 High-Level-Language Debugging Var.set (Modify variable)	**C/C++ parser supports scaled assignments** C/C++ expression parser extended to support scaled assignment (e.g. for ASAP file format).
Build 60219 IMP 02423 Debugger Features Data.EPILOG.state (Display data epilogs) Data.PROLOG.state (Display data prologs) Data.SOFTEPILOG.state (Display data softepilogs) Data.SOFTPROLOG.state (Display data softprologs) Data.STARTUP.state (Startup data state display) Data.TIMER.state (Timer state display) FLASH.EPILOG.state (Display FLASH epilogs) FLASH.PROLOG.state (Display FLASH prologs)	**Full SMP support for auto sequences** Auto-sequences can be supplied for each core in an SMP system.
Build 60219 IMP 02422 Cortex-A/-R Armv8 Debugger SYStem.Option ZoneSPACES (Enable symbol management for Arm zones)	**Onchip Breakpoints ARMv8 are now mode aware** Onchip breakpoints are now secure/nonsecure/hypervisor aware when SYStem.Option ZoneSPACES is enabled.
Build 60219 CHG 02421 Intel~ x86/x64 JTAG Debugger High-Level-Language Debugging Data.LOAD.Elf (Load ELF file)	**Logical memory access for Data.LOAD.Elf** Data.LOAD.Elf use logical memory access when load is not done to a physical address (x86/x64)
Build 60219 IMP 02420 Integrated Development Environment	**XML format supported for WinPrint** The results of TRACE32 commands can now be saved to a file in XML format.
Build 60219 IMP 02419 Intel~ x86/x64 JTAG Debugger RTOS Debugger for Windows	**WinSTD Awareness for x86/x64** Task-aware debugging for Win7/Win8

Build 60219 02418 Trace-based Profiling <trace>.STATistic.INTERRUPT (Interrupt statistic) <trace>.STATistic.TASKORINTERRUPT (Statistic of interrupts and tasks) <trace>.STATistic.TASKVSINTERRUPT (Statistic of interrupts, task-related)	**Improved analysis of interrupt timing** Trace.STATistic.INTERRUPT - analyses timing of interrupts Trace.STATistic.TASKORINTERRUPT - analyses timing of tasks and interrupts Trace.STATistic.TASKVSINTERRUPT - analyses timing of tasks and interrupts, interrupts are analyzed in task context
Build 60219 IMP 02417 Script Language PRACTICE PMACRO.EXPLICIT (Enforce explicit PRACTICE macro declaration) PMACRO.IMPLICIT (Implicit PRACTICE macro declaration) PSTEPOUT (Back to caller)	**New PRACTICE commands** PMACRO.EXPLICIT enforces that macros are declared as PRIVAT, LOCAL or GLOBAL PMACRO.IMPLICIT allows to use an undeclared macro, undeclared macros are LOCAL be default PSTEPOUT allows to step out of the current PRACTICE block
Build 60219 CHG 02416 Script Language PRACTICE	**changed PRACTICE function** CMIBASE(): function parameter <index> introduced

Build 60219 CHG 02415
Script Language PRACTICE
	renamed PRACTICE functions
	COMBIPROBE() replaced by hardware.COMBIPROBE()
	DAS() replaced by interface.DAS()
	ESI() replaced by hardware.ESI()
	FIRE() replaced by hardware.FIRE()
	GDI() replaced by interface.GDI()
	HOSTMCI() replaced by interface.HOSTMCI()
	ICD() replaced by hardware.ICD()
	ICE() replaced by hardware.ICE()
	POWERDEBUG() replaced by hardware.POWERDEBUG()
	POWERINTEGRATOR() replaced by hardware.POWERINTEGRATOR()
	POWERINTEGRATOR2() replaced by hardware.POWERINTEGRATOR2()
	POWERNEXUS() replaced by hardware.POWERNEXUS()
	POWERPROBE() replaced by hardware.POWERPROBE()
	POWERTRACE() replaced by hardware.POWERTRACE()
	POWERTRACE2() replaced by hardware.POWERTRACE2()
	RETURNCODE() replaced by TERM.RETURNCODE()
	SCU() replaced by hardware.SCU()
	STG() replaced by hardware.STG()
	SYStem.UP() replaced by SYStem.Up()
	TA32() replaced by hardware.TA32()
	TO.COUNTER() replaced by TrOnchip.COUNTER()
	UTRACE() replaced by hardware.UTRACE()
	VERSION.SERIAL.PREPRO() replaced by VERSION.SERIAL.PREPROcessor()

Build 60219 IMP 02414 Script Language PRACTICE	**new PRACTICE functions** additional PRACTICE functions introduced: ADDRESS.TRACK.PROGram() synonym for ADDRESS.TRACK.PROG(), ARMARCHVERSION(), CONVert.INTTOADDRESS() synonym for CONV.INTTOADDRESS(), CPU.FEATURE(), GDB.PORT(), PERF.MEMORY.HITS(), PERF.MEMORY.SnoopAddress(), PERF.MEMORY.SnoopSize(), PERF.METHOD(), PERF.MODE(), PERF.PC.HITS(), PERF.RATE(), PERF.RunTime(), PERF.SNOOPFAILS(), PERF.STATE(), PERF.TASK.HITS(), RCL.PORT(), STRing.SPLIT(), SYStem.CONFIG.DCI.Bridge(), SYStem.CONFIG.DCI.BssbClock(), SYStem.GTL.CALLCOUNTER(), Trace.METHOD(), Trace.METHOD.IPROBE(), TrOnchip.Set.CPUBootStall(), VERSION.SERIAL.Integrator(), VERSION.SERIAL.NEXUSadapter(), VERSION.SERIAL.POWERPROBE(),

Build 56057

Build 56057 IMP 02413 Integrated Development Environment HELP (Online help)	**support of additional PDF viewer types added** The TRACE32 online help system officially supports the usage of alternative PDF viewer different from Adobe Reader. More details are described inside the IDE User's Guide.
Build 56057 IMP 02412	**Remote control for POWER DEBUG INTERFACE / USB3** t32tcpusb executable allows remote control for POWER DEBUG INTERFACE / USB3
Build 56057 IMP 02411 Uniform Look-And-Feel for all Processors WELCOME.view (Open the welcome window) WELCOME.SCRIPTS (Open the script search window) WELCOME.STARTUP (Open the welcome window if not disabled)	**Welcome dialog and script search** Welcome dialog for new installations and search dialog for scripts (debugger software only)
Build 56057 IMP 02410 Trace Analyzer <trace>.EXPORT.CSVFunc (Export the function nesting to a CSV file) <trace>.EXPORT.TASKEVENTS (Export task event to CSV)	**New Trace export commands** New commands to export trace information for analysis by a third-party tool.
Build 56057 ODD 02409 Script Language PRACTICE	**changed PRACTICE function** ITMBASE() has function parameter now
Build 56057 CHG 02408 Script Language PRACTICE	**renamed PRACTICE functions** DEBUGPORT() replaced by SYStem.CONFIG.DEBUGPORT(), DEBUGPORT.TYPE() replaced by SYStem.CONFIG.DEBUGPORTTYPE()

Build 56057 IMP 02407 Script Language PRACTICE	**new PRACTICE functions** additional PRACTICE functions introduced: CACHE.DC.LRU(), CACHE.IC.LRU(), CACHE.L2.LRU(), CACHE.L3.LRU(), CAnalyzer.CableTYPE(), CAnalyzer.I2C.DATA(), CHIP.GTMVersion(), Data.HByte(), Data.PByte(), Data.SByte(), MMU.FORMAT(), Onchip(), sYmbol.NAME.AT(), sYmbol.STATE(), SYStem.CONFIG.DEBUGPORT(), SYStem.CONFIG.DEBUGPORTTYPE(), SYStem.CONFIG.ListCORE(), SYStem.CONFIG.ListSIM(), SYStem.GTL.LIBname(), SYStem.GTL.PLUGINVERSION(), SYStem.GTL.VENDORID(), SYStem.GTL.VERSION(), SYStem.Option.MemoryMODEL(), SYStem.ReadPDRL(), SYStem.ReadPDRH(), Trace.METHOD.HAnalyzer(), Trace.STATE(), TrOnchip.Set.SMMINto(), TrOnchip.Set.SMMENtry(), TrOnchip.Set.SMMEXit(), TrOnchip.Set.VMENtry(), TrOnchip.Set.VMEXit(), TrOnchip.Set.VMEXitMask(), TrOnchip.Set.RESet(), TrOnchip.Set.GeneralDetect(), TrOnchip.Set.INIT(), TrOnchip.Set.MachineCheck(), TrOnchip.Set.ShutDown(), TrOnchip.Set.ColdRESet(), TrOnchip.Set.BootStall()
Build 56057 IMP 02406 <trace>.Chart.GROUP (Group activity chart) <trace>.Chart.Line (Graphical HLL lines analysis) <trace>.Chart.sYmbol (Symbol analysis) <trace>.STATistic.Address-GROUP (Address group run-time analysis) <trace>.STATistic.GROUP (Group run-time analysis) <trace>.STATistic.Line (High-level source code line analysis) <trace>.STATistic.sYmbol (Flat run-time analysis)	**Task specific flat statistics and charts** New options "TASK" and "SplitTASK" allows to produce task specific flat trace statistics and charts.
Build 56057 IMP 02405 PER.view (Display peripherals)	**Late PER file compiling** New "WAIT" command allows to delay the compilation for a PER window till certain information from the target is available.
Build 56057 IMP 02404 MMU.INFO (Translation information related to an address)	**New command MMU.INFO** Shows information about MMU translations for a given address.
Build 56057 IMP 02403 PER.Program (Interactive programming) PER.ReProgram (Load default program)	**PER programming extended** New commands ENTRY and INCLUDE allow modular PER files and passing of parameters to PER files.

Build 56057 CHG 02402	**64bit MIPS executable renamed** Renamed from t32mips5 to t32mips64. All related files and IDs also renamed.
Build 56057 IMP 02401	**Trace.FindAll Improved** The window will now show the found addresses in the flow and not the address from the trace cycle.

Build 51144

Build 51144 IMP 02397 Data.STRING (ASCII display)	**New formatting option %CONTinue for Data.STRING command** Allows to add string to output in AREA without having a newline first.
Build 51144 IMP 02396 Register.Set (Modify register contents)	**New pseudo registers for x86** The registers (CSR,DSR,ESR,FSR,GSR,SSR,TRR,LDTRR) allow to modify a segment register without updating the related base, limit and attribute registers.
Build 51144 IMP 02395 <trace>.Chart.sYmbol (Symbol analysis) <trace>.STATistic.sYmbol (Flat run-time analysis)	**Inline function support for Trace.STAT.sYmbol commands** Inline functions are treated as separate functions. The option NoInline can be used to discard inline functions.
Build 51144 IMP 02394 Frame (Call-tree and context)	**Better support for inline function debugging** Active inline functions now shown in status bar and Frame windows.
Build 51144 IMP 02393	**New command Trace.Chart.Address** Chart counterpart to Trace.STATistic.Address command.
Build 51144 CHG 02392 List (Display modes for programs)	**Stepping history visualization changed** The history now just shows where steps have been executed. Already stepped instructions for the Step.Diverge command are also shown. Executing Break while already stopped clears the history.
Build 51144 IMP 02391 Step.Diverge (Step till next unreached line)	**New command Step.Diverge** Command allow to step still the target branches to new locations that have not yet been stepped.
Build 51144 CHG 02390 Register.view (Display registers)	**Stack not shown by default in register window** The command line option Stack enables the stack display. The display can also be toggled by the button in the register window.
Build 51144 IMP 02389 <trace>.STATistic.DistriB (Distribution analysis) <trace>.STATistic.GROUP (Group run-time analysis) <trace>.STATistic.Line (High-level source code line analysis) <trace>.STATistic.MODULE (Code execution broken down by module) <trace>.STATistic.sYmbol (Flat run-time analysis)	**Option SplitTASK for many Trace.STATistic commands** Allows to split up the results for different tasks.

Build 51144 ODD 02388 ARM/Cortex Trace (parallel)	**Analyzer.TSelect EXT** External trigger input on ETM connector is now turned off by default. The command Analyzer.TSelect EXT can enable or disable the trigger function.
Build 51144 IMP 02385 AVX.view (Display AVX registers) AVX512.view (Display AVX512 registers) FPU.view (Display FPU registers) MMX.view (Open MMX register window) SSE.view (Display SSE registers) VPU.view (Display ALTIVEC register window)	**Larger than 64 bit SIMD register sets** FPU, VPE, SPE, MMX, SSE, AVX, AVX512 - Parser for wide registers (>64 bit) unified, values wider than 64 bit possible.
Build 51144 ODD 02384 Intel~ x86/x64 JTAG Debugger SYStem.Option MEMory-MODEL (Define memory model)	**LARGE memory model for Intel x86** Default memory model for Intel x86 is LARGE now.
Build 51144 IMP 02383 Script Language PRACTICE SETUP.EDITEXT (Define an external editor)	**Editor plug-in for PRACTICE syntax highlighting** Syntax highlighting files for TextPad, UltraEdit, Kate and Notepad++ are provided under demo\practice\syntaxhighlighting
Build 51144 IMP 02380 Script Language PRACTICE PBREAK.Delete (Delete breakpoint) PBREAK.DISable (Disable breakpoint) PBREAK.ENable (Enable breakpoint) PBREAK.List (Display breakpoint list) PBREAK.Set (Add breakpoint)	**multiple PRACTICE breakpoints introduced** PRACTICE can handle multiple PRACTICE breakpoints now. Commands PBREAK.Set, PBREAK.Delete, PBREAK.List, PBREAK.ENable, PBREAK.DISable added.

Build 51144 ODD
02379
Script Language PRACTICE
	removed PRACTICE functions
	PRACTICE functions removed:
	Onchip.GAP() replaced by MCDS.TraceBuffer.SIZE(), MCDS.TraceBuffer.LowerGAP() and MCDS.TraceBuffer.UpperGAP()
	FLASH.LIST.TYPE() replaced by FLASH.SECTOR.TYPE()

Build 51144 IMP 02377 Script Language PRACTICE	**new PRACTICE functions** additional PRACTICE functions introduced: ADDRESS.STRACCESS(), ARM64(), AVX(), CAnalyzer.BOTHCables(), CAnalyzer.DebugCable(), CAnalyzer.TracePort(), CHIP.STEPping(), CONVert.INTTOBOOL(), CONVert.INTTOHEX(), CPUIS64BIT(), DWTBASE(), FILE.TYPE(), FLASH.SECTOR.OTP(), FLASH.SECTOR.STATE(), FLASH.SECTOR.TYPE(), FLASH.SECTOR.WIDTH(), FLASH.TARGET.FILE(), FLASHFILE.SPAREADDRESS(), JTAG.ONTRIGGER.STATE(), MCDS.TraceBuffer.LowerGAP(), MCDS.TraceBuffer.SIZE(), MCDS.TraceBuffer.UpperGAP(), MMU.LINEAR(), MMX(), NEXUS.PortMode(), PP(), SSE(), Step.Diverge.ReachedTarget(), SYStem.CPU(), SYStem.OPBT8(), SYStem.RESetBehavior()

Build 50104

Build 50104 IMP 02376 Data.PROfile (Graphical display of data value) Var.PROfile (Graphical display of variable)	**New command Var.PROfile** HLL counterpart to Data.PROfile to show variables over time.
Build 50104 IMP 02375 Data.PROfile (Graphical display of data value)	**Data.PROfile command extended to support more data formats** Allows also wider data and floating point data now.
Build 50104 IMP 02374 ON (Event-controlled PRACTICE script execution)	**ON TIME can stop interruptible commands** This allows to limit the execution time of commands that take too long time.
Build 50104 IMP 02373 ISTATistic.state (Display ISTAT configuration window)	**New command ISTAT.state** Shows and controls various ISTAT related settings.
Build 50104 IMP 02372 COVerage.state (Configure coverage)	**New command COVerage.state** Shows and controls various COVerage related settings.
Build 50104 IMP 02371 COVerage.METHOD (Select code coverage method) ISTATistic.METHOD (Recording method for instruction statistics)	**New SPY method for COVerage and ISTAT commands** Allows to process the trace while recording in STREAM mode.
Build 50104 IMP 02370 ISTATistic.METHOD (Recording method for instruction statistics)	**New command ISTAT.METHOD** Defines the operation of the ISTAT command.
Build 50104 IMP 02369 COVerage.METHOD (Select code coverage method)	**New command COVerage.METHOD** Defines the operation of the COVerage command.
Build 50104 IMP 02368	**New state SPY for STREAM mode traces** The SPY state allows to process and view the trace while the streaming continues.
Build 50104 CHG 02367	**ETM counter values in SNOOPer changed** The value shown is now the virtual counter value (counting up from zero).

Build 50104 CHG 02366 Script Language PRACTICE IF (Conditional script execution) RePeaT (Loop with check at end of loop) WHILE (Loop with check at start of loop)	**Empty lines in PRACTICE allowed** Empty lines are now ignored in PRACTICE. This may change the execution flow of existing programs (e.g. when an IF, REPEAT or WHILE statement is followed by an empty line).
Build 50104 IMP 02365 <trace>.STREAMLOAD (Load streaming file from disk) <trace>.STREAMSAVE (Save streaming file to disk)	**Stream mode save and reload capability added** The commands allow to save and restore the trace data from STREAM mode.
Build 50104 IMP 02362 BMC.state (Display BMC configuration window)	**New Benchmarkcounter state window** Counter display and configuration replaced by a list-like window style. This speeds up the initial buildup and refreshing process of the window.
Build 50104 IMP 02358 Script Language PRACTICE	**new PRACTICE functions** additional PRACTICE functions introduced: ADDRESS.INTERMEDIATE(), ADDRESS.PHYSICAL(), CHIP.EmulationDevice(), CONVert.OCTaltoint(), CPUFLASHTYPE() Data.MAU(), Data.STRingN(), DEBUGPORT(), DEBUGPORT.TYPE(), IFTEST.LATENCY(), IFCONFIG.DEVICENAME(), IFCONFIG.IPADDRESS(), PER.ARG(), PER.ARG.ADDRESS(), PORTSHARING(), RTS.RECORDS(), SYStem.CONFIG.DRPRE(), SYStem.CONFIG.DRPOST(), SYStem.CONFIG.IRPRE(), SYStem.CONFIG.IRPOST(), SYStem.CONFIG.Slave(), SYStem.DETECT.CLTapchain(), SYStem.OPBT(), TIMEOUT(), UTRACE(), VPU(), VPUCR()
Build 50104 IMP 02355 Script Language PRACTICE PRINT (Write text and data to an AREA window (without format decoration))	**PRACTICE command PRINT extended** expression types <address> and <addressrange> full supported now

Build 42354

Build 42354 02356 AREA.OPEN (Open output file)	**option /NoFileCache added** disables TRACE32 file output buffer caching - useful when TRACE32 is killed by testframe
Build 42354 IMP 02354 MPC5xxx and SPC5xx Debugger MPC57xx NEXUS High Speed Serial Trace Port MPC55xx/MPC56xx NEXUS Debugger and Trace	**depletion recovery for affected on-chip flash modules added**
Build 42354 CHG 02353 MPC5xxx and SPC5xx Debugger MPC55xx/MPC56xx NEXUS Debugger and Trace	**short reset sequence (RGM_FESS) disabled during SYStem.Up**
Build 42354 IMP 02352 MPC5xxx and SPC5xx Debugger MPC57xx NEXUS High Speed Serial Trace Port	**run-time performance monitor register access (SNOOPer) added**
Build 42354 IMP 02351 RTOS Debugger for Linux	**support for physical addresses larger than 32bit improved** only 32bit architectures are affected
Build 42354 IMP 02350 RTOS Debugger for Linux	**support for ASLR (Address Space Layout Randomization) added**
Build 42354 IMP 02349 RTOS Debugger for Linux TASK.DMESG (Display the kernel ring buffer) TASK.DTB (Display the device tree blob)	**new commands for Linux awareness** TASK.DMESG to display kernel log TASK.DTB to display device tree blob
Build 42354 CHG 02348 RTOS Debugger for Linux	**Linux awareness splitted for Linux-2.x and Linux-3.x**
Build 42354 IMP 02347 TriCore Debugger TriCore Parallel Trace FLASH.Create (Declare FLASH)	**new command option /BootModeHeaDer**

Build 42354 IMP 02346 FLASH.Create (Declare FLASH)	**new command options /AutoInc and /KEEP** /AutoInc for symplifying FLASH.CREATE commands. /KEEP for protecting flash areas agains overwritting.
Build 42354 IMP 02345 FLASH.TARGET (Define target controlled algorithm)	**new command option /FirmWareRAM** Option supports protection and restoring of on-chip FLASH programming firmware during FLASH programming.
Build 42354 IMP 02344	**new command FLASH.CHANGETYPE for flash programming** Command changes the FLASH type for specific ranges.
Build 42354 IMP 02343 TRACE32 Software	**first official release supporting Power Debug Interface USB3**
Build 42354 IMP 02341 Script Language PRACTICE	**new PRACTICE functions** additional PRACTICE functions introduced: ART.RECORD.ADDRESS(), ART.RECORD.OFFSET(), ART.RECORD.TIME(), CAnalyzer.RECORD.ADDRESS(), CAnalyzer.RECORD.DATA(), CAnalyzer.RECORD.OFFSET(), LOGGER.RECORD.ADDRESS(), LOGGER.RECORD.DATA(), LOGGER.RECORD.OFFSET(), LOGGER.RECORD.TIME(), Onchip.RECORD.ADDRESS(), Onchip.RECORD.DATA(), Onchip.RECORD.OFFSET(), Onchip.RECORD.TIME(), SNOOPer.RECORD.ADDRESS(), SNOOPer.RECORD.DATA(), SNOOPer.RECORD.OFFSET(), SNOOPer.RECORD.TIME(), Trace.RECORD.ADDRESS(), Trace.RECORD.DATA(), Trace.RECORD.OFFSET(), Trace.RECORD.TIME(), CACHE.L2.SHARED(), CACHE.L2.SHAREDMASK(), DAP.Available(), Data.MAU(), Data.STRingN(), FLASH.UNIT(), FLASH.UNIT.BEGIN(), FLASH.UNIT.END(), FLASH.UNIT.EXIST(), FLASH.UNIT.NEXT(), FORMAT.CHAR(), FORMAT.STRing(), INTERFACE.CADI(), INTERFACE.MCD(), INTERFACE.VAST(), INTERFACE.VDI(), STATE.TARGET()
Build 42354 IMP 02340 sYmbol.List.SOURCE (Display source file names)	**Debug info keeps full original source name** The matching algorithm for source names has been extended to match also for files in subdirectories.
Build 42354 CHG 02339	**Merged traces handled differently** The new commands "ETMTrace", "ETMAnalyzer", "ETMCAnalyzer" and "ETMOnchip" show the pure ETM program trace. All other commands show the merged trace.
Build 42354 IMP 02338 LA.IMPORT.TraceFile (Import trace data where processing has failed)	**New command LA.IMPORT.TraceFile** Allows to re-import saved trace data for flow trace re-processing. Useful when processing was not possible when the trace was made.

Build 42354 IMP 02337 TERM.SIZE (Define size of terminal window)	**Terminal window can keep backlog** The third parameter of the TERM.SIZE command defines the size of the backlog. The backlog is updated whenever a line scrolls out of the "real" part of the TERM window.
Build 42354 IMP 02336 <trace>.Chart (Display trace contents graphically)	**Drag & Drop in Trace.Chart windows** Drag & Drop of the symbol name can be used to rearrange the window. Drag & Drop of the symbol to the upper window header removes the symbol from the custom placed symbols.
Build 42354 IMP 02335 <trace>.Timing (Waveform of trace buffer)	**Drag & Drop in Trace Timing Window** Drag and Drop of channels can be used to rearrange channels in the window.
Build 42354 IMP 02334 <trace>.DRAW.Data (Plot data values against time) <trace>.DRAW.Var (Plot variable values against time)	**New Trace Draw commands** Trace.DRAW.Data can draw any memory access, Trace.DRAW.Var can draw any static variable in the trace buffer.
Build 42354 IMP 02333	**Macro expansion in nested PRACTICE block** Macro expansion can now be turned off or on in nested blocks inside a PRACTICE program.
Build 42354 IMP 02332 ON (Event-controlled PRACTICE script execution)	**New ON commands** "ON ATRIGGER" reacts on Analyzer Trigger, "ON OTRIGGER" reacts on Onchip Trigger. "ON ABREAK" and "ON OBREAK" react on the transition to the Break state.
Build 42354 IMP 02331 Data.LOAD.Elf (Load ELF file)	**Load Debug info with Merge** The new option /MERGE <program> allows to merge debug information with existing pure symbols from an already loaded program.
Build 42354 IMP 02324 RTOS Debugger for OSEK/ORTI TASK.ORTI (AUTOSAR/OSEK support)	**TASK.ORTI generates ORTI menu** Depending on the contents of the loaded ORTI file, the TASK.ORTI command automatically generates additional menu entries for displaying OS objects and for performance analysis on selected OS attributes.

Build 38589

Build 38589 IMP 02323 TRACE32 Front-End	**Multi-Core support (AMP & SMP)** Multi-core virtual targets are support via MCD interface in SMP and AMP cases.
Build 38589 IMP 02322 TRACE32 Front-End	**TeakLite via MCD interface supported** The TeakLite DSPs are supported as virtual targets also via MCD interface.
Build 38589 IMP 02320 QorIQ PowerPC 32/64 Bit Debugger	**Improved synchronous break switch in AMP** For further descriptions and details of restrictions of the synchronous hardware based break please refer to the debugger_ppcqoriq.pdf manual.
Build 38589 IMP 02319 QorIQ PowerPC 32/64 Bit Debugger NEXUS.OCeaNport.Mode (Configure Nexus OCeaN message type) NEXUS.OCeaN-port<index>.TraceSELect (Select Nexus OCeaN trace type)	**OCeaN trace support added** The NEXUS window gives easy access to the options.
Build 38589 IMP 02318 Intel~ x86/x64 JTAG Debugger BMC.state (Display BMC configuration window) SNOOPer (Sample-based trace)	**Intel StopAndGo Snooping** For Intel Atom/x86 CPUs it is possible to snoop PC and BMC using StopAndGo. The snoop results can be used for profiling, charting and statistics.
Build 38589 IMP 02317 QorIQ PowerPC 32/64 Bit Debugger	**DDR trace support added** The NEXUS window gives easy access to the options.
Build 38589 FIX 02316 QorIQ PowerPC 32/64 Bit Debugger	**Improved Aurora HSTP and OnChip trace** Fixed problems when switching between HSTP and OnChip trace method under certain conditions, fixed OnChip FIFO mode, improved trace support for various QorIQ CPUs.
Build 38589 FIX 02313 MPC5xx/8xx Debugger Var.set (Modify variable)	**MPC500-Nexus: dualport access** fix dualport access for Var.set command
Build 38589 IMP 02312 ARC Debugger	**Added Support for ARC-EM family**

Build 38589 CHG 02309 C2000 JTAG Debugger C5500 Debugger C6000 Debugger Data.LOAD.COFF (Load COFF file)	**Code loaded to logical address** Data.LOAD.COFF for TI architectures now loads the code at the logical address of the COFF file. The option /PHYSLOAD can force the load to the physical address.
Build 38589 CHG 02308 Data.LOAD (Load file)	**Data.LOAD with verify changed** The verification will now be made after the complete code download.
Build 38589 IMP 02307 WinDEFaultSIZE (Apply a user-defined default size to windows)	**Default window creating size** The command WinDEFaultSIZE can be used to define the default size for new windows.
Build 38589 CHG 02306 WinPrint (Print address or record range of a window)	**Window pulldown item "Print All"** Now "Print All" or "To Clipboard All" will redirect the whole window content and not only the 1st page for the most window types with finite size.
Build 38589 IMP 02305 <trace>.STATistic.MODULE (Code execution broken down by module) <trace>.STATistic.PROGRAM (Code execution broken down by program)	**New flat statistic commands** Trace.STATistic.MODULE and Trace.STATistic.PROGRAM shows which module or program is executing the code.
Build 38589 IMP 02304 <trace>.STATistic.Func (Nesting function runtime analysis) <trace>.STATistic.TREE (Tree display of nesting function run-time analysis)	**Interrupts in nested trace statistics** Interrupts in nested trace statistic windows are now shown under a separate "pseudo" task named "(interrupt)".
Build 38589 IMP 02303 <trace>.SAVE (Save trace for postprocessing in TRACE32) <trace>.SAVE (Save trace for postprocessing in TRACE32)	**Trace save format changed** Now uses the new compression format by default. This provides a higher compression than the previous default format.

Build 38589 IMP 02302 Var.set, Var.Assign, Var.AddWatch, Var.Break.direct, Var.Break.Delete, Var.Break.Pass, Var.Break.Set, Var.Call, Var.CHAIN, Var.DRAW, Var.DUMP, Var.DelWatch, Var.FixedCHAIN, Var.FixedTABle, Var.Go, Var.Go.Back, Var.Go.Change, Var.Go.Till, Var.IF, Var.INFO, Var.Local, Var.LOG, Var.NEW, Var.NEWGLOBAL, Var.NEWLOCAL, Var.PRINT, Var.Ref, Var.Step.BackChange, Var.Step.BackTill, Var.Step.Change, Var.Step.Till, Var.TABle, Var.TYPE, Var.View, Var.Watch, Var.WHILE, Var.WRITE	**HLL expression parser extended** Pointers and address values can be casted to certain segments or memory access classes.
Build 38589 IMP 02301 ARM Debugger	**TrOnchip.Set vector catch capabilites ex** Allows to set secure/nonsecure/monitor vector catches for ARM cores with Trustzone support.
Build 38589 IMP 02300 ARM Debugger ETM.Set (Precise control of ETM trigger events)	**ETM.Set command extended** Allows to supply the data access width to control the comparison.
Build 38589 IMP 02299 GTM Debugger (TriCore) MPC5xxx and SPC5xx Debugger TriCore Debugger	**initinal support for BOSCH GTM added** Initinal support for the BOSCH GTM (Generic Timer Module) added. Available in TriCore and PowerPC devices.
Build 38589 IMP 02298 TriCore Debugger FLASH Programming (Memory-Mapped)	**protection for TriCore Boot Mode Header** Mechanism implemented for preventing unintended deletion or invalidation of a BMI Header to avoid a lock-out of your AURIX device. See the corresponding flash scripts in ~~/demo/tricore/flash/ for details.
Build 38589 IMP 02297 GTM Debugger (TriCore) PCP Debugger (TriCore) TriCore Debugger CLOCK (Display date and time)	**on-chip clock display and settings** The CLOCK command group simplifies the setup for timestamp decoding of an on-chip trace. Can also be used to verify PLL and clock setup.
Build 38589 IMP 02296 MPC5xxx and SPC5xx Debugger TriCore Debugger	**Automotive Debug Cable** Support for a universal automotive debug cable added. Currently TriCore and PowerPC are supported.

Build 38589 IMP 02295 TriCore Parallel Trace	**TriCore AGBT (Serial Trace)** Support for the TriCore AGBT (Aurora GigaBit Trace) has been added. Requirements for this feature is a Serial Trace v2 preprocessor and a TriCore AURIX Emulation Device. Demonstrator devices are not supported.
Build 38589 IMP 02294 Script Language PRACTICE	**new PRACTICE functions** additional PRACTICE functions introduced: Analyzer.MAXSIZE() ART.MAXSIZE(), ART.MODE(), ART.RECORDS(), ART.REF(), ART.SIZE(), ART.STATE(), ART.TRACK.RECORD(), BSDL.GetDRBit(), BSDL.GetPortLevel(), CIProbe.MAXSIZE(), CIProbe.SIZE(), COMPonent.AVAILABLE(), COMPonent.BASE(), COVerage.BDONE(), COVerage.SCOPE(), DIALOG.STRing2(), ETBCORESIGHT(), LA.MAXSIZE(), LA.SIZE(), LOGGER.SIZE(), NEXUS.PortSize(), Onchip.MAXSIZE(), PORT.MAXSIZE(), PORT.SIZE(), SNOOPer.MAXSIZE(), SNOOPer.SIZE(), STRing.Replace(), sYmbol.LIST.PROGRAM(), sYmbol.LIST.SOURCE(), Trace.MAXSIZE(), Trace.RECORDS(), Trace.SIZE(), VMX(), VMX.Guest(),
Build 38589 FIX 02293 PPC440 Debugger	**corrupt TLB (index=5)** TLB (index=5) could have been corrupted after STEP or STOP/GO
Build 38589 FIX 02289 SIM Instruction Set Simulator for TriCore	**dvinit.h simulation issue fixed**

Build 34458

Build 34458 IMP 02464	**new supported host OS version** This is the first TRACE32 software release which supports Windows 8.
Build 34458 IMP 02283 CTS.List (List trace contents) <trace>.List (List trace contents) <trace>.ListNesting (Analyze function nesting)	**New display format with SplitCORE option** Trace is shown separated for the cores.
Build 34458 IMP 02282	**New ENUM item for sYmbol.AddInfo** Allows to show values as enumerations.
Build 34458 IMP 02281 <trace>.TestFocusEye (Check signal integrity)	**New function Analyzer.FOCUS.EYE()** Can check an eye previously scanned by Analyzer.TestFocusEye agains violations.
Build 34458 IMP 02280 PLIST (List PRACTICE script) PMACRO (PRACTICE macros)	**PRACTICE debugging improved** Supporting macro and internal variable display and modification from PLIST and PMACRO windows.
Build 34458 IMP 02279 sYmbol.MARKER.Create (Marker for nesting function run-time analysis)	**New MARKER capabilities for nested stats** New TASKSWITCH marker defines when a taskswitch should take place (delayed taskswitch). Multiple markers can now be set on a single address.
Build 34458 IMP 02278	**New command Trace.MERGEFILE** Can combine two trace files into one. This is useful for traces recorded for different cores working in AMP mode.
Build 34458 IMP 02277 LA.IMPORT.ETB (Import on-chip trace data)	**New command LA.IMPORT.ETB** Can import pure binary ETB buffer contents for offline processing.
Build 34458 IMP 02276	**New command Trace.Chart.Nesting** Shows function call stack as time chart.
Build 34458 IMP 02275	**new ETM command** New command ETM.PortFilter for ARM ETM. Allows to control the hardware filter and compression of trace packets.
Build 34458 IMP 02274	**new trace commands** New commands Trace.STATistic.InterruptIsFunction, Trace.STATistic.InterruptIsKernel and Trace.STATistic.InterruptIsTaskswitch to interpret interrupt code as functions in nested trace statistics.

Build 34458 IMP 02273 sYmbol.MARKER (Fine-tune the nested function run-time analysis)	**new command group sYmbol.MARKER added** New command group sYmbol.MARKER to maintain markers. Markers can be used to handle special cases for nested trace statistics.
Build 34458 IMP 02272 TERM.METHOD (Select terminal protocol)	**TCP method added** New method TCP for TERM.METHOD command to support transfers via TCP. Can be used with TERM.TELNET to open telnet terminals.
Build 34458 IMP 02271 TERM.TELNET (Open TELNET terminal window)	**New command TERM.TELNET to support telne**
Build 34458 IMP 02270 Data.COPY (Copy memory) Data.PATTERN (Fill memory with pattern) Data.Set (Modify memory)	**option DIFF added** New option "DIFF" for commands Data.Set, Data.PATTERN and Data.COPY. Can be used to compare memory against values.
Build 34458 CHG 02269 MMU.DUMP (Dump MMU tables) MMU.List (Compact display of MMU translation table) MMU.SCAN (Scan MMU tables (static snapshot))	**new items for MMU commands** New item "KernelPageTable" for MMU.DUMP, MMU.LIST and MMU.SCAN commands. Shows kernel translations. "PageTable" shows currently active translations.
Build 34458 IMP 02268 Data.View (Display memory) PER.view (Display peripherals)	**command option CORE added** CORE option available in PER, Data.View and Data.PRINT commands.

Build 34458 IMP 02267
Script Language PRACTICE
	new PRACTICE functions
	additional PRACTICE functions introduced:
	FALSE() and TRUE()
	STRing.TRIM()

Build 34458 IMP 02264 Intel~ x86/x64 JTAG Debugger	**Add option "/core" to MMU window for x86**
Build 34458 02259 Uniform Look-And-Feel for all Processors PEDIT (Edit PRACTICE script)	**pedit not closing file after close**
Build 34458 FIX 02256 Uniform Look-And-Feel for all Processors	**b::dlgedit crashes in WIN64 bit**

Build 34458 FIX 02253 Host Driver Software	**online help linking** online help linking fixed for window prefix commands like Win, WinFreeze, WinBack, WinResist, WinPrint, WinSmall, WinMid, WinLarge, WinTrans, WinExt

Software Release from 04-Jun-2011

Build 30461

Build 30461 CHG 02252 Host Driver Software	**t32w95.exe renamed** Windows executable t32w95.exe renamed in t32win.exe
Build 30461 FIX 02251 8 Bit Emulation Compact Controller	**commands TM.S X TM.S TO locked** For hardware configuration ICE and ECC8 only the commands {TrMain||Trigger}.Set {eXception||TimeOut} were locked since build 25948 from 06.10.2010.
Build 30461 FIX 02250 Script Language PRACTICE	**new PRACTICE functions** additional PRACTICE functions introduced: PowerIntegrator: Integrator.COUNTER.EVENT(), Integrator.COUNTER.EXTERN(), Integrator.COUNTER.TIME(), Integrator.FLAG() PowerProbe: Probe.COUNTER.EVENT(), Probe.COUNTER.EXTERN(), Probe.COUNTER.TIME(), Probe.FLAG() IDE: CONVert.TOLOWER(), DIALOG.EXIST(), FILE.EOF(), FILE.EOFLASTREAD(), FILE.EXIST(), FCOS(), FINF(), FNAN(), FSIN(), OS.PED()
Build 30461 IMP 02248 Data.IMAGE (Display image data)	**Displaying image data enhanced** Support zooming via mouse scroll wheel. Advanced pixel operations like dumping data on right mouse button click, display pixel address in status bar, display pixel data in tool-tip.
Build 30461 IMP 02247 Data.WRITESTRING (Write string to PRACTICE file)	**New command Data.WRITESTRING** Writes string from target memory to PRACTICE file.
Build 30461 IMP 02246 OPEN (Open data file) WRITEB (Write binary data to file)	**PRACTICE can now handle binary files** New option /Binary for "OPEN" command allows to open or create binary files. New command "WRITEB" allows to write binary data to a file.
Build 30461 IMP 02245	**New command SETUP.STEPWITHINTASK** When enabled all HLL stepping and temporary breakpoints will be task selective (on the currently active task). This allows to step and debug shared code without stopping in another task.
Build 30461 IMP 02244 <trace>.PROfileChart (Profile charts)	**New PROfileChart window coloring** Names get now a fixed colors based on a hash of the name. Old (dynamic) coloring still available with the "/Color AlternatingColors" option.

Build 30461 IMP 02243 MIPS (Number of instructions per second)	**New trace analysis "MIPS" introduced** Allows to analyze "number of instructions per second" numerically or graphically.
Build 30461 IMP 02242	**New command to color SMP cores** The command "SETUP.COLORCORE" turn color coding single cores in SMP debug mode on or off.
Build 30461 IMP 02241	**Maximum size of windows increased** Allows wider file exports and support of larger multi screen windows.
Build 30461 IMP 02240 Data.dump (Memory dump)	**New STRING option for memory dumps** Allows to display strings in a dump window (newline separated lines).
Build 30461 IMP 02239	**New command ETA.state** Command allows global settings and provides links to the most common analysis commands.
Build 30461 IMP 02238 ARM/Cortex Trace (parallel)	**ETM+HTM trace correlation simplified** New "correlated" modes of "HTM.DataTrace" allow correlation of HTM trace to ETM. "Trace" commands shows merged traces.
Build 30461 IMP 02237 Combiprobe ARM/Cortex Trace (parallel) ITM (CoreSight ITM (Instrumentation Trace Macrocell))	**ETM+ITM trace correlation simplified** "ITM.DataTrace DataCorrelated" selects data trace function with correlation to ETM trace. "Trace" commands will then show merged trace data.
Build 30461 IMP 02236 ETA.PROfileChart (Power consumption by function as function of time)	**New command ETA.PROfileChart** Command shows power consumption by function as function of time.
Build 30461 IMP 02235 BMC.PROfileChart (Profile chart with benchmark counter)	**New command BMC.PROfileChart** Command shows benchmark counts related to certain code graphically as function of time.
Build 30461 IMP 02234 ARM/Cortex Trace (parallel)	**STM traces via Analyzer possible**
Build 30461 IMP 02233 <trace>.PROfileChart.COUNTER (Display a profile chart)	**New command Trace.PROfileChart.COUNTER** Allows to show time profiles of a counter that is traced as data value.

Build 30461 IMP 02232 CTS.List (List trace contents) <trace>.ListNesting (Analyze function nesting) <trace>.STATistic.TREE (Tree display of nesting function run-time analysis)	**Nesting analysis made error tolerant** The analysis will try to detect and fix problems in the function nesting caused by special target code or traceport FIFO overflows. The fixes and detected problems are be marked in the nested trace listing (Trace.ListNesting or CTS.List).
Build 30461 IMP 02231	**Option /MARKPC to debug optimized code.** The option highlights all instructions related to the current HLL line.

Software Release from 10-Nov-2010

Build 26464

Build 26464 CHG 02230 Host Driver Software	**discontinued windows versions** This is the last TRACE32 software release which supports Windows 9x, ME, and NT 4.0.
Build 26464 IMP 02229	**Autoloader window can be sorted by name**
Build 26464 IMP 02228 Data.DRAW (Graphical memory display of arrays) Var.DRAW (Graphical variable display)	**Graphical data display enhanced** Can now show up to six channels in one window. New option "Element" can select certain element of structure in an array of structures.
Build 26464 IMP 02226 <trace>.PROfileSTATistic (Statistical analysis in a table versus time)	**New command group Trace.PROfileSTATistic** Shows results of numerical interval analysis in tabular format.
Build 26464 IMP 02225 ARM/Cortex Trace (parallel) ETM.ReturnStack (Enable return stack tracing mode)	**New command EMU.ETM.ReturnStack** Enables return stack compression in ETM (PFT) trace.
Build 26464 IMP 02224 DTM (DTM trace sources (Data Trace Module))	**New command DTM to support CoreSight** DTM trace source can show the contents of simple CoreSight trace sources in different formats. Trace sources are typically either internal signals, busses or instrumentation traced.
Build 26464 IMP 02223 CombiProbe RTS (Real-time profiling (RTS))	**RTS supported with CombiProbe** Can be used for long time traces and coverage analysis for Cortex-M series.
Build 26464 IMP 02222 <trace>.PROTOcol.List (Display trace buffer for user-defined protocol)	**Protocol analysis API improved** Extended API allows handling of larger traces, faster processing and sharing of processed data among different protocol windows.
Build 26464 IMP 02221	**New command Trace.PROfileChart.PROTOcol** Allows to draw profile charts based on protocol analysis.

Build 26464 IMP 02220
Script Language PRACTICE
	new PRACTICE functions
	additional PRACTICE functions introduced:
	CA.MAXSIZE(), CA.SIZE(), CA.REF(), CA.STATE()
	I.MAXSIZE(), I.SIZE(), IPROBE.SIZE(), PROBE.SIZE()

Build 26464 FIX 02219 PPC440 Debugger	**Fix for Virtex5 dual core designs** two PPC440 core in one JTAG chain with FPGA is working now.

Build 22490

Build 22490 IMP 02218 Host Driver Software	**font installation under LINUX simplified**
Build 22490 IMP 02217 Host Driver Software	**UTF-8 font support under LINUX**
Build 22490 IMP 02216 Host Driver Software	**modified USB driver file for LINUX** The UDEV driver file for USB interface was modified to avoid troubles with newer kernel releases.
Build 22490 ODD 02215 PERF.List (Default profiling)	**PERF.List command behavior changed** Important: The performance analyzer isn't armed automatically per default anymore. The PERF.List* windows display 0.00 % values till the performance analyzer was armed at least once.
Build 22490 IMP 02214 JTAG Debugger	**New command SYStem.DETECT.JtagClock**
Build 22490 ODD 02213 Help System HELP.FILTER.Delete (Delete filter from filter list)	**online help** command HELP.FILTER.Delete syntax change - parameter is filter name instead of number now
Build 22490 IMP 02212 Script Language PRACTICE	**new PRACTICE functions** additional PRACTICE functions introduced: ADDRESS.SEGMENT(), ADDRESS.WIDTH(), CA.RECORD.TIME(), CONFIG.SCREEN(), DEBUGGER(), MCDS.GAP(), MCDS.SIZE(), MCDS.STATE(), ONCHIP.GAP(), ONCHIP.RECORDS(), ONCHIP.REF(), ONCHIP.SIZE(), ONCHIP.STATE(), ONCHIP.TRACK.RECORD() PER.BUFFER.BYTE(), PER.BUFFER.LONG(), PER.BUFFER.LONGLONG(), PER.BUFFER.QUAD(), PER.BUFFER.SHORT(), PER.BUFFER.WORD(), TITLE(), VERSION.DATE(), VERSION.SERIAL.CABLE(), VERSION.SOFTWARE()
Build 22490 IMP 02211 <trace>.STATistic.ParentTREE (Show the call context of a function)	**New command Trace.STATistic.ParentTREE** Allows to show the parents of a function.

Build 22490 IMP 02210 <trace>.STATistic.ChildTREE (Show callee context of a function)	**New command Trace.STATistic.ChildTREE** Allows to show all children of a function as a tree.
Build 22490 IMP 02209 <trace>.STATistic.Func (Nesting function runtime analysis) <trace>.STATistic.TREE (Tree display of nesting function run-time analysis)	**New options for Trace.STAT.TREE** Option /TASK allows a task selective display of information. With /MergeTASK it is possible to "overlay" the information of multiple tasks. The option SplitTASK splits the infomation of different tasks (the default).
Build 22490 IMP 02208 ETM (Embedded Trace Macrocell (ETM)) ITM (CoreSight ITM (Instrumentation Trace Macrocell))	**Merged ETM/ITM analyzer for Cortex-M** The merged analyzer allows a correlated processing of ETM and ITM trace data. This allows details nesting analysis for targets using an RTOS.
Build 22490 IMP 02207 BMC (Benchmark counters)	**Command BMC redesigned** New layout allows more flexibility to cover different target configurations.
Build 22490 IMP 02206 PERF.state (Display state)	**PERF command with new functionality** Performance measures now always all ranges and allows to show the results in different windows concurrently. New features include performance display for single instructions in list windows.
Build 22490 CHG 02205	**New command NEXUS for PowerPC** All traceport related settings are now in this command (removed from SYStem command).
Build 22490 CHG 02204	**New memory access classes for x86/x64** PP: changed to NP:, PD: changed to ND:, PR: changed to N: Old access classes can still be used.
Build 22490 IMP 02203 Var.Break.Set (Set breakpoint to HLL expression)	**Var.Break.Set data capabilities extended** The data expression can now also be a bitmask or range.
Build 22490 CHG 02202	**Sourcelines limited in mixed list window** Only the last view lines are shown. sYmbol.CUTLINE command can be used to change the limits.
Build 22490 IMP 02201 ARM/Cortex Trace (parallel) ETM.PseudoDataTrace (Enable pseudo data trace detection)	**Pseudo data trace for ARM ETMv3** Pseudo data trace allows to transmit data information via a pure program trace by target code instrumentation. This brings limited data trace to Cortex-A8 core.

Build 22490 IMP 02200	**New variable stickers introduced** Variable stickers can show variable values directly in the source window. Stickers are created by a DRAG&DROP operation of an expression to a source window.
Build 22490 CHG 02199 <trace>.STATistic.Address (Time between up to 8 program events) <trace>.STATistic.AddressGROUP (Address group run-time analysis) <trace>.STATistic.DistriB (Distribution analysis) <trace>.STATistic.GROUP (Group run-time analysis) <trace>.STATistic.Line (High-level source code line analysis) <trace>.STATistic.Var (Statistic of variable accesses) <trace>.STATistic.sYmbol (Flat run-time analysis)	**New options to change count behavior** CountChange counts all changes, CountFirst counts just the first location and CountALL counts each event. The defaults for the commands have been changed to provide useful results.
Build 22490 IMP 02198	**New command to time-track multiple GUIs** SYNCH.XTRACK connects window tracking requests of one GUI to other GUIs. This allows to track trace windows of multiple debuggers (replaces the old Analyzer.XTRACK command).
Build 22490 IMP 02197 <trace>.STATistic.GROUP (Group run-time analysis) <trace>.STATistic.TASK (Task activity statistic) <trace>.STATistic.sYmbol (Flat run-time analysis)	**New option Interval for statistics** The InterVal option allows to specify either a fixed time or a tracebuffer filter expression to specify the time slices. The statistic shows the minimum and maximum times and ratios for the events in a slice.
Build 22490 IMP 02196 <trace>.PROfileChart.GROUP (Group profile chart)	**New option Interval for profile charts** The InterVal option allows to specify either a fixed time or a tracebuffer filter expression to specify the time slices of most profile chart windows.
Build 22490 IMP 02195	**New marker types** KFROM/KTO markers allow marking address ranges as "kernel". IGNOREFROM/IGNORETO allows to ignore functions inside a range. CALL marker can mark an instruction to be a "call".
Build 22490 IMP 02194 PPC440 Debugger	**add PPC460GTX, APM821x1, APM83290**
Build 22490 IMP 02193 PPC440 Debugger	**Add PPC460SX**

Build 22490 IMP 02192	**Enhance T32m*.exe CMD line syntax** Implement cmd line options to allow separated argument lists for config file and startup script. Starting t32m*.exe with -? option on shell for more details.
Build 22490 IMP 02191 Script Language PRACTICE OPEN (Open data file)	**maximum opened files PRACTICE files** The limit of maximal 20 parallel opened files increased to 120. It's independent from the maximal number of opened sourcefiles.
Build 22490 IMP 02190 PER.view (Display peripherals)	**New commands for PER files** New command ASSERT to check for a condition before going on with parsing the PER file New command VARX for SGROUPS in PER-files, to assign a PRACTICE expression to the SGOUP buffer, which will be parsed continuously during the displaying of the PER file. New Functions PER.B.B(i), PER.B.W(i), PER.B.L(i) and PER.B.Q(i), to access the SGOUP buffer inside VARX commands.
Build 22490 IMP 02189 MPC5xxx and SPC5xx Debugger MPC55xx/MPC56xx NEXUS Debugger and Trace FLASH.ReProgram (Re-program FLASH)	**MPC55XX/56XX** FLASH.ReProgram now supports programming Flash with ECC errors without prior FLASH.Erase call

Build 19417

Build 19417 IMP 02465	**new supported host OS version** This is the first TRACE32 software release which supports Windows7.
Build 19417 IMP 02188 ARC Debugger <trace>.List (List trace contents)	**Support for ARC onchip trace (SmaRT)**
Build 19417 IMP 02187 MicroBlaze Debugger	**Chipscope supported for Microblaze**
Build 19417 IMP 02186	**Native debugger for Win32 and Linux** Native debug support for Windows (x86) and Linux (x86) is available now.
Build 19417 IMP 02185 TRACE32 Instruction Set Simulator for CORTEX / ARM / XSCALE SIM.CACHE (Cache/MMU simulation and more)	**MMU and Cache support in ARM simulator** MMUs and caches can also be simulated now.
Build 19417 IMP 02184 ARC Debugger TERM.GATE (Terminal with virtual hosting)	**Support for Metaware Hostlink library**
Build 19417 IMP 02183	**Support for API via .NET** The TRACE32 Remote API is also available for .NET now.
Build 19417 IMP 02182 Data.LOAD (Load file) sYmbol (Debug symbols)	**Support for code overlays** Supports different concepts of code overlays for multiple architecures.
Build 19417 CHG 02181 MPC5xxx and SPC5xx Debugger MPC55xx/MPC56xx NEXUS Debugger and Trace SYStem (System configuration) TrOnchip (Onchip triggers)	**SYStem.Up with CPU selection MPC55XX** The behavior of the MPC55XX (auto-selection type) has changed. When this type is selected, the debugger will reset all SYStem, TrOnchip and trace settings to default upon SYStem mode change. If settings are needed to be set before SYStem.Up, e.g. SYStem.Option.WATCHDOG, SYStem.Option.ETK, first select proper CPU type before, or use SYStem.DETect CPU before setting SYStem options.

Build 19417 IMP 02180 MPC5xxx and SPC5xx Debugger MPC55xx/MPC56xx NEXUS Debugger and Trace	**New command SYStem.Option.LPMDebug** Enables the support for the low-power debug handshake protocol to debug through low-power modes.
Build 19417 IMP 02179 ARM/Cortex Trace (parallel) LA.IMPORT.TracePort (Import off-chip trace data)	**New LA.IMPORT.TracePort command** Allows to import pure binary files from ETBs.
Build 19417 IMP 02178 SUBTITLE (Define a window subtitle for AMP debugging)	**New command SUBTITLE** Allows to add text in the header of each window. Can be use to distinguish different GUIs of a multicore target.
Build 19417 CHG 02177	**Register.RESet command renamed** New name is "Register.Init".
Build 19417 IMP 02176 <trace>.SAVE (Save trace for postprocessing in TRACE32)	**New compress option for Trace.SAVE** Compress option stores data in compressed format. The compression reduces the file size, but keeps access to the file quick. Trace save time is usually increased.
Build 19417 IMP 02175 sYmbol.NAMESPACES (Search symbol in C++ namespace)	**New command to define namespaces** The command sYmbol.NAMESPACES defines a list of namespace which are searched by the debugger (similar to "using" statement).
Build 19417 IMP 02174 MAP.BUS16 (Bus width mapping) MAP.BUS32 (Bus width mapping) MAP.BUS8 (Bus width mapping)	**Forcing memory access to access size** The MAP.BUSxx commands will now effect how the debugger accesses target memory.
Build 19417 IMP 02173 ARM/Cortex Trace (parallel) <trace>.AutoFocus (Calibrate AUTOFOCUS preprocessor)	**Autofocus algorithm changed** For high speed traceports and AutoFocus-II preprocessor the old algorithm has choosen the wrong eye (too high delay).
Build 19417 CHG 02172 ARM/Cortex Trace (parallel) Analyzer.SAMPLE (Set AutoFocus sample time offset)	**Sampling information calibrated** The delay values shown in Analyzer.ShowFocus and entered with Analyzer.SAMPLE are now calibrated. This has changed the values for AutoFocus-II significantly. Old saved sampling times are not longer valid.
Build 19417 IMP 02171	**Frequency increase for PowerTrace-I** Internal operation speed of PowerTrace-I for most targets increased from 80MHz to 100MHz.

Build 19417 IMP 02170 MAP.VMREAD (Redirect memory reads to TRACE32 virtual memory)	**Support for non-readable memories** The command MAP.VMREAD can redirect memory reads to the VM memory. This allows code execution and disassembly of code running in memory without data access path.
Build 19417 IMP 02169	**New command Trace.PROfileChart.MIPS** Can show instructions/second as time chart.
Build 19417 IMP 02168 <trace>.STATistic.AddressGROUP (Address group run-time analysis)	**New commands for address group stats** Trace.STATistic.AddressGROUP and related chart commands can analyze the number of accesses to certain memory regions.
Build 19417 IMP 02167 Cortex-M (ARMv6/7/8 32-bit) Debugger BMC (Benchmark counters) ITM (CoreSight ITM (Instrumentation Trace Macrocell))	**Cortex-M benchmark counter support** Benchmark counters can be traced by ITM and displayed graphically.
Build 19417 IMP 02166 MPC56x NEXUS Debugger and Trace CTS (Context tracking system (CTS))	**CTS on PowerPC extended to support FPU** CTS reconstructs now also the FPU registers.
Build 19417 IMP 02165 Data.LOAD.CCSDAT (Load CCSDAT file) Data.SAVE.CCSDAT (Save CCSDAT file)	**New file format CCSDAT supported** Format is compatible with TI Code Composer Studio (R) data file format.
Build 19417 CHG 02164 MMU (Memory management unit)	**MMU command structure changed** Command structure changed to match CACHE command structure.
Build 19417 CHG 02163 MMU (Memory management unit) TRANSlation (Debugger address translation)	**Debugger MMU commands renamed to TRANS** All commands that just effect the debuggers address translation are moved to the new command "TRANSlation".
Build 19417 CHG 02162 In-Circuit Emulator for 386 MMU.view (View MMU registers)	**80386 MMU window changed** Attribute registers shown as 16bit values now.
Build 19417 IMP 02161 ARM/Cortex Trace (parallel) <trace>.TestFocus (Test trace port recording)	**Utilisation test for ETM traceports** "Analyzer.TestFocus /Utilisation" can test the utilisation of the traceport and uncover CoreSight trace bus configuration problems.

Build 19417 IMP 02160 ARM Debugger MIPS32/MIPS64 Debugger MPC5xx/8xx Debugger Data.LOAD (Load file)	**New DIFFLOAD options and targets** New options allow further compression or compressed load (/ZIPLOAD) and quick download checks (/CHECKLOAD and /CHECKONLY). Target agents for MIPS32 and MIPS64 architectures added.
Build 19417 CHG 02159	**Function D.S() renamed to D.STRING()** The long name "DATA.STRING()" remained unchanged.
Build 19417 IMP 02158 Register (Processor registers)	**New options for Register, Frame and List** Allows to select the register context. "REGSET" option selects a registerset for architectures with multiple sets, "TASK" selects the registers of a certain task (when the task awareness supports it) and "CORE" selects a certain core (in SMP configurations).
Build 19417 IMP 02157 Data.LOAD (Load file)	**STRIPPART option can do multiple parts** Option with string parameter can be repeated to strip away multiple different parts.
Build 19417 IMP 02156 ARM/Cortex Trace (parallel) DTM (DTM trace sources (Data Trace Module))	**Custom trace source support** The DTM command provides support for "simple" custom CoreSight trace sources.
Build 19417 IMP 02155 ENCRYPTDO (Encrypt a PRACTICE script (*.cmm))	**Possibility to create encrypted PER file** Encrypts PER files that can only be shown, but not dumped or edited.
Build 19417 IMP 02154 sYmbol.Modify (Modify symbols)	**New sYmbol.MODIFY commands** Allow to convert labels into ranges and vice versa or ranges into functions.
Build 19417 CHG 02153 x386 and x486 Monitor	**Executable t32m86 renamed** New name is t32mx86. Covers all x86 32bit targets.
Build 19417 IMP 02152	**INTERCOM.EVALUATE extended** Can also process functions that return 64bit values or strings now.
Build 19417 IMP 02151 FLASHFILE.DUMP (Dump FLASH)	**New command FLASHFILE.DUMP** Dumps NAND flash contents.
Build 19417 FIX 02150 Analyzer (Trace method Analyzer, recording, and analysis commands)	**Bug fixed in ARM/Thumb trace via GDI** Display of ARM/Thumb mode traces made with GDI API was wrong (swapped).

Build 19417 IMP 02149 Break.Set (Set breakpoints) Var.Break.Set (Set breakpoint to HLL expression)	**Data value breakpoint support** Data value breakpoints are now supported via stop and go when the target has no data comparision capabilities.
Build 19417 IMP 02148 Trace.METHOD (Select trace method)	**New trace methods for SMP traces** MergedAnalyzer and MergedOnchip show the trace for all cores together. Same as supplying the /MergeCORE option to the trace commands.
Build 19417 IMP 02147 CTS.ListNesting (Analyze function nesting) <trace>.ListNesting (Analyze function nesting)	**New command Trace.ListNesting** Command shows execution flow similar to CTS.List window.
Build 19417 IMP 02146 ARM/Cortex Trace (parallel) ETM.Register (Display the ETM registers)	**ETM.Register window enhanced** Contains now also related CoreSight components.
Build 19417 CHG 02145 ARM/Cortex Trace (parallel) Analyzer.TraceCLOCK (Set capture rate for single-wire-trace)	**Command Analyzer.ExportCLOCK renamed** New name is Analyzer.TraceCLOCK.
Build 19417 IMP 02144 Data.LOAD (Load file)	**D.LOAD command address syntax extended** Now many load commands can relocate and limit the download at the same time (supply single address and address range parameters).
Build 19417 IMP 02143 ARM/Cortex Trace (parallel) Analyzer (Trace method Analyzer, recording, and analysis commands) ETM.PortSize (Define trace port width)	**Trace buffer usage for 8bit ETM changed** Doubles available trace records and increases frequency limit of older trace modules for ETM ports with 8 or less bits.
Build 19417 IMP 02142 <trace>.STATistic.CYcle (Analyze cycle types)	**Trace.STAT.CYcle command enhanced** Shows more information about execution times, idles and trace gaps.
Build 19417 CHG 02141 List (Display modes for programs)	**Logger command renamed to LOGGER**

Build 19417 CHG 02140 ARM/Cortex Trace (parallel) <trace>.LOAD (Load trace file for offline processing) <trace>.SAVE (Save trace for postprocessing in TRACE32)	**Trace save/load for cycle accurate trace** Information in file is now using clock information. The display of timestamps in the loaded trace will depend on the Trace.CLOCK command.
Build 19417 IMP 02139 Data.LOAD.Binary (Load binary file)	**Binary file save/load can compress** New options /ZIP and /UNZIP can compress/decompress binary files.
Build 19417 IMP 02138	**New command Trace.ISTATistic** Command group for instruction execution statistics. Operation is similar to Trace.COVerage command. Results can show how much time a single instruction (or block of instruction) take and how often it is executed.
Build 19417 IMP 02137 <trace>.EXPORT (Export trace data for processing in other applications)	**Trace.EXPORT command enhanced** New options to support various compressed binary formats.
Build 19417 IMP 02136 ARM Debugger	**MMU.TableWalk support for WinCE 5.x**
Build 19417 IMP 02134 SYStem.CONFIG (Configure debugger according to target topology)	**New command SYStem.CONFIG ListCORE** Allows to show running platforms for CADI and MCD frontend debuggers.
Build 19417 CHG 02133 ARM Debugger FDX.METHOD (Select communication channel)	**FDX method DCC wrong** Is named "DCC4" now. "DCC3" method introduced.
Build 19417 IMP 02132	**ZERO timestamp correlation improved** All timestamps are now correlated (including SNOOPER, ART and ZERO times).
Build 19417 IMP 02131 ARM/Cortex Trace (parallel) ETM.PortSize (Define trace port width)	**ETM support for special port sizes** Supporting also 3,5,6,7 bits (in normal mode) and unaligned trace data.
Build 19417 CHG 02130 Frame (Call-tree and context)	**Flag command group renamed to FLAG** The abbreviation "F" of the command group "FLAG" removed.
Build 19417 CHG 02129	**Enhance T32m*.exe CMD line syntax** Mips34k_AMP has been removed from the cpu selection list. Both AMP and SMP debugging is supported with cpu selection MIPS34K. Dependent on the number of Power View Instances either AMP or SMP debugging is done.

Build 19417 IMP 02128	**GPR shadow register handling** Display and handling of available GPR shadow register sets, for eg. complete stack back trace.
Build 19417 IMP 02127 MIPS32/MIPS64 Debugger	**SPR overlay handling** Implement read and write access to instruction Scratch Pad RAM (only for ksegs).
Build 19417 IMP 02126 MIPS32/MIPS64 Debugger	**TCB Trigger and Filter** Add TCB Trigger and Filter functionality.
Build 19417 FIX 02125 sYmbol.CREATE.MACRO (Create user-defined macro) sYmbol.NEW.MACRO (Create user-defined macro)	**Macro creation not working** The macro name and contents are mixed up.
Build 19417 FIX 02124 ARM/Cortex Trace (parallel) Break.Set (Set breakpoints) ETM (Embedded Trace Macrocell (ETM))	**ETM selective data trace** Selective data trace may miss cycles in ETMv3.
Build 19417 FIX 02123 ARM/Cortex Trace (parallel) Break.Set (Set breakpoints) ETM (Embedded Trace Macrocell (ETM))	**ETM trace of program ranges** Selective tracing program ranges failed for ETMv3.
Build 19417 FIX 02122	**Performance on Solaris-Sparc poor** The performance is much lower than expected due to a data alignment problem.
Build 19417 FIX 02121 Data.LOAD.Elf (Load ELF file)	**Wrong "double" format for old ARM files** The alternate double format is not selected as default when loading old ARM ELF files. Workaround is to add the /ALTDOUBLE option to the load command.
Build 19417 FIX 02120	**Trace display may crash on 64bit hosts** Application may crash, e.g. when setting reference cursors.
Build 19417 CHG 02119 PPC400 Debugger PPC440 Debugger	**improve bus error handler for PPC4xx**
Build 19417 IMP 02118 PPC400 Debugger SYStem.CPU (Select CPU)	**Add PPC405EX/405EXr**
Build 19417 IMP 02117 PPC400 Debugger PPC440 Debugger	**new command SYS.DETECT.CPU** Add CPU autotedection for PowerPC 440/40x. (command: SYStem.DETECT.CPU)

Build 19417 IMP 02081 PPC440 Debugger	**Add 460GX, 460GT, Virtex5**

Build 13751

Build 13751 IMP 02116 DIALOG.MESSAGE (Create dialog box with an information icon)	**New standard dialog** Creates a dialog with an information message.
Build 13751 IMP 02115 TERM.GATE (Terminal with virtual hosting) TERM.WRITE (Write terminal output to file)	**TERM.WRITE works also for TERM.GATE** Only terminal output is now saved into the file.
Build 13751 IMP 02114 STOre (Store settings as PRACTICE script)	**STORE of global macros possible** STORE <file> GLOBAL stores all global macros.
Build 13751 IMP 02113	**Statusbar extended** Information for core (for SMP systems) and task (for RTOS awareness) added. Both with pulldown menu to change the current core or task.
Build 13751 IMP 02112 C5500 Debugger	**Correlated program & data trace for C55x**
Build 13751 IMP 02111 ARM Debugger MMX (MMX registers (MultiMedia eXtension))	**XScale MMX register support**
Build 13751 IMP 02110 Host Driver Software	**Graphics on Windows improved** Support for ClearType fonts and additional colors for bitmaps. Adapted to Windows Vista.
Build 13751 IMP 02109 Host Driver Software	**Graphics on Motif improved** Support for custom fonts added.
Build 13751 IMP 02108 ARM Debugger ETM.DataTrace (Configure data-trace)	**New data trace modes**
Build 13751 IMP 02107 Data.SAVE.DAB (Save DAB file)	**Save commands support muliple ranges**

Build 13751 IMP 02106 ColdFire Trace	**Trace & Trigger on Coldfire trace** WDDATA and PULSE instructions can be used to control the trace.
Build 13751 IMP 02105 Data.LOAD.DAB (Load DAB file) Data.SAVE.DAB (Save DAB file)	**DAB file format supported**
Build 13751 IMP 02104 TRACE32 Maintenance Check	**Floating licenses supported** Frontend debuggers (via API) are using a floating license.
Build 13751 IMP 02103	**ON command extended** Now ON can also invoke scripts with ON <event> DO <script>. GLOBALON can install an ON handler on PRACTICE toplevel.
Build 13751 IMP 02102 RunTime (Runtime measurement)	**Tolerance information added to RunTime** Tolerance displayed in tooltip of last execution time.
Build 13751 IMP 02101 Data.LOAD.Elf (Load ELF file)	**Support for enum consts improved** New option /ENUMCONSTS to make enum constants available at any place in HLL expressions.
Build 13751 IMP 02100	**Data.LOAD.SYMBIAN extended** Now also generates module address range information.
Build 13751 IMP 02099	**TASK.SELect comand extended** Command now also usable in stop-mode debug (same effect as R.TASK command).
Build 13751 IMP 02098	**Dynamic MMU tablewalk improved** CTS and flowtrace work now also with MMU.TABLEWALK ON.
Build 13751 IMP 02097 Break.Set (Set breakpoints)	**Default breakpoint types changed** For program breakpoints the default is now to try a software implementation and switch to onchip if this fails.
Build 13751 IMP 02096 <trace>.STATistic (Statistic analysis)	**New columns for Trace.STAT commands** "TASK" shows task number. "CORE" shows core number (SMP). The old "TASK" column is renamed to "ExternalTASK".
Build 13751 IMP 02095	**Support for On-Demang-Paging** MMU.SHADOW allows to display paged out code in the list windows by looking at a copy of the code in VM. MMU.PAGER allows to set software breakpoints in paged code areas.
Build 13751 IMP 02094 Data.DRAWFFT (Graphical display of fast fourier transformation)	**FFT display command** FFT (Fast Fourier Transform) supported by command Data.DRAWFFT.

Build 13751 IMP 02093 XC800 Debugger In-Circuit Emulator for 8051	**OMF2 format supported**
Build 13751 IMP 02092 ARM/Cortex Trace (parallel)	**External Timestamping improved** New command Analyzer.ExportCLOCK to define export trace clock for CoreSight targets. Command Analyzer.ExportCLOCKDelay can fix additional delay on some trace ports.
Build 13751 IMP 02091 ICD Solutions for ARM CAnalyzer (Trace features of Compact Analyzer) SystemTrace (MIPI STP and CoreSight ITM)	**CORTEX-M3 Trace Support** Support for Cortex-M3 tracing via SWV and TPIU for ETM and ITM added.
Build 13751 IMP 02090	**OCPWP** New command OCP to support OCPWP trace unit.
Build 13751 IMP 02088 ICD Solutions for ARM STM (Configure STM component on target)	**STM** New command STM to support configuration of STM and SDTI trace ports.
Build 13751 IMP 02087 ICD Solutions for ARM	**Trustzone support extended** New memory access classes allow to access secure or nonsecure memory. This also allows to use the ETM to trace a mixture of secure and nonsecure code.
Build 13751 IMP 02085 ICD Solutions for ARM Break.Set (Set breakpoints) Var.Break.Set (Set breakpoint to HLL expression)	**ARM ETM Breakpoints Extended** ETM Breakpoints now support also combined program/data breakpoints and breakpoints with condition pass/fail.
Build 13751 IMP 02083 TriCore Debugger CACHE.view (Display cache control register)	**Cache Dump added** For the members of the AUDO-FUTURE chip family, the Cache Dump features have been added.
Build 13751 IMP 02082 Integrated Development Environment	**new IDE functions** the following PRACTICE functions were added: CLOCK.DAY(), CLOCK.MONTH(), CLOCK.YEAR(), OS.PCF(), OS.PHELPD(), OS.PLF(), OS.PPD(), OS.PPF()
Build 13751 CHG 02080 PPC400 Debugger PPC440 Debugger TrOnchip (Onchip triggers)	**change mode of r/w data breakpoints** Mode for data BP will be ASYNCHRONOUS instead of SYNCHRONOUS by default from now on. With 'TrOnchip.SYNCHRONOUS ON' could be switched to old mode.

Build 13751 FIX 02077 MPC5xx/8xx Debugger	**Bus error on byte/word write access** Each write to byte or word wide accesses result in a bus error.
Build 13751 FIX 02076 Host Driver Software	**t32rem** t32rem crashed with too long command line or return message strings
Build 13751 FIX 02075 Installation Guide	**Silent mode of TRACE32 installer** Due to an error inside the TRACE32 installer software from InstallShield the silent mode didn't work correctly. Please don't use the default path and filename C:\windows\setup.iss!
Build 13751 IMP 02073 LICENSE.List (Display all license information)	**New command LICENSE.List** Opens window which displays license information.
Build 13751 IMP 02072 Host Driver Software	**new PRACTICE function EOF()** Function returns a boolean whether the last READ command reached the file end or not.
Build 13751 IMP 02069 FLASH.ReProgram (Re-program FLASH)	**New command FLASH.ReProgram** New FLASH.ReProgram command for optimized flash programming performance.
Build 13751 IMP 02068 Host Driver Software	**general command parameter parser** comparisions == and != added for boolean values
Build 13751 FIX 02067 SYStem.Option (Special setup)	**SYStem.Option SOFTLONG fails**
Build 13751 FIX 02066 FIRE Fully Integrated RISC Emulator Out (Output control)	**fire analyser trigger unit - OUT.A** output signal from fire analyser trigger unit didn't work
Build 13751 IMP 02065 Data.LOAD.PureHex (Load hex-byte file) Data.SAVE.PureHex (Save pure HEX file)	**new commands for handling purehex files**
Build 13751 ODD 02064 Integrated Development Environment Eval (Evaluate expression)	**EVAL EVAL() EVAL.TYPE()** behavior of command EVAL changed: now an empty expression parameter sets value to 0 and value type to EMPTY instead of using the old values PRACTICE functions EVAL() and EVAL.TYPE(): expression type EMPTY added
Build 13751 FIX 02063 XScale Debugger	**SYStem.Up for XScale 80219/80321 fails**
Build 13751 IMP 02062 TriCore Debugger	**Cache Analysis added** For the members of the AUDO-NG and TC116x familiy the Cache Analysis feature has been added.

Build 13751 IMP 02061 TriCore Debugger	**FPI Bus Trace for on-chip trace** For the TriCore Emulation Devices, PowerView now supports tracing of the TriCore FPI buses SPB and RPB (System Peripheral Bus, Remote Peripheral Bus).
Build 13751 IMP 02060 TriCore Debugger TrOnchip (Onchip triggers)	**MCBS feature added for TriCore and PCP** Newer TriCore chips (e.g. the AUDO-NG or TC11xx family) have two internal break buses and a suspend bus which can be used to distribute break signals beween different modules (MLI, DMA, ...) and cores (TriCore, PCP). Configuration of this feature can be done via the TrOnchip window or the TrOnchip commands.
Build 13751 CHG 02059 TriCore Debugger	**Trace Enable now handled by Sofware** Enabling the trace output from the core (TriCore, PCP) to the trace port is now done by PowerView, depending on the selected trace source, for off-chip trace (OCDS-L2) as well as for on-chip trace (OCDS-L3). If the chip has a multiplexed traceport (e.g. for GPIO - OCDS-L2), the port configuration still has to be done by the user. See the demo scripts in ~~/demo/tricore/hardware and ~~/demo/tricore/etc/mcds for more details.

Software Release from 12-Sep-2007

Build 9640

Build 9640 IMP 02057 ETM.Set (Precise control of ETM trigger events)	**EMU.Set command extended** Support for context ID comparators, data mismatch compare and executepass/executefail comparisons. Also support for TraceON/TraceOFF control.
Build 9640 IMP 02056 ARM/Cortex Trace (parallel)	**ETM programming dialog extended** Supports now new data, external input, external output and trace controls.
Build 9640 CHG 02055 ARM Debugger C5500 Debugger SH Debugger BMC (Benchmark counters)	**Benchmark counter commands renamed** All benchmark counter related onchip trigger commands have been moved to the new command "BMC".
Build 9640 IMP 02054 <trace>.Chart.Var (Variable chart) <trace>.STATistic.Var (Statistic of variable accesses)	**New variable access statistics** Shows statistics simmilar to Trace.STATistic.sYmbol, but for data access cycles.
Build 9640 IMP 02053 Data.LOAD.COFF (Load COFF file) Data.LOAD.Elf (Load ELF file)	**Const variables supported** Constant variables in DWARF2 debug format are now also loaded.
Build 9640 IMP 02052	**Better support of multiple address lines** Source lines with multiple addresses (e.g. inlined functions) are now handled better in the HLL source view.
Build 9640 IMP 02051 CTS.CACHE.ViewStalls (Display statistics for idles/stalls)	**Stall analysis improved** Differentiates between data and prefetch stalls. New MIPS profile chart.
Build 9640 IMP 02050 <trace>.STATistic.Func (Nesting function runtime analysis) <trace>.STATistic.TASKFunc (Task related function run-time analysis) <trace>.STATistic.TASKTREE (Tree display of task specific functions) <trace>.STATistic.TREE (Tree display of nesting function run-time analysis)	**New options for function statistics** New options /IncludeOwn, /IncludeTASK /IncludeINTR change the way times for interrupts and other tasks are accumulated in the analysis. Default display is now to exclude times of interrupts and other tasks. With /INTRROOT and /INTRTASK it is possible to split up the interrupt times to the different tasks or count them globaly.
Build 9640 IMP 02049 Data.LOAD (Load file)	**New option /cygdrive to convert pathname** Converts cygwin pathnames into regular windows pathnames.
Build 9640 IMP 02048 SYStem.RESetTarget (Release target reset)	**New command SYStem.RESetTarget** Resets target system. On most targets simmilar to system.up and register.reset. On virtual platforms usually activates a target platform reset.

Build 9640 IMP 02046 SYStem.POLLING (Polling mode of CPU)	**New command to control JTAG polling** Allows to increase or decrease polling of target via JTAG when target is running.
Build 9640 IMP 02045	**Autoloader support extended** Works now also for Linux and WinCE.
Build 9640 IMP 02044	**New command GROUP.CreateTask** Allows to group different tasks together. Mainly used for Trace.PROfileChart displays.
Build 9640 IMP 02042 <trace>.STATistic.FIRST (Start point for statistic analysis) <trace>.STATistic.LAST (End point for statistic analysis)	**New commands to limit trace statistics** Allows to limit numerical statistical analysis to a smaller part of the trace.
Build 9640 IMP 02041 Data.LOAD.CORE (Load Linux core dump file)	**New command Data.LOAD.CORE for coredumps** Allows to load Linux coredump files for post-mortem analysis.
Build 9640 CHG 02040 PowerProbe - Logic and Protocol Analyzer	**PowerProbe prompt removed** PowerProbe can now only be controlled with "Probe" command of the debugger.
Build 9640 IMP 02039 <trace>.List (List trace contents)	**New CLOCK item for trace display** Allows to display time values as number of clocks. Syntax simmilar to TIME item.
Build 9640 FIX 02038	**MMU** Bug fixed in MMU translation (also without MMU) for cores with <4GB address space.
Build 9640 FIX 02037 PowerIntegrator - Logic and Bus Analyzer <trace>.Arm (Arm the trace) <trace>.Init (Initialize trace)	**no sampling at all** under circumstances the PowerIntergrator sampled nothing at all
Build 9640 IMP 02036 Trace Analyzer	**A.MODE()** new PRACTICE function A.MODE() introduced which returns the actual mode of the analyzer
Build 9640 IMP 02035 Unspecific Worktime Account APU.GREP (Search for string)	**APU.GREP command implemented** APU.GREP works similar to Data.GREP, but opens APU.* windows instead of Data.* windows.
Build 9640 IMP 02034 TriCore Debugger SYStem.Option STEPSOFT (Step with software breakpoints)	**TriCore soft single-stepping added** TriCore single-stepping can now also be performed by using software breakpoints. Use SYStem.Option STEPSOFT ON to enable this feature. The default is using on-chip breakpoints.
Build 9640 FIX 02033 Host Driver Software	**Systeminfo usage under UNIX failed** menuitem Help -> Support -> Systeminfo... didn't worked properly under UNIX file support.cmm won't be found

Build 9640 FIX 02031 FIRE Fully Integrated RISC Emulator Host Driver Software	**named USB devices on FIRE** storing of devices name caused crash
Build 9640 IMP 02030 Host Driver Software	**support of named USB devices added** Multiple named USB devices can be used simultaneously on Windows and Linux (x86). This feature requires a debugger firmware update (at least V8.0) and for Windows use of the current USB driver (V5.5).
Build 9640 FIX 02029 MPC5xx/8xx Debugger	**MPC500/800 hang on old PowerDebugModules**
Build 9640 FIX 02028 MPC56x NEXUS Debugger and Trace	**MPC565 MaskD:SIUMCR display error** 8-bit burst bit of the SIUMCR register was cleared after step/break. This will only take effect on the MPC565 MaskD!
Build 9640 FIX 02027 MPC5xx/8xx Debugger SYStem.Option (Special setup)	**MPC500/800:SYStem.Option.SCRATCH** independent from the SYStem.Option.SCRATCH setting the AUTO mode was used.
Build 9640 IMP 02023 TriCore Debugger SYStem.RESetOut (Reset peripherals)	**Target reset feature added** SYStem.RESetOut triggers a 2 ms low pulse on the nRESET line on the JTAG connector. This will reset the CPU. Formerly this command was only available in the Simulator.

Build 8248 IMP 02026 PRINT (Write text and data to an AREA window (without format decoration))	**PRINT command** new expression types TIMERANGE and RANGE added
Build 8248 FIX 02025 TriCore Debugger Break.Set (Set breakpoints)	**Support for 4 Onchip Breakpoints in Code**
Build 8248 FIX 02024 SIM Instruction Set Simulator for HC12/MCS12	**EORX instruction bug fixed** Instruction EORX wrote result to register IY instead of IX.
Build 8248 IMP 02022 <trace>.STATistic.TASKKernel (Task analysis with kernel markers (flat))	**New command Trace.STATistic.TASKKernel** Make a runtime statistic of tasks and kernel runtimes. Kernel entry and exit must be marked with sYmbol.NEW.MARKER commands.
Build 8248 IMP 02021	**New command sYmbol.NEW.MARKER** Allows to create markers for statistic analysis. Kernel markers allow the kernel to be separated from the application code. Functions inside the kernel are not tracked. New Trace.STATistic.TARKKernel command is based on these markers.
Build 8248 IMP 02020 CTS.PROfileChart (Profile charts) <trace>.PROfileChart (Profile charts)	**New command Trace.PROfileChart** Allows to display distributions vs. time graphically. Useful to show processing ratio of selected tasks versus time.
Build 8248 FIX 02019 PPC400 Debugger	**PPC in Xilinx FPGA** SW expect a wrong peripheral file for VirtexPPC, VirtexPPC1st and VirexPPC2nd. Debugger do not work at all with VirtexPPC, VirtexPPC1st and VirexPPC2nd.
Build 8248 CHG 02018 <trace>.FILE (Load a file into the file trace buffer) <trace>.LOAD (Load trace file for offline processing)	**Trace.LOAD functionality changed** The loaded trace data is now shown without the need for the /FILE option in place of the real traced data. The old functionality is available with the Trace.FILE command (useful for comparing saved traces with actual traces).
Build 8248 IMP 02017 Var.set (Modify variable)	**Var commands possible without file load** HLL expressions can now also be handled when no HLL file is loaded.

Build 7466 IMP 02016 FLASH.CFI (Generate FLASH declaration by CFI)	**New command FLASH.CFI** FLASH.CFI command is generating flash declaration automatically using CFI (Common Flash Interface) information of the flash device.

Software Release from 11-Jan-2007

Build 7451

Build 7451 FIX 02015 PPC440 Debugger	**440EPx support**

Build 7181 FIX 02013 SIM Instruction Set Simulator for TriCore	**SHA16 instruction fixed** Shift direction flag of SHA16 instruction shifted left instead of right.
Build 7181 IMP 02012 TriCore Debugger	**TriCore: PCP single stepping improved** The single stepping for the PCP subcore debugger is improved. The channel is no more disabled when a software breakpoint is hit. So no dcr error is generated any more, e.g. when single stepping certain instructions.
Build 7181 IMP 02011 Host Driver Software	**general command parameter parser** division and multiplication of single time values added
Build 7181 FIX 02010 TriCore Debugger	**TC1130 window flickering fixed** Data or peripheral windows were flickering when at least one bus error was displayed in some windows. Affected TC1130, TC1115 and TC1100.
Build 7181 CHG 02008 MIPS32/MIPS64 Debugger MIPS32 Trace SIM Instruction Set Simulator for MIPS SIM Instruction Set Simulator for MIPS5K	**accessmodes** additional access modes: ADC: AIC: ANC:
Build 7181 FIX 02007 MPC5xxx and SPC5xx Debugger Step.Hll (HLL single-stepping)	**HLL stepping may ignore BPs on MPC55xx**
Build 7181 FIX 02006 In-Circuit Emulator for the 80186XL Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Analyzer trigger unit programming** DATA event declaration: the modes B0, B1, W0 were locked depending on the CPU derivate (since build 4789)
Build 7181 IMP 02005 PowerProbe - Logic and Protocol Analyzer	**List defaults for PowerProbe changed** List shows raw data when no disassembler configured. Grouped or named channels shown first.
Build 7181 CHG 02004 MPC5xx/8xx Debugger FPU.view (Display FPU registers)	**FPU support for PPC440EP** preliminary 'diag 3201 138 <RAM addr> command is replaced by a official TRACE32 command 'SYStem.Option.SCRATCH <RAM_addr>'
Build 7181 FIX 02003 MPC5xx/8xx Debugger FPU.view (Display FPU registers)	**FPU support for MPC5xx** FPU read/write wasn't working correctly
Build 7181 IMP 02002 Script Language PRACTICE PRIVATE (Create private PRACTICE macro)	**New PRACTICE command PRIVATE** Creates only local accessible macro.

Build 7181 CHG 02001 PowerIntegrator - Logic and Bus Analyzer Integrator.Program (Program trigger unit) Integrator.ReProgram (Program trigger unit)	**complex trigger unit** SELECTOR definition: mode RAISING renamed to RISING
Build 7181 ERR 02000 S12 Debugger SIM Instruction Set Simulator for HC12/MCS12	**Star12X Simulator: tstx and tsty swapped** The instructions tstx and tsty were simulated swapped.
Build 7181 CHG 01999 TriCore Debugger SYStem.Option (Special setup)	**TriCore SYStem.Option change** After a transition period, the following TriCore SYStem.Options will become obsolete: SYStem.Option TC1130FIX SYStem.Option TC1796FIX SYStem.Option TC1900FIX These options were used for disabling the internal watchdog on SYStem.Mode [Up|Go|Attach] as a bugfix. As replacement was introduced: SYStem.Option WATCHDOGFIX The functionality will exactly be the same, except that it has only effect on TriCore Chips that have the corresponding bug. For disabling the watchdog (e.g. for flash programming) please use Data.Set 0xF0000024 %Long 0x00000008 after the SYStem.Up command. See the demo scripts for an example.
Build 7181 IMP 01997 Host Driver Software	**ethernet interface** hostbased software changed to increase the tolerance concerning wrong package order (avoiding sw hang)
Build 7181 IMP 01995 Host Driver Software ICD In-Circuit Debugger	**ethernet interface** new firmware version V6.9 to avoid port remapping problems in NAT configuration
Build 7181 IMP 01994 PPC400 Debugger SYStem.CPU (Select CPU)	**support XILINX designs with JTAGPPC ctrl** support for PPC405 core(s) in chain with FPGA (JTAGPPC controller) Please also see application note in FAQs.
Build 7181 ODD 01993 ARM11 JTAG Debugger	**MMU.SCANALL on ARM11 modified common** Common range is set by accident to 0xa0000000--0xffffffff.
Build 7181 IMP 01992 SNOOPer.Mode (Set operation mode of SNOOPer trace)	**New modes for SNOOPer** StopAndGo mode allows all snooping modes to stop the core temporarily.

Build 7117

Build 7117 ERR 01975 TriCore Debugger	**TriCore Data Cache** For TriCore devices with Data Cache (DCACHE in DMI, e.g. TC10GP, TC11IB, TC1100, TC1115 and TC1130) it is not possible writing data back to memory. Since the debugger can not access the data cache, Data.dump windows with cached memory will contain outdated (wrong) data. Workaround: If possible, disable data cache or link to uncached locations. Note: This does not affect the data cached by the EBU.

Build 5151

Build 5151 ERR 01996 FLASH Programming (Memory-Mapped) FLASH.Create (Declare FLASH)	**Flash family codes AM29x256 not working** Flash programming with family codes AM29LV256, AM29LV256B, AM29M256, AM29M256B, AM29N256 and AM29M2562 is failing after a few words (since 09 March 2006).

Build 4943

Build 4943 IMP 01991 Data.IMAGE (Display image data)	**JPEG format supported by Data.IMAGE**
Build 4943 FIX 01990 C5500 Debugger Break.Set (Set breakpoints) Var.Break.Set (Set breakpoint to HLL expression)	**C55x data breakpoint problems fixed** Breakpoints with data value and long access breakpoints failed sometimes.

Build 4928

Build 4928 IMP 01989 SNOOPer.Mode (Set operation mode of SNOOPer trace)	**SNOOPER can trace MMU spaces** New mode PC+MMU will trace PC and memory space.
Build 4928 IMP 01988 Data.LOAD.Elf (Load ELF file)	**DWARF2 sourcefile/module relation** Workarround for modules with wrong name (includefile instead of main source file) implemented.
Build 4928 IMP 01987 PERF.MMUSPACES (Include space IDs for addresses in the sampling)	**PERF command can consider memory spaces** When enabled, the PERF command will distinguish between different memory spaces.
Build 4928 IMP 01986 <trace>.Chart.DatasYmbol (Analyze pointer contents graphically) <trace>.STATistic.DatasYmbol (Analyze pointer contents numerically)	**New commands for symbolic data statistic** Trace.STAT.DatasYmbol provides symbolic statistics for data similar like Trace.STAT.sYmbol does for addresses.

Build 4790 IMP 01985 <trace>.STATistic.Address-DIStance (Time interval for single program event) <trace>.STATistic.Address-DURation (Time between two program events)	**New commands for address statistic**
Build 4790 IMP 01984 <trace>.STATistic.Address (Time between up to 8 program events)	**New command for address hit statistic**
Build 4790 FIX 01983 Teak/TeakLite Debugger	**Stack backtrace on TEAK not working**
Build 4790 IMP 01982 ARM/Cortex Trace (parallel)	**FCSE (WinCE) supported for ARM trace**
Build 4790 FIX 01981	**Var.Frame with GNU compiler may fail** Fails for functions with stmdb r13,{...,r15}.
Build 4790 FIX 01978 SIM.LOAD (Load simulator module)	**Simulator port callbacks may fail** Callbacks for port changes may be called with wrong value value (=0) for certain 32bit ports.

Build 4728 FIX 01980 PowerIntegrator - Logic and Bus Analyzer Integrator.Program (Program trigger unit) Integrator.ReProgram (Program trigger unit) Integrator.TSYNC (Select trigger line and mode)	**transient recognition changed** for simple and complex trigger unit the transient recognition for the dataselectors was changed and some bugs fixed
Build 4728 CHG 01979 TriCore Debugger TriCore Parallel Trace SIM Instruction Set Simulator for TriCore	**accessmodes** additional access modes: PCD: PCP: EPCD: EPCP: deleted accessmode: PCP:
Build 4728 FIX 01977 SIM Instruction Set Simulator for TriCore	**Various bugs in jump instructions fixed.** Some of them failed under certain conditions.
Build 4728 FIX 01976 PER.view (Display peripherals) PER.Program (Interactive programming) PER.ReProgram (Load default program)	**per programming** saveout command did not restore original value
Build 4728 IMP 01974 ARM Debugger TRACE32 Instruction Set Simulator for CORTEX / ARM / XSCALE	**New command to fix literal attibutes** The command sYmbol.CLEANUP.CodeLiterals fixes literal attributes that really contain code (instructions synthesized by a define word).
Build 4728 IMP 01973 CTS (Context tracking system (CTS))	**CTS can start with known memory state** CTS.CAPTURE can capture the contents of memory, caches and TLBs before the application is started. This allows CTS to start with known values.
Build 4728 IMP 01972 Var.TREE (Display variables in the form of a tree structure)	**New command to show variables in tree**
Build 4728 IMP 01971 Var.PATtern (Display variables allowing wildcards for symbol name and type)	**New command to show variables by name**
Build 4728 IMP 01970 Break.Set (Set breakpoints)	**Onchip breakpoint with data not match** On some cores breakpoints on data mismatch are supported (not operator).
Build 4728 IMP 01969 Data.LOAD.Elf (Load ELF file)	**Support for DWARF2/3 callframe info** Adding the /FRAME option will load DWARF2/DWARF3 call frame information. The Var.Frame window will use this information whenever possible. The implementation is is target architecture specific.
Build 4728 IMP 01968	**New symbian file format loader** Can load low level symbol information from symbian kernel build tool.

Build 4728 IMP 01967 Data.DRAW (Graphical memory display of arrays) Var.DRAW (Graphical variable display)	**Data.DRAW and Var.DRAW extended** Can now display up to six channels.
Build 4728 IMP 01966 sYmbol.Modify.TYPE (Modify type of symbols) sYmbol.NEW.Var (Create user-defined variable)	**New variable create and modify commands** Can create new (target) variables or modify the HLL type of an existing variable or label.
Build 4728 IMP 01965 PER.Program (Interactive programming)	**New capabilites in PER window**
Build 4728 IMP 01964 TERM.view (Terminal display)	**VT100 emulation extended**
Build 4728 IMP 01963 ARM Debugger TRACE32 Instruction Set Simulator for CORTEX / ARM / XSCALE	**Unified Assembly Language supported** SETUP.DIS command extended to choose between traditional disassembly and unified assembly language mode.
Build 4728 FIX 01962 PowerTrace Analyzer.List (List trace contents)	**sporadic ghost records fixed** Under certain circumstances the trace contained "ghost" records (records which actually were never recorded by the hardware).
Build 4728 FIX 01960 TriCore Debugger TrBus.Out (Define source for the external trigger pulse)	**Trigger on target break** Added option to generate a PODBUS trigger pulse when target breaks.
Build 4728 FIX 01959 SIM Instruction Set Simulator for HC12/MCS12	**EMULS instruction wrong results** EMUL instruction behaved like EMUL (no signed multiplys).
Build 4728 CHG 01958 NIOS II Debugger SYStem.Option (Special setup)	**New sytem options BTM,DTM,SYNC** Control behavior of off-chip and on-chip trace via system.option.
Build 4728 CHG 01957 NIOS II Debugger SYStem.Option (Special setup)	**New system option CFGCLK** set clock frequency for FPGA configuration commandos.
Build 4728 CHG 01956 NIOS II Debugger SYStem.Option (Special setup)	**New system option FSS** Support for different trace clock timings.
Build 4728 IMP 01955 NIOS II Debugger	**Trace analysis improved** Better handling of gaps in trace
Build 4728 IMP 01954 NIOS II Debugger	**new command JTAG.LOADRBF**

Build 4728 CHG 01953 S08 Debugger FIRE Emulator for Freescale HC08 / HCS08 In-Circuit Emulator for 68HC05 and 68HC08	**accessmodes** additional access modes: DBG: EDBG: INI:
Build 4728 FIX 01948 MPC56x NEXUS Debugger and Trace	**runtime counter overflow fixed**
Build 4728 FIX 01946 PPC440 Debugger Register.view (Display registers) Register.Set (Modify register contents)	**MCSRR0/1 register write with r.s** MCSRR0/1 register write from the register window (r.s mcsrr0 xxxx) do not take any effect.
Build 4728 IMP 01945 ARM11 JTAG Debugger Break.Set (Set breakpoints)	**ContextID supported by breakpoints+ETM** When TrOnchip.ContextID is enabled task-related breakpoints will use the ContextID comparators of breakpoint unit and ETM.
Build 4728 IMP 01944 Break.Set (Set breakpoints)	**ETM support BusTrigger/BusCount breakp** BusTrigger activates EXTOUT0, BusCount counts the events with ETM counters.
Build 4728 FIX 01942 SIM Instruction Set Simulator for HC12/MCS12	**Flags on load/store operations set wrong** Flag register was set wrong on some load/ store operations (e.g. STD instruction).
Build 4728 FIX 01941 C5500 Debugger Register.Set (Modify register contents)	**Register modification failed** Modification of CFCT, RETA and XARn was not working.
Build 4728 IMP 01939 sYmbol.Browse.Type (Browse HLL types)	**New commands for type/class browsing** sYmbol.Browse.Type browses for any types. sYmbol.Browse.ENUM browses for enumeration types. sYmbol.Browse.Struct browses for structs, unions and classes. sYmbol.Browse.Class browses for C++ classes.
Build 4728 FIX 01937 SIM Instruction Set Simulator for HC12/MCS12	**cpex instruction incorrect zero flag set** Zero Flag was alwas set after execution. Should only be set when set before.
Build 4728 FIX 01936 Host Driver Software OS.screen (Call up the shell or execute host command)	**command OS crashes under circumstances** The command OS.screen crashed if the length of <commandline> exceeded 252 characters.
Build 4728 FIX 01934 Data.Find (Search in memory)	**Data.Find with word or long data fails** May skip areas or search the data wrong aligned. Bytewide search operations are not affected.
Build 4728 FIX 01933 SIM Instruction Set Simulator for PowerPC	**PowerPC simulator DEC register problems** Manual set of DEC register fails. DEC register shown incorrect in register window.

Build 4728 FIX 01932 High-Level-Language Debugging Var.View (Display variables)	**Internal functions in HLL parser fail** When used in windows (e.g. Var.View) the functions will cause strange effect on the command line.
Build 4728 FIX 01931 Data.IMAGE (Display image data)	**Data.IMAGE with large /STRIDE fails** Values larger than 0x1000 cause a crash.
Build 4728 FIX 01929 FIRE Emulator for C166 Family FIRE Emulator for C166 Cell-Based-Core FIRE Emulator for H8S and H8/300H FIRE Emulator for SH2 FIRE Emulator for NEC V850 FIRE Emulator for C166S V2 Family	**RESET command crashes**
Build 4728 FIX 01928 Data.SAVE.<format> (Save data in file with specified format)	**D.SAVE.ASCII... using wrong format**
Build 4728 IMP 01927 PPC440 Debugger	**per440ep: EBC mod., USB2.0 added**
Build 4728 FIX 01926 MPC5xx/8xx Debugger	**runtime counter overflow fixed**
Build 4728 FIX 01925 ARM Debugger Data.LOAD.COFF (Load COFF file)	**Function info missing for ARM TI-COFF** Only symbol name and type information is there.
Build 4728 FIX 01924 MPC5xxx and SPC5xx Debugger Data.LOAD.Elf (Load ELF file)	**PowerPC e500 floating point returns** Floating point function return values are not displayed.
Build 4728 FIX 01923 ARM Debugger Data.LOAD.COFF (Load COFF file)	**TI ARM compiler ARM/Thumb attributes** Attributes may be wrong after literal constants in code.
Build 4728 FIX 01922 GROUP.HIDE (Hide group from debugging) <trace>.STATistic.Func (Nesting function runtime analysis)	**Function statistics with disabled groups** Calls inside the group have caused wrong internal/external times and calling counts.
Build 4728 FIX 01921 <trace>.STATistic.Func (Nesting function runtime analysis) <trace>.STATistic.TREE (Tree display of nesting function run-time analysis)	**Function statistics fails with opt code** When a function jumps (not a call) to a function containing only a return instruction the analysis will fail and the nesting incorrect.
Build 4728 FIX 01920 S12 Debugger Data.LOAD.Ubrof (Load UBROF file)	**UBROF10 register variables wrong** Register variables may be shown wrong when UBROF10 format is used.

Build 4728 IMP 01919 ARM/Cortex Trace (parallel)	**Smarttrace for ARM improved** Can now also generate interrupts in gaps. Several new algorithms added.
Build 4728 CHG 01904 MPC5200 Debugger PowerQUICC II/Pro Debugger SYStem.Option (Special setup)	**new SYStem.Option.IP.BOTH** new MSR[IP] handling for Onchip-BP's on processor with two Onchip-BP's. (G2_LE /ZEPPO core) Default is SYS.O.IP.AUTO where only the Onchip-BP control the active (MSR[IP]) exception handler block.

Build 2940

Build 2940 IMP 01918 ICD Solutions for ARM	**TLB DUMP functions implemented for ARM11** Allows to dump TLB contents during debug.
Build 2940 IMP 01917 CTS.CAPTURE (Copy real memory to the virtual memory for CTS)	**New command CTS.CAPTURE** Allows to capture initial cache and memory contents for a CTS trace when UseVM is enabled.
Build 2940 IMP 01916 Register.RELOAD (Reload the compiler register settings)	**New command Register.RELOAD** Reloads registers as done with last Data.LOAD command.
Build 2940 CHG 01915 ARM Debugger	**access modes** additional access modes: ATE: EATE: ETE: TE: access modes deleted: ASJ: AUJ: EASJ: EAUJ: ESJ: EUJ: SJ: UJ:
Build 2940 CHG 01914 ARM Debugger	**access modes** additional access modes: DAP: EDAP:
Build 2940 FIX 01912 SIM Instruction Set Simulator for PowerPC	**stswi and lswi instructions wrong result** stswi and lswi instructions delivered wrong result when written to an address not modulo 4 aligned.
Build 2940 IMP 01911 JTAG Debugger FIRE Fully Integrated RISC Emulator ICE In-Circuit Emulator PowerTrace for NEXUS TRACE32 Instruction Set Simulators	**Practice functions** synonym D.SUM() for DATA.SUM() added
Build 2940 CHG 01910 Host Driver Software AREA.CLEAR (Clear area)	**area.clear behavior changed** without a given area name the default area 'A000' is cleared now
Build 2940 IMP 01908 Data.LOAD.S3record (Load S3-Record file)	**Data.LOAD.S3record updates initial PC** The option /NOREG can be used to suppress this.
Build 2940 IMP 01907 TPU.Register.Set (Register modification)	**TPU debugger allows FLx setting**
Build 2940 IMP 01906 sYmbol.Browse (Browse symbols)	**Symbol browser uses GROUP hide info** Symbols belonging to hidden groups are not displayed.

Build 2940 IMP 01902 <trace>.Chart (Display trace contents graphically) <trace>.DRAW (Plot trace data against time) <trace>.Timing (Waveform of trace buffer)	**New option /ZoomTrack for timing/chart** Allows to track position and zoom factor between timing and chart windows.
Build 2940 FIX 01900 PODBUS Ethernet Controller	**DHCP problem** boothang of podbus ethernet controllers (old revisions only) solved
Build 2940 FIX 01899 PPC440 Debugger	**new bus error handling for PPC440EP** support for crossbar Arbiter on PPC440EP-Pass2 instead of Core PLB4 Arbiter from PPC440EP-Pass1
Build 2940 FIX 01898 <trace>.SIZE (Define buffer size)	**Tracebuffer too large for RISCTRACE** Was causing illegal trace entry when tracebuffer is full.
Build 2940 FIX 01897 CPU32 Debugger	**CpuAccess Dualport fails** Will crash host software on all passive debuggers (eg. 68k,HC16).
Build 2940 CHG 01896 Data.LOAD.Elf (Load ELF file)	**DWARF2 loader takes care of /Include** The /Include option is now required to show lines from include files.
Build 2940 IMP 01895 SNOOPer.Mode (Set operation mode of SNOOPer trace)	**New SNOOP.Mode PC for all cores** Allows to make statistics or time charts based on PC snoops.
Build 2940 FIX 01894	**TRACE32 API T32_ResetCpu not working** Performs only system down on TRACE32 ICD.
Build 2940 FIX 01893	**Stackbacktrace not correct on Tricore** Registers D8..D11 and A12..A15 where swapped.
Build 2940 IMP 01892 <trace>.STATistic.Func (Nesting function runtime analysis) <trace>.STATistic.TREE (Tree display of nesting function run-time analysis)	**New "Own" items in Trace.STAT commands** "Own" time is the time of function with all called functions, but without the time spend in interrupts or other tasks.
Build 2940 CHG 01891 TrBus.Out (Define source for the external trigger pulse)	**TrBus.Out now off by default** Was enabled by default till now.
Build 2940 IMP 01889 TriCore Debugger RunTime.state (Display results)	**RunTime measurement** Runtime Measurement is more accurate now.
Build 2940 FIX 01885 PPC440 Debugger	**PPC440 not working**

Build 2940 FIX 01883 PowerIntegrator - Logic and Bus Analyzer Integrator.ABCDEF (Sampling configuration for probes ABCDEF) Integrator.JKLMNO (Sampling configuration for probes JKLMNO)	**Transient detection - sample control** transient detection for sample control was working partly wrong under circumstances affected modes: State, StatePLL, StatePLLBoth, DDR
Build 2940 FIX 01881 ARM/Cortex Trace (parallel)	**ETM V3 possible flow errors with BLX** Flow error when BLX <Rn> call does not change operation mode.
Build 2940 FIX 01879 MMU (Memory management unit)	**MMU command structure redesigned** Structure is now more logical and clean. Common command conversions: MMU.DUMP -> MMU.PageTable.dump MMU.SCAN -> MMU.PageTable.SCAN MMU.TLB -> MMU.TLB.dump The old commands are still usable.
Build 2940 FIX 01877 Embedded Trace Buffer Real-Time Trace ARM/Cortex Trace (parallel)	**ETMV3 cycle accurate trace timestamp** Timestamp wrong with selective tracing.
Build 2940 FIX 01876 MPC55xx/MPC56xx NEXUS Debugger and Trace <trace>.List (List trace contents)	**eTPU trace may produce flow errors** Occurs when long traces from other cores are also in the trace or at certain slot instruction combinations during the end of a channel program.
Build 2940 FIX 01875 Data.LOAD.COFF (Load COFF file)	**COFF loader may cause unaligned loads** If the code in the file is not aligned to 32bits then the load blocks may also be not aligned. This degrades download performance and can cause wrong downloads on some architectures (e.g. C6000).
Build 2940 FIX 01874 Data.LOAD.Ubrof (Load UBROF file)	**UBROF loader fails on untagged member** The UBROF file format loader may fail when the file contains untagged structure members.
Build 2940 FIX 01873	**Star12X not working with AUTO cpu select** Star12X core must be selected prior to system.up to have Star12X features working correctly (onchip trace, thread command).

Software Release from 12-Feb-2005

Build 1270

Build 1270 IMP 01872 Data.DRAW (Graphical memory display of arrays) <trace>.DRAW (Plot trace data against time)	**Floating point formats allowed** Floating point formats now allowed for low level graphical displays.

Build 1268

Build 1268 IMP 01871 Adaptation for OCDS Level 2 TriCore Break.Set (Set breakpoints)	**onchip breakpoints mixable** Mixing onchip breakpoints with trace action and other onchip breakpoints was not allowed. When setting them anyway, they did not work. Mixing them is now allowed and works.
Build 1268 IMP 01869 Super10 NEXUS Debugger and Trace Break.Set (Set breakpoints)	**Onchip breakpoints improved for Super10** NOT bit for improved filtering and exclude breakpoints (R304 cell). Data only breakpoints for stack variables (R305 cell).

Build 1264 FIX 01863 PPC400 Debugger PER.view (Display peripherals)	**TBL/TBH address in perfile corrected** Correct TBL/TBH read/write address implemented in per405xx.per files
Build 1264 IMP 01862 Data.LOAD (Load file)	**New ALIGN option for flash downloads** Can align memory operations to blocks of up to 1K size. Useful for FLASH programming.
Build 1264 IMP 01861 ICE In-Circuit Emulator Go.BackEntry (Go back in program to function entry (CTS)) Step.BackOver (Step back)	**new commands for CTS introduced** Go.BackEntry Go.BackTillViolation Go.TillViolation Step.BackOver
Build 1264 CHG 01860 ICE In-Circuit Emulator Step.BackChange (Step back till expression changes)	**commandname changed** longname changed from Step.BackChang to Step.BackChange
Build 1264 IMP 01859 MAP.InitVar (CTS initial variable mapping) MAP.NoInitVar (CTS initial variable mapping)	**New mapping command for CTS** Defines already initialized variables for CTS violation checking features. The mapped areas are considered as having been read already.
Build 1264 IMP 01858	**New violation checking features for CTS** Go.TillViolation searches for access violations by the program. It will find Read-Before-Write accesses, writes to "const" mapped memory, writes to location zero and accesses to already released stack frames.
Build 1264 IMP 01857 Go.BackEntry (Go back in program to function entry (CTS)) Step.BackOver (Step back)	**New CTS backward execution features** Step-Back-Over HLL or assembler and Go-Back-Till_Entry supported.
Build 1264 IMP 01856	**Trace based data coverage** The trace based coverage analysis also supports data coverage (when datatrace is available).

Build 1258

Build 1258 IMP 01853 SETUP.ASCIITEXT (Configure ASCII text display) Data.dump (Memory dump) Var.View (Display variables)	**UTF-8 support for ASCII strings/dumps** UTF-8 characters is enabled with SETUP.ASCIITEXT UTF-8.
Build 1258 FIX 01852 ICD Solutions for ARM TRACE32 Instruction Set Simulator for CORTEX / ARM / XSCALE TERM.GATE (Terminal with virtual hosting)	**ARM semihosting file append not working** Append to existing files (e.g. fopen with "a+" mode) was not working as expected.

Build 1256

Build 1256 IMP 01850 TriCore Debugger PER.view (Display peripherals)	**perfile for TC1796 revised** various errors corrected, support for P-Bit protected registers added, improved descriptions and layout

Build 1253

Build 1253 FIX 01880 PowerIntegrator - Logic and Bus Analyzer Integrator.Program (Program trigger unit) Integrator.ReProgram (Program trigger unit)	**analyzer complex trigger unit** hostbased software may crash under certain circumstances
Build 1253 IMP 01849 CTS.CACHE (CTS cache analysis)	**CTS Cache analysis supports locking** Trace of cache locking is supported for ARM920 and ARM922 cores.
Build 1253 FIX 01847 Break.Set (Set breakpoints) Var.Break.Set (Set breakpoint to HLL expression)	**Onchip program exclude breakpoint failed** Breakpoint refused when no ETM breakpoints usable and range does not fit into single address mask.
Build 1253 IMP 01846 FLASH.TARGET (Define target controlled algorithm)	**target based flash programming for C55x** Support for target based flash programming for C55x family.
Build 1253 FIX 01845 Data.LOAD.Ubrof (Load UBROF file)	**bug fixed UBROF V9 loader for V850** Loader failed.
Build 1253 IMP 01844 <trace>.LOAD (Load trace file for offline processing)	**Probe.LOAD can also recover setti** The option /Config recovers also the configuration of the trace and the channel names.
Build 1253 IMP 01843 <trace>.SAVE (Save trace for postprocessing in TRACE32)	**Probe.SAVE I.SAVE save configuration too** This includes the name settings and simple trigger settings.
Build 1253 IMP 01842 Break.Init (Initialize breakpoints)	**Break.Init reinitializes also onchip bps** The comand can be used to restore overwritten onchip trigger resources.
Build 1253 FIX 01841 DSP56800 and DSP56800E Debugger	**SYStem.Up with SYS.Option.DE off fixed**
Build 1253 IMP 01840 MPC5xx/8xx Debugger	**SEC Lite Module 885/875 implemented** per885.per and per875.per will also support the "SEC Lite" module register

Build 1248

Build 1248 FIX 01837 SIM Instruction Set Simulator for PowerPC	**isel implemented** simulator stopped when reaching isel instruction

Build 1246 FIX 01836 Host Driver Software	**PowerDebug USB2 on USB1 transfer errors** Sporadic transfer errors appear on Windows XP with USB1 interface and USB2 debug module.
Build 1246 FIX 01835 MAC71xx/72xx NEXUS Debugger and Trace <trace>.List (List trace contents)	**Traced flow incorrect on MAC7xxx** One instruction too much displayed before interrupt entry.
Build 1246 CHG 01834 ICE In-Circuit Emulator PowerTrace for NEXUS Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**programming of complex triggerunit** commands renamed: Flag.OFF -> Flag.FALSE Flag.ON -> Flag.TRUE
Build 1246 FIX 01833 SIM Instruction Set Simulator for PowerPC	**divw special case handled** When dividing 0x80000000 / -1, on Windows (and probably other systems) an exception occured (Integer Overflow) or TRACE32 quitted without an error message. This is now fixed. Note: Due to PPC spec, divw returns an undefined value when doing this operation or a division by 0.
Build 1246 CHG 01832 PowerTrace for NEXUS Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**programming of complex triggerunit** keyword for addressevent definition ADDR replaced by ADDRESS
Build 1246 CHG 01831 ICE In-Circuit Emulator Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**programming of analyzer triggerunit** keyword for addressevent definition ADDR replaced by ADDRESS
Build 1246 CHG 01830 FIRE Fully Integrated RISC Emulator Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**programming of analyzer triggerunit** keyword for addressevent definition ADDR replaced by ADDRESS
Build 1246 FIX 01826 CTS.UseMemory (Use memory contents for CTS) CTS.UseRegister (Use the CPU registers for CTS)	**CTS may ignore the current context** This may occur during partial processing.
Build 1246 FIX 01825 SIM Instruction Set Simulator for PowerPC	**mulhwu now takes care of signs** internal handling of signs corrected

Build 1246 FIX 01824 SIM Instruction Set Simulator for PowerPC	**mulhw takes care of signs** internal handling of sign corrected
Build 1246 FIX 01823 SIM Instruction Set Simulator for PowerPC	**mtpmr command now executed** The Simulator now does not stop any more when reaching the mtpmr command; command can also be stepped now
Build 1246 FIX 01822 SIM Instruction Set Simulator for PowerPC	**efsctuiz command is now executed** The simulatur stopped when the assembler command efsctuiz was reached in running mode or single step mode.
Build 1246 FIX 01821 SIM Instruction Set Simulator for SuperH	**SH simulator immediate not sign extended**
Build 1246 FIX 01820 TRACE32 Instruction Set Simulator for CORTEX / ARM / XSCALE	**ARM Simulator LDR pc,xx did no switch** LDR pc,xx did no ARM/Thumb mode switch on ARM architecture V5e and above.
Build 1246 FIX 01819 PowerQUICC II/Pro Debugger SYStem.Option (Special setup)	**SYStem.Option.IP AUTO do not work at all** SYStem.Option.IP.AUTO may fail in conjunction with SYStem.Option.BASE.AUTO
Build 1246 FIX 01818 SIM Instruction Set Simulator for PowerPC	**sraw[.] right shift with 0 corrected** shifting 0x80000000 >> 0 resulted in 0xFFFFFFFF, expected value is 0x80000000
Build 1246 IMP 01816 PPC600/750 Debugger	**Additional 740/750 derivative supported** 750A2/740A2 Support for PVR==0x30082202 and 0x10082202 (740P/750P:Conan/Doyle)
Build 1246 FIX 01815 Data.SAVE.<format> (Save data in file with specified format)	**Data.SAVE not working on Linux host**
Build 1246 FIX 01814 PowerIntegrator - Logic and Bus Analyzer Integrator.ABCDEF (Sampling configuration for probes ABCDEF) Integrator.JKLMNO (Sampling configuration for probes JKLMNO)	**Transient detection - sample control** transient detection for sample control was working partly wrong under circumstances affected modes: State, StatePLL, StatePLLBoth, DDR
Build 1246 CHG 01813 JTAG Debugger FIRE Fully Integrated RISC Emulator ICE In-Circuit Emulator ROM Monitor	**Address range expressions** Behavior of general parameter parser changed The checking of accessmode equality inside address range expressions like address1--address2 is switched off now. Now it's allowed to use different accessmodes for begin and end address of address ranges. The accessmode of the end address will be ignored and will not cause an error message anymore. e.g. UP:1000--SD:2000 will be handled as UP:1000--UP:2000

Build 1246 CHG 01812 ARM/Cortex Trace (parallel) Analyzer.Mode (Set the trace operation mode)	**Analyzer.Mode Prestore for ARM ETM** Selective data traces, setup with /TraceEnable breakpoints, will now display only the data cycles without code information. Enabling "A.Mode Prestore" will display (and trace) also the program information.
Build 1246 FIX 01806 MPC5xx/8xx Debugger	**Debugging in compressed mode may fail** The debugger may access wrong locations for stackframe analysis.

Build 1241 IMP 01811 ARM11 JTAG Debugger ARM7 JTAG Debugger ARM9 JTAG Debugger SYStem.JtagClock (Define JTAG frequency)	**New JTAG clock modes CTCK and CRTCK** New JTAG clock modes CTCK and CRTCK (besides RTCK and ARTCK) and improved download speed.
Build 1241 IMP 01810	**Data.LOAD.EXE can now also load PDB file** This requires no more .DLL or .EXE file to load symbols from .PDB files.
Build 1241 IMP 01809	**Type packing for ARM ELF/DWARF impl.** Type packing is now also possible for ELF/DWARF files with global types (like ARM ADS/RVCT).
Build 1241 FIX 01808 HELP (Online help)	**PDF online help not working on Linux** Causes always "file not found" error message in acrobat reader.
Build 1241 IMP 01807 sYmbol.CREATE.ATTRibute (Create user-defined attribute) sYmbol.NEW.ATTRibute (Create user-defined memory attribute)	**new command sYmbol.NEW.ATTRibute** Creates new attribute entries. Useful when ARM/Thumb awareness is needed and no debug information for it is present.
Build 1241 FIX 01804 ColdFire BDM Debugger	**Onchip breakpoints not working** Onchip breakpoints are not working on passive debug modules.
Build 1241 FIX 01803 PPC440 Debugger	**PPC440GX CPU (re)support** Bug fix for PPC440GX. Software change has had disabled the PPC440GX derivative.
Build 1241 FIX 01802 MPC5xx/8xx Debugger SYStem.Mode (Select mode)	**Sys.M.NoDebug reliable deact. DebugMode** SYS.Mode.NoDebug will not work. Even though if sys.m.nodebug is selected and an external reset will occur. The reason is a permanent running DSCK clock signal.
Build 1241 IMP 01801 TERM.view (Terminal display)	**TERM window accepts ^C, ^X and DEL**
Build 1241 FIX 01800 PERF.Program (Write a performance analyzer program) PERF.ReProgram (Load an existing performance analyzer program)	**PERF.Program/ReProgram may crash**
Build 1241 CHG 01799 <trace>.TestFocus (Test trace port recording)	**SYSTEM.TEST command replaced** The SYStem.TEST command of Super10 nexus has been replaced by the (generic) Analyzer.TestFocus command.
Build 1241 FIX 01798 StarCore Debugger	**SDRAM r/w problem on EB1000 board fixed**
Build 1241 IMP 01796 Data.LOAD.Elf (Load ELF file)	**Loading large ELF/DWARF files improved** Large files require less memory during download.

Build 1241 FIX 01795 XScale Debugger SYStem.Option (Special setup)	**System.Option DynVector does not work** Non working system option may cause problems for operating system awareness. Unintended interrupt may occur because modified vector table is not read during runtime.
Build 1241 FIX 01794 FIRE Fully Integrated RISC Emulator Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Programming of complex triggerunit** hardware and onchip breakpoints configured wrong under circumstances when breakpoints were defined inside a analyzer trigger program via ADDR declaration
Build 1241 FIX 01793 <trace>.Find (Find specified entry in trace) <trace>.FindAll (Find all specified entries in trace)	**Trace.Find for segmented addresses fails** Searching for segmented addresses in trace (e.g. 80x86) does not find matches.
Build 1241 FIX 01792	**IA64 demangler may fail** Demangler may fail on functions with "restrict", "const", or "volatile" qualifiers.
Build 1241 FIX 01791 TriCore Debugger Data.EPILOG (Automatic data modification on program execution halt)	**Data.Epilog did not execute** Data.EPILOG was not executed when breaking
Build 1241 IMP 01790 Data.COPY (Copy memory)	**New options for Data.COPY command** TByte, TWord, Verify and ComPare
Build 1241 IMP 01789	**New WSGROUP command for Per.Program** Defines "shadowed" write only group. Write only I/O port with shadow in RAM memory.
Build 1241 FIX 01788	**Complex breakpoints MPC5200 failed** Breakpoints with conditions or commands did not work.
Build 1241 IMP 01787	**new API added to readout traces** Supports also reading the PowerIntegrator and PowerProbe.
Build 1241 IMP 01786 Adaption for Embedded Trace Macrocell Adaptation for MIPS MPC55xx/MPC56xx NEXUS Debugger and Trace <trace>.AutoFocus (Calibrate AUTOFOCUS preprocessor) <trace>.TestFocus (Test trace port recording)	**New commands to verify trace connection** The Analyzer.AutoTest command verifys the integrity of the trace connection. For adapters with adjustable threshold the command Analyzer.AutoFocus will also choose the threshold in the middle of the data eye.
Build 1241 FIX 01785	**MPC55xx Nexus data trace may be wrong** When tracing 64bit transfers the upper 4 bits are wrong when the data results from data trace message with sync.
Build 1241 FIX 01784	**DWARF2 location descriptions may fail** Some DWARF2 location descriptions for "reference" variables may fail to load correctly and produce a load warning.

Build 1241 FIX 01782 PPC400 Debugger PPC440 Debugger	**1st ICD start get fatal error (timeout)** 1st ICD start after a power cycle may get a fatal error because of a timeout.
Build 1241 FIX 01781 ICD Solutions for PowerPC400 <trace>.THreshold (Optimize threshold for trace lines)	**Threshold voltage set wrong on PPC4xx** The analyzer window settings where not set correctly. Threshold may be wrong.

Build 1228 CHG 01780 TMS320C54XX Debugger C5500 Debugger	**accessmodes** additional access modes ETB: EETB:
Build 1228 CHG 01779 MIPS32/MIPS64 Debugger	**accessmodes** additional access modes CBU: ECBU:
Build 1228 IMP 01778 ARM9 JTAG Debugger	**MOVE Coprocessor Register Access** The peripheral file and the required target routine is available in the system directory of the TRACE32 installation or on the CDROM at demo\arm\etc\coprocessor\move. Read the commends in "move.cmm".
Build 1228 FIX 01777 <trace>.EXPORT (Export trace data for processing in other applications)	**Analyzer.EXPORT may crash** Crashed when output file size was larger than 2GB.
Build 1228 IMP 01776	**File I/O changed to support 64bit files**
Build 1228 IMP 01775 Var (HLL variables and expressions)	**MAP.VOLATILE affects now var display** Memory access width rules for arrays and structures in volatile areas are now handled more strict, i.e. an array of 16bit values will be read with 16bit memory accesses instead of 32bit.
Build 1228 FIX 01774 Data.USRACCESS (Prepare USR access)	**Data.USRACCESS does not load program** The filename was accepted, but the program not loaded. The software breakpoint at the return address was also not set.
Build 1228 FIX 01773	**Var.Frame fails with MMU on SH+MIPS** Var.Frame with option /TASK and SYStem.Option MMU active uses the wrong address space on SH and MIPS.
Build 1228 FIX 01771 MIPS32/MIPS64 Debugger StarCore Debugger	**Some ICDs fail on USB2** MIPS5k and StarCore fail at system.up or single step.
Build 1228 FIX 01770	**Data.Load with verify + FLASH may crash** Data.Load with option /Verify and FLASH.AUTO programming enabled may crash.
Build 1228 FIX 01769 Var (HLL variables and expressions)	**Variable window with inherited may crash** Displaying classes with inherited members and long inherited class names ay crash.
Build 1228 FIX 01768 FIRE Emulator for H8S and H8/300H	**Fire H8S system window store fails** Processor mode may be stored wrong. Some settings missing.
Build 1228 FIX 01767 CTS.STATistic.TREE (Tree display of nesting function run-time analysis)	**Nesting button in CTS.STAT.TREE window** Button does not display CTS nesting.
Build 1228 FIX 01766 M-Core Debugger SYStem (System configuration)	**SYStem.JtagClock was missing in window** Executing the command from the commandline is possible.

Build 1228 IMP 01765 Var (HLL variables and expressions)	**untagged unions in structs supported** Access to untagged unions in HLL expressions is now possible.
Build 1228 IMP 01764 Data.LOAD.Ubrof (Load UBROF file)	**UBROF-10 debug format supported**

Build 1224

Build 1224 FIX 01854 Host Driver Software	**USB under WIN98SE** installation of TRACE32 usb driver failed under Windows 98SE
Build 1224 FIX 01763 Data.Test (Memory integrity test)	**Memory test at page boundaries** Data.Test delivered wrong results at certain memory boundaries with certain processor families.
Build 1224 FIX 01762 PPC400 Debugger	**support for new dongle on passiv modules** The new redesigned PPC4xx dongle (gray ribbon cable) will be also supported by the standard "Debug Interface" (passiv ICD, LA-7700, LA-7701).
Build 1224 IMP 01761 PowerQUICC II/Pro Debugger SYStem.Option PARITY (Generate parity on memory access)	**write parity support for PQ2 (groucho)**
Build 1224 FIX 01760 MIPS32/MIPS64 Debugger PERF (Sample-based profiling)	**Performance Analysis for BCMxx corrected** Broadcom specific DERET handling in BDM_GetPC added. Changes necessary for functionality of performance analysis.
Build 1224 FIX 01759 S12 Debugger	**Disassembler Mnemonics swapped** Mnemonics PSHA und PSHB were swapped
Build 1224 IMP 01758 ARM7 JTAG Debugger	**Flash prog. for PCD8070x implemented**
Build 1224 IMP 01757	**New GROUP.Create commands** GROUP.CreateModules creates groups for all modules that match a wildcard. GROUP.CreateSources creates groups for all sources that match a wildcard.
Build 1224 IMP 01756	**INTERCOM.execute waits for completition** The command waits now (up to 2 seconds) for command completition.
Build 1224 IMP 01755 C5500 Debugger Onchip (Trace method Onchip, recording, and analysis commands)	**ETB support implemented** ETB (Embedded Trace Buffer) is an on-chip trace memory module from ARM, but can also be used on some derivatives to store the trace data produced by the DSP. The trace data can be read out by the debugger using the JTAG interface.
Build 1224 IMP 01754 ARM11 JTAG Debugger	**DFAR, DFSR, IFAR, IFSR saved/restored** DFAR, DFSR, IFAR, IFSR registers will be saved when entering debug mode and restored when restarting the user application. This avoids that memory accesses in debug mode, which would cause an abort, will modify these registers.
Build 1224 IMP 01753 TriCore Debugger SYStem.Mode (Select mode)	**Tricore ATTACH mode implemented** Attach allows to connect to running target. New feature is available on all "Power" tools.

Build 1224 IMP 01752 Var.DRAW (Graphical variable display)	**Data.DRAW and Var.DRAW with new options** Option /Alternate can be used to interpret one-dimensional array as two channels. Options /XY and /YX display X-Y graphics.
Build 1224 IMP 01751	**ETM can be used for ranged P-breakpoints** Ranged program (execution) breakpoints on ARM7/ARM9/ARM11 are mapped to the ETM when ETM.ReadWriteBreak is set to ON and ETM is enabled.
Build 1224 FIX 01750	**Tricore non-stopping breakpoints** Non-stopping breakpoints (watch or trace control) where setting the PERSTOP bit in the event control register.
Build 1224 IMP 01749 PPC440 Debugger	**PPC44x core support**
Build 1224 IMP 01748 PPC400 Debugger	**PPC40x core and Xilinx V2P support** certain CPU cores like 405D3 will be supported as well as in the multicore environment.
Build 1224 IMP 01747 PPC400 Debugger PPC440 Debugger	**Multicore support for PPC4xx**
Build 1224 FIX 01746 PPC400 Debugger PPC440 Debugger Go.direct (Start the program execution) RunTime.state (Display results)	**Runtime counter stop randomly** Internal driver error. Wrong mask for JDSR[PSP] bit.
Build 1224 FIX 01745 HELP.Bookmark.ADD.file (Add file to bookmark list) HELP.Bookmark.ADD.Find (Add file to bookmark list) HELP.Bookmark.ADD.Index (Add file to bookmark list)	**Bugfix help.bookmark.add**
Build 1224 IMP 01744	**"Pause" key activates command line** The "Pause" key activates the command line in any context (MS Windows).

Build 1206 IMP 01743	**PowerProbe pattern generator integrated** The pattern generator of the PowerProbe can now be accessed from debug prompt under the PATTERN command.
Build 1206 CHG 01742	**PULSEProbe command renamed to PULSE** PowerProbe PULSE generator (not available when used with FIRE).
Build 1206 CHG 01741 MPC55xx/MPC56xx NEXUS Debugger and Trace Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Programming of complex triggerunit** supports Copperhead now particularly supports DTM, WTM, OTME with unitnumbers
Build 1206 CHG 01740 M-Core Debugger M-Core NEXUS Debugger and Trace	**accessmodes** additional access modes DBG: EDBG:
Build 1206 IMP 01739	**New function INTERCOM.PING()** Returns boolean TRUE when there is a response from the other side. Can be used to check the existence of debuggers for other cores of a multicore debug application.
Build 1206 IMP 01738	**Timestamp res. for some targets changed** The resolution of the timestamps for PowerTrace with ARM ETM with traceboard versions 6 and PowerNexus for Super10 is improved to 10ns.
Build 1206 IMP 01737	**Modules can consist of multiple ranges** HLL Modules can now consist of multiple address ranges.
Build 1206 IMP 01736 Data.SOFTEPILOG (Automated sequence after setting software breakp.) Data.SOFTPROLOG (Automated sequence before setting software breakp.)	**New commands to prolog software bp write** New command Data.SOFTPROLOG and Data.SOFTEPILOG to execute a memory access sequence before and after writing software breakpoints. Useful to "enable" memory writes to otherwise read-only memory (e.g. C55x PDRAM).
Build 1206 CHG 01735 MPC5xx/8xx Debugger FIRE Emulator MPC8XX In-Circuit Emulator for Power Architecture SIM Instruction Set Simulator for PowerPC	**accessmodes** additional access modes PMR: F: AF: EF: EAF: FP: AFP: EFP: EAFP: FD: AFD: EFD: EAFD: SF: ASF: ESF: EASF: UF: AUF: EUF: EAUF: V: AV: EV: EAV: VP: AVP: EVP: EAVP: VD: AVD: EVD: EAVD: SV: ASV: ESV: EASV: UV: AUV: EUV: EAUV:
Build 1206 CHG 01733 <trace>.Find (Find specified entry in trace) <trace>.FindAll (Find all specified entries in trace)	**Trace.Find Data changed** New default Data width selector "ANY" introduced (default width). Will match any data width.

Build 1206 IMP 01732	**Onchip trace for C55x of OMAP1710** Onchip trace buffer now supported.
Build 1206 FIX 01731	**Star12 simulator IDIVS was wrong** Performed unsigned instead of signed operation.
Build 1206 CHG 01730 S12 Debugger FIRE Emulator for Freescale 68HC12 / MCS12 / S12X In-Circuit Emulator for 68HC11 In-Circuit Emulator for 68HC12 SIM Instruction Set Simulator for HC12/MCS12	**accessmodes** additional access modes X: XD: XP: AX: AXD: AXP: EX: EXD: EXP: EAX: EAXD: EAXP:
Build 1206 FIX 01729	**TERM.RESet removes also all TERM windows**
Build 1206 CHG 01728 MMDSP Debugger	**access modes** deleted accessmodes: P2: Y2: EP2: EY2: AP2: AY2: EAP2: EAY2: additional access modes : DC: IC: L2: NC:
Build 1206 IMP 01727	**Access to PRACTICE functions from Var** PRACTICE functions can be accessed from Var commands. The function name must be prepended with a ´\´ and any ´.´ in the name replaced by ´_´, e.g. Var.View \r(r4)*myvariable

Build 1204

Build 1204 IMP 01848 Host Driver Software	**USB under LINUX** USB interface supported under LINUX (hostbased driver only)
Build 1204 IMP 01726 PowerIntegrator - Logic and Bus Analyzer	**programming of complex triggerunit** design of selector realization changed (more flexible, higher internal number) several supervising modes for inputchannel available
Build 1204 FIX 01725 Super10 NEXUS Debugger and Trace	**breakpoint setting** address events deleted under circumstances (/ONCHIP and starting emulation from software breakpoint only) effects analyzer triggerunit programming too
Build 1204 CHG 01724 S12 Debugger FIRE Emulator for Freescale 68HC12 / MCS12 / S12X In-Circuit Emulator for 68HC11 In-Circuit Emulator for 68HC12 SIM Instruction Set Simulator for HC12/MCS12	**access modes** additional access modes AG: EAG: GD: AGD: EGD: GP: AGP: EGP:

Build 1203 IMP 01723	**Data.IMAGE extended with RGB565 format** New formats RGB565 and RGB565LE.
Build 1203 FIX 01722 Super10 NEXUS Debugger and Trace Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**analyzer trigger programming** address events set wrong under circumstances (/HARD only)
Build 1203 FIX 01721	**SNOOPer trigger not working correctly** On PowertDebug modules the trigger stops the SNOOPer immediatelly, the trigger delay is not taken.
Build 1203 FIX 01720 In-Circuit Emulator for C166/ST10 - Out of Production Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**analyzer trigger programming** address events OD, ODX, ODL, ODLX, ODLH: programming of breakpoints partly wrong
Build 1203 IMP 01719 PowerTrace for NEXUS	**PRACTICE functions** A.COUNTER.TIME(), A.COUNTER.EVENT() implemented for NEXUS now
Build 1203 IMP 01718 sYmbol.SourcePATH.Translate (Replace part of the source path)	**New command sYmbol.SourcePATH.Translate** The command translates parts of filenames or directories from object files. This is useful when directory names change names or are mangled by MSDOS.

Build 1198 FIX 01717 SIM Instruction Set Simulator for PowerPC	**PPC Simulator LWSI/LWSX may fail** LWSI/LWSX give wrong results when bytecount is not aligned to 4.
Build 1198 IMP 01716 PER.Program (Interactive programming)	**New definition HGROUP for PER.Program** HGROUP define a group to be used with HIDE items. No memory access is done during window update.
Build 1198 FIX 01715	**Coldfire software breakpoints in flash** BUG fixed in handling of Coldfire software breakpoints in flash. Onchip breakpoints where not set correctly. PC-relative instructions may executed wrong. Onchip breakpoint was not used for temporary breakpoints.
Build 1198 CHG 01714 NIOS II Debugger SIM Instruction Set Simulator for NIOS II	**access modes** additional access modes DC: IC: NC: EDC: EIC: ENC:
Build 1198 CHG 01713 S12 Debugger FIRE Emulator for Freescale 68HC12 / MCS12 / S12X In-Circuit Emulator for 68HC11 In-Circuit Emulator for 68HC12 SIM Instruction Set Simulator for HC12/MCS12	**access modes** additional access modes G: EG:
Build 1198 CHG 01712 eTPU Debugger	**access modes** additional access modes H: EH:
Build 1198 IMP 01711	**ELF/DWARF loader unknown modules now ASM** Modules with unknown language ID in ELF/DWARF are now considered as assembler modules (relevant for source color coding).
Build 1198 IMP 01710 PRinTer.EXPORT (Export formatted printer output to file)	**new command PRT.EXPORT** Same as PRT.FILE command, but outputs data in CSV (Comma Separated Value) format.
Build 1198 IMP 01709 PRinTer.FileType (Select file format)	**New file and clipboard format CSV** The "Comma Separated Value" format is useful to export windows (e.g. dumps, variables, statistic results) to other applications.
Build 1198 IMP 01708	**SNOOPer now also works without dualport** The SNOOPer software trace now also works on targets without realtime memory access capability. Set "SYStem.CpuAccess Enable" to enable non realtime snooping. The target is stopped periodically for the snoop.

Build 1196 IMP 01734 MPC56x NEXUS Debugger and Trace	**MPC56x 8BitBurst workaround f. RevD ext.** Additional MPC56x 8bit burst workaround for Rev.D for sys.m.standby.
Build 1196 FIX 01707	**enumerations not working with OMF-96**
Build 1196 FIX 01706 PPC400 Debugger PPC440 Debugger	**sync- or async-break fail after GO**
Build 1196 IMP 01705 MPC56x NEXUS Debugger and Trace	**MPC56x 8bit burst workaround for RevD** MPC56x 8bit burst workaround for Rev.D
Build 1196 IMP 01704	**Better selection of fonts on UNIX** Alternate font sizes are tried automatically if the preferred font size is not found.
Build 1196 IMP 01703 Data.LOAD.ASAP2 (Load ASAP2 file) sYmbol.AddInfo.LOADASAP2 (Load scaling information from ASAP2 file)	**New command Data.LOAD.ASAP2** Loads information from ASAP2 database file and creates symbols and type information to display the values WITHOUT the need for a debug symbol file.
Build 1196 IMP 01702	**New command INTERCOM.WAIT** New command INTERCOM.WAIT <intercom_name> to wait for another debugger. Useful for starting up multicore debug sessions.
Build 1196 IMP 01701	**ETB for ARM7/ARM9 supported** Support for ARM7/ARM9 ETB included. Tested configurations: 24bit wide trace memory, ETM with 4/8/16 bit trace port size.
Build 1196 FIX 01699	**MPC56x nexus data cycle width problem** The data cycle width (8/16/32 bits) was not reported correctly (only Rev.D silicon).
Build 1196 FIX 01698	**Software breakpoint bugfix for ARM946** The SYStem.Option BUGFIX was not activated on ARM946 cores.
Build 1196 IMP 01697 SETUP.StepAutoAsm (HLL steps stops at assembler code)	**New command SETUP.StepAutoAsm** Stops HLL stepping when assembler code is reached.
Build 1196 FIX 01696	**MPC56x Nexus data cycle alignment failed** Data cycle alignment failed on mtspr/mfspr data cycles and when branches are reported later then datacycles of instructions executed after the branch.
Build 1196 IMP 01695	**Fan of PowerTrace/USB2 temp. controlled** The fan of new PowerTrace units (with USB2 interface) is now temperature controlled.
Build 1196 FIX 01694	**GNU stabs C++ debug info may crash load** Certain stabs C++ debug info may crash loader (very long lines or classes with types defined in method list).

Build 1190 FIX 01693 CPU32 Debugger	**Data.List mixed mode may fail** Mixed mode Data.List window display may caused a debug port fail in very rare cases.
Build 1190 FIX 01692 Super10 NEXUS Debugger and Trace Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Programming of complex triggerunit** now a possible necessary second programming of complex triggerunit is superfluous in the case of DTM or WTM usage inside an ADDR declaration
Build 1190 IMP 01691 DIALOG.Program (Interactive programming) DIALOG.ReProgram (Dialog programming)	**New dialog element DYNAMIC** The new element allows to display dynamic (changable) text or bitmap fields in the dialog window.
Build 1190 FIX 01690 PER.Program (Interactive programming) PER.ReProgram (Load default program)	**PER.Program crashed with long texts** PER.Program was crashing when the description text was longer than 256 characters.
Build 1190 IMP 01689 Host Driver Software	**Support for Windows 2003 Server** USB drivers are available for x86, ia64 and amd64 versions.
Build 1190 IMP 01688	**PROTOCOL API supports graphical display** New command <trace>.PROTOcol.Draw to display protocol related data as X-t diagram.
Build 1190 IMP 01686	**DSP56xxx MCOFF line assignment improved** The source line information for MCOFF formats (GNU) for DSP56xxx is improved. The start address of a function now always has a line number.
Build 1190 01685 DSP56K Debugger ONCE Debugger for DSP56300 SYStem.Option (Special setup)	**DSP56xxx IMASKASM/IMASKHLL locked** SYStem.Option IMASKASM/IMASKHLL was locked unintentionally.
Build 1190 IMP 01684 ARM Debugger ARM9 JTAG Debugger FIRE Emulator for ARM7 In-Circuit Emulator for ARM7 TRACE32 Instruction Set Simulator for CORTEX / ARM / XSCALE	**new accessmodes for ARM family** EC0:..EC15: EDEBG: EETB: EETM: EICE: EINI: added
Build 1190 IMP 01683 C166CBC Debugger Super10 Debugger FIRE Emulator for C166 Family In-Circuit Emulator for C166/ST10 - Out of Production SIM Instruction Set Simulator for C166/ST10	**new access mode for 80166 family** EDBG: added

Build 1190 IMP 01682 sYmbol.SourceCONVert (Conversion for Japanese font)	**EUC-JP characterset support added** Strings in target memory will be displayed in EUC with the setting "SETUP.ASCIITEXT EUC-JP". Sourcetext is converted from EUC-JP by the setting "sYmbol.SourceCONVert EUC-JP".
Build 1190 FIX 01681	**Problems with large or dual screens** Window borders where not updated correctly. Command NONCPAINTFIX under SCREEN= chapter in config.t32 may now not longer be necessary.
Build 1190 ERR 01680 CTS.List (List trace contents)	**Bug fixed in T.CTS.List with groups** T.CTS.List may not display all group changes. Changing the groups on the fly may cause an incorrect display.
Build 1190 IMP 01679 APPEND (Append to file)	**New command APPEND for logfile support** The command APPEND adds a line to the end of a textfile.
Build 1190 FIX 01678	**Multicore debug with OMAP16xx failed** DSP memory display could get instable when dump memory on the ARM side. Synchronous Go/Break was not working.
Build 1190 IMP 01677	**Online Help Acrobat Plugin** new Plugin Version: 4.10 added new Find function to use the new find algorithm of the Online Help
Build 1190 IMP 01676 HELP.Find (Perform a full-text search)	**Online Help: Find** Complete change of the find algorithm to get better search results.
Build 1190 IMP 01675 Data.IMAGE (Display image data)	**New video image formats** New formats GrayScale8, YUV420 and YUV422P/YUV422PS.
Build 1190 FIX 01674 ICE In-Circuit Emulator STOre (Store settings as PRACTICE script)	**STORE Break BreakAll BreakANH** STORE <filename> BREAK, BREAKALL, BREAKANH Inside the generated file the old syntax was used for storing Break.SELect FOREGROUND and Break.SELect BACKGROUND (since 2003.12.15) and old syntax was locked.
Build 1190 FIX 01673 MENU.AddMenu (Add one standard menu item) MENU.AddTool (Add a button to the main toolbar)	**MENU.ADDxxx failed with quotes in string** The commands failed when quotes where in the parameter strings.
Build 1190 IMP 01672	**Multicore settings missing in store** The multicore settings where missing in the STORE SYSTEM command.
Build 1190 IMP 01671 LA (Logic analyzer)	**New trace method LA** The LA method allows to import trace data from external logic analyzers.
Build 1190 IMP 01670 Data.LOAD.eXe (Load EXE file)	**EXE/PDB loader looks at alternate file** The EXE/PDB (WindowsCE) loader searches also for renamed .PDB files.

Build 1190 IMP 01669 Data.LOAD.Elf (Load ELF file)	**Bugfix for Metrowerks EPPC 8.0** Bugfix implemented for Metrowerks PPC 8.x compilers source lines. Files must be loaded with /MetroWerks /GlobTypes /BUGFIX options.
Build 1190 IMP 01668 TrOnchip (Onchip triggers)	**Benchmarking counters on TI ARM cores** The benchmarking counters no TI ARM cores (ARM925T and ARM926EJ) are now supported by the TrOnchip window.
Build 1190 FIX 01667	**Static members missing for GCC3/RVCT2.X** Static members of classes may have no type information on GCC3.x or RVCT 2.x compiled files with DWARF2 debug information.
Build 1190 FIX 01666 PERF.Mode (Specify sampling object)	**PERF.Mode TASK may crash** PERF.Mode TASK may crash when more than 64 tasks exist.
Build 1190 FIX 01665 FIRE Fully Integrated RISC Emulator Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**analyzer trigger unit programming** hostbased software may crash under certain circumstances
Build 1190 FIX 01664 PowerTrace for NEXUS PowerIntegrator - Logic and Bus Analyzer PowerProbe - Logic and Protocol Analyzer Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**programming of complex triggerunit** hostbased software version only: trigger unit programmed wrong under certain circumstances
Build 1190 IMP 01663 SYStem.MemAccess (Run-time memory access (non-intrusive))	**ARM RealMonitor supported** The new memory access mode "RealMON" supports communication with the ARM RealMonitor.
Build 1190 FIX 01661 Data.LOAD.Elf (Load ELF file)	**Floating point format wrong (METAWARE)** The double floating point format was wrong for Metaware ARM compiler files. Loading requires now the /METAWARE option.
Build 1190 FIX 01660 PowerIntegrator - Logic and Bus Analyzer PowerProbe - Logic and Protocol Analyzer Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**programming of complex triggerunit** since V1.12 from 2004.02.23 the usage of several single channels was blocked inside a SELECTOR definition
Build 1190 FIX 01659 Super10 NEXUS Debugger and Trace Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Programming of complex triggerunit** now a possible necessary second programming of complex triggerunit is superfluous in the case of DTM or WTM usage inside an ADDR declaration

Build 1190 FIX 01658 Super10 NEXUS Debugger and Trace Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Programming of complex triggerunit** the usage of DTM and WTM inside an ADDR declaration failed sometimes under special circumstances without reprogramming of analyzer trigger program
Build 1190 FIX 01657	**Triggerpoint on PowerTrace wrong** The triggerpoint may be shown wrong (not at record 0) when the analyzer was stopped before the trigger delay has elapsed.
Build 1190 FIX 01656	**Target state may not be updated in loop** Target state variable (e.g. PRACTICE run() function) may not be updated in PRACTICE loops that only contain simple PRACTICE commands and assignments.
Build 1190 FIX 01655	**Tricore Disassembler error LOOP16** The 4bit displacement field was sign extended.
Build 1190 FIX 01654	**ARM7DI fails to access 16bit words** ARM7DI (not ARM7TDMI !) failed on PowerDebug units when 16bit data was accessed.
Build 1190 FIX 01653	**Data.Load.eXe /RELOC was not relocating** Data.Load.eXE with /RELOC option was not relocating module addresses and variables in data sections other than .bss. Mainly used for Symbian OS load.
Build 1190 FIX 01651 <trace>.STATistic.LINKage (Per caller statistic of function) <trace>.STATistic.TASK (Task activity statistic) <trace>.STATistic.TASKFunc (Task related function run-time analysis) <trace>.STATistic.TASKState (Performance analysis) <trace>.STATistic.TASKSRV (Analysis of time in OS service routines) <trace>.STATistic.TASKTREE (Tree display of task specific functions)	**some A.STAT commands may be locked** Some A.STAT get locked when using a trace method that supports no timestamps. This lock remains active even when later a trace method with timestamps is selected.
Build 1190 FIX 01650 Super10 NEXUS Debugger and Trace	**Analyzer flow processing by loop endless** Analyzer.Find backward (or simmilar commands) may hang when selective tracing (with TraceON/TraceOFF berakpoints) is made.
Build 1190 FIX 01649	**PERF.List pulldown menu fails** Executing PROfile or View items from the PERF.List pulldown menu may fail when PERF operates in LABEL,VarState,TASK,DistriBution or LeVel mode.
Build 1190 FIX 01648 PPC600/750 Debugger	**Wrong IP value after Register.RESet** PPC6xx: PC was initialized with 0x00 instead of 0xfff00100
Build 1190 CHG 01647 ARM9 JTAG Debugger	**Software breakpoints on ARM966E** Software breakpoints on ARM966E Rev.1 will fail sometimes. Setting of "SYStem.Option BUGFIX ON" is required to activate the workarround in the software. ARM966E Rev.2 has this buf already fixed.

Build 1190 FIX 01646 sYmbol.RELOCate.shift (Relocate symbols)	**sYmbol.RELOcate crashes with ext. addrs** sYmbol.RELOCate.shift crashes with extended addresses (16:32 format) when attribute information is present.
Build 1190 FIX 01645	**KEIL C166 XHUGE pointers handled wrong** "xhuge" data pointers where handled as "far" pointers.
Build 1190 FIX 01644	**ARM710 onchip breakpoints not working** Onchip breakpoints and single stepping on ARM710 may fail. Takes not existing PID register into consideration.

Build 1161 FIX 01643	**Data.Draw and Var.Draw fail with 24bit** Data.Draw and Var.Draw commands fails with 24bit or 48bit wide data.
Build 1161 FIX 01642 FDX.OutChannel (Outchannel state display)	**FDX fails to transfer packets** FDX transfers from target to host fails for packet sizes where the lower byte of the packet size is in the range 0xfc to 0xff.
Build 1161 FIX 01641	**Trace.FindAll does not display timestamp** The Trace.FindAll window does not display timestamp information.
Build 1161 CHG 01640 Timing and Protocol Analyzer	**Programming of complex triggerunit** command renames: Count.* -> Counter.* Count.Enable -> Counter.Increment declaration renames: EVENT -> EVENTCOUNTER TIME -> TIMECOUNTER declarationtype synonymes introduced EVCNT for EVENTCOUNTER TICNT for TIMECOUNTER
Build 1161 CHG 01639 ICD In-Circuit Debugger Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Programming of complex triggerunit** declarationtype synonymes introduced EVCNT for EVENTCOUNTER TICNT for TIMECOUNTER
Build 1161 CHG 01638 FIRE Fully Integrated RISC Emulator Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**analyzer triggerunit programming** declarationtype synonymes introduced EVCNT for EVENTCOUNTER TICNT for TIMECOUNTER
Build 1161 CHG 01637 FIRE Fully Integrated RISC Emulator Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**analyzer triggerunit programming** command renames: Count.* -> Counter.* Count.Enable -> Counter.Increment declaration renames: EVENT -> EVENTCOUNTER TIME -> TIMECOUNTER
Build 1161 CHG 01636 PowerIntegrator - Logic and Bus Analyzer PowerProbe - Logic and Protocol Analyzer Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**programming of complex triggerunit** declarationtype synonymes introduced EVCNT for EVENTCOUNTER EXSYCNT for EXTERNSYNCCOUNTER TICNT for TIMECOUNTER

Build 1161 CHG 01635 ICE In-Circuit Emulator Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**analyzer triggerunit programming** declarationtype synonymes introduced EVCNT for EVENTCOUNTER EXCNT for EXTERNCOUNTER TICNT for TIMECOUNTER
Build 1161 FIX 01634	**PERF may display only "other" time** On some architectures the PERF command (performance profiler) may display only time in the "other" column.
Build 1161 FIX 01633	**FDX T32_FDX_Open call not working** The T32_FDX_Open call fails for dedicated communication cannels (e.g. ARM DCC) in the target to host directory when the channel window is already open.
Build 1161 FIX 01632 Data.LOAD.COFF (Load COFF file) Data.LOAD.Elf (Load ELF file)	**Compilation directory missing in DWARF2** The compilation directory in missing in the source filename in DWARF2 debug information when a directory is already specified for the source.
Build 1161 FIX 01631	**Function prototypes in expressions** Function prototypes in expressions may cause the expression parser to crash.
Build 1161 IMP 01630	**Online Help Acrobat Plugin** This plugin is used for communication between Acrobat Reader and the new online help. It provides navigational commands for the PDF help files. Plugin will work on Acrobat Reader Versions 4.x, 5.x, 6.x and Acrobat Full Versions 4.x, 5.x The current plugin software version is 3.01 (can be seen in the Help menu "About TRACE32...")
Build 1161 IMP 01629 HELP (Online help) STOre (Store settings as PRACTICE script)	**New Online Help PDF** The online help is completely changed from Winhelp to PDF help. The following functions are changed: - all HELP commands and the help dialog - context help - error messages and warnings Until the new PDF help is not installed, the main Winhelp functions will still work.
Build 1161 IMP 01628	**MIPS simulator with compare interrupt** The MIPS simulator now supports generating interrupts on COMPARE register matches.
Build 1161 CHG 01627 FIRE Emulator for TriCore ICD Solutions for TriCore SIM Instruction Set Simulator for TriCore	**Access modes** access modes BBB: and EBBB: renamed to EEC: and EEEC:
Build 1161 FIX 01626	**MIPS32 disassembler shows wrong targets** The disassembler for MIPS32 code shows wrong target addresses for branch instructions.

Build 1161 FIX 01625 <trace>.Timing (Waveform of trace buffer) Probe.Timing (Display trace contents as timing diagram)	**Unexpected zoom out in timing windows** Scrolling timing windows out of the visible area may cause the window to zoom out.
Build 1161 ERR 01624 CTS.ON (Switch on trace-based debugging)	**<trace>.CTS.ON crashes** Executing <trace>.CTS.ON the first time will crash the debugger. Use <trace>.CTS.GOTO instead (or the CTS Goto item in the pulldown menu of the trace windows).
Build 1161 IMP 01623 FIRE Emulator for TriCore ICD Solutions for TriCore SIM Instruction Set Simulator for TriCore	**Access modes** additional access modes BBB: and EBBB:
Build 1161 CHG 01622 ICE In-Circuit Emulator Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Analyzer trigger unit programming** command renames: Count.* -> Counter.* Count.Enable -> Counter.Increment declaration renames: EVENT -> EVENTCOUNTER EXTERN -> EXTERNCOUNTER TIME -> TIMECOUNTER
Build 1161 IMP 01621 FIRE Emulator MPC8XX ICD Solutions for PowerPC and Power Architecture SIM Instruction Set Simulator for PowerPC	**Access modes** additional access mode DBG: and EDBG:
Build 1161 IMP 01620 FIRE Emulator for ARM7 ICD Solutions for ARM TRACE32 Instruction Set Simulator for CORTEX / ARM / XSCALE	**Access modes** additional access mode ETB:
Build 1161 FIX 01619 S12 Debugger SYStem.RESet (Reset configuration)	**SYStem.RESet clears option of MCS12** Executing "SYStem.RESet" clears the "SYStem.Option ROMHM" when a MCS12 cpu is selected. On default this option should be set.
Build 1161 FIX 01618 GOTO (Local script jump)	**PRACTICE may cause nesting error** Executing a GOTO command immediatelly after a PRACTICE block begin may cause a PRACTICE nesting error when executed the second time.
Build 1161 FIX 01616 <trace>.List (List trace contents)	**CPx data cycles in ETM trace shown wrong** Accesses to coprocessor registers are shown with the wrong cycle type.
Build 1161 CHG 01615 Data.LOAD.eXe (Load EXE file)	**EXE code was loaded to physical memory** Data.LOAD.eXe was loading the code to physical memory (A:) also for non-X86 architectures. Now the code is loaded to regular memory - including the MMU translation, when enabled.

Software Release from 10-Jan-2004

Build 1151

Build 1151 FIX 01614 FIRE Fully Integrated RISC Emulator Data.LOAD.Binary (Load binary file)	**Download** download of binary files to monitor control based FIREs may fail
Build 1151 FIX 01613 Var.AddWatch (Add variable to Var.Watch window)	**AddWatch** problem fixed in AddWatch pulldown menu item - may crash debugger
Build 1151 IMP 01612 MIPS32/MIPS64 Debugger SH Debugger Data.LOAD (Load file)	**DWARF1** Changes made to DWARF1 loader for Hitachi-SH compiler and Greenhills-MIPS
Build 1151 FIX 01611 ICE In-Circuit Emulator	old fashioned command B.B for B.BG was locked (since 15.12.2003)
Build 1151 FIX 01610 MPC5xx/8xx Debugger	MPC500: 55x&56x bits in SCCR and PLPRCR of peripherie corrected
Build 1151 FIX 01609 MPC56x NEXUS Debugger and Trace	MPC500: 55x&56x bits in SCCR and PLPRCR of peripherie corrected
Build 1151 IMP 01608 ARM Debugger	ARM SWI compatible semihosting supported (via breakpoint at SWI or via DCC channel)
Build 1151 IMP 01607 ARM Debugger	ARM7/9 ICEbreaker uses ANDNOT combination to better match ranged breakpoints
Build 1151 IMP 01606	**Statistic analysis** Instruction based statistic functions deal better with jumps to libraries at end of function
Build 1151 FIX 01605 ARM Debugger Go (Debug control, program execution, and real-time emulation)	**OMAP1510** bug fixed in OMAP1510 multicore synchronous GO (caused ARM to step only)
Build 1151 FIX 01604 C5500 Debugger Go (Debug control, program execution, and real-time emulation)	bug fixed in OMAP1510 multicore synchronous GO (caused ARM to step only)
Build 1151 FIX 01603 CTS.List (List trace contents)	several bugs fixed in A.CTS.List when GROUPs were used

Build 1151 IMP 01598 Uniform Look-And-Feel for all Processors Data.IMAGE (Display image data)	new formats added to Data.IMAGE command (TrueColor formats RGB24 and YUV422)
Build 1151 IMP 01597 MPC56x NEXUS Debugger and Trace	MPC56x Nexus relocated exception table addresses from trace automatically converted when sys.option vectors is set to the table address.
Build 1151 CHG 01596 PowerIntegrator - Logic and Bus Analyzer PowerProbe - Logic and Protocol Analyzer	**Programming of complex triggerunit** command Counter.Enable renamed to Counter.Increment
Build 1151 FIX 01595 JTAG Debugger	JTAG sequence improved to reach 20MHz from the beginning
Build 1151 FIX 01594	Bugs fixed in PCP single stepping of jump instructions
Build 1151 FIX 01593 ARM/Cortex Trace (parallel)	SWP instructions caused data alignment problems on ARM ETM
Build 1151 FIX 01592 TRACE32 Instruction Set Simulators	PORT analyzer in Simulator with SLAVE mode may cause protection fail
Build 1151 IMP 01591	Selecting virtual function in source selects method
Build 1151 FIX 01590 PPC600/750 Debugger	MPC603e: some 603e derivatives (Projekt99) do not work with PowerDebugUSB/PowerDebugEthernet. Or at least with with low JTAG frequency.
Build 1151 IMP 01589 PowerIntegrator - Logic and Bus Analyzer PowerProbe - Logic and Protocol Analyzer Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	size of all counters increased from 44 to 45 bits
Build 1151 CHG 01588 ICE In-Circuit Emulator Go.Back (Go back in program (CTS))	**commands renamed** Break.SELect FORE -> Break.SELect FOREGROUND Break.SELect BACK -> Break.SELect BACKGROUND Break.Back -> Break.BackGround Go.Back -> Go.BackGround Go.Back is used for CTS now
Build 1151 FIX 01587	Hicross loader /PackedEnum option was not working
Build 1151 IMP 01586	stabs loader accepts 'k' and 'B' items
Build 1151 FIX 01585	DWARF2 loader fails when location list begins with large discarded portion
Build 1151 FIX 01584 SIM Instruction Set Simulator for HC12/MCS12	Simulator for HC12 executed min and max commands wrong

Build 1151 FIX 01583 SIM Instruction Set Simulator for 68K/ColdFire	Simulator for 68k indexed pc-rel addressing wrong for disp >= 0x40
Build 1151 FIX 01582 SIM Instruction Set Simulator for 68K/ColdFire	Simulator for Coldfire did not place vector on stack in exception frame
Build 1151 IMP 01581	Multiple debug instances of same module in stabs debug info handled better
Build 1151 FIX 01580	Macros may not be found in expression parser

Build 1146

Build 1146 FIX 01579 PowerIntegrator - Logic and Bus Analyzer PowerProbe - Logic and Protocol Analyzer Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Programming of complex triggerunit** usage of levelname START may crash hostbased software
Build 1146 CHG 01578 Teak/TeakLite Debugger	OAK: usage of memory class DBG supported now

Build 1145 FIX 01577 ARM9 JTAG Debugger	TrOnchip.Set StepVector was always active on ARM9
Build 1145 FIX 01576	hostdriver didn't start if IP is used directly instead of nodename (since 21.11.03) (error message: fatal error from device-driver: TRACE32 (<ipaddr>) not responding)
Build 1145 FIX 01575	T32_EvalGet API call was crashing
Build 1145 FIX 01574 sYmbol.CLEANUP (Workarounds for redundant symbol information)	y.cleanup removed also "global" module symbols
Build 1145 IMP 01573 ARM Debugger	First implementation of background monitor for ARM
Build 1145 FIX 01572 Sample-based Profiling PERF (Sample-based profiling)	SH/ARM monitors failed with the PERF command
Build 1145 FIX 01571	DWARF loader may crash when loading large classes
Build 1145 FIX 01570	Broadcast configuration was not working on little endian hosts
Build 1145 FIX 01569 TPU Debugger (68332, MPC55x/56x) TPU.Step (Single step TPU)	TPU command was locked on CPU32 BDM debuggers
Build 1145 FIX 01568 JTAG Debugger PowerTrace	PowerDebug/Ethernet was responding to ping commands on multicast addresses
Build 1145 FIX 01567 C5500 Debugger Data.LOAD.COFF (Load COFF file)	**COFF loader** bug fixed in C55x COFF loader (support of large pointers)
Build 1145 FIX 01566 C5500 Debugger	bug fixed in C55x disassembler
Build 1145 IMP 01565	symbol.autoload distinguishes between static and dynamic entries, dynamic do not overwrite static
Build 1145 IMP 01564	new command ENCRPYPT and DECRYPT to encrypt files
Build 1145 IMP 01563 Script Language PRACTICE	new commands DODECRYPT and ENCRYPTDO implemented to handle encrypted PRACTICE files
Build 1145 IMP 01562	presence of temporary directory checked during startup
Build 1145 IMP 01561 PSTEP (Execute single line)	PSTEP with same parameters as "DO" command will start PRACTICE file for debugging

Build 1145 IMP 01560 STOre (Store settings as PRACTICE script)	STO SOURCEPATHCACHE implemented to store cached directories from y.spath command
Build 1145 FIX 01559	y.autoload.checkepoc may crash
Build 1145 IMP 01558	Software breakpoint instruction used on ALL ARM9E cores (can be disabled with sys.o bugfix)
Build 1145 FIX 01557	y.spath.list may crash when pressing "Delete" in window multiple times
Build 1145 IMP 01556	timestamps in Analyzer.List window improved to support "artifical" timestamps for flowtraces
Build 1145 IMP 01555	sys.reset does not change CPU type setting
Build 1145 FIX 01554	MPC82xx/G2_LE: DCACHE was corrupted with each step/go/break. Bit 1616 fix on LSRL
Build 1145 FIX 01553	ICE for 68302 and fulldualported RAM: under circumstances wrong flagram data
Build 1145 FIX 01552	Simulators accepted no input after menubar was opened when in GO mode
Build 1145 FIX 01551	Scrolling bejoind end of file in Data.ListMix window now possible on Motif
Build 1145 FIX 01550	New option PPCLittleEnd to support PPC little endian mode memory access
Build 1145 FIX 01549 FIRE Fully Integrated RISC Emulator Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Analyzer trigger unit programming** output events BUS.A..BUS.D programmed with inverse level (only FEC hardware release > 5 effected)
Build 1145 IMP 01548	Data.ReRoute implemented for ARM
Build 1145 FIX 01547	MPC56x Nexus in 2bit mode was not working
Build 1145 FIX 01546	A.L L.TASK or A.CTS.L may display byte swapped task ID on PC hosts
Build 1145 IMP 01545	New functions PROBE|INTEGRATOR.RECORDS|REF|TRACK.RECORD|STATE|GET
Build 1145 IMP 01544	All commands of selected method possible with Trace.<command>
Build 1145 CHG 01543	MPC500/MPC800: Write to a locked DEC (TBK) (also R.S to <any CPU register>) will be ignored.
Build 1145 FIX 01542	Display of recursive nested based structures may crash debugger
Build 1145 FIX 01541	d.load.elf with /PACK or /FASTPACK was crashing after second load
Build 1145 IMP 01540	return-of-function detection for performance/cts improved (especially for ARM)
Build 1145 IMP 01539	RunTime command improved (tooltip,changed reset behavior)
Build 1145 FIX 01538	Coldfire Disassembler+Simulator implements Rev A+ instructions
Build 1145 IMP 01537	Break.Delete|DIsable|ENable /HARD /SOFT /ONCHIP selectively deletes breakpoints

Build 1138

Build 1138 FIX 01536	Drag&Drop on Windows with text was not working with other applications
Build 1138 IMP 01535	Analyzer based statistic and chart functions improved, work now also without Break.SetFunc, better graphics, support for multiple branches on flowtraces, support for artificial timestamps, new Nesting sort mode, incremental window update, support for special return and function chaining constructs, better support for recursive function calls.

Build 1133 IMP 01534	Analyzer based code coverage extended to support 16:32 address models
Build 1133 CHG 01533	Temporary dialog windows now always in MWI mode (outside MDI window)
Build 1133 IMP 01532	TrOnchip.VarCONVert implemented for PowerPC, is now default
Build 1133 IMP 01531	programming of complex triggerunit of PowerProbe and PowerIntegrator released (V1.02)
Build 1133 IMP 01530	A.FindAll was limited to 10000 found records
Build 1133 CHG 01529	general commandline parser: accessmodes EDC:, EIC: and ENC: added (only PPC affected)
Build 1133 IMP 01528	new commands MAP.UpdateOnce and Data.UPDATE to prevent debugger from updating memory displays
Build 1133 IMP 01527	A.Chart/A.STAT improved for long symbol names
Build 1133 FIX 01526 <trace>.Chart (Display trace contents graphically) <trace>.STATistic (Statistic analysis)	A.Chart/A.STAT may crash with long symbol/function names
Build 1133 FIX 01525 <trace>.List (List trace contents) <trace>.View (Display single record)	A.List/View /FILE may crash when MAP.DMUX/MAP.ADELAY is active
Build 1133 IMP 01524	New prefix "WinExt" to open external window (MWI window in MDI environment)
Build 1133 IMP 01523	File selection dialog windows on Motif new closed immediatelly after selection
Build 1133 FIX 01522	canceling standard dialogs (e.g DIALOG.FILE) left parameters in undefined state
Build 1133 FIX 01521 Data.PROLOG (Automatic data modification on program execution start)	Data.PROLOG not working correctly when Go made on breakpoint location (Step-Go sequence)
Build 1133 FIX 01520 PER.Program (Interactive programming)	TEXTFLD item in PER.Program was not working as expected
Build 1133 FIX 01519	WinPrint.Flag.ListFunction may hang when functions without range exist in window
Build 1133 CHG 01518	/GHILLS option now also required for MIPS/Tricore/V850 Greenhills compiler
Build 1133 IMP 01517	Better support for Japaneese Windows (Fonts)
Build 1133 FIX 01516	display of a.chart/a.stat may toggle when sorting is active and items have same values

Build 1133 FIX 01515 Data.LOAD.SysRof (Load RENESAS SYSROF file)	d.load.sysrof may crash
Build 1133 FIX 01514 ARM/Cortex Trace (parallel)	Bug fixed in ARM ETM preprocessor may work in wrong mode or with wrong threshold level
Build 1133 FIX 01513 STOre (Store settings as PRACTICE script)	STOre xxx FLAG was not working correctly
Build 1133 IMP 01512 CTS.SmartTrace (CTS smart trace)	New command Trace.CTS.SelectiveTrace to support CTS with selective tracing
Build 1133 FIX 01511 <trace>.STATistic (Statistic analysis)	Bug fixed in a.stat commands - may crash with long symbols names
Build 1133 CHG 01510	text find "cancel" keeps old window display position
Build 1133 IMP 01509 C5500 Debugger	y.autoload.checkdll implemented for automatic loading of DLLs for C55x
Build 1133 FIX 01508 Data.ComPare (Compare memory)	D.COMPARE xxx--0xffffffff crashed
Build 1133 IMP 01507 TriCore Debugger	fract, sfract, laccum data types supported (for Tricore)
Build 1133 IMP 01506	Tabs positions kept when window commandline is modified
Build 1133 FIX 01505	"Trace" dialog window may crash after switching Method on Windows
Build 1133 IMP 01504	Improved timing display during single stepping
Build 1133 FIX 01503 ColdFire BDM Debugger	Coldfire 5606e was not recognized as MAC coldfire
Build 1133 IMP 01502 <trace>.STATistic.Measure (Analyze the performance of a single signal)	New command A.STAT.Measure to get signal statistics
Build 1133 FIX 01501	Iconified dialog windows where restored incorrect on Windows
Build 1133 CHG 01500 PPC400 Debugger	PPC400: the DBSR[MRR] bit will be mask out and not be cleared by the debugger anymore.
Build 1133 FIX 01499 H8S and H8/300H Monitor	standard H8S rom monitor was not working correctly with extended interrupt mode for CPU-type = 7
Build 1133 FIX 01498 sYmbol.RELOCate (Relocate symbols)	Y.RELOC crashed when attribute or stack table was not empty
Build 1133 FIX 01497 S12 Debugger	Star12C breakpoint to location 0 with trap action active after breakpoints removed

Build 1133 FIX 01496 C5500 Debugger	Software breakpoints on C55x may not hit when realtime memory access is in use
Build 1133 IMP 01495 ARM Debugger SYStem.Option (Special setup)	SYStem.Option StepVector and ICEbreaker for single stepping changed for ARM
Build 1133 FIX 01494 ARM/Cortex Trace (parallel)	Bug fixed in ETM disassembly of interrupts and exceptions
Build 1133 FIX 01493 PowerTrace for NEXUS	Frequency counter for Nexus messages was multiplied by 8
Build 1133 CHG 01492 Data.LOAD.COFF (Load COFF file)	COFF loader const symbols now saved as C: memory class
Build 1133 IMP 01491 C5500 Debugger	New command TrOnchip.PROfile for time and event profiling on C55x
Build 1133 IMP 01490 List (Display modes for programs) SNOOPer (Sample-based trace)	SNOOPer/Logger now always 32bit
Build 1133 IMP 01489 ART (Trace method for Advanced Register Trace)	New trace method ART
Build 1133 IMP 01488 FDX (Trace method FDX)	New trace method FDX
Build 1133 IMP 01487 PowerProbe - Logic and Protocol Analyzer	PowerProbe complex trigger polarities selectable with Probe.TOUT.A..D
Build 1133 FIX 01486 Host Driver Software	**USB Win98** USB disconnect while operating was causing bluescreen on Win98
Build 1133 FIX 01485 S12 Debugger	Star12 C32 onchip breakpoints may be set wrong when starting on a breakpoint
Build 1133 IMP 01484	Busbreaks implemented for Alchemy processors
Build 1133 IMP 01483 Data.Test (Memory integrity test)	Data.TEST now set to QUAD/LONG/WORD as default memory width
Build 1133 FIX 01482	**USB2** Slow commands may cause communication problems on USB2
Build 1133 FIX 01481 Super10 NEXUS Debugger and Trace	Super10 local registersets where wrong in register window, L3 was missing for M345
Build 1133 FIX 01480	**USB2** Lost communication (due to bad PC hardware) may cause USB2 communication to hang

Build 1133 FIX 01479	**USB2** Booting of PowerDebug with USB2 interface may fail on some PCs
Build 1133 IMP 01478 ARM/Cortex Trace (parallel)	onchip breakpoints for ETM make use of A0 and A1 address ignore bits to avoid full range resource when possible
Build 1133 IMP 01477 Var.Break.Set (Set breakpoint to HLL expression)	New command TrOnchip.VarCONVert to convert Var.Break.Set commands into single address when possible
Build 1133 FIX 01476 ARM/Cortex Trace (parallel)	ETM trigger dialog fails for TraceEnable (Global) with two addresses or ranges
Build 1133 FIX 01475 sYmbol (Debug symbols)	Clicking on module that has same name as valid variable in symbol browser may select wrong symbol
Build 1133 FIX 01474 ARM Debugger	A.COV.List display on ARM with flowtrace was not conforming to the coverage conventions for ARM
Build 1133 FIX 01473 <trace>.Chart.VarState (Variable activity chart) <trace>.STATistic.Sort (Specify sorting criteria for statistic commands)	A.Chart.VarState display was wrong when A.STAT.Sort was not set to OFF
Build 1133 FIX 01472 Super10 NEXUS Debugger and Trace	Super10 trace disassembly/alignment failed in certain repeat block constructs
Build 1133 IMP 01471	**PowerPC L2** commandline parser: accessmode L2: added (only PPC affected)
Build 1133 IMP 01470	**Ethernet** New configuration option "SMALLBLOCKS" for ICD with Ethernet interface
Build 1133 IMP 01469 C5500 Debugger	Advanced onchip breakpoints and benchmark counters on TMS320C55x supported
Build 1133 CHG 01468 Super10 NEXUS Debugger and Trace PER (Peripheral files)	peripheral files for R303 and super10 renamed to perm340.per
Build 1133 FIX 01467 <trace>.FindAll (Find all specified entries in trace)	A.FINDALL may crash when more than 1000 hits on ICE or Ethernet-Controller-based ICD/FIRE software
Build 1133 FIX 01466	R.TASK or task selection in Var.Frame failed on ICE
Build 1133 IMP 01465 NAME (Logical names for physical connections)	NAME.WORD and all internal trace lines extended to 64 bits
Build 1133 FIX 01464	STRING.MID() function crashed when "len" parameter was negative

Build 1133 FIX 01463 <trace>.Find (Find specified entry in trace)	Trace.Find WORD.xxx <value|range> was not working on debugger
Build 1133 IMP 01462	**Software breakpoints** Software breakpoints may also be set to physical addresses (where supported)
Build 1133 FIX 01461 SYStem.Option (Special setup)	SYStem.Option IP now taken as operation value for breakpoints no PPC6xx, PPC7xx and PPC8xxx
Build 1133 FIX 01460	Locking of JTAG via TRACE32 API call may fail
Build 1133 FIX 01459 C5500 Debugger PowerProbe - Logic and Protocol Analyzer	PowerProbe was not working as "Probe" command in TMS320C55x debugger

Build 1096 IMP 01458 Super10 NEXUS Debugger and Trace	Data cycle alignment for direct address selective write tracing improved on Super10 Nexus
Build 1096 FIX 01457 SH Trace	tracebased performance on SH3/4 flowtrace not working correctly
Build 1096 FIX 01456 FIRE Emulator for SH2	Single step on SH2 Fire was not working
Build 1096 IMP 01455 PowerProbe - Logic and Protocol Analyzer Count (Universal counter)	frequency meassurement extended beyond 20MHz on PowerProbe, new counter window
Build 1096 FIX 01454 ColdFire BDM Debugger	Condition/Command/Spot not working for read/write breakpoints on Coldfire
Build 1096 IMP 01453 Power Architecture PQII Trace	New option NOTC for RiscTrace for for MPC82xx to support tracing without TC lines
Build 1096 FIX 01452 JTAG Debugger	Problem fixed in JTAG PLD for Super10, Tricore, 166SV2, 166CBC, H8S, PPC400, MCORE, DSP56XXX, may cause instable JTAG connection
Build 1096 FIX 01451 ColdFire BDM Debugger PERF.Mode (Specify sampling object)	PERF with StopAndGo method not working on Coldfire with PowerDebug
Build 1096 IMP 01450 TriCore Debugger	separate perfile for PCP2 (Tricore)
Build 1096 FIX 01449 FIRE Emulator for H8S and H8/300H	H8 internal memory is only shadow memory on Fire
Build 1096 IMP 01448 PowerPC 400 Trace	Debug cycles in RiscTrace for PPC440 now suppressed
Build 1096 IMP 01447 TERM (Terminal emulation)	TERM window with CPU-stop protocol changed to fully stop when nothing is ready
Build 1096 FIX 01446 TERM (Terminal emulation)	TERM window with spotpoints may skip spotpoint instruction
Build 1096 IMP 01445	**PowerPC** Memory access width for variables on PowerPC improved for better performance
Build 1096 FIX 01444 PERF.List (Default profiling)	Clicking on line in PERF.List window fails for addresses >0xa0000000
Build 1096 FIX 01443 C5500 Debugger	RCL/InterCom handling not suspended during long commands (caused subcore debug time on C55x/ARM debugger)
Build 1096 IMP 01442 DSP56K Debugger ONCE Debugger for DSP56300	DSP5600x/DSP563xx strings displayed as "WideString"

Build 1096 FIX 01441 FIRE Emulator for H8S and H8/300H	Dualport access problem to bit variables denied for H8S/H8300H in 64K and 1M addressing modes
Build 1096 FIX 01440	Bug fixed in program controlled data filering when last filter range end with address 0xffffffff
Build 1096 IMP 01439 Super10 NEXUS Debugger and Trace	Super10 filter assignment improved to have no gaps in filter usage
Build 1096 IMP 01438 Break.Set (Set breakpoints)	Break.Set /VCONDition expression checked on command entry
Build 1096 FIX 01437 ICD In-Circuit Debugger Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Trigger Programming** ANAICD triggerprogramming: program crash fixed
Build 1096 FIX 01436 PowerProbe - Logic and Protocol Analyzer	PowerProbe timestamps in 50MHz trace mode wrong
Build 1096 FIX 01435	**ELF/STABS** Relocation of ELF/STABS files not working correctly
Build 1096 FIX 01434 FIRE Emulator MPC8XX MPC5xx/MPC8xx Trace	MPC5xx/MPC8xx flow trace display may hang when PTR cycles are missing
Build 1096 FIX 01433 Super10 NEXUS Debugger and Trace	Selectively tracing overlayed programs on Super10 fails when last message before SBM was OTR message
Build 1096 FIX 01432	Demangler for Borland C++ signatures not working
Build 1096 IMP 01431 sYmbol.Browse (Browse symbols)	sYmbol.Browse window did not show all symbols when symbol number increases while window is open
Build 1096 FIX 01430	**WinCE** WinCE PDB loader, references to types that have a single ´_´ in the name may not be resolved
Build 1096 FIX 01429 ARM Debugger	ARM MMU translation wrong for 64K pages sizes
Build 1096 FIX 01428	API function "T32_GetPracticeState" was not working
Build 1096 BUG 01427 C166CBC Debugger C166 Family Real-Time Trace Break.SetFunc (Mark HLL functions)	breakpoints set with Break.SetFunc command may not be stored on CBC166 Risctrace

Build 1085

Build 1085 FIX 01426 DSP56K Debugger MPC5xx/8xx Debugger SH Debugger	missing files after software installation - install database corrected files are on the CD, but weren't not copied FIRE/ICD-C166/ST10: men7860t.men ICD-DSP56K : per56802.per per56807.per per56809.per per56367.per ICD-PPC : per8280.per ICD-SH2 : per7294.per
Build 1085 FIX 01425 ARM Debugger	Stackframe for ARM frames was wrong for framesizes >= 256
Build 1085 FIX 01424	**Win95/98/ME** File timestamp reading functions (e.g. EDIT, DIR) may cause sporadic fails on Win95/98/ME
Build 1085 IMP 01423 FIRE Emulator for C166 Family FIRE Emulator for ST10 FIRE Emulator for NEC V850 Break.SetFunc (Mark HLL functions)	C167/ST10 and V850 fire Break.SetFunc improved to save breakpoint resources
Build 1085 IMP 01422 MPC56x NEXUS Debugger and Trace	Selective program trace of single locations supported on MPC56x Nexus
Build 1085 FIX 01421 sYmbol.EXIT(<symbol>)	Y.EXIT(<func>) function may fail with error message

Build 1084 IMP 01420 MPC56x NEXUS Debugger and Trace	SmartTrace for MPC56x Nexus improved to better deal with very long fifo full gaps
Build 1084 FIX 01419 FIRE Emulator for C166 Family FIRE Emulator for ST10	C167/ST10 Fire second block of bondout flag memories did not work
Build 1084 FIX 01418 DSP56K Debugger ONCE Debugger for DSP56300 DSP56800 and DSP56800E Debugger	DSP56xxx IEEE loaded produced wrong function address ranges
Build 1084 IMP 01417 ONCE Debugger for DSP56300	DSP563xx ONCHIP trace command implemented
Build 1084 FIX 01416 DSP56K Debugger	DSP56xxx bug fixed in stack backtrace for internal stack
Build 1084 FIX 01415 DSP56K Debugger ONCE Debugger for DSP56300 DSP56800 and DSP56800E Debugger	DSP56xxx bug fixed in IEEE loader for OLD compiler formats
Build 1084 FIX 01414 ONCE Debugger for DSP56300	DSP563xx bug fixed in stepping single word branch instructions
Build 1084 FIX 01413 Super10 NEXUS Debugger and Trace	Super10 TraceOFF breakpoint at end of trace may cause long time "tracking" display
Build 1084 FIX 01412 Super10 NEXUS Debugger and Trace	Super10 TraceOFF breakpoint on odd address causes flow error in trace
Build 1084 FIX 01411	**Trace buffer** some system options clear the trace buffer size setting
Build 1084 FIX 01410 DSP56K Debugger ONCE Debugger for DSP56300	**DSP563xx disassembler error** DSP563xx and DSP5600x jump targets calculated wrong (HLL Step into fails)
Build 1084 IMP 01409	**Drag&Drop** Drop of filenames to commandline works also with spaces in filename
Build 1084 IMP 01408 Universal EPROM/Flash Adapter	ESICON trace supports two baseaddresses for address reconstruction (sys.o ts ...)
Build 1084 IMP 01407	New multicore start sequence and INTERCOM protocol

Build 1081 FIX 01406 MPC5xx/8xx Debugger	MPC8xx/MPC5xx small data breakpoint range bugfix was not working correctly
Build 1081 IMP 01405 In-Circuit Emulator for MC68020/30 SYStem.Mode (Select mode)	speed of sys.mode standby for ICE68020 improved
Build 1081 FIX 01404 Var.Break.Set (Set breakpoint to HLL expression)	Default data width for Var.Break.Set /DATA option not used for /MemoryRead and /MemoryWrite breakpoints
Build 1081 FIX 01403 MPC5xx/8xx Debugger	MPC8xx/MPC5xx combined /MemoryRead and /MemoryWrite breakpoint was not allowed

Build 1079 IMP 01402 Super10 NEXUS Debugger and Trace	Super10 TraceON / TraceOFF breakpoints can be set on-the-fly
Build 1079 FIX 01401 Data.LOAD.Ieee (Load IEEE-695 file)	**Tasking VX compiler** D.LOAD.I with new tasking VX compiler failed
Build 1079 FIX 01400	**Drag&Drop** Drop of filename with extension in capital letters was for giving correct command
Build 1079 IMP 01399 ARM/Cortex Trace (parallel) SmartTrace	SmartTrace for ARM improved
Build 1079 IMP 01398 ARM/Cortex Trace (parallel)	ARM ETM trace is now viewable with simulator
Build 1079 FIX 01397 DIR (List subdirectories and files) LS (Display directory)	Dir command showed wrong modification time (not local time)
Build 1079 CHG 01396 SmartTrace	Saved data for flowtraces changed when FIFOFULLs are saved (to keep smarttrace info)
Build 1079 IMP 01395 sYmbol.NEW (Create new symbol)	Y.CREATE/Y.NEW can now also create address range labels
Build 1079 IMP 01394 sYmbol.EXIT(<symbol>)	New function Y.EXIT(<function_name>) to get the last statement of a function
Build 1079 FIX 01393 sYmbol.CREATE (Create and modify user-defined symbols) sYmbol.NEW (Create new symbol)	Accessing user defined functions (Y.CREATE) with Var commands may produce crash
Build 1079 FIX 01392 Super10 NEXUS Debugger and Trace	Super10: new command TrOnchip.BusTriggerDE for on/off triggers on PODBUS
Build 1079 FIX 01391 ARM Debugger	Simple range breakpoints on ARM that need to be converted into two masks may fail
Build 1079 IMP 01390 MPC5xx/8xx Debugger	MPC8xx and MPC5xx flowtrace correlation of PTR cycles improved
Build 1079 IMP 01389 ARM/Cortex Trace (parallel)	ETM.ON also clears all ETM registers
Build 1079 FIX 01388 Var.DRAW (Graphical variable display)	Var.DRAW without parameters caused software to crash

Build 1079 FIX 01387 PowerTrace for NEXUS <trace>.List (List trace contents)	A.L NEXUS may not show last nexus message of trace window
Build 1079 FIX 01386 <trace>.Chart (Display trace contents graphically)	a.c.xxx / a.t windows ignore timescale parameter when given range is outside trace

Build 1077 FIX 01385 ColdFire BDM Debugger	several bugs fixed in ColdfireV4 onchip breakpoints, watch and exclude breakpoints implemented
Build 1077 FIX 01384 <trace>.FindAll (Find all specified entries in trace)	A.FINDALL may stop searching in flowtrace
Build 1077 FIX 01383 ARM Debugger	**Linux** ARM on linux host may crash when analyzing certain stack frames
Build 1077 FIX 01382 ARM Debugger	ARM stepping ldr pc,[rx,xx] may set breakpoint to wrong address (not using D-cache for memory)
Build 1077 IMP 01381	**ARP** address resolution can be done by broadcast instead of using ARP command
Build 1077 IMP 01380	**DHCP** New name resolution possibilities for Ethernet with Broadcast (to support DHCP without DNS)
Build 1077 FIX 01379	Very long symbols in stateline (PC display) may cause debugger to crash
Build 1077 FIX 01378	**PowerPC** MPC82XX, MPC7XX, MPC74XX, PPC750: fixed problems with sys.go
Build 1077 FIX 01377 PowerQUICC II/Pro Debugger	MPC82XX: enabled watchdog servicing via debugger
Build 1077 FIX 01376 C5500 Debugger	Nonaligned memory writes without TargetServer write wrong data on TMS320C55x
Build 1077 FIX 01375 PowerProbe - Logic and Protocol Analyzer	PowerProbe timestamps where wrong in 50MHz mode
Build 1077 FIX 01374 MPC56x NEXUS Debugger and Trace Super10 NEXUS Debugger and Trace	A.VIEW may crash on Super10 Nexus or PPC Nexus
Build 1077 FIX 01373 Super10 NEXUS Debugger and Trace	Super10 register tracing mode was not enabling register trace with watchpoints
Build 1077 FIX 01372 ChDir (Change directory)	chdir <wildcard> starts now on correct directory on Windows
Build 1077 FIX 01371 Super10 NEXUS Debugger and Trace	Super10 watchpoints and filters where not set during realtime run
Build 1077 IMP 01370 LICENSE (Manage TRACE32 licenses)	License window comes up when file license is newer than debug cable license

Build 1077 CHG 01369 ARM/Cortex Trace (parallel)	SYS.O ETMSWAP ON is now default (only 8&16 bit demux mode configuration)
Build 1077 IMP 01368 ColdFire Trace PowerTrace	Coldfire supported on PowerTrace
Build 1077 IMP 01367 TriCore Debugger	Tricore DBGSR register used to detect break reason
Build 1077 FIX 01366	**PODBUS** PODBUS trigger level was not initialized correctly (before any setting was done)
Build 1077 IMP 01365	**DWARF2** Main source name only accepted as ID=1 in DWARF2 debug info
Build 1077 IMP 01364	**DWARF2** Address range for modules taken from functions when not available in DWARF2
Build 1077 FIX 01363 FIRE Fully Integrated RISC Emulator	MAP.PROTECT was locked on FIREs which have memory protection
Build 1077 IMP 01362	Arbitration for RCL/IC changed to prevent it from pulling down the GUI
Build 1077 IMP 01361 <trace>.STATistic (Statistic analysis)	A.STAT max. stack nesting increased to 200
Build 1077 IMP 01360	Switching to "NonStop" cpu access mode disables now any set stopping breakpoints
Build 1077 FIX 01359 CTS.List (List trace contents)	Bug fixed in CTS.List, may crash on ICE when instruction causes more than 63 bytes data transfer
Build 1077 IMP 01358 sYmbol.Modify.CutFunction (Reduce function address information)	New command Y.Modify.CutFunction to remove the address range from functions
Build 1077 FIX 01357 In-Circuit Emulator for the 80186XL x186 Monitor	**MMU** Handling of MMU translation for banked 80186 targets fixed
Build 1077 IMP 01356	**M-Systems** API for M-Systems Flash extended
Build 1077 IMP 01355	**Analyzer functions** Analyzer functions of API return now default analyzer data (FLOW or BUS trace)
Build 1077 IMP 01354 DIALOG (Custom dialogs)	Dialog window size stored correctly
Build 1077 CHG 01353	**Ethernet** Default Warp packet cluster size reduced to 8 to avoit ethernet buffer overflows

Build 1077 FIX 01352 PowerTrace	Analyzer upload of PowerTrace on Ethernet sporadically decompressed wrong data 1:100000000
Build 1077 IMP 01351	**Runtime PowerPC** PPC75X, PPC74XX, MGT5X00, MPC82XX: improved runtime measurement
Build 1077 FIX 01350 PowerQUICC II/Pro Debugger	MPC82XX manual setting of base address. IOBASE() does not worked.
Build 1077 FIX 01349	Setting software breakpoints at spotpoint with CpuAccess enabled may disable all set breakpoints
Build 1077 IMP 01348 Real-time Trace PowerTrace	Trigger record now always marked in RiscTrace/PowerTrace
Build 1077 CHG 01347 MPC5xx/MPC8xx Trace MPC56x NEXUS Debugger and Trace	MPC55x/56x: DMBR/DBOR improved display.
Build 1077 IMP 01346 MPC5200 Debugger PowerQUICC II/Pro Debugger	MPC82XX, MGT5100/5200 ICache decoder added
Build 1077 FIX 01345	/StripPART <string> not working
Build 1077 IMP 01344	Input caret at end of edit field for default focus edit control on windows
Build 1077 FIX 01343 C5500 Debugger	C55x fast stack model implemented
Build 1077 FIX 01342 C5500 Debugger	C55x pointers treated as 16bit now
Build 1077 FIX 01341 C5500 Debugger	C55x bug fixed in multicore stepping when DSP was triggered before start
Build 1077 IMP 01340 TriCore Debugger	Advanced breakpoints for Tricore implemented (Trace control, MEMORY, STACK)
Build 1077 FIX 01339 MPC5xx/8xx Debugger	MPC500/800: TrB trigger for AICD (Proj.97) and ICD (8254/8282) changed and tested. m?800f.tdf necessary.
Build 1077 FIX 01338	Time cursor could flicker when modal dialog or menu was open
Build 1077 CHG 01337 <trace>.AutoInit (Automatic initialization)	A.AutoInit does not clear trace at spotpoints
Build 1077 IMP 01336	**ARM DCC** ARM DCC channel now named DCC (PERF/SNOOP/TERM)
Build 1077 IMP 01335 PowerQUICC II/Pro Debugger	MPC82xx step/onchip breakpoint pipeline fix changed
Build 1077 CHG 01334 PowerQUICC II/Pro Debugger	MPC82xx: moved from project 96 to 91 (fccppc82.t32 -> fccppc42.t32)

Build 1077 FIX 01333 TRACE32 Instruction Set Simulator for CORTEX / ARM / XSCALE	**ARM** ARM simulator was reacting wrong when CPSR_C equal zero
Build 1077 FIX 01332 MPC5xx/MPC8xx Trace	MPC5xx flow trace error fixed (burst size was 4 instead of 8)
Build 1077 IMP 01331 ARM Debugger	Onchip address ranges use both ICEbreaker registers to better convert the range
Build 1077 FIX 01330	Onchip breakpoints that are converted into address makes may cause a too large address mask
Build 1077 FIX 01329 Data.DRAW (Graphical memory display of arrays)	Data.DRAW may crash when window is made larger and do data to update
Build 1077 FIX 01328 ARM Debugger	ETM based breakpoints where not working without trace hardware
Build 1077 IMP 01327 ARM Debugger	Advanced breakpoints for ARM7 and ARM9 implemented (TASK, MEMORY, STACK, EXCLUDE)
Build 1077 IMP 01326	Better error messages when onchip/hardware breakpoints cannot be set
Build 1077 IMP 01325 MPC5xx/8xx Debugger	Advanced breakpoints for MPC5xx/MPC8xx cover now all onchip capabilities
Build 1077 IMP 01324 ARM Debugger Break.SetFunc (Mark HLL functions)	B.SF on ARM was causing wrong memory access class (no other effects)
Build 1077 IMP 01323 ARM Debugger	**Microware ROF** Microware ROF format implemented for ARM
Build 1077 FIX 01322	Stackframe backtrace for x86 with Turbo Debug format changed
Build 1077 IMP 01321 MPC56x NEXUS Debugger and Trace	Frequency Counter selection changed for PPC Nexus
Build 1077 FIX 01320 Step.BackChange (Step back till expression changes)	StepBackChange was stepping forward (CTS)
Build 1077 FIX 01319 Data.SAVE.<format> (Save data in file with specified format)	D.SAVE.xx may crash on some processors (e.g. C166) with PowerDebug USB/Ethernet
Build 1077 FIX 01318 FIRE Emulator for C166 Family C166 Family Real-Time Trace	CTS for C166 was tagging byte writes to registers wrong

Build 1077 IMP 01317 <trace>.state (Display trace configuration window) List.auto (Display program listing) <trace>.state (Display trace configuration window)	WR and DATA.xx items enabled for Logger, SNOOPer and Integrator
Build 1077 IMP 01316	**CTS** CTS memory requirements reduced
Build 1077 IMP 01315 FIRE Fully Integrated RISC Emulator PowerTrace	New analyzer mode Leach for PowerTrace and FIRE
Build 1077 FIX 01314	MMU.CLEANUP command may produce protection fail
Build 1077 FIX 01313	**DWARF2** DWARF2 location descriptions with larger then 4 byte pieces now accepted
Build 1077 FIX 01312	Ethernet timeouts may cause "trashing" of Ethernet packets and LINK ERROR display
Build 1077 FIX 01311	Taskaware charts may crash when last record of trace holds task information
Build 1077 FIX 01310	**UBROF V9.0** Function return points where missing for UBROF V9.0 and up compilers
Build 1077 FIX 01309	**MPC7410** MPC7410: SystemUp update (problems with empty flash devices)
Build 1077 FIX 01308 MPC56x NEXUS Debugger and Trace	PPC Nexus data breakpoint reason display fixed
Build 1077 IMP 01307 Data.LOAD (Load file)	/NOCODE and /PLUSVM options work when used together
Build 1077 FIX 01306 ARM Debugger FIRE Emulator for ARM7	Target function call from Var commands for ARM fixed
Build 1077 IMP 01305	Binary output and HLL input made compatible
Build 1077 IMP 01304	**IDE** When existing window is reopened then it reactivates its original parameters
Build 1077 IMP 01303 Super10 NEXUS Debugger and Trace	Several enhancements for Super10 filtered program traceing
Build 1077 IMP 01302 Break.List (Display list of breakpoints)	breakpoint display highlights hit breakpoint
Build 1077 IMP 01301	**Option DISableHIT** new option /DISableHIT to disable hit breakpoints

Build 1077 IMP 01300 Break.Set (Set breakpoints)	**New breakpoint types** Several new breakpoint types implemented
Build 1077 FIX 01299 MPC74XX Debugger	**Software Breakpoints did not work** MPC74XX: Software Breakpoints did not work properly with MSR_VEC = 1
Build 1077 IMP 01298 MPC5200 Debugger	**Boot Low/High Auto detection** MGT5100, 5200, MPC8280: Boot Low/High Auto detection
Build 1077 FIX 01297 PPC600/750 Debugger	**L2: DEC value partially inverted** PPC603E, PPC750A2, A3, L2: DEC value partially inverted
Build 1077 FIX 01296 PPC600/750 Debugger	**set PC here failure** PPC603E, PPC750A2, A3, L2: set PC here failure
Build 1077 FIX 01295 ONCE Debugger for DSP56300 PER.view (Display peripherals)	**All per Files reworked** 563xx All per Files reworked. Last changes has wrong include parameter.
Build 1077 FIX 01294 MPC5200 Debugger PowerQUICC II/Pro Debugger	**SystemUp failed when Boot** MGT5200/MPC8280: SystemUp failed when Boot from 0x00000100
Build 1077 FIX 01293 ARM Debugger C5500 Debugger	**Bugfix implemented for Multicore** Bugfix implemented for Multicore start/stop ARM+C55x
Build 1077 FIX 01292	**TERM.PROTOCOL COM** TERM.PROTOCOL COM may refuse to open channel for baudrates > 38400
Build 1077 FIX 01291	**TERM.PROTOCOL COM** TERM.PROTOCOL COM with wrong parameters kept com port open without chance to retry
Build 1077 FIX 01290 ARM Debugger	**ELF/DWARF1** Stack backtrace on ARM changed for ELF/DWARF1 format and generally
Build 1077 FIX 01289 MPC56x NEXUS Debugger and Trace SYStem.Mode (Select mode)	Change of DTM setting for MPC5xx nexus was not affecting next sys.mode standby
Build 1077 FIX 01288 MPC5xx/MPC8xx Trace	**Flowtrace error** Flowtrace error on MPC8xx/MPC5xx mtmsr was not considered as indirect branch
Build 1077 FIX 01287	**SR (MSR/PSW) register modification** possible SR (MSR/PSW) register modification problem
Build 1077 FIX 01286 MPC56x NEXUS Debugger and Trace	**StandBy mode** PPC Nexus was not tracing in StandBy mode (trace switched to stack mode)
Build 1077 FIX 01285	**Flow processing table** Flow processing table was allocated one entry too small, may be overwritten
Build 1077 FIX 01284 PPC600/750 Debugger	**Updated Memory access functions** MPC755 Updated Memory access functions (for MPC107 companion chip access)

Build 1077 FIX 01283 MPC5200 Debugger	**Fixed single step over rfci instructions** MGT5100/5200 Fixed single step over rfci instructions
Build 1077 FIX 01282 PPC600/750 Debugger MPC74XX Debugger	**SRR/CSRR Bug** Fixed SRR/CSRR Bug (wrong values displayed) (750,755,74XX,MGT)
Build 1077 FIX 01281 ARM/Cortex Trace (parallel)	**SYSTEM.Option ETM** SYSTEM.Option ETM... was not setting configuration on PowerTrace immediately
Build 1077 FIX 01280 MPC56x NEXUS Debugger and Trace	563xx: per Files. group address of PLL was wrong. PLL DSCR register missed. ESSIx hand over parameter was wrong.
Build 1077 FIX 01279 FIRE Fully Integrated RISC Emulator	Problem on FCC may not produce coredump on USB but instead stall USB
Build 1077 IMP 01278 PowerTrace <trace>.SAVE (Save trace for postprocessing in TRACE32)	**A.SAVE changed** A.SAVE changed to a compressed format for PowerTrace
Build 1077 IMP 01277	**USB2** USB2 support implemented
Build 1077 IMP 01276	**Scaling option added to variable formatt** Scaling option added to variable formatting
Build 1077 IMP 01275	**Floating point numbers** Scalar control "Hex" setting display floating point numbers as hex dump
Build 1077 FIX 01274	**Floating point numbers** Floating point numbers output max. precision increased to 19 digits
Build 1077 FIX 01273	**IEEE Double** IEEE Double +/- INF displayed correctly
Build 1077 IMP 01272	**USB** USB makes communication test on startup
Build 1077 IMP 01271	**Breakpoints handled faster** Spotpoint/Cond breakpoints handled faster also with limited update rate
Build 1077 IMP 01270	Conditional/Counting/Spot breakpoints where very slow on simulator
Build 1077 FIX 01269	**Counting data breakpoints** Counting data breakpoints did not show current count in break.list window
Build 1077 CHG 01268 PPC600/750 Debugger	**HW/SW-BP handling changed** HW/SW-BP handling changed. Will be set with BDM_start.
Build 1077 IMP 01267 Teak/TeakLite Debugger	**Assembly source level debug** OAK assembly source level debug added
Build 1077 IMP 01266	**two-state dialog buttons implemented** two-state dialog buttons implemented

Build 1077 FIX 01265 ARM9 JTAG Debugger	**ICEbreaker** FCSE handing for ICEbreaker changed
Build 1077 FIX 01264	**Rounding error** Rounding error in extended 64bit divide fixed
Build 1077 FIX 01263 PPC600/750 Debugger	**ICache+DCache** MPC755 L1 ICache+DCache r/w and *BAT4..7 write fixed
Build 1077 IMP 01262	**ASAP2** ASAP2 scaling supported
Build 1077 IMP 01261 FramePOS (Controls the position of TRACE32 in MDI window mode)	**FRAMEPOS** new command FramePOS to define size, position and mode of main window
Build 1077 FIX 01260 ARM7 Family Real-Time Trace	**ARM bus trace disassembly** serveral fixes for ARM bus trace disassembly and CTS, new option EXEC to analyze EXEC signal
Build 1077 FIX 01259	**printing** printing wp.f.lf may hang infinite
Build 1077 FIX 01258 C166CBC Debugger FIRE Emulator for C166 Family In-Circuit Emulator for C166/ST10 - Out of Production C166 Monitor Var.Call (Call a new procedure)	bugs fixed in Var.Call for C166 architecture
Build 1077 FIX 01257	**Bus disassembly** problem fixed in speculative bus disassembly
Build 1077 IMP 01256 PPC600/750 Debugger MPC74XX Debugger	**Multicore debugging** PPC75X, MPC740X Multicore debugging supported
Build 1077 FIX 01255	**CTS** VM: and USR: access where not working when CTS was active
Build 1077 CHG 01254 DSP56800 and DSP56800E Debugger	56800: EPLD support 32bit register, TLM (56800E) and no dummy shifts necessary any more. drvbdm59.c adjust.
Build 1077 FIX 01253	**License key** License key stored in dongle was too short (only March/April versions)
Build 1077 FIX 01252	**Update of windows** Update of windows and handling of communication stopped on Windows when pulldown open
Build 1077 FIX 01251 MPC5xx/8xx Debugger MPC56x NEXUS Debugger and Trace	**RSTCONF** MPC500: MEN55x/MEN56x update for RSTCONF programming. Old routine do not work if ISB base is != 0x0.
Build 1077 IMP 01250	**HLL expressions** HLL expressions can include line numbers

Build 1077 IMP 01249 FIRE Emulator for ST10	**Onchip program breakpoints** Onchip program breakpoints implemented for ST10 bondout, also used for stepping
Build 1077 IMP 01248	OS.AREA and OS.HIDDEN made faster
Build 1077 IMP 01247	**Japaneese font handling changed** Japaneese font handling changed
Build 1077 IMP 01246	**IC and RCL** IC and RCL could be used simultaneously now (implemented end of March)
Build 1077 FIX 01245 TRACE32 Instruction Set Simulator for CORTEX / ARM / XSCALE	**ARM** Bug fixed in ARM simulator exception handling (PSR.I and PSR.F where not cleared)
Build 1077 IMP 01244 FIRE Emulator for C166 Family FIRE Emulator for ST10 In-Circuit Emulator for C166/ST10 - Out of Production C166 Family Real-Time Trace	**CTS** handling of volatile memory and atomic instructions improved for CTS with C166
Build 1077 FIX 01243	**Subcore debug license** Subcore debug license checking was not working
Build 1077 FIX 01242 PPC440 Debugger	**TLB** TLB replacement caused debug TLBs to overwrite target TLBs for PPC440
Build 1077 FIX 01241 PPC440 Debugger	**Buserror handling** Buserror handling of PPC440 fixed
Build 1077 FIX 01240	**DWARF2** Lines of assembler module at loaction zero where ignored in DWARF2 loader
Build 1077 IMP 01239	**Saving** More compact saving of multiple window to file
Build 1077 FIX 01238 Super10 NEXUS Debugger and Trace	Several fixes and workarrounds for Super10 Nexus
Build 1077 IMP 01237 JTAG Debugger PowerTrace	**Ethernet** Lost packets on PowerDebug Ethernet handled better
Build 1077 IMP 01236	**warp** analyzer warp update made auto correcting (reduces packet size if it fails)
Build 1077 CHG 01235	**Breakpoints** internal structure of breakpoints changed to prepare for new features
Build 1077 FIX 01234 Data.ComPare (Compare memory)	Data.COMPARE without second parameter compares with same address
Build 1077 FIX 01233	a static member that contains an array of the class itself caused the debugger to crash

Build 1077 FIX 01232 C166CBC Debugger FIRE Emulator for C166 Family In-Circuit Emulator for C166/ST10 - Out of Production C166 Monitor	**Symbol comment** Symbol comment for C166 family direct addressing was wrong or missing
Build 1077 FIX 01231 Super10 NEXUS Debugger and Trace	Cycle alignment for Super10 coxxx instructions fixed
Build 1077 FIX 01230	**MDI interface** Setting of CPU registers failed with MDI interface
Build 1077 IMP 01229	**Dialog labels** Dialog labels can be extended with window name
Build 1077 IMP 01228	**flow trace** Search for "mid-frame" addresses in flow trace implemented
Build 1077 IMP 01227 Super10 NEXUS Debugger and Trace	**WAF** Super10 WAF bit supported
Build 1077 IMP 01226 Super10 NEXUS Debugger and Trace	**Trace control** Super10 trace control flipflop for TraceON/TraceOFF breakpoint cleared/set before use
Build 1077 CHG 01225 C166 Family Real-Time Trace TrOnchip.state (Display onchip trigger window)	C166 Risctrace trace control breakpoints implemented default settings of TrOnchip changed
Build 1077 FIX 01224	Input focus stays on default edit field when pressing return in dialog box on Windows
Build 1077 FIX 01223 PowerProbe - Logic and Protocol Analyzer	CPU selection caused channel modes of PROBE command to be wrong
Build 1077 FIX 01222 Data.LOAD (Load file)	**64bit targets** data.load command /RELOC was not working correctly on 64bit targets
Build 1077 IMP 01221	**module local register variables** basic support for module local register variables implemented
Build 1077 FIX 01220 ARM/Cortex Trace (parallel)	ARM ETM data messages after long trace gaps may cause flow errors (only in noncompressed trace mode)
Build 1077 FIX 01219	**Cosmic** Handling of banked cosmic V4 symbols changed
Build 1077 IMP 01218	**IFTEST** IFTEST extended to include WARP transfer statistics
Build 1077 FIX 01217	**Ethernet** Bug fixed in soft-retries on Ethernet with hostbased software

Build 1077 FIX 01216 <trace>.FindAll (Find all specified entries in trace)	**"AT" item** Trace.FindAll was not working with "AT" item
Build 1077 IMP 01215	On-Site license upgrade supported by LICENSE.ADD command
Build 1077 FIX 01214 CACHE.ComPare (Compare CACHE with memory)	CACHE.COMPARE command was not using physical addressing for the noncached reads
Build 1077 FIX 01213 PPC440 Debugger	**ICache** ICache recoding for PPC440 implemented
Build 1077 FIX 01212 PPC440 Debugger	bug fixed in PPC440 logical memory access with TS=1
Build 1077 IMP 01211 SIM Instruction Set Simulator for C166/ST10	**C166** C166 Simulator supports interrupts
Build 1077 FIX 01210 SIM Instruction Set Simulator for C166/ST10	**C167/ST10** Bug fixed in C167/ST10 simulator when reading from register context memory
Build 1077 FIX 01209 Data.dump (Memory dump)	**Spotlight** Selection fails in data.dump windows with spotlight option
Build 1077 IMP 01208 PPC440 Debugger	**Peripherial file** Peripherial file for PPC440 rewriten
Build 1077 FIX 01207 PPC440 Debugger	**MMU translation** Bug fixed in MMU translation for PPC440 with ERPN!=0
Build 1077 FIX 01206 PPC440 Debugger	**physical memory access** Bug fixed in physical memory access for PPC440
Build 1077 FIX 01205 PPC400 Debugger PPC440 Debugger	**Bus error handling** Bus error handling for OPB for PPC405 and PPC440 improved
Build 1077 FIX 01204	**NAME window** Executing name store from NAME window may produce wrong device selection prompt
Build 1077 FIX 01203 MPC5xx/8xx Debugger	PPC500/800 data write breakpoints with data value disallowed (chip bug)
Build 1077 IMP 01201 LICENSE.UPDATE (Update the maintenance contract)	**New command** New command LICENSE.UPDATE to update license in debug cable (from license file)
Build 1077 FIX 01200 DIALOG.YESNO (Create dialog box with YES and NO buttons)	**Dialog.YESNO may crash** Dialog.YESNO may crash for very long strings
Build 1077 IMP 01199 Data.LOAD (Load file)	**MetroWerks** New option /MetroWerks as workarround to access function local statics for Metrowerks compiler

Build 1077 FIX 01198 Go (Debug control, program execution, and real-time emulation)	**80186** bug fixed 80186, caused GO to fail when stack was in onchip memory
Build 1077 FIX 01197 ColdFire BDM Debugger	**Coldfire** R.SWAP and R.UNDO may fail for A7 on Coldfire BDM
Build 1077 FIX 01196 CACHE.DUMP (Dump CACHE)	**Cache** printing CACHE.DUMP window was now showing whole cache contents
Build 1077 FIX 01195	**Nexus** Nexus BRK trace message may be displayed one branch too early
Build 1077 FIX 01194 MPC5xx/8xx Debugger	passive PPC500/800 debugger was ignoring powerfail while running
Build 1077 FIX 01193	**Simulator** Simulator was ignoring breakpoints after data breakpoint with resume option was hit
Build 1077 FIX 01192	**C++ method** Search for C++ method may fail when methods within classes also exist
Build 1077 FIX 01191	**Mapper** bug fixed in mapper, caused MAP.Noxxx with range to fail
Build 1077 FIX 01190 Data.LOAD (Load file)	problem fixed in .stabs loader with "r" items with "@" size attribute
Build 1077 CHG 01189	**Flowtrace** Internal flowtrace code information harmonized
Build 1077 FIX 01188	**Loading of large files** Loading of large files (1GB) failed with hostbased software
Build 1077 IMP 01187 Super10 Debugger Super10 NEXUS Debugger and Trace	Super10 cycle alignment improved
Build 1077 FIX 01186 SH Debugger	**SH bondout fetch decoding** SH bondout fetch decoding bug fixed
Build 1077 FIX 01185 Super10 NEXUS Debugger and Trace	Super10 tracing of trap #xx and comac with opcode #83 fixed
Build 1077 IMP 01184 WRITE (Write to data file)	WRITE #x %CONT will write to file without line feed
Build 1077 FIX 01183 PPC400 Debugger	**D-Cache** Bug fixed in fast download for PPC405 when D-Cache was dirty
Build 1077 IMP 01182 PPC600/750 Debugger PER.view (Display peripherals)	MPC755: own per file.

Build 1077 IMP 01181	**Simulator** MAP.Protect supported by simulator
Build 1077 FIX 01180	**PRACTICE** Windows shuffling when pages switched by PRACTICE
Build 1077 IMP 01179 JTAG Debugger	**Multiple serial numbers** Supporting multiple serial numbers in one debug cable
Build 1077 FIX 01178	**CTS** bug fixed in ARM CTS condition prediction
Build 1077 IMP 01177 MPC74XX Debugger Register (Processor registers)	ALTIVEC registers and variables implememted
Build 1077 FIX 01176	**Hardcopy** Hardcopies of windows dit not contain PC or selection bars
Build 1077 FIX 01175 MMU.DUMP (Dump MMU tables) MMU.SCAN (Scan MMU tables (static snapshot))	MMU.SCAN/MMU.DUMP without default format issues now an error
Build 1077 FIX 01174 Data.AllocList (Static memory allocation analysis)	**Printing** WinPrint.Data.AllocList was not printing all contents
Build 1077 CHG 01173 ONCE Debugger for DSP56300 Data.LOAD (Load file)	DSP56300: speed up download speed. changed status read in drvbdm58.c
Build 1077 FIX 01172 MPC56x NEXUS Debugger and Trace	Bug fixed in data cycle alignment on PPC Nexus when many pipelined cycles span across multiple jumps
Build 1077 FIX 01171	**Fast memory management** Bug fixed in internal fast memory management
Build 1077 FIX 01170 MPC56x NEXUS Debugger and Trace	Assignment of halfword and byte pipelined memory accesses failed on PowerPC nexus
Build 1077 FIX 01169	**Win2000** Bug fixed in windows font file T32FONT.FON (23. July 2002) may not run on some Win2000 versions (will cause empty windows e.g. version.hardware or error message "ixed width font not found") salvage: copy new file t32font.fon into TRACE32 system directory and reboot PC before using TRACE32 the next time
Build 1077 IMP 01168	**CTS** CTS III implemented (CTS while running, better incremental processing, speculative disassembly, constant memory support).
Build 1077 FIX 01167	**Additional buttons** Additional buttons in Data.List windows where not working

Build 1077 FIX 01166	**Simulator trace** Simulator trace fails in stack mode when trace buffer size is set smaller
Build 1077 FIX 01165 Trace-based Debugging (CTS) MPC56x NEXUS Debugger and Trace SmartTrace	**cts** Several fixed in CTS/Smarttrace for PowerPC Nexus
Build 1077 IMP 01164	**Ethernet** Soft retries implemented for Ethernet debug communication
Build 1077 FIX 01163 SIM Instruction Set Simulator for 68K/ColdFire	**Exception handling** Exception handling (stackframes) fixed in 68k simulator
Build 1077 IMP 01162 SIM Instruction Set Simulator for 68K/ColdFire	Simulator for 68k uses prefetching (solves wrong trace disassembly problem)
Build 1077 FIX 01161 TriCore Debugger TriCore Parallel Trace	**Error handling** Error handling in Tricore OCDS-L2 trace may cause software to crash
Build 1077 IMP 01160 TriCore Debugger TriCore Parallel Trace	**Tricore OCDS-L2** Tricore OCDS-L2 TraceON and TraceOFF breakpoints implemented
Build 1077 FIX 01159	**Temporary license keys for ICE** Temporary license keys for ICE worked only when date in license code was all zeros
Build 1077 CHG 01158	InterCom "file" mode removed
Build 1077 FIX 01157 ARM Debugger MPC56x NEXUS Debugger and Trace Break.Set (Set breakpoints)	**B.Set dialog** B.Set dialog may crash on Sun for ARM or PPC-Nexus
Build 1077 FIX 01156	**Pulldown menus** Pulldown menus may be placed wrong on dual-screen displays
Build 1077 FIX 01155 SIM Instruction Set Simulator for C166/ST10	C166 simulator handled short esfr addressing to some registers wrong
Build 1077 FIX 01154	**USB driver W2000** USB driver may hang on some Windows2000 versions
Build 1077 FIX 01153 FIRE Fully Integrated RISC Emulator ICE In-Circuit Emulator PowerIntegrator - Logic and Bus Analyzer PowerTrace PowerProbe - Logic and Protocol Analyzer <trace>.AutoInit (Automatic initialization)	Automatic init of analyzers (Integrator,PowerTrace,FIRE,ICE) changed

Build 1077 IMP 01152	**Enumeration type** Enumeration type size taken from stabs debug info
Build 1077 FIX 01151	**Metrowerks DWARF1** Mangled names from Metrowerks DWARF1 loaded
Build 1077 FIX 01150 SIM Instruction Set Simulator for x86	**80x86 simulator** 80x86 simulator scas was taking SI register instead of DI
Build 1077 FIX 01149 Data.LOAD (Load file)	STABS loader changed '@' item handling
Build 1077 FIX 01148 XScale Debugger	**Go at 0** XScale Go at 0 makes step first
Build 1077 FIX 01147 In-Circuit Emulator for 68HC11	**dualported RAM** ICE 68HC11: flag information of full dualported RAM may be wrong
Build 1077 IMP 01146 PPC600/750 Debugger	**PPC750CXdd2** PPC750CX: Added Support for PPC750CXdd2
Build 1077 FIX 01145 Trace-based Debugging (CTS)	**Function nesting display** Function nesting display in CTS may not be correct
Build 1077 FIX 01144 Super10 Debugger Super10 NEXUS Debugger and Trace	scxt was wrong dequeued in Super10 Nexus
Build 1077 FIX 01143 Super10 Debugger Super10 NEXUS Debugger and Trace	**Disassembler** Missing Super10 specific instructions added to disassembler
Build 1077 IMP 01142 C5500 Debugger Data.LOAD.COFF (Load COFF file)	**Reset PC** Reset PC taken from COFF file for C55x
Build 1077 FIX 01141 C5500 Debugger	**Update of XARn registers** update of XARn registers in register window failed on C55x
Build 1077 FIX 01140 C5500 Debugger	**C5xx disassembler** Bug fixed in C5xx disassembler, may crash on xcc and xccpart constructs
Build 1077 IMP 01139	**eXception command** eXception command locked (where not available)
Build 1077 FIX 01138 <trace>.STATistic.DIStance (Time interval for a single event)	analyzer distribution statistic has wrong total time
Build 1077 FIX 01137 ONCE Debugger for DSP56300	DSP56300: BIT SCR[REIE] changed. [ena,dis] -> [dis,ena]
Build 1077 IMP 01136 PPC400 Debugger	**Powerdebug** PPC405 SLER!=0 supported on powerdebug
Build 1077 IMP 01135 PPC400 Debugger	**logical memory access via TLBs** PPC40x logical memory access via TLBs supported on PowerDebug

Build 1077 IMP 01134 PPC440 Debugger	**Full memory model** PPC440 full memory model supported
Build 1077 IMP 01133	**SPRs of PowerPC** all SPRs of PowerPC virtualized
Build 1077 FIX 01132 PPC600/750 Debugger	**RUNN counter enlarged** PPC750L3, MPC755: RUNN counter enlarged for slow MEM accesses + wait after RUNN (PDE).
Build 1077 IMP 01131	**COFF/DWARF2 compiler from TI** basic support for COFF/DWARF2 compiler from TI implemented
Build 1077 FIX 01130	**&register_variable** &register_variable was not reporting an error
Build 1077 FIX 01129 ARM9 JTAG Debugger	**ICEbreaker setting** ICEbreaker setting for FCSE on ARM926EJ was wrong
Build 1077 FIX 01128	**Quick search in source window** Quick search in source window did only find first occurence of string in HLL block
Build 1077 FIX 01127	Data.List etc. may show wrong breakpoints in extended (16:32) address mode
Build 1077 IMP 01126 MPC56x NEXUS Debugger and Trace	**MPC56x Rev.D** MPC56x Rev.D supported
Build 1077 IMP 01125 PowerTrace for NEXUS	**data trace modes** New nexus data trace modes
Build 1077 FIX 01124 MPC56x NEXUS Debugger and Trace	**ISYNC** ISYNC was not considered as an indirect branch for NexusPPC565
Build 1077 FIX 01123 XScale Debugger SETUP.IMASKASM (Mask interrupts during assembler step) SETUP.IMASKHLL (Mask interrupts during HLL step)	**not working** XScale: IMaskAsm and IMaskHll not working
Build 1077 FIX 01122	**Multi-Dimensional arrays in DWARF2** Multi-Dimensional arrays in DWARF2 may have wrong row/col ordering
Build 1077 IMP 01121 DSP56800 and DSP56800E Debugger	**Second onchip breakpoint unit** DSP56800 Debugger supports second onchip breakpoint unit
Build 1077 IMP 01120 DSP56800 and DSP56800E Debugger	**Software breakpoints in FLASH** DSP56800 Debugger supports software breakpoints in FLASH
Build 1077 IMP 01119 DSP56800 and DSP56800E Debugger	DSP56800 JTAG access optimized for better performance
Build 1077 IMP 01118 Data.LOAD.IntelHex (Load INTEL-HEX file)	**Optimized** D.LOAD.IntelHex optimized for better performance

Build 1077 FIX 01117 sYmbol.Delete (Delete symbols of one program)	**extended addressing** y.delete may cause crash when removing programs in extended addressing (16:32 format)
Build 1077 CHG 01116 PPC400 Debugger	**debug cable version check** debug cable version check for PPC4xx only done in PowerDebug III
Build 1077 IMP 01115	**hidden flash program** target calls for hidden flash program / prolog / epilog not in trace now
Build 1077 IMP 01114	**VFP9** Support for VFP9 implemented
Build 1077 FIX 01113 Register.view (Display registers)	**Spotlight display** Spotlight display in register window may not mark changes when stack changes
Build 1077 FIX 01112 Data.LOAD.COFF (Load COFF file)	**COFF loader** COFF loader may fail when HLL functions contain no block
Build 1077 FIX 01111 TPU Debugger (68332, MPC55x/56x) TPU.List (View microcode)	**TPU Disassembler** TPU Disassembler bug fixed for constant +1/-1 add
Build 1077 FIX 01110 Data.IMAGE (Display image data)	Data.IMAGE /BottomUp did not draw correct when window size not aligned
Build 1077 IMP 01109 Script Language PRACTICE	PRACTICE parameter passing made reentrant
Build 1077 CHG 01108 Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Trigger.Puls renamed** ICE: SA120 only: analyzer trigger unit programming: Trigger.Puls renamed to Trigger.Pulse
Build 1077 CHG 01107 FIRE Fully Integrated RISC Emulator Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Renamed** FIRE: analyzer triggerunit programming: Trigger.Puls and Trigger.PULSHIGH renamed to Trigger.Pulse and Trigger.PULSEHIGH
Build 1077 CHG 01106 <trace>.Chart (Display trace contents graphically)	**commands changed** Graphics of Chart commands changed
Build 1077 FIX 01105 FIRE Emulator for H8S and H8/300H	**USB interface during boot** Fire H8S may cause link error or fail on USB interface during boot

Build 1077 IMP 01104 MPC56x NEXUS Debugger and Trace	**High memory accesses** new option SYStem.Option HighMemory for Nexus-56x support high memory accesses
Build 1077 FIX 01103 PowerTrace for NEXUS	NEXUS data message addresses sign extended
Build 1077 IMP 01102	**Drag buttons** Drag buttons have four scrolling speeds
Build 1077 FIX 01101 Data.LOAD (Load file)	**ELF/DWARF** ELF/DWARF loader may crash after load when many complex lifetimes are loaded
Build 1077 FIX 01100 MIPS32/MIPS64 Debugger	**FlowError with Mips Preprocessor** SETUP.IMASKASM / SETUP.IMASKHLL not working on MIPS64
Build 1077 FIX 01099	**Adding licenses** Adding licenses to separate license file may fail when lines follow the LICENSE=<file> line in config.t32
Build 1077 CHG 01098	**Separators in HLL lines** Separators in HLL lines moved to top of HLL block
Build 1077 FIX 01097	**Change of license** Change of license in LICENSE window crashes when opening of license file fails.
Build 1077 FIX 01096	**Handling of prefix character** Handling of prefix character during external/global symbol match wrong, may cause wrong demangling of static members and vtabs
Build 1077 IMP 01095 Data.LOAD (Load file)	**GNU compilers** Better automatic detection of demangling variants for GNU compilers
Build 1077 CHG 01094 Step (Steps through the program)	**Step.XXX restores context** Step.XXX restores context (from R.UP etc.) before executing
Build 1077 FIX 01093 PowerProbe - Logic and Protocol Analyzer	PowerProbe displays wrong records in Stack mode, time correlation lost
Build 1077 FIX 01092 MIPS32/MIPS64 Debugger	Register.RESet not working on MIPS32/MIPS64
Build 1077 FIX 01091 MIPS32/MIPS64 Debugger	**32-bit stack backtrace** Values from 32-bit stack backtrace where not sign extended on MIPS64
Build 1077 FIX 01090 ARM Debugger FIRE Emulator for ARM7 In-Circuit Emulator for ARM7	**Disassembler** condition code in disassembler was missing on some instructions for ARM
Build 1077 FIX 01089 C166 Family Real-Time Trace	**Disassembler** Disassembler was not working for C166 RiscTrace
Build 1077 FIX 01088 ARM/Cortex Trace (parallel)	**ETM counter programming failed** ETM counter programming failed when no RESTART condition was specified

Build 1077 CHG 01087 Data.LOAD.COFF (Load COFF file) Data.LOAD.Elf (Load ELF file)	**COFF/ELF** COFF/ELF option /BUGFIX changed, option /ChainedStab for .stabs chaining
Build 1077 FIX 01086	**Inherited classes** Handling of inherited classes wrong when it contains no visible members
Build 1077 IMP 01085 PowerProbe - Logic and Protocol Analyzer	A.STATE() and A.RECORDS() implemented for PowerProbe
Build 1077 FIX 01084	**MACRO contents** MACRO contents length was limited to 256 characters (now 4K)
Build 1077 FIX 01083	**Expressions** expressions in the form "reference_variable.class::member" fail
Build 1077 FIX 01082 FIRE Portanalyzer FIRE Emulator for ST10	FIRE Port analyzer channels for ST10F27x where those of ST10F280
Build 1077 CHG 01081	**Renamed** Commands PULSe.Puls, PULSe2.Puls renamed to Pulse.Pulse, Pulse2.Pulse
Build 1077 FIX 01080	**Scalar members** Scalar members of structures are treated as bitfields in stabs debug format for little endian targets (for ARM on Win32 only big endian targets)
Build 1077 FIX 01079	**Modification of 32-bit wide bitfields** Modification of 32-bit wide bitfields fails on Windows host
Build 1077 FIX 01078 MIPS32/MIPS64 Debugger	**IEEE** MIPS5K: 32bit addresses of IEEE files where not sign-extended
Build 1077 FIX 01077 Step.Over (Step over call)	**indirect jump** Step.Over HLL made regular Step when target function was called by indirect jump instruction
Build 1077 FIX 01076	**Caption bar** Caption bar of Data.List window may display "active" when the input focus is already in the command line
Build 1077 FIX 01075	**Display** Displaying an HLL source window shows no PC when the PC is in an assembler section short after the HLL section
Build 1077 FIX 01074 Var.set (Modify variable)	**Clicking on methods** Clicking on methods in Var windows fails
Build 1077 FIX 01073 Var.set (Modify variable)	**Brackets** Brackets may be missing when selecting pointer contents in variables windows
Build 1077 IMP 01072	**Class display extended** Derived class display extended to reference variables

Build 1077 IMP 01071	**Class display extended** Derived class display extended to support IA64 mangling and ARM RVCT 2.0
Build 1077 IMP 01070	**Class method names** Double clicking on class method names takes also classname
Build 1077 FIX 01069	**Windows tracking** Windows tracking to wrong address when selecting functions/variables in Data.List window
Build 1077 FIX 01068 Var.INFO (View information about HLL variable or HLL expression)	window creation size of Var.INFO may be wrong
Build 1077 IMP 01067	**RCL** RCL can support multiple clients
Build 1077 IMP 01066 AREA.Create (Create or modify message area)	**AREA size** AREA size limits extended to 32767 lines and 4096 columns
Build 1077 IMP 01065 Var.LOG (Log variables)	**Extended** Var.LOG max line length extended to 4K
Build 1077 IMP 01064 TERM.Mode (Define terminal type)	TERM.MODE HEX implemented
Build 1077 CHG 01063 Super10 NEXUS Debugger and Trace	Disassembler / cycle info C166 changed to support Super10

Sorted by Commands

<trace>.Arm

Build 9640 FIX 02037 PowerIntegrator - Logic and Bus Analyzer <trace>.Arm (Arm the trace) <trace>.Init (Initialize trace)	**no sampling at all** under circumstances the PowerIntergrator sampled nothing at all

<trace>.AutoFocus

Build 19417 IMP 02173 ARM/Cortex Trace (parallel) <trace>.AutoFocus (Calibrate AUTOFOCUS preprocessor)	**Autofocus algorithm changed** For high speed traceports and AutoFocus-II preprocessor the old algorithm has choosen the wrong eye (too high delay).
Build 1241 IMP 01786 Adaption for Embedded Trace Macrocell Adaptation for MIPS MPC55xx/MPC56xx NEXUS Debugger and Trace <trace>.AutoFocus (Calibrate AUTOFOCUS preprocessor) <trace>.TestFocus (Test trace port recording)	**New commands to verify trace connection** The Analyzer.AutoTest command verifys the integrity of the trace connection. For adapters with adjustable threshold the command Analyzer.AutoFocus will also choose the threshold in the middle of the data eye.

Build 1077 CHG 01337 <trace>.AutoInit (Automatic initialization)	A.AutoInit does not clear trace at spotpoints
Build 1077 FIX 01153 FIRE Fully Integrated RISC Emulator ICE In-Circuit Emulator PowerIntegrator - Logic and Bus Analyzer PowerTrace PowerProbe - Logic and Protocol Analyzer <trace>.AutoInit (Automatic initialization)	Automatic init of analyzers (Integrator,PowerTrace,FIRE,ICE) changed

<trace>.Chart

Build 130863 IMP 02607 <trace>.Chart (Display trace contents graphically) <trace>.STATistic (Statistic analysis)	**new option /MACHINE for command Trace commands** New option /MACHINE for Trace.STATistic and Trace.Chart commands.
Build 93173 IMP 02513 Integrated Development Environment <trace>.Chart (Display trace contents graphically) <trace>.Chart.TASK (Task activity chart) <trace>.Chart.sYmbol (Symbol analysis) <trace>.Timing (Waveform of trace buffer)	**Mouse interface changed for Trace.Chart/Timing windows** Zooming into the selected area is now done via double-click with the left mouse button. Drag and drop of the cursor to another trace window aligns this window to the same cursor position. Drag and drop of the selected area to another trace window with graphical display aligns this window to the same area.
Build 42354 IMP 02336 <trace>.Chart (Display trace contents graphically)	**Drag & Drop in Trace.Chart windows** Drag & Drop of the symbol name can be used to rearrange the window. Drag & Drop of the symbol to the upper window header removes the symbol from the custom placed symbols.
Build 2940 IMP 01902 <trace>.Chart (Display trace contents graphically) <trace>.DRAW (Plot trace data against time) <trace>.Timing (Waveform of trace buffer)	**New option /ZoomTrack for timing/chart** Allows to track position and zoom factor between timing and chart windows.
Build 1077 CHG 01106 <trace>.Chart (Display trace contents graphically)	**commands changed** Graphics of Chart commands changed

<trace>.Chart.DatasYmbol

Build 4928 IMP 01986 <trace>.Chart.DatasYmbol (Analyze pointer contents graphically) <trace>.STATistic.DatasYmbol (Analyze pointer contents numerically)	**New commands for symbolic data statistic** Trace.STAT.DatasYmbol provides symbolic statistics for data similar like Trace.STAT.sYmbol does for addresses.

<trace>.Chart.GROUP

Build 88288 02496 Trace Analyzer <trace>.Chart.DatasYmbol (Analyze pointer contents graphically) <trace>.Chart.GROUP (Group activity chart) <trace>.Chart.TASK (Task activity chart) <trace>.Chart.sYmbol (Symbol analysis) <trace>.Timing (Waveform of trace buffer)	**Chart and timing windows with Track option keep zoom** The zoom factor of the <trace>.Chart and <trace>.Timing windows is retained, even if the trace content changes, if the /Track option is used.
Build 56057 IMP 02406 <trace>.Chart.GROUP (Group activity chart) <trace>.Chart.Line (Graphical HLL lines analysis) <trace>.Chart.sYmbol (Symbol analysis) <trace>.STATistic.Address-GROUP (Address group run-time analysis) <trace>.STATistic.GROUP (Group run-time analysis) <trace>.STATistic.Line (High-level source code line analysis) <trace>.STATistic.sYmbol (Flat run-time analysis)	**Task specific flat statistics and charts** New options "TASK" and "SplitTASK" allows to produce task specific flat trace statistics and charts.

<trace>.Chart.Line

Build 56057 IMP 02406 <trace>.Chart.GROUP (Group activity chart) <trace>.Chart.Line (Graphical HLL lines analysis) <trace>.Chart.sYmbol (Symbol analysis) <trace>.STATistic.Address-GROUP (Address group run-time analysis) <trace>.STATistic.GROUP (Group run-time analysis) <trace>.STATistic.Line (High-level source code line analysis) <trace>.STATistic.sYmbol (Flat run-time analysis)	**Task specific flat statistics and charts** New options "TASK" and "SplitTASK" allows to produce task specific flat trace statistics and charts.

<trace>.Chart.RUNNABLE

Build 105499 IMP 02553 Trace Analyzer <trace>.Chart.RUNNABLE (Runnable activity chart)	**New commands to analyze AUTOSAR runnable trace information**

<trace>.Chart.TASK

Build 93173 IMP 02513 Integrated Development Environment <trace>.Chart (Display trace contents graphically) <trace>.Chart.TASK (Task activity chart) <trace>.Chart.sYmbol (Symbol analysis) <trace>.Timing (Waveform of trace buffer)	**Mouse interface changed for Trace.Chart/Timing windows** Zooming into the selected area is now done via double-click with the left mouse button. Drag and drop of the cursor to another trace window aligns this window to the same cursor position. Drag and drop of the selected area to another trace window with graphical display aligns this window to the same area.
Build 88288 02496 Trace Analyzer <trace>.Chart.DatasYmbol (Analyze pointer contents graphically) <trace>.Chart.GROUP (Group activity chart) <trace>.Chart.TASK (Task activity chart) <trace>.Chart.sYmbol (Symbol analysis) <trace>.Timing (Waveform of trace buffer)	**Chart and timing windows with Track option keep zoom** The zoom factor of the <trace>.Chart and <trace>.Timing windows is retained, even if the trace content changes, if the /Track option is used.

<trace>.Chart.Var

Build 9640 IMP 02054 <trace>.Chart.Var (Variable chart) <trace>.STATistic.Var (Statistic of variable accesses)	**New variable access statistics** Shows statistics simmilar to Trace.STATistic.sYmbol, but for data access cycles.

Build 1133 FIX 01473 <trace>.Chart.VarState (Variable activity chart) <trace>.STATistic.Sort (Specify sorting criteria for statistic commands)	A.Chart.VarState display was wrong when A.STAT.Sort was not set to OFF

Build 93173 IMP 02513 Integrated Development Environment <trace>.Chart (Display trace contents graphically) <trace>.Chart.TASK (Task activity chart) <trace>.Chart.sYmbol (Symbol analysis) <trace>.Timing (Waveform of trace buffer)	**Mouse interface changed for Trace.Chart/Timing windows** Zooming into the selected area is now done via double-click with the left mouse button. Drag and drop of the cursor to another trace window aligns this window to the same cursor position. Drag and drop of the selected area to another trace window with graphical display aligns this window to the same area.
Build 88288 02496 Trace Analyzer <trace>.Chart.DatasYmbol (Analyze pointer contents graphically) <trace>.Chart.GROUP (Group activity chart) <trace>.Chart.TASK (Task activity chart) <trace>.Chart.sYmbol (Symbol analysis) <trace>.Timing (Waveform of trace buffer)	**Chart and timing windows with Track option keep zoom** The zoom factor of the <trace>.Chart and <trace>.Timing windows is retained, even if the trace content changes, if the /Track option is used.
Build 56057 IMP 02406 <trace>.Chart.GROUP (Group activity chart) <trace>.Chart.Line (Graphical HLL lines analysis) <trace>.Chart.sYmbol (Symbol analysis) <trace>.STATistic.Address-GROUP (Address group run-time analysis) <trace>.STATistic.GROUP (Group run-time analysis) <trace>.STATistic.Line (High-level source code line analysis) <trace>.STATistic.sYmbol (Flat run-time analysis)	**Task specific flat statistics and charts** New options "TASK" and "SplitTASK" allows to produce task specific flat trace statistics and charts.
Build 51144 IMP 02395 <trace>.Chart.sYmbol (Symbol analysis) <trace>.STATistic.sYmbol (Flat run-time analysis)	**Inline function support for Trace.STAT.sYmbol commands** Inline functions are treated as separate functions. The option NoInline can be used to discard inline functions.

Build 2940 IMP 01902 <trace>.Chart (Display trace contents graphically) <trace>.DRAW (Plot trace data against time) <trace>.Timing (Waveform of trace buffer)	**New option /ZoomTrack for timing/chart** Allows to track position and zoom factor between timing and chart windows.
Build 1270 IMP 01872 Data.DRAW (Graphical memory display of arrays) <trace>.DRAW (Plot trace data against time)	**Floating point formats allowed** Floating point formats now allowed for low level graphical displays.

Build 93173 IMP 02514 Integrated Development Environment <trace>.DRAW.channel (Plot no-data values against time) <trace>.DRAW.Data (Plot data values against time) <trace>.DRAW.Var (Plot variable values against time)	**Mouse interface changed for Trace.Draw command group** Zooming into the selected area is now done via double-click with the left mouse button. Free selection is now possible in the draw area. Value range or time/record range selection is possible via the scale area. Drag and drop of the selected area to another trace window with graphical display aligns this window to the same area.
Build 69655 IMP 02442 High-Level-Language Debugging Trace Features <trace>.DRAW.channel (Plot no-data values against time) <trace>.DRAW.Data (Plot data values against time) <trace>.DRAW.Var (Plot variable values against time) Data.DRAW (Graphical memory display of arrays) Data.PROfile (Graphical display of data value) Var.PROfile (Graphical display of variable)	**Multiple variables of different type can be drawn** TRACE32 allows to draw multiple variables/channels with different formats e.g. integers and floats for all listed commands.
Build 42354 IMP 02334 <trace>.DRAW.Data (Plot data values against time) <trace>.DRAW.Var (Plot variable values against time)	**New Trace Draw commands** Trace.DRAW.Data can draw any memory access, Trace.DRAW.Var can draw any static variable in the trace buffer.

Build 93173 IMP 02514 Integrated Development Environment <trace>.DRAW.channel (Plot no-data values against time) <trace>.DRAW.Data (Plot data values against time) <trace>.DRAW.Var (Plot variable values against time)	**Mouse interface changed for Trace.Draw command group** Zooming into the selected area is now done via double-click with the left mouse button. Free selection is now possible in the draw area. Value range or time/record range selection is possible via the scale area. Drag and drop of the selected area to another trace window with graphical display aligns this window to the same area.
Build 69655 IMP 02442 High-Level-Language Debugging Trace Features <trace>.DRAW.channel (Plot no-data values against time) <trace>.DRAW.Data (Plot data values against time) <trace>.DRAW.Var (Plot variable values against time) Data.DRAW (Graphical memory display of arrays) Data.PROfile (Graphical display of data value) Var.PROfile (Graphical display of variable)	**Multiple variables of different type can be drawn** TRACE32 allows to draw multiple variables/channels with different formats e.g. integers and floats for all listed commands.
Build 42354 IMP 02334 <trace>.DRAW.Data (Plot data values against time) <trace>.DRAW.Var (Plot variable values against time)	**New Trace Draw commands** Trace.DRAW.Data can draw any memory access, Trace.DRAW.Var can draw any static variable in the trace buffer.

<trace>.DRAW.channel

Build 93173 IMP 02514 Integrated Development Environment <trace>.DRAW.channel (Plot no-data values against time) <trace>.DRAW.Data (Plot data values against time) <trace>.DRAW.Var (Plot variable values against time)	**Mouse interface changed for Trace.Draw command group** Zooming into the selected area is now done via double-click with the left mouse button. Free selection is now possible in the draw area. Value range or time/record range selection is possible via the scale area. Drag and drop of the selected area to another trace window with graphical display aligns this window to the same area.
Build 69655 IMP 02442 High-Level-Language Debugging Trace Features <trace>.DRAW.channel (Plot no-data values against time) <trace>.DRAW.Data (Plot data values against time) <trace>.DRAW.Var (Plot variable values against time) Data.DRAW (Graphical memory display of arrays) Data.PROfile (Graphical display of data value) Var.PROfile (Graphical display of variable)	**Multiple variables of different type can be drawn** TRACE32 allows to draw multiple variables/channels with different formats e.g. integers and floats for all listed commands.

<trace>.EXPORT

Build 19417 IMP 02137 <trace>.EXPORT (Export trace data for processing in other applications)	**Trace.EXPORT command enhanced** New options to support various compressed binary formats.

<trace>.EXPORT.CSVFunc

Build 56057 IMP 02410 Trace Analyzer <trace>.EXPORT.CSVFunc (Export the function nesting to a CSV file) <trace>.EXPORT.TASKEVENTS (Export task event to CSV)	**New Trace export commands** New commands to export trace information for analysis by a third-party tool.

<trace>.EXPORT.TASKEVENTS

Build 56057 IMP 02410 Trace Analyzer <trace>.EXPORT.CSVFunc (Export the function nesting to a CSV file) <trace>.EXPORT.TASKEVENTS (Export task event to CSV)	**New Trace export commands** New commands to export trace information for analysis by a third-party tool.

<trace>.EXPORT.TracePort

Build 130863 IMP 02614 <trace>.EXPORT.TracePort (Export trace packets as recorded at trace port)	**new option for command CoreSightTrace.EXPORT.TracePort** New option /CoreSightByteStream for the command CoreSightTrace.EXPORT.TracePort to export single CoreSight stream.

<trace>.EXPORT.cycles

Build 125398 CHG 02591 Trace Analyzer <trace>.EXPORT.cycles (Export trace data) LA.IMPORT.cycles (Import bus trace data)	**renamed commands Trace.EXPORT.flow and LA.IMPORT.flow** The commands Trace.EXPORT.flow and LA.IMPORT.flow have been renamed to Trace.EXPORT.cycles and LA.IMPORT.cycles.

<trace>.FILE

Build 8248 CHG 02018 <trace>.FILE (Load a file into the file trace buffer) <trace>.LOAD (Load trace file for offline processing)	**Trace.LOAD functionality changed** The loaded trace data is now shown without the need for the /FILE option in place of the real traced data. The old functionality is available with the Trace.FILE command (useful for comparing saved traces with actual traces).

<trace>.Find

Build 125398 IMP 02594 Trace Analyzer <trace>.Find (Find specified entry in trace) <trace>.FindAll (Find all specified entries in trace) <trace>.List (List trace contents) <trace>.ListVar (List variable recorded to trace)	**new option /TASK for Trace commands** new option /TASK for Trace.List, Trace.Find, Trace.FindAll and Trace.ListVar to filter for tasks.
Build 69655 IMP 02448 Trace Analyzer Data.Find (Search in memory) Data.FindCODE (Execute command on specified code type) <trace>.Find (Find specified entry in trace)	**New option /ALL for Trace.Find command** With the option /ALL the TRACE32 Message Line displays how often the searched item was found. The number of occurrences can be processed in a script via the function FOUND.COUNT().
Build 1241 FIX 01793 <trace>.Find (Find specified entry in trace) <trace>.FindAll (Find all specified entries in trace)	**Trace.Find for segmented addresses fails** Searching for segmented addresses in trace (e.g. 80x86) does not find matches.
Build 1206 CHG 01733 <trace>.Find (Find specified entry in trace) <trace>.FindAll (Find all specified entries in trace)	**Trace.Find Data changed** New default Data width selector "ANY" introduced (default width). Will match any data width.
Build 1133 FIX 01463 <trace>.Find (Find specified entry in trace)	Trace.Find WORD.xxx <value|range> was not working on debugger

<trace>.FindAll

Build 125398 IMP 02594 Trace Analyzer <trace>.Find (Find specified entry in trace) <trace>.FindAll (Find all specified entries in trace) <trace>.List (List trace contents) <trace>.ListVar (List variable recorded to trace)	**new option /TASK for Trace commands** new option /TASK for Trace.List, Trace.Find, Trace.FindAll and Trace.ListVar to filter for tasks.
Build 1241 FIX 01793 <trace>.Find (Find specified entry in trace) <trace>.FindAll (Find all specified entries in trace)	**Trace.Find for segmented addresses fails** Searching for segmented addresses in trace (e.g. 80x86) does not find matches.
Build 1206 CHG 01733 <trace>.Find (Find specified entry in trace) <trace>.FindAll (Find all specified entries in trace)	**Trace.Find Data changed** New default Data width selector "ANY" introduced (default width). Will match any data width.
Build 1077 FIX 01216 <trace>.FindAll (Find all specified entries in trace)	**"AT" item** Trace.FindAll was not working with "AT" item

<trace>.Init

Build 9640 FIX 02037 PowerIntegrator - Logic and Bus Analyzer <trace>.Arm (Arm the trace) <trace>.Init (Initialize trace)	**no sampling at all** under circumstances the PowerIntergrator sampled nothing at all

<trace>.LOAD

Build 1253 IMP 01844 <trace>.LOAD (Load trace file for offline processing)	**Probe.LOAD can also recover setti** The option /Config recovers also the configuration of the trace and the channel names.

Build 65657 02435 ARM/Cortex Trace (parallel) MPC55xx/MPC56xx NEXUS Debugger and Trace <trace>.List (List trace contents) <trace>.STATistic.PAddress (Which instructions accessed data address) <trace>.STATistic.PsYmbol (Shows which functions accessed data address)	**Prestore address information for ARM-ETM/NEXUS** The columns PAddress/PsYmbol in the trace listing show the address of the instruction that was executed before a read/write access was performed. The statistic function Trace.STATistic.PsYmbol analyses which functions performed read/write accesses. The statistic function Trace.STATistic.PAddress analyses which assembler instruction performed the read/write accesses.
Build 2940 FIX 01876 MPC55xx/MPC56xx NEXUS Debugger and Trace <trace>.List (List trace contents)	**eTPU trace may produce flow errors** Occurs when long traces from other cores are also in the trace or at certain slot instruction combinations during the end of a channel program.
Build 1246 FIX 01835 MAC71xx/72xx NEXUS Debugger and Trace <trace>.List (List trace contents)	**Traced flow incorrect on MAC7xxx** One instruction too much displayed before interrupt entry.
Build 1161 FIX 01616 <trace>.List (List trace contents)	**CPx data cycles in ETM trace shown wrong** Accesses to coprocessor registers are shown with the wrong cycle type.
Build 1133 FIX 01525 <trace>.List (List trace contents) <trace>.View (Display single record)	A.List/View /FILE may crash when MAP.DMUX/MAP.ADELAY is active
Build 1079 FIX 01387 PowerTrace for NEXUS <trace>.List (List trace contents)	A.L NEXUS may not show last nexus message of trace window

<trace>.ListNesting

Build 34458 IMP 02283 CTS.List (List trace contents) <trace>.List (List trace contents) <trace>.ListNesting (Analyze function nesting)	**New display format with SplitCORE option** Trace is shown separated for the cores.
Build 30461 IMP 02232 CTS.List (List trace contents) <trace>.ListNesting (Analyze function nesting) <trace>.STATistic.TREE (Tree display of nesting function run-time analysis)	**Nesting analysis made error tolerant** The analysis will try to detect and fix problems in the function nesting caused by special target code or traceport FIFO overflows. The fixes and detected problems are be marked in the nested trace listing (Trace.ListNesting or CTS.List).
Build 19417 IMP 02147 CTS.ListNesting (Analyze function nesting) <trace>.ListNesting (Analyze function nesting)	**New command Trace.ListNesting** Command shows execution flow similar to CTS.List window.

<trace>.ListVar

Build 125398 IMP 02594 Trace Analyzer <trace>.Find (Find specified entry in trace) <trace>.FindAll (Find all specified entries in trace) <trace>.List (List trace contents) <trace>.ListVar (List variable recorded to trace)	**new option /TASK for Trace commands** new option /TASK for Trace.List, Trace.Find, Trace.FindAll and Trace.ListVar to filter for tasks.

<trace>.PROTOcol.List

Build 26464 IMP 02222 <trace>.PROTOcol.List (Display trace buffer for user-defined protocol)	**Protocol analysis API improved** Extended API allows handling of larger traces, faster processing and sharing of processed data among different protocol windows.

<trace>.PROfileChart

Build 30461 IMP 02244 <trace>.PROfileChart (Profile charts)	**New PROfileChart window coloring** Names get now a fixed colors based on a hash of the name. Old (dynamic) coloring still available with the "/Color AlternatingColors" option.
Build 8248 IMP 02020 CTS.PROfileChart (Profile charts) <trace>.PROfileChart (Profile charts)	**New command Trace.PROfileChart** Allows to display distributions vs. time graphically. Useful to show processing ratio of selected tasks versus time.

<trace>.PROfileChart.COUNTER

Build 30461 IMP 02233 <trace>.PROfileChart.COUNTER (Display a profile chart)	**New command Trace.PROfileChart.COUNTER** Allows to show time profiles of a counter that is traced as data value.

<trace>.PROfileChart.GROUP

Build 22490 IMP 02196 <trace>.PROfileChart.GROUP (Group profile chart)	**New option Interval for profile charts** The InterVal option allows to specify either a fixed time or a tracebuffer filter expression to specify the time slices of most profile chart windows.

<trace>.PROfileSTATistic

Build 26464 IMP 02226 <trace>.PROfileSTATistic (Statistical analysis in a table versus time)	**New command group Trace.PROfileSTATistic** Shows results of numerical interval analysis in tabular format.

<trace>.SAVE

Build 38589 IMP 02303 <trace>.SAVE (Save trace for postprocessing in TRACE32) <trace>.SAVE (Save trace for postprocessing in TRACE32)	**Trace save format changed** Now uses the new compression format by default. This provides a higher compression than the previous default format.
Build 1077 IMP 01278 PowerTrace <trace>.SAVE (Save trace for postprocessing in TRACE32)	**A.SAVE changed** A.SAVE changed to a compressed format for PowerTrace

<trace>.SIZE

Build 2940 FIX 01898 <trace>.SIZE (Define buffer size)	**Tracebuffer too large for RISCTRACE** Was causing illegal trace entry when tracebuffer is full.

<trace>.STATistic

Build 13751 IMP 02096 <trace>.STATistic (Statistic analysis)	**New columns for Trace.STAT commands** "TASK" shows task number. "CORE" shows core number (SMP). The old "TASK" column is renamed to "ExternalTASK".
Build 1133 FIX 01526 <trace>.Chart (Display trace contents graphically) <trace>.STATistic (Statistic analysis)	A.Chart/A.STAT may crash with long symbol/function names
Build 1133 FIX 01511 <trace>.STATistic (Statistic analysis)	Bug fixed in a.stat commands - may crash with long symbols names
Build 1077 IMP 01361 <trace>.STATistic (Statistic analysis)	A.STAT max. stack nesting increased to 200

<trace>.STATistic.Address

Build 4790 IMP 01984 <trace>.STATistic.Address (Time between up to 8 program events)	**New command for address hit statistic**

<trace>.STATistic.AddressDIStance

Build 4790 IMP 01985 <trace>.STATistic.Address-DIStance (Time interval for single program event) <trace>.STATistic.Address-DURation (Time between two program events)	**New commands for address statistic**

<trace>.STATistic.AddressDURation

Build 4790 IMP 01985 <trace>.STATistic.Address-DIStance (Time interval for single program event) <trace>.STATistic.Address-DURation (Time between two program events)	**New commands for address statistic**

<trace>.STATistic.AddressGROUP

Build 56057 IMP 02406 <trace>.Chart.GROUP (Group activity chart) <trace>.Chart.Line (Graphical HLL lines analysis) <trace>.Chart.sYmbol (Symbol analysis) <trace>.STATistic.AddressGROUP (Address group run-time analysis) <trace>.STATistic.GROUP (Group run-time analysis) <trace>.STATistic.Line (High-level source code line analysis) <trace>.STATistic.sYmbol (Flat run-time analysis)	**Task specific flat statistics and charts** New options "TASK" and "SplitTASK" allows to produce task specific flat trace statistics and charts.
Build 22490 CHG 02199 <trace>.STATistic.Address (Time between up to 8 program events) <trace>.STATistic.AddressGROUP (Address group run-time analysis) <trace>.STATistic.DistriB (Distribution analysis) <trace>.STATistic.GROUP (Group run-time analysis) <trace>.STATistic.Line (High-level source code line analysis) <trace>.STATistic.Var (Statistic of variable accesses) <trace>.STATistic.sYmbol (Flat run-time analysis)	**New options to change count behavior** CountChange counts all changes, CountFirst counts just the first location and CountALL counts each event. The defaults for the commands have been changed to provide useful results.
Build 19417 IMP 02168 <trace>.STATistic.AddressGROUP (Address group run-time analysis)	**New commands for address group stats** Trace.STATistic.AddressGROUP and related chart commands can analyze the number of accesses to certain memory regions.

<trace>.STATistic.CYcle

Build 19417 IMP 02142 <trace>.STATistic.CYcle (Analyze cycle types)	**Trace.STAT.CYcle command enhanced** Shows more information about execution times, idles and trace gaps.

<trace>.STATistic.ChildTREE

Build 22490 IMP 02210 <trace>.STATistic.ChildTREE (Show callee context of a function)	**New command Trace.STATistic.ChildTREE** Allows to show all children of a function as a tree.

<trace>.STATistic.DIStance

Build 1077 FIX 01138 <trace>.STATistic.DIStance (Time interval for a single event)	analyzer distribution statistic has wrong total time

<trace>.STATistic.DatasYmbol

Build 4928 IMP 01986 <trace>.Chart.DatasYmbol (Analyze pointer contents graphically) <trace>.STATistic.DatasYmbol (Analyze pointer contents numerically)	**New commands for symbolic data statistic** Trace.STAT.DatasYmbol provides symbolic statistics for data similar like Trace.STAT.sYmbol does for addresses.

<trace>.STATistic.DistriB

Build 51144 IMP 02389 <trace>.STATistic.DistriB (Distribution analysis) <trace>.STATistic.GROUP (Group run-time analysis) <trace>.STATistic.Line (High-level source code line analysis) <trace>.STATistic.MODULE (Code execution broken down by module) <trace>.STATistic.sYmbol (Flat run-time analysis)	**Option SplitTASK for many Trace.STATistic commands** Allows to split up the results for different tasks.
Build 22490 CHG 02199 <trace>.STATistic.Address (Time between up to 8 program events) <trace>.STATistic.AddressGROUP (Address group run-time analysis) <trace>.STATistic.DistriB (Distribution analysis) <trace>.STATistic.GROUP (Group run-time analysis) <trace>.STATistic.Line (High-level source code line analysis) <trace>.STATistic.Var (Statistic of variable accesses) <trace>.STATistic.sYmbol (Flat run-time analysis)	**New options to change count behavior** CountChange counts all changes, CountFirst counts just the first location and CountALL counts each event. The defaults for the commands have been changed to provide useful results.

<trace>.STATistic.FIRST

Build 9640 IMP 02042 <trace>.STATistic.FIRST (Start point for statistic analysis) <trace>.STATistic.LAST (End point for statistic analysis)	**New commands to limit trace statistics** Allows to limit numerical statistical analysis to a smaller part of the trace.

Build 38589 IMP 02304 <trace>.STATistic.Func (Nesting function runtime analysis) <trace>.STATistic.TREE (Tree display of nesting function run-time analysis)	**Interrupts in nested trace statistics** Interrupts in nested trace statistic windows are now shown under a separate "pseudo" task named "(interrupt)".
Build 22490 IMP 02209 <trace>.STATistic.Func (Nesting function runtime analysis) <trace>.STATistic.TREE (Tree display of nesting function run-time analysis)	**New options for Trace.STAT.TREE** Option /TASK allows a task selective display of information. With /MergeTASK it is possible to "overlay" the information of multiple tasks. The option SplitTASK splits the infomation of different tasks (the default).
Build 9640 IMP 02050 <trace>.STATistic.Func (Nesting function runtime analysis) <trace>.STATistic.TASKFunc (Task related function run-time analysis) <trace>.STATistic.TASKTREE (Tree display of task specific functions) <trace>.STATistic.TREE (Tree display of nesting function run-time analysis)	**New options for function statistics** New options /IncludeOwn, /IncludeTASK /IncludeINTR change the way times for interrupts and other tasks are accumulated in the analysis. Default display is now to exclude times of interrupts and other tasks. With /INTRROOT and /INTRTASK it is possible to split up the interrupt times to the different tasks or count them globaly.
Build 4728 FIX 01922 GROUP.HIDE (Hide group from debugging) <trace>.STATistic.Func (Nesting function runtime analysis)	**Function statistics with disabled groups** Calls inside the group have caused wrong internal/external times and calling counts.
Build 2940 IMP 01892 <trace>.STATistic.Func (Nesting function runtime analysis) <trace>.STATistic.TREE (Tree display of nesting function run-time analysis)	**New "Own" items in Trace.STAT commands** "Own" time is the time of function with all called functions, but without the time spend in interrupts or other tasks.

<trace>.STATistic.FuncDURation

Build 80996 IMP 02469 Trace-based Profiling <trace>.STATistic.FuncDURation (Statistic analysis of single function) <trace>.STATistic.FuncDURationInternal (Statistic analysis of single func.)	**New commands to analyse run-time of a single function** Detailed analysis of a single function. Trace.STATistic.FuncDURation shows time between function entry and exit without time is interrupts and other tasks. Trace.STATistic.FuncDURationInternal show net time of function.

<trace>.STATistic.FuncDURationInternal

Build 80996 IMP 02469 Trace-based Profiling <trace>.STATistic.FuncDURation (Statistic analysis of single function) <trace>.STATistic.FuncDURationInternal (Statistic analysis of single func.)	**New commands to analyse run-time of a single function** Detailed analysis of a single function. Trace.STATistic.FuncDURation shows time between function entry and exit without time is interrupts and other tasks. Trace.STATistic.FuncDURationInternal show net time of function.

Build 56057 IMP 02406 <trace>.Chart.GROUP (Group activity chart) <trace>.Chart.Line (Graphical HLL lines analysis) <trace>.Chart.sYmbol (Symbol analysis) <trace>.STATistic.Address-GROUP (Address group run-time analysis) <trace>.STATistic.GROUP (Group run-time analysis) <trace>.STATistic.Line (High-level source code line analysis) <trace>.STATistic.sYmbol (Flat run-time analysis)	**Task specific flat statistics and charts** New options "TASK" and "SplitTASK" allows to produce task specific flat trace statistics and charts.
Build 51144 IMP 02389 <trace>.STATistic.DistriB (Distribution analysis) <trace>.STATistic.GROUP (Group run-time analysis) <trace>.STATistic.Line (High-level source code line analysis) <trace>.STATistic.MODULE (Code execution broken down by module) <trace>.STATistic.sYmbol (Flat run-time analysis)	**Option SplitTASK for many Trace.STATistic commands** Allows to split up the results for different tasks.
Build 22490 CHG 02199 <trace>.STATistic.Address (Time between up to 8 program events) <trace>.STATistic.Address-GROUP (Address group run-time analysis) <trace>.STATistic.DistriB (Distribution analysis) <trace>.STATistic.GROUP (Group run-time analysis) <trace>.STATistic.Line (High-level source code line analysis) <trace>.STATistic.Var (Statistic of variable accesses) <trace>.STATistic.sYmbol (Flat run-time analysis)	**New options to change count behavior** CountChange counts all changes, CountFirst counts just the first location and CountALL counts each event. The defaults for the commands have been changed to provide useful results.

Build 22490 IMP 02197 <trace>.STATistic.GROUP (Group run-time analysis) <trace>.STATistic.TASK (Task activity statistic) <trace>.STATistic.sYmbol (Flat run-time analysis)	**New option Interval for statistics** The InterVal option allows to specify either a fixed time or a tracebuffer filter expression to specify the time slices. The statistic shows the minimum and maximum times and ratios for the events in a slice.

<trace>.STATistic.INTERRUPT

Build 60219 02418 Trace-based Profiling <trace>.STATistic.INTERRUPT (Interrupt statistic) <trace>.STATistic.TASKORINTERRUPT (Statistic of interrupts and tasks) <trace>.STATistic.TASKVSINTERRUPT (Statistic of interrupts, task-related)	**Improved analysis of interrupt timing** Trace.STATistic.INTERRUPT - analyses timing of interrupts Trace.STATistic.TASKORINTERRUPT - analyses timing of tasks and interrupts Trace.STATistic.TASKVSINTERRUPT - analyses timing of tasks and interrupts, interrupts are analyzed in task context

<trace>.STATistic.LAST

Build 9640 IMP 02042 <trace>.STATistic.FIRST (Start point for statistic analysis) <trace>.STATistic.LAST (End point for statistic analysis)	**New commands to limit trace statistics** Allows to limit numerical statistical analysis to a smaller part of the trace.

Build 1190 FIX 01651 <trace>.STATistic.LINKage (Per caller statistic of function) <trace>.STATistic.TASK (Task activity statistic) <trace>.STATistic.TASKFunc (Task related function run-time analysis) <trace>.STATistic.TASKState (Performance analysis) <trace>.STATistic.TASKSRV (Analysis of time in OS service routines) <trace>.STATistic.TASKTREE (Tree display of task specific functions)	**some A.STAT commands may be locked** Some A.STAT get locked when using a trace method that supports no timestamps. This lock remains active even when later a trace method with timestamps is selected.

Build 56057 IMP 02406 <trace>.Chart.GROUP (Group activity chart) <trace>.Chart.Line (Graphical HLL lines analysis) <trace>.Chart.sYmbol (Symbol analysis) <trace>.STATistic.AddressGROUP (Address group run-time analysis) <trace>.STATistic.GROUP (Group run-time analysis) <trace>.STATistic.Line (High-level source code line analysis) <trace>.STATistic.sYmbol (Flat run-time analysis)	**Task specific flat statistics and charts** New options "TASK" and "SplitTASK" allows to produce task specific flat trace statistics and charts.
Build 51144 IMP 02389 <trace>.STATistic.DistriB (Distribution analysis) <trace>.STATistic.GROUP (Group run-time analysis) <trace>.STATistic.Line (High-level source code line analysis) <trace>.STATistic.MODULE (Code execution broken down by module) <trace>.STATistic.sYmbol (Flat run-time analysis)	**Option SplitTASK for many Trace.STATistic commands** Allows to split up the results for different tasks.
Build 22490 CHG 02199 <trace>.STATistic.Address (Time between up to 8 program events) <trace>.STATistic.AddressGROUP (Address group run-time analysis) <trace>.STATistic.DistriB (Distribution analysis) <trace>.STATistic.GROUP (Group run-time analysis) <trace>.STATistic.Line (High-level source code line analysis) <trace>.STATistic.Var (Statistic of variable accesses) <trace>.STATistic.sYmbol (Flat run-time analysis)	**New options to change count behavior** CountChange counts all changes, CountFirst counts just the first location and CountALL counts each event. The defaults for the commands have been changed to provide useful results.

<trace>.STATistic.MODULE

Build 51144 IMP 02389 <trace>.STATistic.DistriB (Distribution analysis) <trace>.STATistic.GROUP (Group run-time analysis) <trace>.STATistic.Line (High-level source code line analysis) <trace>.STATistic.MODULE (Code execution broken down by module) <trace>.STATistic.sYmbol (Flat run-time analysis)	**Option SplitTASK for many Trace.STATistic commands** Allows to split up the results for different tasks.
Build 38589 IMP 02305 <trace>.STATistic.MODULE (Code execution broken down by module) <trace>.STATistic.PROGRAM (Code execution broken down by program)	**New flat statistic commands** Trace.STATistic.MODULE and Trace.STATistic.PROGRAM shows which module or program is executing the code.

<trace>.STATistic.Measure

Build 1133 IMP 01502 <trace>.STATistic.Measure (Analyze the performance of a single signal)	New command A.STAT.Measure to get signal statistics

<trace>.STATistic.PAddress

Build 65657 02435 ARM/Cortex Trace (parallel) MPC55xx/MPC56xx NEXUS Debugger and Trace <trace>.List (List trace contents) <trace>.STATistic.PAddress (Which instructions accessed data address) <trace>.STATistic.PsYmbol (Shows which functions accessed data address)	**Prestore address information for ARM-ETM/NEXUS** The columns PAddress/PsYmbol in the trace listing show the address of the instruction that was executed before a read/write access was performed. The statistic function Trace.STATistic.PsYmbol analyses which functions performed read/write accesses. The statistic function Trace.STATistic.PAddress analyses which assembler instruction performed the read/write accesses.

<trace>.STATistic.PROGRAM

Build 38589 IMP 02305 <trace>.STATistic.MODULE (Code execution broken down by module) <trace>.STATistic.PROGRAM (Code execution broken down by program)	**New flat statistic commands** Trace.STATistic.MODULE and Trace.STATistic.PROGRAM shows which module or program is executing the code.

<trace>.STATistic.ParentTREE

Build 22490 IMP 02211 <trace>.STATistic.ParentTREE (Show the call context of a function)	**New command Trace.STATistic.ParentTREE** Allows to show the parents of a function.

<trace>.STATistic.PsYmbol

Build 65657 02435 ARM/Cortex Trace (parallel) MPC55xx/MPC56xx NEXUS Debugger and Trace <trace>.List (List trace contents) <trace>.STATistic.PAddress (Which instructions accessed data address) <trace>.STATistic.PsYmbol (Shows which functions accessed data address)	**Prestore address information for ARM-ETM/NEXUS** The columns PAddress/PsYmbol in the trace listing show the address of the instruction that was executed before a read/write access was performed. The statistic function Trace.STATistic.PsYmbol analyses which functions performed read/write accesses. The statistic function Trace.STATistic.PAddress analyses which assembler instruction performed the read/write accesses.

<trace>.STATistic.Sort

Build 1133 FIX 01473 <trace>.Chart.VarState (Variable activity chart) <trace>.STATistic.Sort (Specify sorting criteria for statistic commands)	A.Chart.VarState display was wrong when A.STAT.Sort was not set to OFF

<trace>.STATistic.TASK

Build 22490 IMP 02197 <trace>.STATistic.GROUP (Group run-time analysis) <trace>.STATistic.TASK (Task activity statistic) <trace>.STATistic.sYmbol (Flat run-time analysis)	**New option Interval for statistics** The InterVal option allows to specify either a fixed time or a tracebuffer filter expression to specify the time slices. The statistic shows the minimum and maximum times and ratios for the events in a slice.

<trace>.STATistic.TASKFunc

Build 1190 FIX 01651 <trace>.STATistic.LINKage (Per caller statistic of function) <trace>.STATistic.TASK (Task activity statistic) <trace>.STATistic.TASKFunc (Task related function run-time analysis) <trace>.STATistic.TASKState (Performance analysis) <trace>.STATistic.TASKSRV (Analysis of time in OS service routines) <trace>.STATistic.TASKTREE (Tree display of task specific functions)	**some A.STAT commands may be locked** Some A.STAT get locked when using a trace method that supports no timestamps. This lock remains active even when later a trace method with timestamps is selected.

<trace>.STATistic.TASKKernel

Build 8248 IMP 02022 <trace>.STATistic.TASKKernel (Task analysis with kernel markers (flat))	**New command Trace.STATistic.TASKKernel** Make a runtime statistic of tasks and kernel runtimes. Kernel entry and exit must be marked with sYmbol.NEW.MARKER commands.

<trace>.STATistic.TASKLOCK

Build 125398 IMP 02584 Trace Analyzer <trace>.STATistic.TASKLOCK (Analyze lock accesses from tasks)	**new command Trace.STATistic.TASKLOCK** New command introduced to analyze lock accesses from tasks in the trace.

<trace>.STATistic.TASKORINTERRUPT

Build 60219 02418 Trace-based Profiling <trace>.STATistic.INTERRUPT (Interrupt statistic) <trace>.STATistic.TASKORINTERRUPT (Statistic of interrupts and tasks) <trace>.STATistic.TASKVSINTERRUPT (Statistic of interrupts, task-related)	**Improved analysis of interrupt timing** Trace.STATistic.INTERRUPT - analyses timing of interrupts Trace.STATistic.TASKORINTERRUPT - analyses timing of tasks and interrupts Trace.STATistic.TASKVSINTERRUPT - analyses timing of tasks and interrupts, interrupts are analyzed in task context

<trace>.STATistic.TASKORINTRState

Build 117056 IMP 02577 Trace Analyzer <trace>.STATistic.TASKORINTRState (Task and ISR2 statistic analysis)	**new trace commands** New commands Trace.STATistic.TASKORINTRState and Trace.Chart.TASKORINTRState introduced to analyze tasks and (ORTI) interrupts.

<trace>.STATistic.TASKSRV

Build 1190 FIX 01651 <trace>.STATistic.LINKage (Per caller statistic of function) <trace>.STATistic.TASK (Task activity statistic) <trace>.STATistic.TASKFunc (Task related function run-time analysis) <trace>.STATistic.TASKState (Performance analysis) <trace>.STATistic.TASKSRV (Analysis of time in OS service routines) <trace>.STATistic.TASKTREE (Tree display of task specific functions)	**some A.STAT commands may be locked** Some A.STAT get locked when using a trace method that supports no timestamps. This lock remains active even when later a trace method with timestamps is selected.

<trace>.STATistic.TASKState

Build 1190 FIX 01651 <trace>.STATistic.LINKage (Per caller statistic of function) <trace>.STATistic.TASK (Task activity statistic) <trace>.STATistic.TASKFunc (Task related function run-time analysis) <trace>.STATistic.TASKState (Performance analysis) <trace>.STATistic.TASKSRV (Analysis of time in OS service routines) <trace>.STATistic.TASKTREE (Tree display of task specific functions)	**some A.STAT commands may be locked** Some A.STAT get locked when using a trace method that supports no timestamps. This lock remains active even when later a trace method with timestamps is selected.

<trace>.STATistic.TASKTREE

Build 9640 IMP 02050 <trace>.STATistic.Func (Nesting function runtime analysis) <trace>.STATistic.TASKFunc (Task related function run-time analysis) <trace>.STATistic.TASKTREE (Tree display of task specific functions) <trace>.STATistic.TREE (Tree display of nesting function run-time analysis)	**New options for function statistics** New options /IncludeOwn, /IncludeTASK /IncludeINTR change the way times for interrupts and other tasks are accumulated in the analysis. Default display is now to exclude times of interrupts and other tasks. With /INTRROOT and /INTRTASK it is possible to split up the interrupt times to the different tasks or count them globaly.

<trace>.STATistic.TASKVSINTERRUPT

Build 60219 02418 Trace-based Profiling <trace>.STATistic.INTERRUPT (Interrupt statistic) <trace>.STATistic.TASKORINTERRUPT (Statistic of interrupts and tasks) <trace>.STATistic.TASKVSINTERRUPT (Statistic of interrupts, task-related)	**Improved analysis of interrupt timing** Trace.STATistic.INTERRUPT - analyses timing of interrupts Trace.STATistic.TASKORINTERRUPT - analyses timing of tasks and interrupts Trace.STATistic.TASKVSINTERRUPT - analyses timing of tasks and interrupts, interrupts are analyzed in task context

Build 38589 IMP 02304 <trace>.STATistic.Func (Nesting function runtime analysis) <trace>.STATistic.TREE (Tree display of nesting function run-time analysis)	**Interrupts in nested trace statistics** Interrupts in nested trace statistic windows are now shown under a separate "pseudo" task named "(interrupt)".
Build 30461 IMP 02232 CTS.List (List trace contents) <trace>.ListNesting (Analyze function nesting) <trace>.STATistic.TREE (Tree display of nesting function run-time analysis)	**Nesting analysis made error tolerant** The analysis will try to detect and fix problems in the function nesting caused by special target code or traceport FIFO overflows. The fixes and detected problems are be marked in the nested trace listing (Trace.ListNesting or CTS.List).
Build 22490 IMP 02209 <trace>.STATistic.Func (Nesting function runtime analysis) <trace>.STATistic.TREE (Tree display of nesting function run-time analysis)	**New options for Trace.STAT.TREE** Option /TASK allows a task selective display of information. With /MergeTASK it is possible to "overlay" the information of multiple tasks. The option SplitTASK splits the infomation of different tasks (the default).
Build 9640 IMP 02050 <trace>.STATistic.Func (Nesting function runtime analysis) <trace>.STATistic.TASKFunc (Task related function run-time analysis) <trace>.STATistic.TASKTREE (Tree display of task specific functions) <trace>.STATistic.TREE (Tree display of nesting function run-time analysis)	**New options for function statistics** New options /IncludeOwn, /IncludeTASK /IncludeINTR change the way times for interrupts and other tasks are accumulated in the analysis. Default display is now to exclude times of interrupts and other tasks. With /INTRROOT and /INTRTASK it is possible to split up the interrupt times to the different tasks or count them globaly.
Build 4728 FIX 01921 <trace>.STATistic.Func (Nesting function runtime analysis) <trace>.STATistic.TREE (Tree display of nesting function run-time analysis)	**Function statistics fails with opt code** When a function jumps (not a call) to a function containing only a return instruction the analysis will fail and the nesting incorrect.
Build 2940 IMP 01892 <trace>.STATistic.Func (Nesting function runtime analysis) <trace>.STATistic.TREE (Tree display of nesting function run-time analysis)	**New "Own" items in Trace.STAT commands** "Own" time is the time of function with all called functions, but without the time spend in interrupts or other tasks.

Build 22490 CHG 02199 <trace>.STATistic.Address (Time between up to 8 program events) <trace>.STATistic.Address-GROUP (Address group run-time analysis) <trace>.STATistic.DistriB (Distribution analysis) <trace>.STATistic.GROUP (Group run-time analysis) <trace>.STATistic.Line (High-level source code line analysis) <trace>.STATistic.Var (Statistic of variable accesses) <trace>.STATistic.sYmbol (Flat run-time analysis)	**New options to change count behavior** CountChange counts all changes, CountFirst counts just the first location and CountALL counts each event. The defaults for the commands have been changed to provide useful results.
Build 9640 IMP 02054 <trace>.Chart.Var (Variable chart) <trace>.STATistic.Var (Statistic of variable accesses)	**New variable access statistics** Shows statistics simmilar to Trace.STATistic.sYmbol, but for data access cycles.

Build 56057 IMP 02406 <trace>.Chart.GROUP (Group activity chart) <trace>.Chart.Line (Graphical HLL lines analysis) <trace>.Chart.sYmbol (Symbol analysis) <trace>.STATistic.AddressGROUP (Address group run-time analysis) <trace>.STATistic.GROUP (Group run-time analysis) <trace>.STATistic.Line (High-level source code line analysis) <trace>.STATistic.sYmbol (Flat run-time analysis)	**Task specific flat statistics and charts** New options "TASK" and "SplitTASK" allows to produce task specific flat trace statistics and charts.
Build 51144 IMP 02395 <trace>.Chart.sYmbol (Symbol analysis) <trace>.STATistic.sYmbol (Flat run-time analysis)	**Inline function support for Trace.STAT.sYmbol commands** Inline functions are treated as separate functions. The option NoInline can be used to discard inline functions.
Build 51144 IMP 02389 <trace>.STATistic.DistriB (Distribution analysis) <trace>.STATistic.GROUP (Group run-time analysis) <trace>.STATistic.Line (High-level source code line analysis) <trace>.STATistic.MODULE (Code execution broken down by module) <trace>.STATistic.sYmbol (Flat run-time analysis)	**Option SplitTASK for many Trace.STATistic commands** Allows to split up the results for different tasks.
Build 22490 CHG 02199 <trace>.STATistic.Address (Time between up to 8 program events) <trace>.STATistic.AddressGROUP (Address group run-time analysis) <trace>.STATistic.DistriB (Distribution analysis) <trace>.STATistic.GROUP (Group run-time analysis) <trace>.STATistic.Line (High-level source code line analysis) <trace>.STATistic.Var (Statistic of variable accesses) <trace>.STATistic.sYmbol (Flat run-time analysis)	**New options to change count behavior** CountChange counts all changes, CountFirst counts just the first location and CountALL counts each event. The defaults for the commands have been changed to provide useful results.

Build 22490 IMP 02197 <trace>.STATistic.GROUP (Group run-time analysis) <trace>.STATistic.TASK (Task activity statistic) <trace>.STATistic.sYmbol (Flat run-time analysis)	**New option Interval for statistics** The InterVal option allows to specify either a fixed time or a tracebuffer filter expression to specify the time slices. The statistic shows the minimum and maximum times and ratios for the events in a slice.

<trace>.STREAMLOAD

Build 50104 IMP 02365 <trace>.STREAMLOAD (Load streaming file from disk) <trace>.STREAMSAVE (Save streaming file to disk)	**Stream mode save and reload capability added** The commands allow to save and restore the trace data from STREAM mode.

<trace>.STREAMSAVE

Build 50104 IMP 02365 <trace>.STREAMLOAD (Load streaming file from disk) <trace>.STREAMSAVE (Save streaming file to disk)	**Stream mode save and reload capability added** The commands allow to save and restore the trace data from STREAM mode.

<trace>.THreshold

Build 1241 FIX 01781 ICD Solutions for PowerPC400 <trace>.THreshold (Optimize threshold for trace lines)	**Threshold voltage set wrong on PPC4xx** The analyzer window settings where not set correctly. Threshold may be wrong.

<trace>.TestFocus

Build 19417 IMP 02161 ARM/Cortex Trace (parallel) <trace>.TestFocus (Test trace port recording)	**Utilisation test for ETM traceports** "Analyzer.TestFocus /Utilisation" can test the utilisation of the traceport and uncover CoreSight trace bus configuration problems.
Build 1241 CHG 01799 <trace>.TestFocus (Test trace port recording)	**SYSTEM.TEST command replaced** The SYStem.TEST command of Super10 nexus has been replaced by the (generic) Analyzer.TestFocus command.
Build 1241 IMP 01786 Adaption for Embedded Trace Macrocell Adaptation for MIPS MPC55xx/MPC56xx NEXUS Debugger and Trace <trace>.AutoFocus (Calibrate AUTOFOCUS preprocessor) <trace>.TestFocus (Test trace port recording)	**New commands to verify trace connection** The Analyzer.AutoTest command verifys the integrity of the trace connection. For adapters with adjustable threshold the command Analyzer.AutoFocus will also choose the threshold in the middle of the data eye.

<trace>.TestFocusEye

Build 34458 IMP 02281 <trace>.TestFocusEye (Check signal integrity)	**New function Analyzer.FOCUS.EYE()** Can check an eye previously scanned by Analyzer.TestFocusEye agains violations.

Build 93173 IMP 02513 Integrated Development Environment <trace>.Chart (Display trace contents graphically) <trace>.Chart.TASK (Task activity chart) <trace>.Chart.sYmbol (Symbol analysis) <trace>.Timing (Waveform of trace buffer)	**Mouse interface changed for Trace.Chart/Timing windows** Zooming into the selected area is now done via double-click with the left mouse button. Drag and drop of the cursor to another trace window aligns this window to the same cursor position. Drag and drop of the selected area to another trace window with graphical display aligns this window to the same area.
Build 88288 02496 Trace Analyzer <trace>.Chart.DatasYmbol (Analyze pointer contents graphically) <trace>.Chart.GROUP (Group activity chart) <trace>.Chart.TASK (Task activity chart) <trace>.Chart.sYmbol (Symbol analysis) <trace>.Timing (Waveform of trace buffer)	**Chart and timing windows with Track option keep zoom** The zoom factor of the <trace>.Chart and <trace>.Timing windows is retained, even if the trace content changes, if the /Track option is used.
Build 42354 IMP 02335 <trace>.Timing (Waveform of trace buffer)	**Drag & Drop in Trace Timing Window** Drag and Drop of channels can be used to rearrange channels in the window.
Build 2940 IMP 01902 <trace>.Chart (Display trace contents graphically) <trace>.DRAW (Plot trace data against time) <trace>.Timing (Waveform of trace buffer)	**New option /ZoomTrack for timing/chart** Allows to track position and zoom factor between timing and chart windows.

<trace>.View

Build 1133 FIX 01525 <trace>.List (List trace contents) <trace>.View (Display single record)	A.List/View /FILE may crash when MAP.DMUX/MAP.ADELAY is active

<trace>.state

Build 1077 IMP 01317 <trace>.state (Display trace configuration window) List.auto (Display program listing) <trace>.state (Display trace configuration window)	WR and DATA.xx items enabled for Logger, SNOOPer and Integrator

APPEND

Build 1190 IMP 01679 APPEND (Append to file)	**New command APPEND for logfile support** The command APPEND adds a line to the end of a textfile.

APU.GREP

Build 9640 IMP 02035 Unspecific Worktime Account APU.GREP (Search for string)	**APU.GREP command implemented** APU.GREP works similar to Data.GREP, but opens APU.* windows instead of Data.* windows.

AREA.CLEAR

Build 2940 CHG 01910 Host Driver Software AREA.CLEAR (Clear area)	**area.clear behavior changed** without a given area name the default area 'A000' is cleared now

AREA.Create

Build 80996 IMP 02474 AREA.Create (Create or modify message area) AREA.Delete (Delete message area) AREA.List (Display a detailed list off all message areas)	**Improvements and changes for AREA command group** The new command AREA.List displays a detailed list about all created AREA windows. The new command AREA.Delete <area_name> allows to remove an specific AREA window created by the AREA.Create command. The following characters are no longer allowed in AREA names: * \ / ' " ; , &. " are automatically removed whenever possible.
Build 1077 IMP 01066 AREA.Create (Create or modify message area)	**AREA size** AREA size limits extended to 32767 lines and 4096 columns

AREA.Delete

Build 80996 IMP 02474 AREA.Create (Create or modify message area) AREA.Delete (Delete message area) AREA.List (Display a detailed list off all message areas)	**Improvements and changes for AREA command group** The new command AREA.List displays a detailed list about all created AREA windows. The new command AREA.Delete <area_name> allows to remove an specific AREA window created by the AREA.Create command. The following characters are no longer allowed in AREA names: * \ / ' " ; , &. " are automatically removed whenever possible.

AREA.List

Build 80996 IMP 02474 AREA.Create (Create or modify message area) AREA.Delete (Delete message area) AREA.List (Display a detailed list off all message areas)	**Improvements and changes for AREA command group** The new command AREA.List displays a detailed list about all created AREA windows. The new command AREA.Delete <area_name> allows to remove an specific AREA window created by the AREA.Create command. The following characters are no longer allowed in AREA names: * \ / ' " ; , &. " are automatically removed whenever possible.

AREA.OPEN

Build 42354 02356 AREA.OPEN (Open output file)	**option /NoFileCache added** disables TRACE32 file output buffer caching - useful when TRACE32 is killed by testframe

ART

Build 1133 IMP 01489 ART (Trace method for Advanced Register Trace)	New trace method ART

AUTOINDENT

Build 100486 IMP 02530 Logical Display of Peripherals AUTOINDENT (Indent content of peripheral file automatically)	**Automatic indentation for PER files** The AUTOIDENT command allows activating various indentation styles.

AVX.view

Build 51144 IMP 02385 AVX.view (Display AVX registers) AVX512.view (Display AVX512 registers) FPU.view (Display FPU registers) MMX.view (Open MMX register window) SSE.view (Display SSE registers) VPU.view (Display ALTIVEC register window)	**Larger than 64 bit SIMD register sets** FPU, VPE, SPE, MMX, SSE, AVX, AVX512 - Parser for wide registers (>64 bit) unified, values wider than 64 bit possible.

AVX512.view

Build 51144 IMP 02385 AVX.view (Display AVX registers) AVX512.view (Display AVX512 registers) FPU.view (Display FPU registers) MMX.view (Open MMX register window) SSE.view (Display SSE registers) VPU.view (Display ALTIVEC register window)	**Larger than 64 bit SIMD register sets** FPU, VPE, SPE, MMX, SSE, AVX, AVX512 - Parser for wide registers (>64 bit) unified, values wider than 64 bit possible.

Analyzer

Build 19417 FIX 02150 Analyzer (Trace method Analyzer, recording, and analysis commands)	**Bug fixed in ARM/Thumb trace via GDI** Display of ARM/Thumb mode traces made with GDI API was wrong (swapped).
Build 19417 IMP 02143 ARM/Cortex Trace (parallel) Analyzer (Trace method Analyzer, recording, and analysis commands) ETM.PortSize (Define trace port width)	**Trace buffer usage for 8bit ETM changed** Doubles available trace records and increases frequency limit of older trace modules for ETM ports with 8 or less bits.

Analyzer.List

Build 4728 FIX 01962 PowerTrace Analyzer.List (List trace contents)	**sporadic ghost records fixed** Under certain circumstances the trace contained "ghost" records (records which actually were never recorded by the hardware).

Build 1246 CHG 01812 ARM/Cortex Trace (parallel) Analyzer.Mode (Set the trace operation mode)	**Analyzer.Mode Prestore for ARM ETM** Selective data traces, setup with /TraceEnable breakpoints, will now display only the data cycles without code information. Enabling "A.Mode Prestore" will display (and trace) also the program information.

Build 105499 IMP 02556 Integrated Development Environment DIALOG.Program (Interactive programming) EDIT (TRACE32 editor) MENU.Program (Interactive programming) PEDIT (Edit PRACTICE script) Analyzer.Program (Program trigger unit) Data.PROGRAM (Editor for writing assembler program) Integrator.Program (Program trigger unit) Probe.Program (Program trigger unit) PER.Program (Interactive programming)	**new internal text editor** New text editor version with syntax highlighting, nesting awareness, undo/redo and other features activated.
Build 7181 FIX 02006 In-Circuit Emulator for the 80186XL Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Analyzer trigger unit programming** DATA event declaration: the modes B0, B1, W0 were locked depending on the CPU derivate (since build 4789)
Build 1246 CHG 01834 ICE In-Circuit Emulator PowerTrace for NEXUS Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**programming of complex triggerunit** commands renamed: Flag.OFF -> Flag.FALSE Flag.ON -> Flag.TRUE
Build 1246 CHG 01832 PowerTrace for NEXUS Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**programming of complex triggerunit** keyword for addressevent definition ADDR replaced by ADDRESS
Build 1246 CHG 01830 FIRE Fully Integrated RISC Emulator Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**programming of analyzer triggerunit** keyword for addressevent definition ADDR replaced by ADDRESS

Build 1241 FIX 01794 FIRE Fully Integrated RISC Emulator Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Programming of complex triggerunit** hardware and onchip breakpoints configured wrong under circumstances when breakpoints were defined inside a analyzer trigger program via ADDR declaration
Build 1206 CHG 01741 MPC55xx/MPC56xx NEXUS Debugger and Trace Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Programming of complex triggerunit** supports Copperhead now particularly supports DTM, WTM, OTME with unitnumbers
Build 1203 FIX 01722 Super10 NEXUS Debugger and Trace Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**analyzer trigger programming** address events set wrong under circumstances (/HARD only)
Build 1203 FIX 01720 In-Circuit Emulator for C166/ST10 - Out of Production Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**analyzer trigger programming** address events OD, ODX, ODL, ODLX, ODLH: programming of breakpoints partly wrong
Build 1190 FIX 01692 Super10 NEXUS Debugger and Trace Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Programming of complex triggerunit** now a possible necessary second programming of complex triggerunit is superfluous in the case of DTM or WTM usage inside an ADDR declaration
Build 1190 FIX 01665 FIRE Fully Integrated RISC Emulator Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**analyzer trigger unit programming** hostbased software may crash under certain circumstances
Build 1190 FIX 01660 PowerIntegrator - Logic and Bus Analyzer PowerProbe - Logic and Protocol Analyzer Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**programming of complex triggerunit** since V1.12 from 2004.02.23 the usage of several single channels was blocked inside a SELECTOR definition
Build 1190 FIX 01658 Super10 NEXUS Debugger and Trace Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Programming of complex triggerunit** the usage of DTM and WTM inside an ADDR declaration failed sometimes under special circumstances without reprogramming of analyzer trigger program

Build 1161 CHG 01639 ICD In-Circuit Debugger Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Programming of complex triggerunit** declarationtype synonymes introduced EVCNT for EVENTCOUNTER TICNT for TIMECOUNTER
Build 1161 CHG 01637 FIRE Fully Integrated RISC Emulator Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**analyzer triggerunit programming** command renames: Count.* -> Counter.* Count.Enable -> Counter.Increment declaration renames: EVENT -> EVENTCOUNTER TIME -> TIMECOUNTER
Build 1161 CHG 01635 ICE In-Circuit Emulator Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**analyzer triggerunit programming** declarationtype synonymes introduced EVCNT for EVENTCOUNTER EXCNT for EXTERNCOUNTER TICNT for TIMECOUNTER
Build 1161 CHG 01622 ICE In-Circuit Emulator Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Analyzer trigger unit programming** command renames: Count.* -> Counter.* Count.Enable -> Counter.Increment declaration renames: EVENT -> EVENTCOUNTER EXTERN -> EXTERNCOUNTER TIME -> TIMECOUNTER
Build 1151 IMP 01589 PowerIntegrator - Logic and Bus Analyzer PowerProbe - Logic and Protocol Analyzer Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	size of all counters increased from 44 to 45 bits
Build 1146 FIX 01579 PowerIntegrator - Logic and Bus Analyzer PowerProbe - Logic and Protocol Analyzer Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Programming of complex triggerunit** usage of levelname START may crash hostbased software
Build 1145 FIX 01549 FIRE Fully Integrated RISC Emulator Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Analyzer trigger unit programming** output events BUS.A..BUS.D programmed with inverse level (only FEC hardware release > 5 effected)

Build 1096 FIX 01437 ICD In-Circuit Debugger Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Trigger Programming** ANAICD triggerprogramming: program crash fixed
Build 1077 CHG 01108 Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Trigger.Puls renamed** ICE: SA120 only: analyzer trigger unit programming: Trigger.Puls renamed to Trigger.Pulse

Build 7181 FIX 02006 In-Circuit Emulator for the 80186XL Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Analyzer trigger unit programming** DATA event declaration: the modes B0, B1, W0 were locked depending on the CPU derivate (since build 4789)
Build 1246 CHG 01834 ICE In-Circuit Emulator PowerTrace for NEXUS Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**programming of complex triggerunit** commands renamed: Flag.OFF -> Flag.FALSE Flag.ON -> Flag.TRUE
Build 1246 CHG 01832 PowerTrace for NEXUS Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**programming of complex triggerunit** keyword for addressevent definition ADDR replaced by ADDRESS
Build 1246 CHG 01830 FIRE Fully Integrated RISC Emulator Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**programming of analyzer triggerunit** keyword for addressevent definition ADDR replaced by ADDRESS
Build 1241 FIX 01794 FIRE Fully Integrated RISC Emulator Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Programming of complex triggerunit** hardware and onchip breakpoints configured wrong under circumstances when breakpoints were defined inside a analyzer trigger program via ADDR declaration
Build 1206 CHG 01741 MPC55xx/MPC56xx NEXUS Debugger and Trace Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Programming of complex triggerunit** supports Copperhead now particularly supports DTM, WTM, OTME with unitnumbers
Build 1203 FIX 01722 Super10 NEXUS Debugger and Trace Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**analyzer trigger programming** address events set wrong under circumstances (/HARD only)

Build 1203 FIX 01720 In-Circuit Emulator for C166/ST10 - Out of Production Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**analyzer trigger programming** address events OD, ODX, ODL, ODLX, ODLH: programming of breakpoints partly wrong
Build 1190 FIX 01692 Super10 NEXUS Debugger and Trace Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Programming of complex triggerunit** now a possible necessary second programming of complex triggerunit is superfluous in the case of DTM or WTM usage inside an ADDR declaration
Build 1190 FIX 01665 FIRE Fully Integrated RISC Emulator Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**analyzer trigger unit programming** hostbased software may crash under certain circumstances
Build 1190 FIX 01660 PowerIntegrator - Logic and Bus Analyzer PowerProbe - Logic and Protocol Analyzer Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**programming of complex triggerunit** since V1.12 from 2004.02.23 the usage of several single channels was blocked inside a SELECTOR definition
Build 1190 FIX 01658 Super10 NEXUS Debugger and Trace Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Programming of complex triggerunit** the usage of DTM and WTM inside an ADDR declaration failed sometimes under special circumstances without reprogramming of analyzer trigger program
Build 1161 CHG 01639 ICD In-Circuit Debugger Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Programming of complex triggerunit** declarationtype synonymes introduced EVCNT for EVENTCOUNTER TICNT for TIMECOUNTER
Build 1161 CHG 01637 FIRE Fully Integrated RISC Emulator Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**analyzer triggerunit programming** command renames: Count.* -> Counter.* Count.Enable -> Counter.Increment declaration renames: EVENT -> EVENTCOUNTER TIME -> TIMECOUNTER
Build 1161 CHG 01635 ICE In-Circuit Emulator Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**analyzer triggerunit programming** declarationtype synonymes introduced EVCNT for EVENTCOUNTER EXCNT for EXTERNCOUNTER TICNT for TIMECOUNTER

Build 1161 CHG 01622 ICE In-Circuit Emulator Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Analyzer trigger unit programming** command renames: Count.* -> Counter.* Count.Enable -> Counter.Increment declaration renames: EVENT -> EVENTCOUNTER EXTERN -> EXTERNCOUNTER TIME -> TIMECOUNTER
Build 1151 IMP 01589 PowerIntegrator - Logic and Bus Analyzer PowerProbe - Logic and Protocol Analyzer Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	size of all counters increased from 44 to 45 bits
Build 1146 FIX 01579 PowerIntegrator - Logic and Bus Analyzer PowerProbe - Logic and Protocol Analyzer Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Programming of complex triggerunit** usage of levelname START may crash hostbased software
Build 1145 FIX 01549 FIRE Fully Integrated RISC Emulator Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Analyzer trigger unit programming** output events BUS.A..BUS.D programmed with inverse level (only FEC hardware release > 5 effected)
Build 1096 FIX 01437 ICD In-Circuit Debugger Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Trigger Programming** ANAICD triggerprogramming: program crash fixed
Build 1077 CHG 01108 Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Trigger.Puls renamed** ICE: SA120 only: analyzer trigger unit programming: Trigger.Puls renamed to Trigger.Pulse

Analyzer.SAMPLE

Build 19417 CHG 02172 ARM/Cortex Trace (parallel) Analyzer.SAMPLE (Set AutoFocus sample time offset)	**Sampling information calibrated** The delay values shown in Analyzer.ShowFocus and entered with Analyzer.SAMPLE are now calibrated. This has changed the values for AutoFocus-II significantly. Old saved sampling times are not longer valid.

Analyzer.TraceCLOCK

Build 19417 CHG 02145 ARM/Cortex Trace (parallel) Analyzer.TraceCLOCK (Set capture rate for single-wire-trace)	**Command Analyzer.ExportCLOCK renamed** New name is Analyzer.TraceCLOCK.

BMC

Build 22490 IMP 02207 BMC (Benchmark counters)	**Command BMC redesigned** New layout allows more flexibility to cover different target configurations.
Build 19417 IMP 02167 Cortex-M (ARMv6/7/8 32-bit) Debugger BMC (Benchmark counters) ITM (CoreSight ITM (Instrumentation Trace Macrocell))	**Cortex-M benchmark counter support** Benchmark counters can be traced by ITM and displayed graphically.
Build 9640 CHG 02055 ARM Debugger C5500 Debugger SH Debugger BMC (Benchmark counters)	**Benchmark counter commands renamed** All benchmark counter related onchip trigger commands have been moved to the new command "BMC".

BMC.PROfileChart

Build 30461 IMP 02235 BMC.PROfileChart (Profile chart with benchmark counter)	**New command BMC.PROfileChart** Command shows benchmark counts related to certain code graphically as function of time.

BMC.state

Build 50104 IMP 02362 BMC.state (Display BMC configuration window)	**New Benchmarkcounter state window** Counter display and configuration replaced by a list-like window style. This speeds up the initial buildup and refreshing process of the window.
Build 38589 IMP 02318 Intel~ x86/x64 JTAG Debugger BMC.state (Display BMC configuration window) SNOOPer (Sample-based trace)	**Intel StopAndGo Snooping** For Intel Atom/x86 CPUs it is possible to snoop PC and BMC using StopAndGo. The snoop results can be used for profiling, charting and statistics.

BTrace.state

Build 93173 IMP 02516 Trace Analyzer BTrace.state (Display BTrace configuration window)	**Command group BTrace enables script controlled trace sink** New command group BTrace allows a script controlled trace sink. First use case: apply Trace.STATistic commands to function run-times measured with the benchmark counters of a RH850 debugger.

Break.CONFIG.InexactAddress

Build 112182 CHG 02571 JTAG Debugger Break.CONFIG.InexactAddress Break.CONFIG.MatchASID Break.CONFIG.MatchMachine Break.CONFIG.MatchZone Break.CONFIG.UseContextID Break.CONFIG.UseMachineID Break.CONFIG.VarConvert	**trigger onchip command group and dialog changed** The trigger onchip dialog and the command parser was changed for all architectures, because commands were renamed and moved to the global Break.CONFIG.state dialog and parser. The Break.CONFIG.state dialog was adapted to show the new settings made by the commands. New Break.CONFIG commands were introduced. The following list gives an overview of the deprecated TrOnchip commands which were replaced by new Break.CONFIG commands: TrOnchip.ContextID was renamed to Break.CONFIG.UseContextID, TrOnchip.CONVert was renamed to Break.CONFIG.InexactAddress, TrOnchip.CONVert0cycle was renamed to Break.CONFIG.InexactAddress0cycle, TrOnchip.MachineID was renamed to Break.CONFIG.UseMachineID, TrOnchip.MatchASID was renamed to Break.CONFIG.MatchASID, TrOnchip.MatchMachine was renamed to Break.CONFIG.MatchMachine, TrOnchip.MatchZone was renamed to Break.CONFIG.MatchZone, TrOnchip.VarCONVert was renamed to Break.CONFIG.VarConvert. The new commands introduced for selected architectures are Break.InexactData, Break.InexactTrigger and Break.InexactResume.

Break.CONFIG.MatchASID

Build 112182 CHG 02571 JTAG Debugger Break.CONFIG.InexactAddress Break.CONFIG.MatchASID Break.CONFIG.MatchMachine Break.CONFIG.MatchZone Break.CONFIG.UseContextID Break.CONFIG.UseMachineID Break.CONFIG.VarConvert	**trigger onchip command group and dialog changed** The trigger onchip dialog and the command parser was changed for all architectures, because commands were renamed and moved to the global Break.CONFIG.state dialog and parser. The Break.CONFIG.state dialog was adapted to show the new settings made by the commands. New Break.CONFIG commands were introduced. The following list gives an overview of the deprecated TrOnchip commands which were replaced by new Break.CONFIG commands: TrOnchip.ContextID was renamed to Break.CONFIG.UseContextID, TrOnchip.CONVert was renamed to Break.CONFIG.InexactAddress, TrOnchip.CONVert0cycle was renamed to Break.CONFIG.InexactAddress0cycle, TrOnchip.MachineID was renamed to Break.CONFIG.UseMachineID, TrOnchip.MatchASID was renamed to Break.CONFIG.MatchASID, TrOnchip.MatchMachine was renamed to Break.CONFIG.MatchMachine, TrOnchip.MatchZone was renamed to Break.CONFIG.MatchZone, TrOnchip.VarCONVert was renamed to Break.CONFIG.VarConvert. The new commands introduced for selected architectures are Break.InexactData, Break.InexactTrigger and Break.InexactResume.

Break.CONFIG.MatchMachine

Build 112182 CHG 02571 JTAG Debugger Break.CONFIG.InexactAddress Break.CONFIG.MatchASID Break.CONFIG.MatchMachine Break.CONFIG.MatchZone Break.CONFIG.UseContextID Break.CONFIG.UseMachineID Break.CONFIG.VarConvert	**trigger onchip command group and dialog changed** The trigger onchip dialog and the command parser was changed for all architectures, because commands were renamed and moved to the global Break.CONFIG.state dialog and parser. The Break.CONFIG.state dialog was adapted to show the new settings made by the commands. New Break.CONFIG commands were introduced. The following list gives an overview of the deprecated TrOnchip commands which were replaced by new Break.CONFIG commands: TrOnchip.ContextID was renamed to Break.CONFIG.UseContextID, TrOnchip.CONVert was renamed to Break.CONFIG.InexactAddress, TrOnchip.CONVert0cycle was renamed to Break.CONFIG.InexactAddress0cycle, TrOnchip.MachineID was renamed to Break.CONFIG.UseMachineID, TrOnchip.MatchASID was renamed to Break.CONFIG.MatchASID, TrOnchip.MatchMachine was renamed to Break.CONFIG.MatchMachine, TrOnchip.MatchZone was renamed to Break.CONFIG.MatchZone, TrOnchip.VarCONVert was renamed to Break.CONFIG.VarConvert. The new commands introduced for selected architectures are Break.InexactData, Break.InexactTrigger and Break.InexactResume.

Break.CONFIG.MatchZone

Build 112182 CHG 02571 JTAG Debugger Break.CONFIG.InexactAddress Break.CONFIG.MatchASID Break.CONFIG.MatchMachine Break.CONFIG.MatchZone Break.CONFIG.UseContextID Break.CONFIG.UseMachineID Break.CONFIG.VarConvert	**trigger onchip command group and dialog changed** The trigger onchip dialog and the command parser was changed for all architectures, because commands were renamed and moved to the global Break.CONFIG.state dialog and parser. The Break.CONFIG.state dialog was adapted to show the new settings made by the commands. New Break.CONFIG commands were introduced. The following list gives an overview of the deprecated TrOnchip commands which were replaced by new Break.CONFIG commands: TrOnchip.ContextID was renamed to Break.CONFIG.UseContextID, TrOnchip.CONVert was renamed to Break.CONFIG.InexactAddress, TrOnchip.CONVert0cycle was renamed to Break.CONFIG.InexactAddress0cycle, TrOnchip.MachineID was renamed to Break.CONFIG.UseMachineID, TrOnchip.MatchASID was renamed to Break.CONFIG.MatchASID, TrOnchip.MatchMachine was renamed to Break.CONFIG.MatchMachine, TrOnchip.MatchZone was renamed to Break.CONFIG.MatchZone, TrOnchip.VarCONVert was renamed to Break.CONFIG.VarConvert. The new commands introduced for selected architectures are Break.InexactData, Break.InexactTrigger and Break.InexactResume.

Break.CONFIG.UseContextID

Build 112182 CHG 02571 JTAG Debugger Break.CONFIG.InexactAddress Break.CONFIG.MatchASID Break.CONFIG.MatchMachine Break.CONFIG.MatchZone Break.CONFIG.UseContextID Break.CONFIG.UseMachineID Break.CONFIG.VarConvert	**trigger onchip command group and dialog changed** The trigger onchip dialog and the command parser was changed for all architectures, because commands were renamed and moved to the global Break.CONFIG.state dialog and parser. The Break.CONFIG.state dialog was adapted to show the new settings made by the commands. New Break.CONFIG commands were introduced. The following list gives an overview of the deprecated TrOnchip commands which were replaced by new Break.CONFIG commands: TrOnchip.ContextID was renamed to Break.CONFIG.UseContextID, TrOnchip.CONVert was renamed to Break.CONFIG.InexactAddress, TrOnchip.CONVert0cycle was renamed to Break.CONFIG.InexactAddress0cycle, TrOnchip.MachineID was renamed to Break.CONFIG.UseMachineID, TrOnchip.MatchASID was renamed to Break.CONFIG.MatchASID, TrOnchip.MatchMachine was renamed to Break.CONFIG.MatchMachine, TrOnchip.MatchZone was renamed to Break.CONFIG.MatchZone, TrOnchip.VarCONVert was renamed to Break.CONFIG.VarConvert. The new commands introduced for selected architectures are Break.InexactData, Break.InexactTrigger and Break.InexactResume.

Break.CONFIG.UseMachineID

Build 112182 CHG 02571 JTAG Debugger Break.CONFIG.InexactAddress Break.CONFIG.MatchASID Break.CONFIG.MatchMachine Break.CONFIG.MatchZone Break.CONFIG.UseContextID Break.CONFIG.UseMachineID Break.CONFIG.VarConvert	**trigger onchip command group and dialog changed** The trigger onchip dialog and the command parser was changed for all architectures, because commands were renamed and moved to the global Break.CONFIG.state dialog and parser. The Break.CONFIG.state dialog was adapted to show the new settings made by the commands. New Break.CONFIG commands were introduced. The following list gives an overview of the deprecated TrOnchip commands which were replaced by new Break.CONFIG commands: TrOnchip.ContextID was renamed to Break.CONFIG.UseContextID, TrOnchip.CONVert was renamed to Break.CONFIG.InexactAddress, TrOnchip.CONVert0cycle was renamed to Break.CONFIG.InexactAddress0cycle, TrOnchip.MachineID was renamed to Break.CONFIG.UseMachineID, TrOnchip.MatchASID was renamed to Break.CONFIG.MatchASID, TrOnchip.MatchMachine was renamed to Break.CONFIG.MatchMachine, TrOnchip.MatchZone was renamed to Break.CONFIG.MatchZone, TrOnchip.VarCONVert was renamed to Break.CONFIG.VarConvert. The new commands introduced for selected architectures are Break.InexactData, Break.InexactTrigger and Break.InexactResume.

Break.CONFIG.VarConvert

Build 112182 CHG 02571 JTAG Debugger Break.CONFIG.InexactAddress Break.CONFIG.MatchASID Break.CONFIG.MatchMachine Break.CONFIG.MatchZone Break.CONFIG.UseContextID Break.CONFIG.UseMachineID Break.CONFIG.VarConvert	**trigger onchip command group and dialog changed** The trigger onchip dialog and the command parser was changed for all architectures, because commands were renamed and moved to the global Break.CONFIG.state dialog and parser. The Break.CONFIG.state dialog was adapted to show the new settings made by the commands. New Break.CONFIG commands were introduced. The following list gives an overview of the deprecated TrOnchip commands which were replaced by new Break.CONFIG commands: TrOnchip.ContextID was renamed to Break.CONFIG.UseContextID, TrOnchip.CONVert was renamed to Break.CONFIG.InexactAddress, TrOnchip.CONVert0cycle was renamed to Break.CONFIG.InexactAddress0cycle, TrOnchip.MachineID was renamed to Break.CONFIG.UseMachineID, TrOnchip.MatchASID was renamed to Break.CONFIG.MatchASID, TrOnchip.MatchMachine was renamed to Break.CONFIG.MatchMachine, TrOnchip.MatchZone was renamed to Break.CONFIG.MatchZone, TrOnchip.VarCONVert was renamed to Break.CONFIG.VarConvert. The new commands introduced for selected architectures are Break.InexactData, Break.InexactTrigger and Break.InexactResume.

Break.DeletePATtern

Build 100486 02525 Uniform Look-And-Feel for all Processors Break.DeletePATtern (Delete breakpoints allowing wildcards) Break.PATtern (Set temporary breakpoints allowing wildcards) Break.SetPATtern (Set breakpoints allowing wildcards)	**Set/delete breakpoints on symbol pattern** Breakpoints can be set on symbol pattern.

Break.Init

Build 1253 IMP 01842 Break.Init (Initialize breakpoints)	**Break.Init reinitializes also onchip bps** The comand can be used to restore overwritten onchip trigger resources.

Break.List

Build 100486 IMP 02521 Uniform Look-And-Feel for all Processors Break.List (Display list of breakpoints)	**Break.List window was revised** Enabling/disabling breakpoints is now possible with a single click to the check mark in the new "icon" column. Change Breakpoint window is opened by pushing the Change button. Double click to a Program breakpoint opens now source list window (List command). Double click to a Read or Write breakpoint opens Data.View window.
Build 1077 IMP 01302 Break.List (Display list of breakpoints)	breakpoint display highlights hit breakpoint

Break.PATtern

Build 100486 02525 Uniform Look-And-Feel for all Processors Break.DeletePATtern (Delete breakpoints allowing wildcards) Break.PATtern (Set temporary breakpoints allowing wildcards) Break.SetPATtern (Set breakpoints allowing wildcards)	**Set/delete breakpoints on symbol pattern** Breakpoints can be set on symbol pattern.

Build 19417 IMP 02149 Break.Set (Set breakpoints) Var.Break.Set (Set breakpoint to HLL expression)	**Data value breakpoint support** Data value breakpoints are now supported via stop and go when the target has no data comparision capabilities.
Build 19417 FIX 02124 ARM/Cortex Trace (parallel) Break.Set (Set breakpoints) ETM (Embedded Trace Macrocell (ETM))	**ETM selective data trace** Selective data trace may miss cycles in ETMv3.
Build 13751 IMP 02097 Break.Set (Set breakpoints)	**Default breakpoint types changed** For program breakpoints the default is now to try a software implementation and switch to onchip if this fails.
Build 13751 IMP 02085 ICD Solutions for ARM Break.Set (Set breakpoints) Var.Break.Set (Set breakpoint to HLL expression)	**ARM ETM Breakpoints Extended** ETM Breakpoints now support also combined program/data breakpoints and breakpoints with condition pass/fail.
Build 8248 FIX 02025 TriCore Debugger Break.Set (Set breakpoints)	**Support for 4 Onchip Breakpoints in Code**
Build 4943 FIX 01990 C5500 Debugger Break.Set (Set breakpoints) Var.Break.Set (Set breakpoint to HLL expression)	**C55x data breakpoint problems fixed** Breakpoints with data value and long access breakpoints failed sometimes.
Build 4728 IMP 01970 Break.Set (Set breakpoints)	**Onchip breakpoint with data not match** On some cores breakpoints on data mismatch are supported (not operator).
Build 4728 IMP 01945 ARM11 JTAG Debugger Break.Set (Set breakpoints)	**ContextID supported by breakpoints+ETM** When TrOnchip.ContextID is enabled task-related breakpoints will use the ContextID comparators of breakpoint unit and ETM.
Build 1268 IMP 01871 Adaptation for OCDS Level 2 TriCore Break.Set (Set breakpoints)	**onchip breakpoints mixable** Mixing onchip breakpoints with trace action and other onchip breakpoints was not allowed. When setting them anyway, they did not work. Mixing them is now allowed and works.

Build 1253 FIX 01847 Break.Set (Set breakpoints) Var.Break.Set (Set breakpoint to HLL expression)	**Onchip program exclude breakpoint failed** Breakpoint refused when no ETM breakpoints usable and range does not fit into single address mask.
Build 1096 IMP 01438 Break.Set (Set breakpoints)	Break.Set /VCONDition expression checked on command entry
Build 1077 IMP 01300 Break.Set (Set breakpoints)	**New breakpoint types** Several new breakpoint types implemented
Build 1077 FIX 01157 ARM Debugger MPC56x NEXUS Debugger and Trace Break.Set (Set breakpoints)	**B.Set dialog** B.Set dialog may crash on Sun for ARM or PPC-Nexus

Break.SetFunc

Build 1096 BUG 01427 C166CBC Debugger C166 Family Real-Time Trace Break.SetFunc (Mark HLL functions)	breakpoints set with Break.SetFunc command may not be stored on CBC166 Risctrace
Build 1085 IMP 01423 FIRE Emulator for C166 Family FIRE Emulator for ST10 FIRE Emulator for NEC V850 Break.SetFunc (Mark HLL functions)	C167/ST10 and V850 fire Break.SetFunc improved to save breakpoint resources
Build 1077 IMP 01324 ARM Debugger Break.SetFunc (Mark HLL functions)	B.SF on ARM was causing wrong memory access class (no other effects)

Break.SetPATtern

Build 100486 02525 Uniform Look-And-Feel for all Processors Break.DeletePATtern (Delete breakpoints allowing wildcards) Break.PATtern (Set temporary breakpoints allowing wildcards) Break.SetPATtern (Set breakpoints allowing wildcards)	**Set/delete breakpoints on symbol pattern** Breakpoints can be set on symbol pattern.

CACHE.ComPare

Build 1077 FIX 01214 CACHE.ComPare (Compare CACHE with memory)	CACHE.COMPARE command was not using physical addressing for the noncached reads

CACHE.DUMP

Build 1077 FIX 01196 CACHE.DUMP (Dump CACHE)	**Cache** printing CACHE.DUMP window was now showing whole cache contents

CACHE.view

Build 13751 IMP 02083 TriCore Debugger CACHE.view (Display cache control register)	**Cache Dump added** For the members of the AUDO-FUTURE chip family, the Cache Dump features have been added.

CAnalyzer

Build 13751 IMP 02091 ICD Solutions for ARM CAnalyzer (Trace features of Compact Analyzer) SystemTrace (MIPI STP and CoreSight ITM)	**CORTEX-M3 Trace Support** Support for Cortex-M3 tracing via SWV and TPIU for ETM and ITM added.

CAnalyzer.SAMPLE

Build 100486 IMP 02541 AUTOFOCUS Self Calibration CAnalyzer.SAMPLE (Set sample time offset) CAnalyzer.ShowFocus (Display data eye) CAnalyzer.ShowFocusClockEye (Show clock eye) CAnalyzer.TERMination (Configure parallel trace termination)	**Support for Whisker MIPI20T-HS CombiProbe/uTrace Arm** New commands to use new capability of Whisker MIPI20T-HS CombiProbe/uTrace Arm.

CAnalyzer.ShowFocus

Build 100486 IMP 02541 AUTOFOCUS Self Calibration CAnalyzer.SAMPLE (Set sample time offset) CAnalyzer.ShowFocus (Display data eye) CAnalyzer.ShowFocusClockEye (Show clock eye) CAnalyzer.TERMination (Configure parallel trace termination)	**Support for Whisker MIPI20T-HS CombiProbe/uTrace Arm** New commands to use new capability of Whisker MIPI20T-HS CombiProbe/uTrace Arm.

CAnalyzer.ShowFocusClockEye

Build 100486 IMP 02541 AUTOFOCUS Self Calibration CAnalyzer.SAMPLE (Set sample time offset) CAnalyzer.ShowFocus (Display data eye) CAnalyzer.ShowFocusClockEye (Show clock eye) CAnalyzer.TERMination (Configure parallel trace termination)	**Support for Whisker MIPI20T-HS CombiProbe/uTrace Arm** New commands to use new capability of Whisker MIPI20T-HS CombiProbe/uTrace Arm.

CAnalyzer.TERMination

Build 100486 IMP 02541 AUTOFOCUS Self Calibration CAnalyzer.SAMPLE (Set sample time offset) CAnalyzer.ShowFocus (Display data eye) CAnalyzer.ShowFocusClockEye (Show clock eye) CAnalyzer.TERMination (Configure parallel trace termination)	**Support for Whisker MIPI20T-HS CombiProbe/uTrace Arm** New commands to use new capability of Whisker MIPI20T-HS CombiProbe/uTrace Arm.

CLOCK

Build 38589 IMP 02297 GTM Debugger (TriCore) PCP Debugger (TriCore) TriCore Debugger CLOCK (Display date and time)	**on-chip clock display and settings** The CLOCK command group simplifies the setup for timestamp decoding of an on-chip trace. Can also be used to verify PLL and clock setup.

COVerage.EXPORT

Build 93173 IMP 02517 Trace-based Code Coverage COVerage.EXPORT (Export code coverage information to an XML file)	**Improved COVerage.EXPORT for C++ projects** Unique identifiers are provided as additional property for functions. They can be used to identify functions in case of overloading.

COVerage.EXPORT.CSV

Build 125398 IMP 02586 Trace Analyzer COVerage.EXPORT.CSV (Export coverage data in CSV format) ISTATistic.EXPORT.CSV (Export instruction statistics in CSV format)	**new commands COVerage.EXPORT.CSV and ISTAT.EXPORT.CSV** New commands COVerage.EXPORT.CSV and ISTAT.EXPORT.CSV introduced to export coverage/instruction statistic data in comma separated values (CSV) format.

COVerage.EXPORT.JSON

Build 130863 IMP 02603 COVerage.EXPORT.JSON (Export code coverage in JSON format)	**new command COVerage.EXPORT.JSON added** The command COVerage.EXPORT.JSON exports coverage information about functions and lines to a file in JSON format compatible to Gcov.

COVerage.EXPORT.ListCalleEs

Build 130863 IMP 02608 Trace-based Code Coverage COVerage.EXPORT.ListCalleEs (Export the function callees) COVerage.EXPORT.ListCalleRs (Export the function callers) COVerage.ListCalleEs (Display coverage for callees function) COVerage.ListCalleRs (Display coverage for callers function)	**new COVerage commands** New commands COVerage.ListCalleRs, COVerage.ListCalleRs, COVerage.EXPORT.ListCalleEs and COVerage.EXPORT.ListCalleRs. The new commands allow to list/export coverage information for callers and callees of functions.

COVerage.EXPORT.ListCalleRs

Build 130863 IMP 02608 Trace-based Code Coverage COVerage.EXPORT.ListCalleEs (Export the function callees) COVerage.EXPORT.ListCalleRs (Export the function callers) COVerage.ListCalleEs (Display coverage for callees function) COVerage.ListCalleRs (Display coverage for callers function)	**new COVerage commands** New commands COVerage.ListCalleRs, COVerage.ListCalleRs, COVerage.EXPORT.ListCalleEs and COVerage.EXPORT.ListCalleRs. The new commands allow to list/export coverage information for callers and callees of functions.

COVerage.ListCalleEs

Build 130863 IMP 02608 Trace-based Code Coverage COVerage.EXPORT.ListCalleEs (Export the function callees) COVerage.EXPORT.ListCalleRs (Export the function callers) COVerage.ListCalleEs (Display coverage for callees function) COVerage.ListCalleRs (Display coverage for callers function)	**new COVerage commands** New commands COVerage.ListCalleRs, COVerage.ListCalleRs, COVerage.EXPORT.ListCalleEs and COVerage.EXPORT.ListCalleRs. The new commands allow to list/export coverage information for callers and callees of functions.

COVerage.ListCalleRs

Build 130863 IMP 02608 Trace-based Code Coverage COVerage.EXPORT.ListCalleEs (Export the function callees) COVerage.EXPORT.ListCalleRs (Export the function callers) COVerage.ListCalleEs (Display coverage for callees function) COVerage.ListCalleRs (Display coverage for callers function)	**new COVerage commands** New commands COVerage.ListCalleRs, COVerage.ListCalleRs, COVerage.EXPORT.ListCalleEs and COVerage.EXPORT.ListCalleRs. The new commands allow to list/export coverage information for callers and callees of functions.

COVerage.ListFunc

Build 80996 IMP 02472 Trace-based Code Coverage COVerage.ListFunc (Display coverage for functions) COVerage.ListLine (Display coverage for HLL lines) COVerage.ListModule (Display coverage for modules) COVerage.Option.SourceMetric (Select code coverage metric)	**Statement and decision coverage now supported** TRACE32 Code Coverage now supports Statement and Decision coverage.

COVerage.ListLine

Build 80996 IMP 02472 Trace-based Code Coverage COVerage.ListFunc (Display coverage for functions) COVerage.ListLine (Display coverage for HLL lines) COVerage.ListModule (Display coverage for modules) COVerage.Option.SourceMetric (Select code coverage metric)	**Statement and decision coverage now supported** TRACE32 Code Coverage now supports Statement and Decision coverage.

COVerage.ListModule

Build 80996 IMP 02472 Trace-based Code Coverage COVerage.ListFunc (Display coverage for functions) COVerage.ListLine (Display coverage for HLL lines) COVerage.ListModule (Display coverage for modules) COVerage.Option.SourceMetric (Select code coverage metric)	**Statement and decision coverage now supported** TRACE32 Code Coverage now supports Statement and Decision coverage.

COVerage.METHOD

Build 50104 IMP 02371 COVerage.METHOD (Select code coverage method) ISTATistic.METHOD (Recording method for instruction statistics)	**New SPY method for COVerage and ISTAT commands** Allows to process the trace while recording in STREAM mode.
Build 50104 IMP 02369 COVerage.METHOD (Select code coverage method)	**New command COVerage.METHOD** Defines the operation of the COVerage command.

COVerage.Option.SourceMetric

Build 93173 IMP 02515 Trace-based Code Coverage COVerage.Option.SourceMetric (Select code coverage metric)	**TRACE32 supports now MCDC coverage** TRACE32 supports now MCDC coverage without code instrumentation. Requires generation of *.eca (Extended Code Analysis) files with preprocessor t32cast.exe. t32cast.exe can be found in ~~/demo/t32cast/bin.
Build 80996 IMP 02472 Trace-based Code Coverage COVerage.ListFunc (Display coverage for functions) COVerage.ListLine (Display coverage for HLL lines) COVerage.ListModule (Display coverage for modules) COVerage.Option.SourceMetric (Select code coverage metric)	**Statement and decision coverage now supported** TRACE32 Code Coverage now supports Statement and Decision coverage.

COVerage.state

Build 50104 IMP 02372 COVerage.state (Configure coverage)	**New command COVerage.state** Shows and controls various COVerage related settings.

CSV

Build 100486 IMP 02529 Logical Display of Peripherals PER.view (Display peripherals) CSV (Enables CSV capabilities) PERCMD (Row definition in CSV-formatted *.per file)	**CSV format allowed for PER files** PER files can be written within a spreadsheet editor, since the PER.view command can now also compile CSV files.

CTS

Build 19417 IMP 02166 MPC56x NEXUS Debugger and Trace CTS (Context tracking system (CTS))	**CTS on PowerPC extended to support FPU** CTS reconstructs now also the FPU registers.
Build 4728 IMP 01973 CTS (Context tracking system (CTS))	**CTS can start with known memory state** CTS.CAPTURE can capture the contents of memory, caches and TLBs before the application is started. This allows CTS to start with known values.

CTS.CACHE

Build 1253 IMP 01849 CTS.CACHE (CTS cache analysis)	**CTS Cache analysis supports locking** Trace of cache locking is supported for ARM920 and ARM922 cores.

CTS.CACHE.ViewStalls

Build 9640 IMP 02051 CTS.CACHE.ViewStalls (Display statistics for idles/stalls)	**Stall analysis improved** Differentiates between data and prefetch stalls. New MIPS profile chart.

CTS.CAPTURE

Build 2940 IMP 01917 CTS.CAPTURE (Copy real memory to the virtual memory for CTS)	**New command CTS.CAPTURE** Allows to capture initial cache and memory contents for a CTS trace when UseVM is enabled.

Build 34458 IMP 02283 CTS.List (List trace contents) <trace>.List (List trace contents) <trace>.ListNesting (Analyze function nesting)	**New display format with SplitCORE option** Trace is shown separated for the cores.
Build 30461 IMP 02232 CTS.List (List trace contents) <trace>.ListNesting (Analyze function nesting) <trace>.STATistic.TREE (Tree display of nesting function run-time analysis)	**Nesting analysis made error tolerant** The analysis will try to detect and fix problems in the function nesting caused by special target code or traceport FIFO overflows. The fixes and detected problems are be marked in the nested trace listing (Trace.ListNesting or CTS.List).
Build 1190 ERR 01680 CTS.List (List trace contents)	**Bug fixed in T.CTS.List with groups** T.CTS.List may not display all group changes. Changing the groups on the fly may cause an incorrect display.
Build 1151 FIX 01603 CTS.List (List trace contents)	several bugs fixed in A.CTS.List when GROUPs were used
Build 1077 FIX 01359 CTS.List (List trace contents)	Bug fixed in CTS.List, may crash on ICE when instruction causes more than 63 bytes data transfer

CTS.ListNesting

Build 19417 IMP 02147 CTS.ListNesting (Analyze function nesting) <trace>.ListNesting (Analyze function nesting)	**New command Trace.ListNesting** Command shows execution flow similar to CTS.List window.

CTS.ON

Build 1161 ERR 01624 CTS.ON (Switch on trace-based debugging)	**<trace>.CTS.ON crashes** Executing <trace>.CTS.ON the first time will crash the debugger. Use <trace>.CTS.GOTO instead (or the CTS Goto item in the pulldown menu of the trace windows).

CTS.PROfileChart

Build 8248 IMP 02020 CTS.PROfileChart (Profile charts) <trace>.PROfileChart (Profile charts)	**New command Trace.PROfileChart** Allows to display distributions vs. time graphically. Useful to show processing ratio of selected tasks versus time.

CTS.STATistic.TREE

Build 1228 FIX 01767 CTS.STATistic.TREE (Tree display of nesting function run-time analysis)	**Nesting button in CTS.STAT.TREE window** Button does not display CTS nesting.

CTS.SmartTrace

Build 1133 IMP 01512 CTS.SmartTrace (CTS smart trace)	New command Trace.CTS.SelectiveTrace to support CTS with selective tracing

CTS.UseMemory

Build 1246 FIX 01826 CTS.UseMemory (Use memory contents for CTS) CTS.UseRegister (Use the CPU registers for CTS)	**CTS may ignore the current context** This may occur during partial processing.

CTS.UseRegister

Build 1246 FIX 01826 CTS.UseMemory (Use memory contents for CTS) CTS.UseRegister (Use the CPU registers for CTS)	**CTS may ignore the current context** This may occur during partial processing.

ChDir

Build 100486 CHG 02545 Integrated Development Environment ChDir (Change directory)	**The prefix ChDir. can now be used for all commands** The ChDir. command prefix can now be used for all commands. Previously it was limited to IDE commands.
Build 1077 FIX 01372 ChDir (Change directory)	chdir <wildcard> starts now on correct directory on Windows

CmdPOS

Build 100486 IMP 02526 Integrated Development Environment CmdPOS (Controls the position of TRACE32 in MWI window mode) FramePOS (Controls the position of TRACE32 in MDI window mode) WinPOS (Define window dimensions and window name)	**Improved support for multimonitor desktops under MS Windows** The maximum possible window creation size was enlarged to cover the full multimonitor desktop size under MS Windows.

ComPare

Build 130863 IMP 02610 Integrated Development Environment ComPare (Compare files)	**the option of the command ComPare can now be combined** The options /Case, /IgnoreSpace and IgnoreCRLF of the command ComPare can now be combined. Previously only one option was allowed.

Count

Build 1096 IMP 01455 PowerProbe - Logic and Protocol Analyzer Count (Universal counter)	frequency meassurement extended beyond 20MHz on PowerProbe, new counter window

DIALOG

Build 1077 IMP 01354 DIALOG (Custom dialogs)	Dialog window size stored correctly

DIALOG.MESSAGE

Build 13751 IMP 02116 DIALOG.MESSAGE (Create dialog box with an information icon)	**New standard dialog** Creates a dialog with an information message.

Build 105499 IMP 02556 Integrated Development Environment DIALOG.Program (Interactive programming) EDIT (TRACE32 editor) MENU.Program (Interactive programming) PEDIT (Edit PRACTICE script) Analyzer.Program (Program trigger unit) Data.PROGRAM (Editor for writing assembler program) Integrator.Program (Program trigger unit) Probe.Program (Program trigger unit) PER.Program (Interactive programming)	**new internal text editor** New text editor version with syntax highlighting, nesting awareness, undo/redo and other features activated.
Build 1190 IMP 01691 DIALOG.Program (Interactive programming) DIALOG.ReProgram (Dialog programming)	**New dialog element DYNAMIC** The new element allows to display dynamic (changable) text or bitmap fields in the dialog window.

DIALOG.ReProgram

Build 1190 IMP 01691 DIALOG.Program (Interactive programming) DIALOG.ReProgram (Dialog programming)	**New dialog element DYNAMIC** The new element allows to display dynamic (changable) text or bitmap fields in the dialog window.

DIALOG.YESNO

Build 1077 FIX 01200 DIALOG.YESNO (Create dialog box with YES and NO buttons)	**Dialog.YESNO may crash** Dialog.YESNO may crash for very long strings

DIR

Build 1079 FIX 01397 DIR (List subdirectories and files) LS (Display directory)	Dir command showed wrong modification time (not local time)

DO

Build 100486 CHG 02528 Integrated Development Environment DO (Start PRACTICE script)	**autostart.cmm replaces t32.cmm** autostart.cmm is located in the system directory and always executed when TRACE32 instance (e.g. t32marm.exe) is started. If no autostart.cmm exists, TRACE32 falls back to the outdated behavior: t32.cmm is executed from either the working or the system directory, but only if not start-up script was explicitly specified as command line option. For details refer to "Automatic Start-up Scripts" in practice_user.pdf
Build 93173 ODD 02505 Script Language PRACTICE DO (Start PRACTICE script)	**PRACTICE error handling** PRACTICE scripts containing syntax errors are automatically popped (ENDDO) from PRACTICE stack during loadtime (DO <scriptname>) now.

DTM

Build 26464 IMP 02224 DTM (DTM trace sources (Data Trace Module))	**New command DTM to support CoreSight** DTM trace source can show the contents of simple CoreSight trace sources in different formats. Trace sources are typically either internal signals, busses or instrumentation traced.
Build 19417 IMP 02156 ARM/Cortex Trace (parallel) DTM (DTM trace sources (Data Trace Module))	**Custom trace source support** The DTM command provides support for "simple" custom CoreSight trace sources.

Data.ATTACH

Build 117056 IMP 02574 JTAG Debugger Data.ATTACH (Attach data sequence)	**new command group Data.ATTACH** The new Data.ATTACH command group allows to define a sequence of Data.Set commands that are executed when the debugger is activated with SYStem.Mode Attach.

Data.AllocList

Build 1077 FIX 01174 Data.AllocList (Static memory allocation analysis)	**Printing** WinPrint.Data.AllocList was not printing all contents

Data.CLEARVM

Build 88288 CHG 02503 ASM Debugger Data.CLEARVM (Clear the TRACE32 virtual memory (VM:))	**New behavior for TRACE32 Virtual Memory** The TRACE32 Virtual Memory (VM:) is no longer initialized by 64KB blocks upon a write access. Instead, the data are just written to the specified addess/address range. The command Data.CLEARVM clears now exactly the address/address range specified.

Data.COPY

Build 34458 IMP 02270 Data.COPY (Copy memory) Data.PATTERN (Fill memory with pattern) Data.Set (Modify memory)	**option DIFF added** New option "DIFF" for commands Data.Set, Data.PATTERN and Data.COPY. Can be used to compare memory against values.
Build 1241 IMP 01790 Data.COPY (Copy memory)	**New options for Data.COPY command** TByte, TWord, Verify and ComPare

Build 1133 FIX 01508 Data.ComPare (Compare memory)	D.COMPARE xxx--0xffffffff crashed
Build 1077 FIX 01234 Data.ComPare (Compare memory)	Data.COMPARE without second parameter compares with same address

Build 69655 IMP 02442 High-Level-Language Debugging Trace Features <trace>.DRAW.channel (Plot no-data values against time) <trace>.DRAW.Data (Plot data values against time) <trace>.DRAW.Var (Plot variable values against time) Data.DRAW (Graphical memory display of arrays) Data.PROfile (Graphical display of data value) Var.PROfile (Graphical display of variable)	**Multiple variables of different type can be drawn** TRACE32 allows to draw multiple variables/channels with different formats e.g. integers and floats for all listed commands.
Build 26464 IMP 02228 Data.DRAW (Graphical memory display of arrays) Var.DRAW (Graphical variable display)	**Graphical data display enhanced** Can now show up to six channels in one window. New option "Element" can select certain element of structure in an array of structures.
Build 4728 IMP 01967 Data.DRAW (Graphical memory display of arrays) Var.DRAW (Graphical variable display)	**Data.DRAW and Var.DRAW extended** Can now display up to six channels.
Build 1270 IMP 01872 Data.DRAW (Graphical memory display of arrays) <trace>.DRAW (Plot trace data against time)	**Floating point formats allowed** Floating point formats now allowed for low level graphical displays.
Build 1077 FIX 01329 Data.DRAW (Graphical memory display of arrays)	Data.DRAW may crash when window is made larger and do data to update

Data.DRAWFFT

Build 13751 IMP 02094 Data.DRAWFFT (Graphical display of fast fourier transformation)	**FFT display command** FFT (Fast Fourier Transform) supported by command Data.DRAWFFT.

Data.EPILOG

Build 1241 FIX 01791 TriCore Debugger Data.EPILOG (Automatic data modification on program execution halt)	**Data.Epilog did not execute** Data.EPILOG was not executed when breaking

Data.EPILOG.state

Build 60219 IMP 02423 Debugger Features Data.EPILOG.state (Display data epilogs) Data.PROLOG.state (Display data prologs) Data.SOFTEPILOG.state (Display data softepilogs) Data.SOFTPROLOG.state (Display data softprologs) Data.STARTUP.state (Startup data state display) Data.TIMER.state (Timer state display) FLASH.EPILOG.state (Display FLASH epilogs) FLASH.PROLOG.state (Display FLASH prologs)	**Full SMP support for auto sequences** Auto-sequences can be supplied for each core in an SMP system.

Data.Find

Build 69655 IMP 02448 Trace Analyzer Data.Find (Search in memory) Data.FindCODE (Execute command on specified code type) <trace>.Find (Find specified entry in trace)	**New option /ALL for Trace.Find command** With the option /ALL the TRACE32 Message Line displays how often the searched item was found. The number of occurrences can be processed in a script via the function FOUND.COUNT().
Build 4728 FIX 01934 Data.Find (Search in memory)	**Data.Find with word or long data fails** May skip areas or search the data wrong aligned. Bytewide search operations are not affected.

Data.FindCODE

Build 69655 IMP 02448 Trace Analyzer Data.Find (Search in memory) Data.FindCODE (Execute command on specified code type) <trace>.Find (Find specified entry in trace)	**New option /ALL for Trace.Find command** With the option /ALL the TRACE32 Message Line displays how often the searched item was found. The number of occurrences can be processed in a script via the function FOUND.COUNT().

Build 105499 IMP 02558 Data.IMAGE (Display image data)	**Options to display RGB images with user defined format** new options /RGBCUSTOM and /RGBBITS added
Build 30461 IMP 02248 Data.IMAGE (Display image data)	**Displaying image data enhanced** Support zooming via mouse scroll wheel. Advanced pixel operations like dumping data on right mouse button click, display pixel address in status bar, display pixel data in tool-tip.
Build 4943 IMP 01991 Data.IMAGE (Display image data)	**JPEG format supported by Data.IMAGE**
Build 4728 FIX 01931 Data.IMAGE (Display image data)	**Data.IMAGE with large /STRIDE fails** Values larger than 0x1000 cause a crash.
Build 1190 IMP 01675 Data.IMAGE (Display image data)	**New video image formats** New formats GrayScale8, YUV420 and YUV422P/YUV422PS.
Build 1151 IMP 01598 Uniform Look-And-Feel for all Processors Data.IMAGE (Display image data)	new formats added to Data.IMAGE command (TrueColor formats RGB24 and YUV422)
Build 1077 FIX 01110 Data.IMAGE (Display image data)	Data.IMAGE /BottomUp did not draw correct when window size not aligned

Build 38589 CHG 02308 Data.LOAD (Load file)	**Data.LOAD with verify changed** The verification will now be made after the complete code download.
Build 19417 IMP 02182 Data.LOAD (Load file) sYmbol (Debug symbols)	**Support for code overlays** Supports different concepts of code overlays for multiple architecures.
Build 19417 IMP 02160 ARM Debugger MIPS32/MIPS64 Debugger MPC5xx/8xx Debugger Data.LOAD (Load file)	**New DIFFLOAD options and targets** New options allow further compression or compressed load (/ZIPLOAD) and quick download checks (/CHECKLOAD and /CHECKONLY). Target agents for MIPS32 and MIPS64 architectures added.
Build 19417 IMP 02157 Data.LOAD (Load file)	**STRIPPART option can do multiple parts** Option with string parameter can be repeated to strip away multiple different parts.
Build 19417 IMP 02144 Data.LOAD (Load file)	**D.LOAD command address syntax extended** Now many load commands can relocate and limit the download at the same time (supply single address and address range parameters).
Build 9640 IMP 02049 Data.LOAD (Load file)	**New option /cygdrive to convert pathname** Converts cygwin pathnames into regular windows pathnames.
Build 1264 IMP 01862 Data.LOAD (Load file)	**New ALIGN option for flash downloads** Can align memory operations to blocks of up to 1K size. Useful for FLASH programming.
Build 1151 IMP 01612 MIPS32/MIPS64 Debugger SH Debugger Data.LOAD (Load file)	**DWARF1** Changes made to DWARF1 loader for Hitachi-SH compiler and Greenhills-MIPS
Build 1077 IMP 01307 Data.LOAD (Load file)	/NOCODE and /PLUSVM options work when used together
Build 1077 FIX 01222 Data.LOAD (Load file)	**64bit targets** data.load command /RELOC was not working correctly on 64bit targets
Build 1077 IMP 01199 Data.LOAD (Load file)	**MetroWerks** New option /MetroWerks as workarround to access function local statics for Metrowerks compiler
Build 1077 FIX 01190 Data.LOAD (Load file)	problem fixed in .stabs loader with "r" items with "@" size attribute

Build 1077 CHG 01173 ONCE Debugger for DSP56300 Data.LOAD (Load file)	DSP56300: speed up download speed. changed status read in drvbdm58.c
Build 1077 FIX 01149 Data.LOAD (Load file)	STABS loader changed '@' item handling
Build 1077 FIX 01101 Data.LOAD (Load file)	**ELF/DWARF** ELF/DWARF loader may crash after load when many complex lifetimes are loaded
Build 1077 IMP 01095 Data.LOAD (Load file)	**GNU compilers** Better automatic detection of demangling variants for GNU compilers

Data.LOAD.ASAP2

Build 1196 IMP 01703 Data.LOAD.ASAP2 (Load ASAP2 file) sYmbol.Add-Info.LOADASAP2 (Load scaling information from ASAP2 file)	**New command Data.LOAD.ASAP2** Loads information from ASAP2 database file and creates symbols and type information to display the values WITHOUT the need for a debug symbol file.

Data.LOAD.Binary

Build 69655 CHG 02444 ASM Debugger Data.LOAD.Binary (Load binary file)	**Data.LOAD.Binary requires now start address or range** The command Data.LOAD.Binary now requires a start address or a range. Before a load to address zero was performed when no start address or range was provided.
Build 19417 IMP 02139 Data.LOAD.Binary (Load binary file)	**Binary file save/load can compress** New options /ZIP and /UNZIP can compress/decompress binary files.
Build 1151 FIX 01614 FIRE Fully Integrated RISC Emulator Data.LOAD.Binary (Load binary file)	**Download** download of binary files to monitor control based FIREs may fail

Build 19417 IMP 02165 Data.LOAD.CCSDAT (Load CCSDAT file) Data.SAVE.CCSDAT (Save CCSDAT file)	**New file format CCSDAT supported** Format is compatible with TI Code Composer Studio (R) data file format.

Build 38589 CHG 02309 C2000 JTAG Debugger C5500 Debugger C6000 Debugger Data.LOAD.COFF (Load COFF file)	**Code loaded to logical address** Data.LOAD.COFF for TI architectures now loads the code at the logical address of the COFF file. The option /PHYSLOAD can force the load to the physical address.
Build 9640 IMP 02053 Data.LOAD.COFF (Load COFF file) Data.LOAD.Elf (Load ELF file)	**Const variables supported** Constant variables in DWARF2 debug format are now also loaded.
Build 4728 FIX 01925 ARM Debugger Data.LOAD.COFF (Load COFF file)	**Function info missing for ARM TI-COFF** Only symbol name and type information is there.
Build 4728 FIX 01923 ARM Debugger Data.LOAD.COFF (Load COFF file)	**TI ARM compiler ARM/Thumb attributes** Attributes may be wrong after literal constants in code.
Build 2940 FIX 01875 Data.LOAD.COFF (Load COFF file)	**COFF loader may cause unaligned loads** If the code in the file is not aligned to 32bits then the load blocks may also be not aligned. This degrades download performance and can cause wrong downloads on some architectures (e.g. C6000).
Build 1161 FIX 01632 Data.LOAD.COFF (Load COFF file) Data.LOAD.Elf (Load ELF file)	**Compilation directory missing in DWARF2** The compilation directory in missing in the source filename in DWARF2 debug information when a directory is already specified for the source.
Build 1145 FIX 01567 C5500 Debugger Data.LOAD.COFF (Load COFF file)	**COFF loader** bug fixed in C55x COFF loader (support of large pointers)
Build 1133 CHG 01492 Data.LOAD.COFF (Load COFF file)	COFF loader const symbols now saved as C: memory class
Build 1077 IMP 01142 C5500 Debugger Data.LOAD.COFF (Load COFF file)	**Reset PC** Reset PC taken from COFF file for C55x
Build 1077 FIX 01112 Data.LOAD.COFF (Load COFF file)	**COFF loader** COFF loader may fail when HLL functions contain no block
Build 1077 CHG 01087 Data.LOAD.COFF (Load COFF file) Data.LOAD.Elf (Load ELF file)	**COFF/ELF** COFF/ELF option /BUGFIX changed, option /ChainedStab for .stabs chaining

Data.LOAD.CORE

Build 9640 IMP 02041 Data.LOAD.CORE (Load Linux core dump file)	**New command Data.LOAD.CORE for coredumps** Allows to load Linux coredump files for post-mortem analysis.

Data.LOAD.CrashDump

Build 93173 IMP 02507 RTOS Debugger for Windows Data.LOAD.CrashDump (Load MS Windows Crash Dump file) TASK.CrashDump (Windows crash dump analysis)	**Support for loading Windows crash dump** After loading the full Crash Dump file into the simulator or the target memory with the command Data.LOAD.CrashDump, the Windows awareness command TASK.CrashDump.LOADNT is used to auto-load the kernel debug symbols. After that the command TASK.CrashDump.LOADREG could be used to load the context of all the available cores.

Data.LOAD.DAB

Build 13751 IMP 02105 Data.LOAD.DAB (Load DAB file) Data.SAVE.DAB (Save DAB file)	**DAB file format supported**

Build 88288 IMP 02497 Data.LOAD.Elf (Load ELF file) EXTension.LOAD (Load extension) Frame.view (Display stack frame) MMU.FORMAT (Define MMU table structure) Register.view (Display registers) TASK.List.MACHINES (List machines) TASK.List.TREE (Display tasks in a tree structure) TRANSlation.List (List MMU translation table) TRANSlation.TableWalk (Automatic MMU page table walk)	**Hypervisor-aware debugging feature introduced (Part 2-2)** - the debugger address translation (MMU and TRANS command groups) can be individually configured for each virtual machine by specifying a machineid in the address parameters. Where appropriate, MMU commands accept the new option /MACHINE <machineid>. Page tables of all machines can be viewed. Access class I: is used to specify intermediate addresses. - individual symbol sets can be loaded for each virtual machine by loading them onto the appropriate extended address: Data.LOAD.<type> <file> <accessclass>:<machineid>:::0 <options> Data.LOAD.<type> <file> <accessclass>:<machineid>:::<spaceID>::0 <options> - multiple awarenesses can be loaded to individual machines using the EXTension.LOAD command with the /machine option: EXTension.LOAD <awarenessfile> /MACHINE <machineid> [/NAME <extname>] - extension commands may use an infix to specify uniquely a command of a specific extension: TASK.[<extname>.]<extcommand> EXTension.[<extname>.]<extcommand> - new commands TASK.List.MACHINES and TASK.List.TREE are available to display the target's hypervisor/guest/process/thread structure - task names may be extended by the machine name, separated by three colons, to uniquely identify them: <machinename>:::<taskname> - register sets and stack frames of machines and tasks of arbitrary guests can be viewed, e.g.: Register.view /MACHINE <machineid> Frame.view /TASK "<machinename>:::<taskname>"
Build 60219 CHG 02421 Intel~ x86/x64 JTAG Debugger High-Level-Language Debugging Data.LOAD.Elf (Load ELF file)	**Logical memory access for Data.LOAD.Elf** Data.LOAD.Elf use logical memory access when load is not done to a physical address (x86/x64)
Build 42354 IMP 02331 Data.LOAD.Elf (Load ELF file)	**Load Debug info with Merge** The new option /MERGE <program> allows to merge debug information with existing pure symbols from an already loaded program.
Build 19417 FIX 02121 Data.LOAD.Elf (Load ELF file)	**Wrong "double" format for old ARM files** The alternate double format is not selected as default when loading old ARM ELF files. Workaround is to add the /ALTDOUBLE option to the load command.
Build 13751 IMP 02101 Data.LOAD.Elf (Load ELF file)	**Support for enum consts improved** New option /ENUMCONSTS to make enum constants available at any place in HLL expressions.

Build 9640 IMP 02053 Data.LOAD.COFF (Load COFF file) Data.LOAD.Elf (Load ELF file)	**Const variables supported** Constant variables in DWARF2 debug format are now also loaded.
Build 4928 IMP 01988 Data.LOAD.Elf (Load ELF file)	**DWARF2 sourcefile/module relation** Workarround for modules with wrong name (includefile instead of main source file) implemented.
Build 4728 IMP 01969 Data.LOAD.Elf (Load ELF file)	**Support for DWARF2/3 callframe info** Adding the /FRAME option will load DWARF2/DWARF3 call frame information. The Var.Frame window will use this information whenever possible. The implementation is is target architecture specific.
Build 4728 FIX 01924 MPC5xxx and SPC5xx Debugger Data.LOAD.Elf (Load ELF file)	**PowerPC e500 floating point returns** Floating point function return values are not displayed.
Build 2940 CHG 01896 Data.LOAD.Elf (Load ELF file)	**DWARF2 loader takes care of /Include** The /Include option is now required to show lines from include files.
Build 1241 IMP 01796 Data.LOAD.Elf (Load ELF file)	**Loading large ELF/DWARF files improved** Large files require less memory during download.
Build 1190 IMP 01669 Data.LOAD.Elf (Load ELF file)	**Bugfix for Metrowerks EPPC 8.0** Bugfix implemented for Metrowerks PPC 8.x compilers source lines. Files must be loaded with /MetroWerks /GlobTypes /BUGFIX options.
Build 1190 FIX 01661 Data.LOAD.Elf (Load ELF file)	**Floating point format wrong (METAWARE)** The double floating point format was wrong for Metaware ARM compiler files. Loading requires now the /METAWARE option.
Build 1161 FIX 01632 Data.LOAD.COFF (Load COFF file) Data.LOAD.Elf (Load ELF file)	**Compilation directory missing in DWARF2** The compilation directory in missing in the source filename in DWARF2 debug information when a directory is already specified for the source.
Build 1077 CHG 01087 Data.LOAD.COFF (Load COFF file) Data.LOAD.Elf (Load ELF file)	**COFF/ELF** COFF/ELF option /BUGFIX changed, option /ChainedStab for .stabs chaining

Data.LOAD.Ieee

Build 1079 FIX 01401 Data.LOAD.Ieee (Load IEEE-695 file)	**Tasking VX compiler** D.LOAD.I with new tasking VX compiler failed

Data.LOAD.IntelHex

Build 1077 IMP 01118 Data.LOAD.IntelHex (Load INTEL-HEX file)	**Optimized** D.LOAD.IntelHex optimized for better performance

Data.LOAD.OAT

Build 80996 IMP 02475 Android Debugging ARM Debugger Cortex-A/-R Armv8 Debugger Data.LOAD.OAT (Load OAT file)	**Android Debugging** The TRACE32 Android debug support has been extended for Android versions based on the Android RunTime (ART). This includes Android versions L, M and N. The new Android support allows the debugging of the ahead-of-time compiled Android framework and apps. TRACE32 automatically detects ahead-of-time compiled objects and and loads the DWARF/ELF info if available. Otherwise the debugger can parse the OAT data to extract the debug info. Additionally, TRACE32 supports the hybrid compilation introduced in Android N. For interpreted code, it is possible to display the stack frame with native to Java and Java to native transitions. A double click on a Java method displays the high level code together with the Dalvik disassembly. In case the code is just-in-time compiled, TRACE32 uses the symbols of the Android libart.so library to parse the JIT cache in order to get the names and ranges of the so-called hot methods. The new command Data.LOAD.OAT allows to load files generated by Android RunTime (ART).

Data.LOAD.PureHex

Build 13751 IMP 02065 Data.LOAD.PureHex (Load hex-byte file) Data.SAVE.PureHex (Save pure HEX file)	**new commands for handling purehex files**

Data.LOAD.S3record

Build 2940 IMP 01908 Data.LOAD.S3record (Load S3-Record file)	**Data.LOAD.S3record updates initial PC** The option /NOREG can be used to suppress this.

Data.LOAD.SysRof

Build 1133 FIX 01515 Data.LOAD.SysRof (Load RENESAS SYSROF file)	d.load.sysrof may crash

Data.LOAD.Ubrof

Build 4728 FIX 01920 S12 Debugger Data.LOAD.Ubrof (Load UBROF file)	**UBROF10 register variables wrong** Register variables may be shown wrong when UBROF10 format is used.
Build 2940 FIX 01874 Data.LOAD.Ubrof (Load UBROF file)	**UBROF loader fails on untagged member** The UBROF file format loader may fail when the file contains untagged structure members.
Build 1253 FIX 01845 Data.LOAD.Ubrof (Load UBROF file)	**bug fixed UBROF V9 loader for V850** Loader failed.
Build 1228 IMP 01764 Data.LOAD.Ubrof (Load UBROF file)	**UBROF-10 debug format supported**

Data.LOAD.eXe

Build 1190 IMP 01670 Data.LOAD.eXe (Load EXE file)	**EXE/PDB loader looks at alternate file** The EXE/PDB (WindowsCE) loader searches also for renamed .PDB files.
Build 1161 CHG 01615 Data.LOAD.eXe (Load EXE file)	**EXE code was loaded to physical memory** Data.LOAD.eXe was loading the code to physical memory (A:) also for non-X86 architectures. Now the code is loaded to regular memory - including the MMU translation, when enabled.

Data.Out

Build 88288 CHG 02502 ASM Debugger Data.Out (Write port) Data.Set (Modify memory)	**Data.Set/Data.Out commands allow different access/data width** Data.Set and Data.Out commands allow specifying an access width additionally to the data width.

Data.PATTERN

Build 34458 IMP 02270 Data.COPY (Copy memory) Data.PATTERN (Fill memory with pattern) Data.Set (Modify memory)	**option DIFF added** New option "DIFF" for commands Data.Set, Data.PATTERN and Data.COPY. Can be used to compare memory against values.

Data.PROGRAM

Build 105499 IMP 02556 Integrated Development Environment DIALOG.Program (Interactive programming) EDIT (TRACE32 editor) MENU.Program (Interactive programming) PEDIT (Edit PRACTICE script) Analyzer.Program (Program trigger unit) Data.PROGRAM (Editor for writing assembler program) Integrator.Program (Program trigger unit) Probe.Program (Program trigger unit) PER.Program (Interactive programming)	**new internal text editor** New text editor version with syntax highlighting, nesting awareness, undo/redo and other features activated.

Data.PROLOG

Build 1133 FIX 01521 Data.PROLOG (Automatic data modification on program execution start)	Data.PROLOG not working correctly when Go made on breakpoint location (Step-Go sequence)

Data.PROLOG.state

Build 60219 IMP 02423 Debugger Features Data.EPILOG.state (Display data epilogs) Data.PROLOG.state (Display data prologs) Data.SOFTEPILOG.state (Display data softepilogs) Data.SOFTPROLOG.state (Display data softprologs) Data.STARTUP.state (Startup data state display) Data.TIMER.state (Timer state display) FLASH.EPILOG.state (Display FLASH epilogs) FLASH.PROLOG.state (Display FLASH prologs)	**Full SMP support for auto sequences** Auto-sequences can be supplied for each core in an SMP system.

Build 69655 IMP 02442 High-Level-Language Debugging Trace Features <trace>.DRAW.channel (Plot no-data values against time) <trace>.DRAW.Data (Plot data values against time) <trace>.DRAW.Var (Plot variable values against time) Data.DRAW (Graphical memory display of arrays) Data.PROfile (Graphical display of data value) Var.PROfile (Graphical display of variable)	**Multiple variables of different type can be drawn** TRACE32 allows to draw multiple variables/channels with different formats e.g. integers and floats for all listed commands.
Build 50104 IMP 02376 Data.PROfile (Graphical display of data value) Var.PROfile (Graphical display of variable)	**New command Var.PROfile** HLL counterpart to Data.PROfile to show variables over time.

Data.SAVE.<format>

Build 4728 FIX 01928 Data.SAVE.<format> (Save data in file with specified format)	**D.SAVE.ASCII... using wrong format**
Build 1246 FIX 01815 Data.SAVE.<format> (Save data in file with specified format)	**Data.SAVE not working on Linux host**
Build 1077 FIX 01319 Data.SAVE.<format> (Save data in file with specified format)	D.SAVE.xx may crash on some processors (e.g. C166) with PowerDebug USB/Ethernet

Data.SAVE.CCSDAT

Build 19417 IMP 02165 Data.LOAD.CCSDAT (Load CCSDAT file) Data.SAVE.CCSDAT (Save CCSDAT file)	**New file format CCSDAT supported** Format is compatible with TI Code Composer Studio (R) data file format.

Data.SAVE.DAB

Build 13751 IMP 02107 Data.SAVE.DAB (Save DAB file)	**Save commands support muliple ranges**
Build 13751 IMP 02105 Data.LOAD.DAB (Load DAB file) Data.SAVE.DAB (Save DAB file)	**DAB file format supported**

Data.SAVE.PureHex

Build 13751 IMP 02065 Data.LOAD.PureHex (Load hex-byte file) Data.SAVE.PureHex (Save pure HEX file)	**new commands for handling purehex files**

Data.SOFTEPILOG

Build 1206 IMP 01736 Data.SOFTEPILOG (Automated sequence after setting software breakp.) Data.SOFTPROLOG (Automated sequence before setting software breakp.)	**New commands to prolog software bp write** New command Data.SOFTPROLOG and Data.SOFTEPILOG to execute a memory access sequence before and after writing software breakpoints. Useful to "enable" memory writes to otherwise read-only memory (e.g. C55x PDRAM).

Data.SOFTEPILOG.state

Build 60219 IMP 02423 Debugger Features Data.EPILOG.state (Display data epilogs) Data.PROLOG.state (Display data prologs) Data.SOFTEPILOG.state (Display data softepilogs) Data.SOFTPROLOG.state (Display data softprologs) Data.STARTUP.state (Startup data state display) Data.TIMER.state (Timer state display) FLASH.EPILOG.state (Display FLASH epilogs) FLASH.PROLOG.state (Display FLASH prologs)	**Full SMP support for auto sequences** Auto-sequences can be supplied for each core in an SMP system.

Data.SOFTPROLOG

Build 1206 IMP 01736 Data.SOFTEPILOG (Automated sequence after setting software breakp.) Data.SOFTPROLOG (Automated sequence before setting software breakp.)	**New commands to prolog software bp write** New command Data.SOFTPROLOG and Data.SOFTEPILOG to execute a memory access sequence before and after writing software breakpoints. Useful to "enable" memory writes to otherwise read-only memory (e.g. C55x PDRAM).

Data.SOFTPROLOG.state

Build 60219 IMP 02423 Debugger Features Data.EPILOG.state (Display data epilogs) Data.PROLOG.state (Display data prologs) Data.SOFTEPILOG.state (Display data softepilogs) Data.SOFTPROLOG.state (Display data softprologs) Data.STARTUP.state (Startup data state display) Data.TIMER.state (Timer state display) FLASH.EPILOG.state (Display FLASH epilogs) FLASH.PROLOG.state (Display FLASH prologs)	**Full SMP support for auto sequences** Auto-sequences can be supplied for each core in an SMP system.

Data.STANDBY

Build 105499 IMP 02557 TriCore Debugger Data.STANDBY (Standby data-sequences)	**new command Data.STANDBY added**

Data.STARTUP.state

Build 60219 IMP 02423 Debugger Features Data.EPILOG.state (Display data epilogs) Data.PROLOG.state (Display data prologs) Data.SOFTEPILOG.state (Display data softepilogs) Data.SOFTPROLOG.state (Display data softprologs) Data.STARTUP.state (Startup data state display) Data.TIMER.state (Timer state display) FLASH.EPILOG.state (Display FLASH epilogs) FLASH.PROLOG.state (Display FLASH prologs)	**Full SMP support for auto sequences** Auto-sequences can be supplied for each core in an SMP system.

Data.STRING

Build 51144 IMP 02397 Data.STRING (ASCII display)	**New formatting option %CONTinue for Data.STRING command** Allows to add string to output in AREA without having a newline first.

Data.Set

Build 88288 CHG 02502 ASM Debugger Data.Out (Write port) Data.Set (Modify memory)	**Data.Set/Data.Out commands allow different access/data width** Data.Set and Data.Out commands allow specifying an access width additionally to the data width.
Build 34458 IMP 02270 Data.COPY (Copy memory) Data.PATTERN (Fill memory with pattern) Data.Set (Modify memory)	**option DIFF added** New option "DIFF" for commands Data.Set, Data.PATTERN and Data.COPY. Can be used to compare memory against values.

Data.TIMER.state

Build 60219 IMP 02423 Debugger Features Data.EPILOG.state (Display data epilogs) Data.PROLOG.state (Display data prologs) Data.SOFTEPILOG.state (Display data softepilogs) Data.SOFTPROLOG.state (Display data softprologs) Data.STARTUP.state (Startup data state display) Data.TIMER.state (Timer state display) FLASH.EPILOG.state (Display FLASH epilogs) FLASH.PROLOG.state (Display FLASH prologs)	**Full SMP support for auto sequences** Auto-sequences can be supplied for each core in an SMP system.

Data.Test

Build 1224 FIX 01763 Data.Test (Memory integrity test)	**Memory test at page boundaries** Data.Test delivered wrong results at certain memory boundaries with certain processor families.
Build 1133 IMP 01483 Data.Test (Memory integrity test)	Data.TEST now set to QUAD/LONG/WORD as default memory width

Data.USRACCESS

Build 100486 IMP 02531 ASM Debugger Data.USRACCESS (Prepare USR access)	**Set-up of Data.USRACCESS is reset on SYStem.Mode change** Set-up of Data.USRACCESS is reset on every change of the SYStem.Mode. If Data.USRACCESS is not configured, data accesses to access class USR cause an error.
Build 1228 FIX 01774 Data.USRACCESS (Prepare USR access)	**Data.USRACCESS does not load program** The filename was accepted, but the program not loaded. The software breakpoint at the return address was also not set.

Data.VECTOR

Build 117056 IMP 02575 JTAG Debugger Data.VECTOR (Display memory as vectors)	**new command Data.VECTOR** The new Data.VECTOR command allows to display memory contents as vectors side by side.

Data.View

Build 34458 IMP 02268 Data.View (Display memory) PER.view (Display peripherals)	**command option CORE added** CORE option available in PER, Data.View and Data.PRINT commands.

Data.WRITESTRING

Build 30461 IMP 02247 Data.WRITESTRING (Write string to PRACTICE file)	**New command Data.WRITESTRING** Writes string from target memory to PRACTICE file.

Data.dump

Build 30461 IMP 02240 Data.dump (Memory dump)	**New STRING option for memory dumps** Allows to display strings in a dump window (newline separated lines).
Build 1258 IMP 01853 SETUP.ASCIITEXT (Configure ASCII text display) Data.dump (Memory dump) Var.View (Display variables)	**UTF-8 support for ASCII strings/dumps** UTF-8 characters is enabled with SETUP.ASCIITEXT UTF-8.
Build 1077 FIX 01209 Data.dump (Memory dump)	**Spotlight** Selection fails in data.dump windows with spotlight option

ECHO

Build 88288 IMP 02487 Script Language PRACTICE ECHO (Write text and data to an AREA window (with format decoration)) PRINTF (Write formatted data to an AREA window)	**Streamlined output to AREA window from PRACTICE** The ECHO command keeps type and radix information of data and PRACTICE macros. This improves the readablity of the output. The PRINTF command allows to format outputs in the style of the printf() function in C/C++.

EDIT

Build 105499 IMP 02556 Integrated Development Environment DIALOG.Program (Interactive programming) EDIT (TRACE32 editor) MENU.Program (Interactive programming) PEDIT (Edit PRACTICE script) Analyzer.Program (Program trigger unit) Data.PROGRAM (Editor for writing assembler program) Integrator.Program (Program trigger unit) Probe.Program (Program trigger unit) PER.Program (Interactive programming)	**new internal text editor** New text editor version with syntax highlighting, nesting awareness, undo/redo and other features activated.

ENCRYPTDO

Build 19417 IMP 02155 ENCRYPTDO (Encrypt a PRACTICE script (*.cmm))	**Possibility to create encrypted PER file** Encrypts PER files that can only be shown, but not dumped or edited.

ENTRY

Build 65657 IMP 02430 Script Language PRACTICE ENTRY (Parameter passing) PARAMETERS (Parameter fetching) RETURNVALUES (Take return values)	**New commands PARAMETERS and RETURNVALUES** The PARAMETER command first creates PRIVATE macros for all macro names not found in the current scope and then assigns the passed values to the macros. RETURNVALUES takes the return values of a PRACTICE script/subroutine. An error message is generated, if a macro name is used, that cannot be found in the current scope. Both commands fix known issues of the ENTRY command.

ETA.PROfileChart

Build 30461 IMP 02236 ETA.PROfileChart (Power consumption by function as function of time)	**New command ETA.PROfileChart** Command shows power consumption by function as function of time.

ETM

Build 22490 IMP 02208 ETM (Embedded Trace Macrocell (ETM)) ITM (CoreSight ITM (Instrumentation Trace Macrocell))	**Merged ETM/ITM analyzer for Cortex-M** The merged analyzer allows a correlated processing of ETM and ITM trace data. This allows details nesting analysis for targets using an RTOS.
Build 19417 FIX 02124 ARM/Cortex Trace (parallel) Break.Set (Set breakpoints) ETM (Embedded Trace Macrocell (ETM))	**ETM selective data trace** Selective data trace may miss cycles in ETMv3.

ETM.DataTrace

Build 88288 IMP 02504 ARM/Cortex Trace (parallel) ETM.DataTrace (Configure data-trace) ETM.TimeStampsTrace (Specify data trace correlation method (ETMv4))	**ETMv4 data trace correlation extended to use timestamps** ETM.TimeStampTrace ON advises TRACE32 to assigns the data address/data value information of the data trace stream to the appropriate load/store instruction by using the timestamps. This requires that a timestamp unit is available.
Build 13751 IMP 02108 ARM Debugger ETM.DataTrace (Configure data-trace)	**New data trace modes**

ETM.PortSize

Build 19417 IMP 02143 ARM/Cortex Trace (parallel) Analyzer (Trace method Analyzer, recording, and analysis commands) ETM.PortSize (Define trace port width)	**Trace buffer usage for 8bit ETM changed** Doubles available trace records and increases frequency limit of older trace modules for ETM ports with 8 or less bits.
Build 19417 IMP 02131 ARM/Cortex Trace (parallel) ETM.PortSize (Define trace port width)	**ETM support for special port sizes** Supporting also 3,5,6,7 bits (in normal mode) and unaligned trace data.

ETM.PseudoDataTrace

Build 22490 IMP 02201 ARM/Cortex Trace (parallel) ETM.PseudoDataTrace (Enable pseudo data trace detection)	**Pseudo data trace for ARM ETMv3** Pseudo data trace allows to transmit data information via a pure program trace by target code instrumentation. This brings limited data trace to Cortex-A8 core.

ETM.Register

Build 19417 IMP 02146 ARM/Cortex Trace (parallel) ETM.Register (Display the ETM registers)	**ETM.Register window enhanced** Contains now also related CoreSight components.

ETM.ReturnStack

Build 26464 IMP 02225 ARM/Cortex Trace (parallel) ETM.ReturnStack (Enable return stack tracing mode)	**New command EMU.ETM.ReturnStack** Enables return stack compression in ETM (PFT) trace.

ETM.Set

Build 38589 IMP 02300 ARM Debugger ETM.Set (Precise control of ETM trigger events)	**ETM.Set command extended** Allows to supply the data access width to control the comparison.
Build 9640 IMP 02057 ETM.Set (Precise control of ETM trigger events)	**EMU.Set command extended** Support for context ID comparators, data mismatch compare and executepass/executefail comparisons. Also support for TraceON/TraceOFF control.

ETM.TimeStampsTrace

Build 88288 IMP 02504 ARM/Cortex Trace (parallel) ETM.DataTrace (Configure data-trace) ETM.TimeStampsTrace (Specify data trace correlation method (ETMv4))	**ETMv4 data trace correlation extended to use timestamps** ETM.TimeStampTrace ON advises TRACE32 to assigns the data address/data value information of the data trace stream to the appropriate load/store instruction by using the timestamps. This requires that a timestamp unit is available.

ETM.TraceCORE

Build 125398 IMP 02587 ARM Debugger CEVA-X Debugger Trace Analyzer ETM.TraceCORE (Core specific default tracing)	**new command ETM.TraceCORE** New command ETM.TraceCORE to allow core specific default tracing.

ETM.TraceDataPriority

Build 125398 IMP 02589 Trace Analyzer ETM.TraceDataPriority (Define data trace priority)	**new command ETM.TraceDataPriority** New command ETM.TraceDataPriority to define data trace priority on ETMv4.

Build 88288 IMP 02497 Data.LOAD.Elf (Load ELF file) EXTension.LOAD (Load extension) Frame.view (Display stack frame) MMU.FORMAT (Define MMU table structure) Register.view (Display registers) TASK.List.MACHINES (List machines) TASK.List.TREE (Display tasks in a tree structure) TRANSlation.List (List MMU translation table) TRANSlation.TableWalk (Automatic MMU page table walk)	**Hypervisor-aware debugging feature introduced (Part 2-2)** - the debugger address translation (MMU and TRANS command groups) can be individually configured for each virtual machine by specifying a machineid in the address parameters. Where appropriate, MMU commands accept the new option /MACHINE <machineid>. Page tables of all machines can be viewed. Access class I: is used to specify intermediate addresses. - individual symbol sets can be loaded for each virtual machine by loading them onto the appropriate extended address: Data.LOAD.<type> <file> <accessclass>:<machineid>:::0 <options> Data.LOAD.<type> <file> <accessclass>:<machineid>:::<spaceID>::0 <options> - multiple awarenesses can be loaded to individual machines using the EXTension.LOAD command with the /machine option: EXTension.LOAD <awarenessfile> /MACHINE <machineid> [/NAME <extname>] - extension commands may use an infix to specify uniquely a command of a specific extension: TASK.[<extname>.]<extcommand> EXTension.[<extname>.]<extcommand> - new commands TASK.List.MACHINES and TASK.List.TREE are available to display the target's hypervisor/guest/process/thread structure - task names may be extended by the machine name, separated by three colons, to uniquely identify them: <machinename>:::<taskname> - register sets and stack frames of machines and tasks of arbitrary guests can be viewed, e.g.: Register.view /MACHINE <machineid> Frame.view /TASK "<machinename>:::<taskname>"

Eval

Build 13751 ODD 02064 Integrated Development Environment Eval (Evaluate expression)	**EVAL EVAL() EVAL.TYPE()** behavior of command EVAL changed: now an empty expression parameter sets value to 0 and value type to EMPTY instead of using the old values PRACTICE functions EVAL() and EVAL.TYPE(): expression type EMPTY added

FDX

Build 1133 IMP 01488 FDX (Trace method FDX)	New trace method FDX

FDX.METHOD

Build 19417 CHG 02133 ARM Debugger FDX.METHOD (Select communication channel)	**FDX method DCC wrong** Is named "DCC4" now. "DCC3" method introduced.

FDX.OutChannel

Build 1161 FIX 01642 FDX.OutChannel (Outchannel state display)	**FDX fails to transfer packets** FDX transfers from target to host fails for packet sizes where the lower byte of the packet size is in the range 0xfc to 0xff.

Build 105499 IMP 02555 ICD In-Circuit Debugger FLASH.Create (Declare FLASH) FLASH.CFI (Generate FLASH declaration by CFI) FLASH.CHANGEtype (Changes the FLASH type) STOre (Store settings as PRACTICE script)	**new option /DISableBulkErase added** For commands FLASH.Create and FLASH.CHANGEtype option /DISableBulkErase added. Now it is also possible to declare it explicitly for non-CFI flash devices to avoid unintended chip erase for truncated flash address range.
Build 69655 IMP 02441 FLASH Programming (Memory-Mapped) FLASH.CFI (Generate FLASH declaration by CFI) FLASH.TARGET2 (Define second target controlled algorithm)	**Simultaneous programming of multiple NOR flash devices** The new command FLASH.TARGET2 allows the simultaneous programming of multiple flash devices with two different flash algorithms. This is needed for example to program processor internal flash and processor external NOR flash/HyperFlash/QSPI flash together. The new option /TARGET2 for the FLASH.CFI command allows to link a second flash algorithm to a CFI conform flash device. The new function FLASH.TARGET2.FILE() is provided to get the filename of the second flash algorithm.
Build 7466 IMP 02016 FLASH.CFI (Generate FLASH declaration by CFI)	**New command FLASH.CFI** FLASH.CFI command is generating flash declaration automatically using CFI (Common Flash Interface) information of the flash device.

FLASH.CHANGEtype

Build 105499 IMP 02555 ICD In-Circuit Debugger FLASH.Create (Declare FLASH) FLASH.CFI (Generate FLASH declaration by CFI) FLASH.CHANGEtype (Changes the FLASH type) STOre (Store settings as PRACTICE script)	**new option /DISableBulkErase added** For commands FLASH.Create and FLASH.CHANGEtype option /DISableBulkErase added. Now it is also possible to declare it explicitly for non-CFI flash devices to avoid unintended chip erase for truncated flash address range.

FLASH.Create

Build 105499 IMP 02555 ICD In-Circuit Debugger FLASH.Create (Declare FLASH) FLASH.CFI (Generate FLASH declaration by CFI) FLASH.CHANGEtype (Changes the FLASH type) STOre (Store settings as PRACTICE script)	**new option /DISableBulkErase added** For commands FLASH.Create and FLASH.CHANGEtype option /DISableBulkErase added. Now it is also possible to declare it explicitly for non-CFI flash devices to avoid unintended chip erase for truncated flash address range.
Build 42354 IMP 02347 TriCore Debugger TriCore Parallel Trace FLASH.Create (Declare FLASH)	**new command option /BootModeHeaDer**
Build 5151 ERR 01996 FLASH Programming (Memory-Mapped) FLASH.Create (Declare FLASH)	**Flash family codes AM29x256 not working** Flash programming with family codes AM29LV256, AM29LV256B, AM29M256, AM29M256B, AM29N256 and AM29M2562 is failing after a few words (since 09 March 2006).

FLASH.EPILOG.state

Build 60219 IMP 02423 Debugger Features Data.EPILOG.state (Display data epilogs) Data.PROLOG.state (Display data prologs) Data.SOFTEPILOG.state (Display data softepilogs) Data.SOFTPROLOG.state (Display data softprologs) Data.STARTUP.state (Startup data state display) Data.TIMER.state (Timer state display) FLASH.EPILOG.state (Display FLASH epilogs) FLASH.PROLOG.state (Display FLASH prologs)	**Full SMP support for auto sequences** Auto-sequences can be supplied for each core in an SMP system.

FLASH.HOOKSCRIPT

Build 93173 IMP 02506 FLASH Programming (Memory-Mapped) FLASH.HOOKSCRIPT (PRACTICE script based FLASH programming prolog)	**PRACTICE script based FLASH programming prolog** The <hook_script> can perform checks, set-ups etc. to guarantee that the FLASH programming works properly afterwards, e.g. to avoid fatal problems that might occur when the FLASH programming erases or modifies FLASH sectors that contain information that is necessary to operate the debug interface or the chip. First use case: TriCore TC3xx.

FLASH.PROLOG.state

Build 60219 IMP 02423 Debugger Features Data.EPILOG.state (Display data epilogs) Data.PROLOG.state (Display data prologs) Data.SOFTEPILOG.state (Display data softepilogs) Data.SOFTPROLOG.state (Display data softprologs) Data.STARTUP.state (Startup data state display) Data.TIMER.state (Timer state display) FLASH.EPILOG.state (Display FLASH epilogs) FLASH.PROLOG.state (Display FLASH prologs)	**Full SMP support for auto sequences** Auto-sequences can be supplied for each core in an SMP system.

FLASH.ReProgram

Build 22490 IMP 02189 MPC5xxx and SPC5xx Debugger MPC55xx/MPC56xx NEXUS Debugger and Trace FLASH.ReProgram (Re-program FLASH)	**MPC55XX/56XX** FLASH.ReProgram now supports programming Flash with ECC errors without prior FLASH.Erase call
Build 13751 IMP 02069 FLASH.ReProgram (Re-program FLASH)	**New command FLASH.ReProgram** New FLASH.ReProgram command for optimized flash programming performance.

FLASH.SPI.CFI

Build 130863 IMP 02613 FLASH.SPI.CFI (Generate SPI FLASH sector declaration by CFI)	**new options /QuadPI and /OctalPI for command FLASH.SPI.CFI**
Build 125398 IMP 02585 JTAG Debugger FLASH.SPI.CFI (Generate SPI FLASH sector declaration by CFI) FLASHFILE.SPI.CFI (Generate SPI FLASH sector declaration by CFI)	**new commands FLASH.SPI.CFI and FLASHFILE.SPI.CFI** New commands introduced to generate the FLASH declaration by using CFI information.

FLASH.SPI.CMD

Build 125398 CHG 02592 JTAG Debugger FLASH.SPI.CMD (Send data to SPI FLASH device) FLASHFILE.SPI.CMD (Send data to SPI FLASH device)	**renamed commands FLASH.SPICMD and FLASHFILE.SPICMD** The commands FLASH.SPICMD and FLASHFILE.SPICMD have been renamed to FLASH.SPI.CMD and FLASHFILE.SPI.CMD.

FLASH.SPI.GETSFDP

Build 125398 IMP 02588 JTAG Debugger FLASH.SPI.GETSFDP (Read FLASH discovery parameters) FLASHFILE.SPI.GETSFDP (Read FLASH discovery parameters)	**new commands FLASH.SPI.GETSFDP and FLASHFILE.SPI.GETSFDP** New commands introduced to read Serial Flash Discovery Parameters from SPI FLASH. The several parameter values can be utilized afterwards using the FLASH.SPI.SFDP() and FLASHFILE.SPI.SFDP() functions.

FLASH.TARGET

Build 42354 IMP 02345 FLASH.TARGET (Define target controlled algorithm)	**new command option /FirmWareRAM** Option supports protection and restoring of on-chip FLASH programming firmware during FLASH programming.
Build 1253 IMP 01846 FLASH.TARGET (Define target controlled algorithm)	**target based flash programming for C55x** Support for target based flash programming for C55x family.

FLASH.TARGET2

Build 69655 IMP 02441 FLASH Programming (Memory-Mapped) FLASH.CFI (Generate FLASH declaration by CFI) FLASH.TARGET2 (Define second target controlled algorithm)	**Simultaneous programming of multiple NOR flash devices** The new command FLASH.TARGET2 allows the simultaneous programming of multiple flash devices with two different flash algorithms. This is needed for example to program processor internal flash and processor external NOR flash/HyperFlash/QSPI flash together. The new option /TARGET2 for the FLASH.CFI command allows to link a second flash algorithm to a CFI conform flash device. The new function FLASH.TARGET2.FILE() is provided to get the filename of the second flash algorithm.

FLASH.state

Build 65657 02433 FLASH Programming (Memory-Mapped) FLASH.state (FLASH programming dialog)	**FLASH.state command** New dialog to create, check and modify the setup for NOR and onchip flash programming.

FLASHFILE.DUMP

Build 19417 IMP 02151 FLASHFILE.DUMP (Dump FLASH)	**New command FLASHFILE.DUMP** Dumps NAND flash contents.

FLASHFILE.SPI.CFI

Build 125398 IMP 02585 JTAG Debugger FLASH.SPI.CFI (Generate SPI FLASH sector declaration by CFI) FLASHFILE.SPI.CFI (Generate SPI FLASH sector declaration by CFI)	**new commands FLASH.SPI.CFI and FLASHFILE.SPI.CFI** New commands introduced to generate the FLASH declaration by using CFI information.

FLASHFILE.SPI.CMD

Build 125398 CHG 02592 JTAG Debugger FLASH.SPI.CMD (Send data to SPI FLASH device) FLASHFILE.SPI.CMD (Send data to SPI FLASH device)	**renamed commands FLASH.SPICMD and FLASHFILE.SPICMD** The commands FLASH.SPICMD and FLASHFILE.SPICMD have been renamed to FLASH.SPI.CMD and FLASHFILE.SPI.CMD.

FLASHFILE.SPI.GETSFDP

Build 125398 IMP 02588 JTAG Debugger FLASH.SPI.GETSFDP (Read FLASH discovery parameters) FLASHFILE.SPI.GETSFDP (Read FLASH discovery parameters)	**new commands FLASH.SPI.GETSFDP and FLASHFILE.SPI.GETSFDP** New commands introduced to read Serial Flash Discovery Parameters from SPI FLASH. The several parameter values can be utilized afterwards using the FLASH.SPI.SFDP() and FLASHFILE.SPI.SFDP() functions.

FPU.view

Build 51144 IMP 02385 AVX.view (Display AVX registers) AVX512.view (Display AVX512 registers) FPU.view (Display FPU registers) MMX.view (Open MMX register window) SSE.view (Display SSE registers) VPU.view (Display ALTIVEC register window)	**Larger than 64 bit SIMD register sets** FPU, VPE, SPE, MMX, SSE, AVX, AVX512 - Parser for wide registers (>64 bit) unified, values wider than 64 bit possible.
Build 7181 CHG 02004 MPC5xx/8xx Debugger FPU.view (Display FPU registers)	**FPU support for PPC440EP** preliminary 'diag 3201 138 <RAM addr> command is replaced by a official TRACE32 command 'SYStem.Option.SCRATCH <RAM_addr>'

Frame

Build 51144 IMP 02394 Frame (Call-tree and context)	**Better support for inline function debugging** Active inline functions now shown in status bar and Frame windows.
Build 19417 CHG 02130 Frame (Call-tree and context)	**Flag command group renamed to FLAG** The abbreviation "F" of the command group "FLAG" removed.

Build 93173 IMP 02512 Integrated Development Environment Frame.view (Display stack frame) List.auto (Display program listing) Var.View (Display variables) Var.Watch (Open Var.Watch window)	**Mouse interface changed for all text windows** Free text selection is now possible with the left mouse. Click and drag on a selection allows drag and drop.
Build 88288 IMP 02497 Data.LOAD.Elf (Load ELF file) EXTension.LOAD (Load extension) Frame.view (Display stack frame) MMU.FORMAT (Define MMU table structure) Register.view (Display registers) TASK.List.MACHINES (List machines) TASK.List.TREE (Display tasks in a tree structure) TRANSlation.List (List MMU translation table) TRANSlation.TableWalk (Automatic MMU page table walk)	**Hypervisor-aware debugging feature introduced (Part 2-2)** - the debugger address translation (MMU and TRANS command groups) can be individually configured for each virtual machine by specifying a machineid in the address parameters. Where appropriate, MMU commands accept the new option /MACHINE <machineid>. Page tables of all machines can be viewed. Access class I: is used to specify intermediate addresses. - individual symbol sets can be loaded for each virtual machine by loading them onto the appropriate extended address: Data.LOAD.<type> <file> <accessclass>:<machineid>:::0 <options> Data.LOAD.<type> <file> <accessclass>:<machineid>:::<spaceID>::0 <options> - multiple awarenesses can be loaded to individual machines using the EXTension.LOAD command with the /machine option: EXTension.LOAD <awarenessfile> /MACHINE <machineid> [/NAME <extname>] - extension commands may use an infix to specify uniquely a command of a specific extension: TASK.[<extname>.]<extcommand> EXTension.[<extname>.]<extcommand> - new commands TASK.List.MACHINES and TASK.List.TREE are available to display the target's hypervisor/guest/process/thread structure - task names may be extended by the machine name, separated by three colons, to uniquely identify them: <machinename>:::<taskname> - register sets and stack frames of machines and tasks of arbitrary guests can be viewed, e.g.: Register.view /MACHINE <machineid> Frame.view /TASK "<machinename>:::<taskname>"

FramePOS

Build 100486 IMP 02526 Integrated Development Environment CmdPOS (Controls the position of TRACE32 in MWI window mode) FramePOS (Controls the position of TRACE32 in MDI window mode) WinPOS (Define window dimensions and window name)	**Improved support for multimonitor desktops under MS Windows** The maximum possible window creation size was enlarged to cover the full multimonitor desktop size under MS Windows.
Build 1077 IMP 01261 FramePOS (Controls the position of TRACE32 in MDI window mode)	**FRAMEPOS** new command FramePOS to define size, position and mode of main window

GOTO

Build 1161 FIX 01618 GOTO (Local script jump)	**PRACTICE may cause nesting error** Executing a GOTO command immediatelly after a PRACTICE block begin may cause a PRACTICE nesting error when executed the second time.

GROUP.Create

Build 93173 IMP 02518 Trace Analyzer GROUP.Create (Create a new group)	**GROUP command group supports access classes and machineIDs** GROUP command group now supports access classes (such as N: Z: H: for the ARM architecture) and MachineIDs.

GROUP.HIDE

Build 4728 FIX 01922 GROUP.HIDE (Hide group from debugging) <trace>.STATistic.Func (Nesting function runtime analysis)	**Function statistics with disabled groups** Calls inside the group have caused wrong internal/external times and calling counts.

Go

Build 1151 FIX 01605 ARM Debugger Go (Debug control, program execution, and real-time emulation)	**OMAP1510** bug fixed in OMAP1510 multicore synchronous GO (caused ARM to step only)
Build 1077 FIX 01198 Go (Debug control, program execution, and real-time emulation)	**80186** bug fixed 80186, caused GO to fail when stack was in onchip memory

Go.Back

Build 1151 CHG 01588 ICE In-Circuit Emulator Go.Back (Go back in program (CTS))	**commands renamed** Break.SELect FORE -> Break.SELect FOREGROUND Break.SELect BACK -> Break.SELect BACKGROUND Break.Back -> Break.BackGround Go.Back -> Go.BackGround Go.Back is used for CTS now

Go.BackEntry

Build 1264 IMP 01861 ICE In-Circuit Emulator Go.BackEntry (Go back in program to function entry (CTS)) Step.BackOver (Step back)	**new commands for CTS introduced** Go.BackEntry Go.BackTillViolation Go.TillViolation Step.BackOver
Build 1264 IMP 01857 Go.BackEntry (Go back in program to function entry (CTS)) Step.BackOver (Step back)	**New CTS backward execution features** Step-Back-Over HLL or assembler and Go-Back-Till_Entry supported.

Go.Return

Build 76594 ODD 02453 High-Level-Language Debugging Go.Return (Complete HLL function)	**Behavior of Go.Return command was changed** First Go.Return stops now at function epilog, here local variables are still valid. Second Go.Return stops at the return of the function.

Go.direct

Build 1224 FIX 01746 PPC400 Debugger PPC440 Debugger Go.direct (Start the program execution) RunTime.state (Display results)	**Runtime counter stop randomly** Internal driver error. Wrong mask for JDSR[PSP] bit.

HELP

Build 56057 IMP 02413 Integrated Development Environment HELP (Online help)	**support of additional PDF viewer types added** The TRACE32 online help system officially supports the usage of alternative PDF viewer different from Adobe Reader. More details are described inside the IDE User's Guide.
Build 1241 FIX 01808 HELP (Online help)	**PDF online help not working on Linux** Causes always "file not found" error message in acrobat reader.
Build 1161 IMP 01629 HELP (Online help) STOre (Store settings as PRACTICE script)	**New Online Help PDF** The online help is completely changed from Winhelp to PDF help. The following functions are changed: - all HELP commands and the help dialog - context help - error messages and warnings Until the new PDF help is not installed, the main Winhelp functions will still work.

HELP.Bookmark.ADD.Find

Build 1224 FIX 01745 HELP.Bookmark.ADD.file (Add file to bookmark list) HELP.Bookmark.ADD.Find (Add file to bookmark list) HELP.Bookmark.ADD.Index (Add file to bookmark list)	**Bugfix help.bookmark.add**

HELP.Bookmark.ADD.Index

Build 1224 FIX 01745 HELP.Bookmark.ADD.file (Add file to bookmark list) HELP.Bookmark.ADD.Find (Add file to bookmark list) HELP.Bookmark.ADD.Index (Add file to bookmark list)	**Bugfix help.bookmark.add**

HELP.Bookmark.ADD.file

Build 1224 FIX 01745 HELP.Bookmark.ADD.file (Add file to bookmark list) HELP.Bookmark.ADD.Find (Add file to bookmark list) HELP.Bookmark.ADD.Index (Add file to bookmark list)	**Bugfix help.bookmark.add**

HELP.FILTER.Delete

Build 22490 ODD 02213 Help System HELP.FILTER.Delete (Delete filter from filter list)	**online help** command HELP.FILTER.Delete syntax change - parameter is filter name instead of number now

HELP.Find

Build 1190 IMP 01676 HELP.Find (Perform a full-text search)	**Online Help: Find** Complete change of the find algorithm to get better search results.

IF

Build 50104 CHG 02366 Script Language PRACTICE IF (Conditional script execution) RePeaT (Loop with check at end of loop) WHILE (Loop with check at start of loop)	**Empty lines in PRACTICE allowed** Empty lines are now ignored in PRACTICE. This may change the execution flow of existing programs (e.g. when an IF, REPEAT or WHILE statement is followed by an empty line).

IProbe.state

Build 100486 IMP 02527 Trace Analyzer IProbe.state (Display the IProbe configuration window)	**IProbe command group is supported by PowerTrace Serial** As soon as the IProbe is switched from the DISable state to OFF/Arm state, the trace memory is split up. 50% of the trace memory can still be used to record the trace data exported via the Aurora/PCIe trace port. The other 50% are assigned to the IProbe.

ISTATistic.ADD

Build 100486 CHG 02542 Trace-based Profiling ISTATistic.ADD (Add trace contents to ISTAT database)	**PC snooping can be used by ISTAT commands** TRACE32 takes instruction execution information from Analyzer/Onchip trace and PC/timestamp information from the SNOOPer trace in order to improve timestamp accuracy.

ISTATistic.EXPORT.CSV

Build 125398 IMP 02586 Trace Analyzer COVerage.EXPORT.CSV (Export coverage data in CSV format) ISTATistic.EXPORT.CSV (Export instruction statistics in CSV format)	**new commands COVerage.EXPORT.CSV and ISTAT.EXPORT.CSV** New commands COVerage.EXPORT.CSV and ISTAT.EXPORT.CSV introduced to export coverage/instruction statistic data in comma separated values (CSV) format.

ISTATistic.METHOD

Build 50104 IMP 02371 COVerage.METHOD (Select code coverage method) ISTATistic.METHOD (Recording method for instruction statistics)	**New SPY method for COVerage and ISTAT commands** Allows to process the trace while recording in STREAM mode.

ISTATistic.state

Build 50104 IMP 02373 ISTATistic.state (Display ISTAT configuration window)	**New command ISTAT.state** Shows and controls various ISTAT related settings.

Build 30461 IMP 02237 Combiprobe ARM/Cortex Trace (parallel) ITM (CoreSight ITM (Instrumentation Trace Macrocell))	**ETM+ITM trace correlation simplified** "ITM.DataTrace DataCorrelated" selects data trace function with correlation to ETM trace. "Trace" commands will then show merged trace data.
Build 22490 IMP 02208 ETM (Embedded Trace Macrocell (ETM)) ITM (CoreSight ITM (Instrumentation Trace Macrocell))	**Merged ETM/ITM analyzer for Cortex-M** The merged analyzer allows a correlated processing of ETM and ITM trace data. This allows details nesting analysis for targets using an RTOS.
Build 19417 IMP 02167 Cortex-M (ARMv6/7/8 32-bit) Debugger BMC (Benchmark counters) ITM (CoreSight ITM (Instrumentation Trace Macrocell))	**Cortex-M benchmark counter support** Benchmark counters can be traced by ITM and displayed graphically.

Integrator.ABCDEF

Build 2940 FIX 01883 PowerIntegrator - Logic and Bus Analyzer Integrator.ABCDEF (Sampling configuration for probes ABCDEF) Integrator.JKLMNO (Sampling configuration for probes JKLMNO)	**Transient detection - sample control** transient detection for sample control was working partly wrong under circumstances affected modes: State, StatePLL, StatePLLBoth, DDR
Build 1246 FIX 01814 PowerIntegrator - Logic and Bus Analyzer Integrator.ABCDEF (Sampling configuration for probes ABCDEF) Integrator.JKLMNO (Sampling configuration for probes JKLMNO)	**Transient detection - sample control** transient detection for sample control was working partly wrong under circumstances affected modes: State, StatePLL, StatePLLBoth, DDR

Build 2940 FIX 01883 PowerIntegrator - Logic and Bus Analyzer Integrator.ABCDEF (Sampling configuration for probes ABCDEF) Integrator.JKLMNO (Sampling configuration for probes JKLMNO)	**Transient detection - sample control** transient detection for sample control was working partly wrong under circumstances affected modes: State, StatePLL, StatePLLBoth, DDR
Build 1246 FIX 01814 PowerIntegrator - Logic and Bus Analyzer Integrator.ABCDEF (Sampling configuration for probes ABCDEF) Integrator.JKLMNO (Sampling configuration for probes JKLMNO)	**Transient detection - sample control** transient detection for sample control was working partly wrong under circumstances affected modes: State, StatePLL, StatePLLBoth, DDR

Build 105499 IMP 02556 Integrated Development Environment DIALOG.Program (Interactive programming) EDIT (TRACE32 editor) MENU.Program (Interactive programming) PEDIT (Edit PRACTICE script) Analyzer.Program (Program trigger unit) Data.PROGRAM (Editor for writing assembler program) Integrator.Program (Program trigger unit) Probe.Program (Program trigger unit) PER.Program (Interactive programming)	**new internal text editor** New text editor version with syntax highlighting, nesting awareness, undo/redo and other features activated.
Build 7181 CHG 02001 PowerIntegrator - Logic and Bus Analyzer Integrator.Program (Program trigger unit) Integrator.ReProgram (Program trigger unit)	**complex trigger unit** SELECTOR definition: mode RAISING renamed to RISING
Build 4728 FIX 01980 PowerIntegrator - Logic and Bus Analyzer Integrator.Program (Program trigger unit) Integrator.ReProgram (Program trigger unit) Integrator.TSYNC (Select trigger line and mode)	**transient recognition changed** for simple and complex trigger unit the transient recognition for the dataselectors was changed and some bugs fixed
Build 1253 FIX 01880 PowerIntegrator - Logic and Bus Analyzer Integrator.Program (Program trigger unit) Integrator.ReProgram (Program trigger unit)	**analyzer complex trigger unit** hostbased software may crash under certain circumstances

Integrator.ReProgram

Build 7181 CHG 02001 PowerIntegrator - Logic and Bus Analyzer Integrator.Program (Program trigger unit) Integrator.ReProgram (Program trigger unit)	**complex trigger unit** SELECTOR definition: mode RAISING renamed to RISING
Build 4728 FIX 01980 PowerIntegrator - Logic and Bus Analyzer Integrator.Program (Program trigger unit) Integrator.ReProgram (Program trigger unit) Integrator.TSYNC (Select trigger line and mode)	**transient recognition changed** for simple and complex trigger unit the transient recognition for the dataselectors was changed and some bugs fixed
Build 1253 FIX 01880 PowerIntegrator - Logic and Bus Analyzer Integrator.Program (Program trigger unit) Integrator.ReProgram (Program trigger unit)	**analyzer complex trigger unit** hostbased software may crash under certain circumstances

Integrator.TSYNC

Build 4728 FIX 01980 PowerIntegrator - Logic and Bus Analyzer Integrator.Program (Program trigger unit) Integrator.ReProgram (Program trigger unit) Integrator.TSYNC (Select trigger line and mode)	**transient recognition changed** for simple and complex trigger unit the transient recognition for the dataselectors was changed and some bugs fixed

InterCom.ENable

Build 100486 IMP 02539 Multicore Debugging InterCom.ENable (User-defined InterCom name, auto-assigned port number) InterCom.NAME (Assign user-defined InterCom name)	**Assign name to TRACE32 instance** The command InterCom.Enable assigns a name and a port number to a TRACE32 instance. The command InterCom.Name assigns a name to a TRACE32 instance. The given name can then be used by the following commands to simply their use: InterCom.execute, InterCom.executeNoWait, InterCom.Evaluate, InterCom.PING, InterCom.WAIT and SYnch.Connect.

InterCom.Evaluate

Build 100486 IMP 02538 Multicore Debugging InterCom.execute (Execute command via InterCom system) InterCom.Evaluate (Evaluate function via InterCom system) InterCom.executeNoWait (Execute command via InterCom system) InterCom.WAIT (Wait for remote InterCom system)	**Keywords and names simplify use of InterCom commands** All listed InterCom commands allow using the keywords SELF, ALL and OTHERS to address TRACE32 instance(s), as well as the name assigned to a TRACE32 instance and a name pattern.

InterCom.NAME

Build 100486 IMP 02539 Multicore Debugging InterCom.ENable (User-defined InterCom name, auto-assigned port number) InterCom.NAME (Assign user-defined InterCom name)	**Assign name to TRACE32 instance** The command InterCom.Enable assigns a name and a port number to a TRACE32 instance. The command InterCom.Name assigns a name to a TRACE32 instance. The given name can then be used by the following commands to simply their use: InterCom.execute, InterCom.executeNoWait, InterCom.Evaluate, InterCom.PING, InterCom.WAIT and SYnch.Connect.

InterCom.WAIT

Build 100486 IMP 02538 Multicore Debugging InterCom.execute (Execute command via InterCom system) InterCom.Evaluate (Evaluate function via InterCom system) InterCom.executeNoWait (Execute command via InterCom system) InterCom.WAIT (Wait for remote InterCom system)	**Keywords and names simplify use of InterCom commands** All listed InterCom commands allow using the keywords SELF, ALL and OTHERS to address TRACE32 instance(s), as well as the name assigned to a TRACE32 instance and a name pattern.

InterCom.execute

Build 100486 IMP 02538 Multicore Debugging InterCom.execute (Execute command via InterCom system) InterCom.Evaluate (Evaluate function via InterCom system) InterCom.executeNoWait (Execute command via InterCom system) InterCom.WAIT (Wait for remote InterCom system)	**Keywords and names simplify use of InterCom commands** All listed InterCom commands allow using the keywords SELF, ALL and OTHERS to address TRACE32 instance(s), as well as the name assigned to a TRACE32 instance and a name pattern.

InterCom.executeNoWait

Build 100486 IMP 02538 Multicore Debugging InterCom.execute (Execute command via InterCom system) InterCom.Evaluate (Evaluate function via InterCom system) InterCom.executeNoWait (Execute command via InterCom system) InterCom.WAIT (Wait for remote InterCom system)	**Keywords and names simplify use of InterCom commands** All listed InterCom commands allow using the keywords SELF, ALL and OTHERS to address TRACE32 instance(s), as well as the name assigned to a TRACE32 instance and a name pattern.

JTAG.SWD.Init

Build 130863 IMP 02609 JTAG.SWD.Init (Initialize the debug port) JTAG.SWD.ReadDapBus (Read register from DAP) JTAG.SWD.ReadScan (Read register from DAP) JTAG.SWD.SHIFT (Shift data by using the SWIO pin) JTAG.SWD.WriteDapBus (Write register to DAP) JTAG.SWD.WriteScan (Write register to DAP)	**new JTAG.SWD command group** new JTAG.SWD commands for manual Serial Wire Debug control.

JTAG.SWD.ReadDapBus

Build 130863 IMP 02609 JTAG.SWD.Init (Initialize the debug port) JTAG.SWD.ReadDapBus (Read register from DAP) JTAG.SWD.ReadScan (Read register from DAP) JTAG.SWD.SHIFT (Shift data by using the SWIO pin) JTAG.SWD.WriteDapBus (Write register to DAP) JTAG.SWD.WriteScan (Write register to DAP)	**new JTAG.SWD command group** new JTAG.SWD commands for manual Serial Wire Debug control.

JTAG.SWD.ReadScan

Build 130863 IMP 02609 JTAG.SWD.Init (Initialize the debug port) JTAG.SWD.ReadDapBus (Read register from DAP) JTAG.SWD.ReadScan (Read register from DAP) JTAG.SWD.SHIFT (Shift data by using the SWIO pin) JTAG.SWD.WriteDapBus (Write register to DAP) JTAG.SWD.WriteScan (Write register to DAP)	**new JTAG.SWD command group** new JTAG.SWD commands for manual Serial Wire Debug control.

JTAG.SWD.SHIFT

Build 130863 IMP 02609 JTAG.SWD.Init (Initialize the debug port) JTAG.SWD.ReadDapBus (Read register from DAP) JTAG.SWD.ReadScan (Read register from DAP) JTAG.SWD.SHIFT (Shift data by using the SWIO pin) JTAG.SWD.WriteDapBus (Write register to DAP) JTAG.SWD.WriteScan (Write register to DAP)	**new JTAG.SWD command group** new JTAG.SWD commands for manual Serial Wire Debug control.

JTAG.SWD.WriteDapBus

Build 130863 IMP 02609 JTAG.SWD.Init (Initialize the debug port) JTAG.SWD.ReadDapBus (Read register from DAP) JTAG.SWD.ReadScan (Read register from DAP) JTAG.SWD.SHIFT (Shift data by using the SWIO pin) JTAG.SWD.WriteDapBus (Write register to DAP) JTAG.SWD.WriteScan (Write register to DAP)	**new JTAG.SWD command group** new JTAG.SWD commands for manual Serial Wire Debug control.

JTAG.SWD.WriteScan

Build 130863 IMP 02609 JTAG.SWD.Init (Initialize the debug port) JTAG.SWD.ReadDapBus (Read register from DAP) JTAG.SWD.ReadScan (Read register from DAP) JTAG.SWD.SHIFT (Shift data by using the SWIO pin) JTAG.SWD.WriteDapBus (Write register to DAP) JTAG.SWD.WriteScan (Write register to DAP)	**new JTAG.SWD command group** new JTAG.SWD commands for manual Serial Wire Debug control.

JTAG.XUSEFUSE

Build 112182 IMP 02569 JTAG Debugger JTAG.XUSEFUSE (Program Xilinx UltraScale eFUSEs)	**new command JTAG.XUSEFUSE added** new command introduced to burn eFuses of Xilinx UltraScale FPGAs

LA

Build 1190 IMP 01671 LA (Logic analyzer)	**New trace method LA** The LA method allows to import trace data from external logic analyzers.

LA.IMPORT.ETB

Build 34458 IMP 02277 LA.IMPORT.ETB (Import on-chip trace data)	**New command LA.IMPORT.ETB** Can import pure binary ETB buffer contents for offline processing.

LA.IMPORT.STPByteStream

Build 76594 IMP 02457 System Trace LA.IMPORT.STPByteStream (Import STP recording from file (byte))	**Import STP byte stream for processing in TRACE32** The command allows to import a STP byte stream from a file into TRACE32. The imported STP byte stream can then be displayed and analyzed.

LA.IMPORT.TraceFile

Build 42354 IMP 02338 LA.IMPORT.TraceFile (Import trace data where processing has failed)	**New command LA.IMPORT.TraceFile** Allows to re-import saved trace data for flow trace re-processing. Useful when processing was not possible when the trace was made.

LA.IMPORT.TracePort

Build 19417 IMP 02179 ARM/Cortex Trace (parallel) LA.IMPORT.TracePort (Import off-chip trace data)	**New LA.IMPORT.TracePort command** Allows to import pure binary files from ETBs.

LA.IMPORT.cycles

Build 125398 CHG 02591 Trace Analyzer <trace>.EXPORT.cycles (Export trace data) LA.IMPORT.cycles (Import bus trace data)	**renamed commands Trace.EXPORT.flow and LA.IMPORT.flow** The commands Trace.EXPORT.flow and LA.IMPORT.flow have been renamed to Trace.EXPORT.cycles and LA.IMPORT.cycles.

LICENSE

Build 1077 IMP 01370 LICENSE (Manage TRACE32 licenses)	License window comes up when file license is newer than debug cable license

LICENSE.List

Build 13751 IMP 02073 LICENSE.List (Display all license information)	**New command LICENSE.List** Opens window which displays license information.

LICENSE.UPDATE

Build 1077 IMP 01201 LICENSE.UPDATE (Update the maintenance contract)	**New command** New command LICENSE.UPDATE to update license in debug cable (from license file)

LOG.DO

Build 100486 IMP 02546 Script Language PRACTICE LOG.DO (Log calls of PRACTICE scripts)	**Log file for PRACTICE scripts** The command LOG.DO allows logging all PRACTICE script calls. The command was mainly introduced for support purposes.

LS

Build 1079 FIX 01397 DIR (List subdirectories and files) LS (Display directory)	Dir command showed wrong modification time (not local time)

List

Build 51144 CHG 02392 List (Display modes for programs)	**Stepping history visualization changed** The history now just shows where steps have been executed. Already stepped instructions for the Step.Diverge command are also shown. Executing Break while already stopped clears the history.
Build 19417 CHG 02141 List (Display modes for programs)	**Logger command renamed to LOGGER**
Build 1133 IMP 01490 List (Display modes for programs) SNOOPer (Sample-based trace)	SNOOPer/Logger now always 32bit

List.auto

Build 117056 IMP 02576 Integrated Development Environment List.auto (Display program listing)	**new mouse feature** New functionality on middle mouse button for List and Var windows: single click performs an add to watch window and double click a view in a new window.
Build 93173 IMP 02512 Integrated Development Environment Frame.view (Display stack frame) List.auto (Display program listing) Var.View (Display variables) Var.Watch (Open Var.Watch window)	**Mouse interface changed for all text windows** Free text selection is now possible with the left mouse. Click and drag on a selection allows drag and drop.
Build 1077 IMP 01317 <trace>.state (Display trace configuration window) List.auto (Display program listing) <trace>.state (Display trace configuration window)	WR and DATA.xx items enabled for Logger, SNOOPer and Integrator

MAP.BUS16

Build 19417 IMP 02174 MAP.BUS16 (Bus width mapping) MAP.BUS32 (Bus width mapping) MAP.BUS8 (Bus width mapping)	**Forcing memory access to access size** The MAP.BUSxx commands will now effect how the debugger accesses target memory.

MAP.BUS32

Build 19417 IMP 02174 MAP.BUS16 (Bus width mapping) MAP.BUS32 (Bus width mapping) MAP.BUS8 (Bus width mapping)	**Forcing memory access to access size** The MAP.BUSxx commands will now effect how the debugger accesses target memory.

MAP.BUS8

Build 19417 IMP 02174 MAP.BUS16 (Bus width mapping) MAP.BUS32 (Bus width mapping) MAP.BUS8 (Bus width mapping)	**Forcing memory access to access size** The MAP.BUSxx commands will now effect how the debugger accesses target memory.

MAP.InitVar

Build 1264 IMP 01859 MAP.InitVar (CTS initial variable mapping) MAP.NoInitVar (CTS initial variable mapping)	**New mapping command for CTS** Defines already initialized variables for CTS violation checking features. The mapped areas are considered as having been read already.

MAP.NoInitVar

Build 1264 IMP 01859 MAP.InitVar (CTS initial variable mapping) MAP.NoInitVar (CTS initial variable mapping)	**New mapping command for CTS** Defines already initialized variables for CTS violation checking features. The mapped areas are considered as having been read already.

MAP.VMREAD

Build 19417 IMP 02170 MAP.VMREAD (Redirect memory reads to TRACE32 virtual memory)	**Support for non-readable memories** The command MAP.VMREAD can redirect memory reads to the VM memory. This allows code execution and disassembly of code running in memory without data access path.

MENU.AddMenu

Build 1190 FIX 01673 MENU.AddMenu (Add one standard menu item) MENU.AddTool (Add a button to the main toolbar)	**MENU.ADDxxx failed with quotes in string** The commands failed when quotes where in the parameter strings.

MENU.AddTool

Build 1190 FIX 01673 MENU.AddMenu (Add one standard menu item) MENU.AddTool (Add a button to the main toolbar)	**MENU.ADDxxx failed with quotes in string** The commands failed when quotes where in the parameter strings.

Build 112182 IMP 02567 Script Language PRACTICE MENU.Program (Interactive programming) MENU.ReProgram (Menu programming)	**new command PERMENU for menu files added** The new menu programming command PERMENU converts the tree structure of per files to men files.
Build 105499 IMP 02556 Integrated Development Environment DIALOG.Program (Interactive programming) EDIT (TRACE32 editor) MENU.Program (Interactive programming) PEDIT (Edit PRACTICE script) Analyzer.Program (Program trigger unit) Data.PROGRAM (Editor for writing assembler program) Integrator.Program (Program trigger unit) Probe.Program (Program trigger unit) PER.Program (Interactive programming)	**new internal text editor** New text editor version with syntax highlighting, nesting awareness, undo/redo and other features activated.

MENU.ReProgram

Build 112182 IMP 02567 Script Language PRACTICE MENU.Program (Interactive programming) MENU.ReProgram (Menu programming)	**new command PERMENU for menu files added** The new menu programming command PERMENU converts the tree structure of per files to men files.

MIPS

Build 30461 IMP 02243 MIPS (Number of instructions per second)	**New trace analysis "MIPS" introduced** Allows to analyze "number of instructions per second" numerically or graphically.

MMU

Build 19417 CHG 02164 MMU (Memory management unit)	**MMU command structure changed** Command structure changed to match CACHE command structure.
Build 2940 FIX 01879 MMU (Memory management unit)	**MMU command structure redesigned** Structure is now more logical and clean. Common command conversions: MMU.DUMP -> MMU.PageTable.dump MMU.SCAN -> MMU.PageTable.SCAN MMU.TLB -> MMU.TLB.dump The old commands are still usable.

MMU.DUMP

Build 34458 CHG 02269 MMU.DUMP (Dump MMU tables) MMU.List (Compact display of MMU translation table) MMU.SCAN (Scan MMU tables (static snapshot))	**new items for MMU commands** New item "KernelPageTable" for MMU.DUMP, MMU.LIST and MMU.SCAN commands. Shows kernel translations. "PageTable" shows currently active translations.
Build 1077 FIX 01175 MMU.DUMP (Dump MMU tables) MMU.SCAN (Scan MMU tables (static snapshot))	MMU.SCAN/MMU.DUMP without default format issues now an error

MMU.FORMAT

Build 88288 IMP 02497 Data.LOAD.Elf (Load ELF file) EXTension.LOAD (Load extension) Frame.view (Display stack frame) MMU.FORMAT (Define MMU table structure) Register.view (Display registers) TASK.List.MACHINES (List machines) TASK.List.TREE (Display tasks in a tree structure) TRANSlation.List (List MMU translation table) TRANSlation.TableWalk (Automatic MMU page table walk)	**Hypervisor-aware debugging feature introduced (Part 2-2)** - the debugger address translation (MMU and TRANS command groups) can be individually configured for each virtual machine by specifying a machineid in the address parameters. Where appropriate, MMU commands accept the new option /MACHINE <machineid>. Page tables of all machines can be viewed. Access class I: is used to specify intermediate addresses. - individual symbol sets can be loaded for each virtual machine by loading them onto the appropriate extended address: Data.LOAD.<type> <file> <accessclass>:<machineid>:::0 <options> Data.LOAD.<type> <file> <accessclass>:<machineid>:::<spaceID>::0 <options> - multiple awarenesses can be loaded to individual machines using the EXTension.LOAD command with the /machine option: EXTension.LOAD <awarenessfile> /MACHINE <machineid> [/NAME <extname>] - extension commands may use an infix to specify uniquely a command of a specific extension: TASK.[<extname>.]<extcommand> EXTension.[<extname>.]<extcommand> - new commands TASK.List.MACHINES and TASK.List.TREE are available to display the target's hypervisor/guest/process/thread structure - task names may be extended by the machine name, separated by three colons, to uniquely identify them: <machinename>:::<taskname> - register sets and stack frames of machines and tasks of arbitrary guests can be viewed, e.g.: Register.view /MACHINE <machineid> Frame.view /TASK "<machinename>:::<taskname>"

MMU.INFO

Build 56057 IMP 02404 MMU.INFO (Translation information related to an address)	**New command MMU.INFO** Shows information about MMU translations for a given address.

MMU.List

Build 34458 CHG 02269 MMU.DUMP (Dump MMU tables) MMU.List (Compact display of MMU translation table) MMU.SCAN (Scan MMU tables (static snapshot))	**new items for MMU commands** New item "KernelPageTable" for MMU.DUMP, MMU.LIST and MMU.SCAN commands. Shows kernel translations. "PageTable" shows currently active translations.

MMU.SCAN

Build 34458 CHG 02269 MMU.DUMP (Dump MMU tables) MMU.List (Compact display of MMU translation table) MMU.SCAN (Scan MMU tables (static snapshot))	**new items for MMU commands** New item "KernelPageTable" for MMU.DUMP, MMU.LIST and MMU.SCAN commands. Shows kernel translations. "PageTable" shows currently active translations.
Build 1077 FIX 01175 MMU.DUMP (Dump MMU tables) MMU.SCAN (Scan MMU tables (static snapshot))	MMU.SCAN/MMU.DUMP without default format issues now an error

MMU.view

Build 19417 CHG 02162 In-Circuit Emulator for 386 MMU.view (View MMU registers)	**80386 MMU window changed** Attribute registers shown as 16bit values now.

MMX

Build 13751 IMP 02111 ARM Debugger MMX (MMX registers (MultiMedia eXtension))	**XScale MMX register support**

MMX.view

Build 51144 IMP 02385 AVX.view (Display AVX registers) AVX512.view (Display AVX512 registers) FPU.view (Display FPU registers) MMX.view (Open MMX register window) SSE.view (Display SSE registers) VPU.view (Display ALTIVEC register window)	**Larger than 64 bit SIMD register sets** FPU, VPE, SPE, MMX, SSE, AVX, AVX512 - Parser for wide registers (>64 bit) unified, values wider than 64 bit possible.

NAME

Build 1133 IMP 01465 NAME (Logical names for physical connections)	NAME.WORD and all internal trace lines extended to 64 bits

NAME.User

Build 130863 IMP 02612 NAME.User (Create new user channel)	**new command NAME.User for adding user-defined channels** User channels allow to combine (trace) channels via mathematical operations. The resulting new channel can then be referenced in any trace command by 'User.<name>'.

NEXUS.OCeaNport.Mode

Build 38589 IMP 02319 QorIQ PowerPC 32/64 Bit Debugger NEXUS.OCeaNport.Mode (Configure Nexus OCeaN message type) NEXUS.OCeaNport<index>.TraceSELect (Select Nexus OCeaN trace type)	**OCeaN trace support added** The NEXUS window gives easy access to the options.

NEXUS.OCeaNport<index>.TraceSELect

Build 38589 IMP 02319 QorIQ PowerPC 32/64 Bit Debugger NEXUS.OCeaNport.Mode (Configure Nexus OCeaN message type) NEXUS.OCeaNport<index>.TraceSELect (Select Nexus OCeaN trace type)	**OCeaN trace support added** The NEXUS window gives easy access to the options.

ON

Build 50104 IMP 02374 ON (Event-controlled PRACTICE script execution)	**ON TIME can stop interruptible commands** This allows to limit the execution time of commands that take too long time.

OPEN

Build 30461 IMP 02246 OPEN (Open data file) WRITEB (Write binary data to file)	**PRACTICE can now handle binary files** New option /Binary for "OPEN" command allows to open or create binary files. New command "WRITEB" allows to write binary data to a file.
Build 22490 IMP 02191 Script Language PRACTICE OPEN (Open data file)	**maximum opened files PRACTICE files** The limit of maximal 20 parallel opened files increased to 120. It's independent from the maximal number of opened sourcefiles.

OS.OPEN

Build 112182 IMP 02564 Script Language PRACTICE OS.OPEN (Open file in default application)	**new command OS.OPEN added**

OS.screen

Build 4728 FIX 01936 Host Driver Software OS.screen (Call up the shell or execute host command)	**command OS crashes under circumstances** The command OS.screen crashed if the length of <commandline> exceeded 252 characters.

Onchip

Build 1224 IMP 01755 C5500 Debugger Onchip (Trace method Onchip, recording, and analysis commands)	**ETB support implemented** ETB (Embedded Trace Buffer) is an on-chip trace memory module from ARM, but can also be used on some derivatives to store the trace data produced by the DSP. The trace data can be read out by the debugger using the JTAG interface.

Out

Build 13751 FIX 02066 FIRE Fully Integrated RISC Emulator Out (Output control)	**fire analyser trigger unit - OUT.A** output signal from fire analyser trigger unit didn't work

PARAMETERS

Build 65657 IMP 02430 Script Language PRACTICE ENTRY (Parameter passing) PARAMETERS (Parameter fetching) RETURNVALUES (Take return values)	**New commands PARAMETERS and RETURNVALUES** The PARAMETER command first creates PRIVATE macros for all macro names not found in the current scope and then assigns the passed values to the macros. RETURNVALUES takes the return values of a PRACTICE script/subroutine. An error message is generated, if a macro name is used, that cannot be found in the current scope. Both commands fix known issues of the ENTRY command.

PATH.DOWN

Build 100486 IMP 02544 Integrated Development Environment PATH.Delete (Delete search path) PATH.DOWN (Define search path at end of list) PATH.List (List search path) PATH.RESet (Reset search path) PATH.Set (Define search path) PATH.UP (Define search path at top of list)	**Improved usage of search paths for scripts, per files ...** Single command PATH is replaced by command group PATH.

PATH.Delete

Build 100486 IMP 02544 Integrated Development Environment PATH.Delete (Delete search path) PATH.DOWN (Define search path at end of list) PATH.List (List search path) PATH.RESet (Reset search path) PATH.Set (Define search path) PATH.UP (Define search path at top of list)	**Improved usage of search paths for scripts, per files ...** Single command PATH is replaced by command group PATH.

PATH.List

Build 100486 IMP 02544 Integrated Development Environment PATH.Delete (Delete search path) PATH.DOWN (Define search path at end of list) PATH.List (List search path) PATH.RESet (Reset search path) PATH.Set (Define search path) PATH.UP (Define search path at top of list)	**Improved usage of search paths for scripts, per files ...** Single command PATH is replaced by command group PATH.

PATH.RESet

Build 100486 IMP 02544 Integrated Development Environment PATH.Delete (Delete search path) PATH.DOWN (Define search path at end of list) PATH.List (List search path) PATH.RESet (Reset search path) PATH.Set (Define search path) PATH.UP (Define search path at top of list)	**Improved usage of search paths for scripts, per files ...** Single command PATH is replaced by command group PATH.

PATH.Set

Build 100486 IMP 02544 Integrated Development Environment PATH.Delete (Delete search path) PATH.DOWN (Define search path at end of list) PATH.List (List search path) PATH.RESet (Reset search path) PATH.Set (Define search path) PATH.UP (Define search path at top of list)	**Improved usage of search paths for scripts, per files ...** Single command PATH is replaced by command group PATH.

PATH.UP

Build 100486 IMP 02544 Integrated Development Environment PATH.Delete (Delete search path) PATH.DOWN (Define search path at end of list) PATH.List (List search path) PATH.RESet (Reset search path) PATH.Set (Define search path) PATH.UP (Define search path at top of list)	**Improved usage of search paths for scripts, per files ...** Single command PATH is replaced by command group PATH.

PBREAK.DISable

Build 51144 IMP 02380 Script Language PRACTICE PBREAK.Delete (Delete breakpoint) PBREAK.DISable (Disable breakpoint) PBREAK.ENable (Enable breakpoint) PBREAK.List (Display breakpoint list) PBREAK.Set (Add breakpoint)	**multiple PRACTICE breakpoints introduced** PRACTICE can handle multiple PRACTICE breakpoints now. Commands PBREAK.Set, PBREAK.Delete, PBREAK.List, PBREAK.ENable, PBREAK.DISable added.

PBREAK.Delete

Build 51144 IMP 02380 Script Language PRACTICE PBREAK.Delete (Delete breakpoint) PBREAK.DISable (Disable breakpoint) PBREAK.ENable (Enable breakpoint) PBREAK.List (Display breakpoint list) PBREAK.Set (Add breakpoint)	**multiple PRACTICE breakpoints introduced** PRACTICE can handle multiple PRACTICE breakpoints now. Commands PBREAK.Set, PBREAK.Delete, PBREAK.List, PBREAK.ENable, PBREAK.DISable added.

PBREAK.ENable

Build 51144 IMP 02380 Script Language PRACTICE PBREAK.Delete (Delete breakpoint) PBREAK.DISable (Disable breakpoint) PBREAK.ENable (Enable breakpoint) PBREAK.List (Display breakpoint list) PBREAK.Set (Add breakpoint)	**multiple PRACTICE breakpoints introduced** PRACTICE can handle multiple PRACTICE breakpoints now. Commands PBREAK.Set, PBREAK.Delete, PBREAK.List, PBREAK.ENable, PBREAK.DISable added.

PBREAK.List

Build 51144 IMP 02380 Script Language PRACTICE PBREAK.Delete (Delete breakpoint) PBREAK.DISable (Disable breakpoint) PBREAK.ENable (Enable breakpoint) PBREAK.List (Display breakpoint list) PBREAK.Set (Add breakpoint)	**multiple PRACTICE breakpoints introduced** PRACTICE can handle multiple PRACTICE breakpoints now. Commands PBREAK.Set, PBREAK.Delete, PBREAK.List, PBREAK.ENable, PBREAK.DISable added.

Build 51144 IMP 02380 Script Language PRACTICE PBREAK.Delete (Delete breakpoint) PBREAK.DISable (Disable breakpoint) PBREAK.ENable (Enable breakpoint) PBREAK.List (Display breakpoint list) PBREAK.Set (Add breakpoint)	**multiple PRACTICE breakpoints introduced** PRACTICE can handle multiple PRACTICE breakpoints now. Commands PBREAK.Set, PBREAK.Delete, PBREAK.List, PBREAK.ENable, PBREAK.DISable added.

PEDIT

Build 105499 IMP 02556 Integrated Development Environment DIALOG.Program (Interactive programming) EDIT (TRACE32 editor) MENU.Program (Interactive programming) PEDIT (Edit PRACTICE script) Analyzer.Program (Program trigger unit) Data.PROGRAM (Editor for writing assembler program) Integrator.Program (Program trigger unit) Probe.Program (Program trigger unit) PER.Program (Interactive programming)	**new internal text editor** New text editor version with syntax highlighting, nesting awareness, undo/redo and other features activated.
Build 34458 02259 Uniform Look-And-Feel for all Processors PEDIT (Edit PRACTICE script)	**pedit not closing file after close**

Build 1133 CHG 01468 Super10 NEXUS Debugger and Trace PER (Peripheral files)	peripheral files for R303 and super10 renamed to perm340.per

Build 105499 IMP 02556 Integrated Development Environment DIALOG.Program (Interactive programming) EDIT (TRACE32 editor) MENU.Program (Interactive programming) PEDIT (Edit PRACTICE script) Analyzer.Program (Program trigger unit) Data.PROGRAM (Editor for writing assembler program) Integrator.Program (Program trigger unit) Probe.Program (Program trigger unit) PER.Program (Interactive programming)	**new internal text editor** New text editor version with syntax highlighting, nesting awareness, undo/redo and other features activated.
Build 56057 IMP 02403 PER.Program (Interactive programming) PER.ReProgram (Load default program)	**PER programming extended** New commands ENTRY and INCLUDE allow modular PER files and passing of parameters to PER files.
Build 4728 FIX 01976 PER.view (Display peripherals) PER.Program (Interactive programming) PER.ReProgram (Load default program)	**per programming** saveout command did not restore original value
Build 4728 IMP 01965 PER.Program (Interactive programming)	**New capabilites in PER window**
Build 1198 IMP 01716 PER.Program (Interactive programming)	**New definition HGROUP for PER.Program** HGROUP define a group to be used with HIDE items. No memory access is done during window update.
Build 1190 FIX 01690 PER.Program (Interactive programming) PER.ReProgram (Load default program)	**PER.Program crashed with long texts** PER.Program was crashing when the description text was longer than 256 characters.
Build 1133 FIX 01520 PER.Program (Interactive programming)	TEXTFLD item in PER.Program was not working as expected

PER.ReProgram

Build 56057 IMP 02403 PER.Program (Interactive programming) PER.ReProgram (Load default program)	**PER programming extended** New commands ENTRY and INCLUDE allow modular PER files and passing of parameters to PER files.
Build 4728 FIX 01976 PER.view (Display peripherals) PER.Program (Interactive programming) PER.ReProgram (Load default program)	**per programming** saveout command did not restore original value
Build 1190 FIX 01690 PER.Program (Interactive programming) PER.ReProgram (Load default program)	**PER.Program crashed with long texts** PER.Program was crashing when the description text was longer than 256 characters.

PER.STOre

Build 80996 IMP 02481 Logical Display of Peripherals PER.STOre (Generate PRACTICE script from PER settings)	**Generate PRACTICE script for PER settings** New command PER.STOre and extended local pulldown menu in PER.View window to generated script/command sequence for current PER settings.

Build 125398 IMP 02593 PER.view (Display peripherals)	**new option /AlternatingBackGround for PER.view command** The new /AlternatingBackGround option allows to display an alternating background color in the PER.View window.
Build 100486 IMP 02529 Logical Display of Peripherals PER.view (Display peripherals) CSV (Enables CSV capabilities) PERCMD (Row definition in CSV-formatted *.per file)	**CSV format allowed for PER files** PER files can be written within a spreadsheet editor, since the PER.view command can now also compile CSV files.
Build 56057 IMP 02405 PER.view (Display peripherals)	**Late PER file compiling** New "WAIT" command allows to delay the compilation for a PER window till certain information from the target is available.
Build 34458 IMP 02268 Data.View (Display memory) PER.view (Display peripherals)	**command option CORE added** CORE option available in PER, Data.View and Data.PRINT commands.
Build 22490 IMP 02190 PER.view (Display peripherals)	**New commands for PER files** New command ASSERT to check for a condition before going on with parsing the PER file New command VARX for SGROUPS in PER-files, to assign a PRACTICE expression to the SGOUP buffer, which will be parsed continuously during the displaying of the PER file. New Functions PER.B.B(i), PER.B.W(i), PER.B.L(i) and PER.B.Q(i), to access the SGOUP buffer inside VARX commands.
Build 4728 FIX 01976 PER.view (Display peripherals) PER.Program (Interactive programming) PER.ReProgram (Load default program)	**per programming** saveout command did not restore original value
Build 1264 FIX 01863 PPC400 Debugger PER.view (Display peripherals)	**TBL/TBH address in perfile corrected** Correct TBL/TBH read/write address implemented in per405xx.per files
Build 1256 IMP 01850 TriCore Debugger PER.view (Display peripherals)	**perfile for TC1796 revised** various errors corrected, support for P-Bit protected registers added, improved descriptions and layout
Build 1077 FIX 01295 ONCE Debugger for DSP56300 PER.view (Display peripherals)	**All per Files reworked** 563xx All per Files reworked. Last changes has wrong include parameter.

Build 1077 IMP 01182 PPC600/750 Debugger PER.view (Display peripherals)	MPC755: own per file.

PERCMD

Build 100486 IMP 02529 Logical Display of Peripherals PER.view (Display peripherals) CSV (Enables CSV capabilities) PERCMD (Row definition in CSV-formatted *.per file)	**CSV format allowed for PER files** PER files can be written within a spreadsheet editor, since the PER.view command can now also compile CSV files.

PERF

Build 1224 FIX 01760 MIPS32/MIPS64 Debugger PERF (Sample-based profiling)	**Performance Analysis for BCMxx corrected** Broadcom specific DERET handling in BDM_GetPC added. Changes necessary for functionality of performance analysis.
Build 1145 FIX 01572 Sample-based Profiling PERF (Sample-based profiling)	SH/ARM monitors failed with the PERF command

PERF.List

Build 22490 ODD 02215 PERF.List (Default profiling)	**PERF.List command behavior changed** Important: The performance analyzer isn't armed automatically per default anymore. The PERF.List* windows display 0.00 % values till the performance analyzer was armed at least once.
Build 1096 FIX 01444 PERF.List (Default profiling)	Clicking on line in PERF.List window fails for addresses >0xa0000000

PERF.MMUSPACES

Build 4928 IMP 01987 PERF.MMUSPACES (Include space IDs for addresses in the sampling)	**PERF command can consider memory spaces** When enabled, the PERF command will distinguish between different memory spaces.

PERF.Mode

Build 1190 FIX 01666 PERF.Mode (Specify sampling object)	**PERF.Mode TASK may crash** PERF.Mode TASK may crash when more than 64 tasks exist.
Build 1096 FIX 01451 ColdFire BDM Debugger PERF.Mode (Specify sampling object)	PERF with StopAndGo method not working on Coldfire with PowerDebug

PERF.Program

Build 1241 FIX 01800 PERF.Program (Write a performance analyzer program) PERF.ReProgram (Load an existing performance analyzer program)	**PERF.Program/ReProgram may crash**

PERF.ReProgram

Build 1241 FIX 01800 PERF.Program (Write a performance analyzer program) PERF.ReProgram (Load an existing performance analyzer program)	**PERF.Program/ReProgram may crash**

PERF.state

Build 22490 IMP 02206 PERF.state (Display state)	**PERF command with new functionality** Performance measures now always all ranges and allows to show the results in different windows concurrently. New features include performance display for single instructions in list windows.

PLIST

Build 34458 IMP 02280 PLIST (List PRACTICE script) PMACRO (PRACTICE macros)	**PRACTICE debugging improved** Supporting macro and internal variable display and modification from PLIST and PMACRO windows.

PMACRO

Build 34458 IMP 02280 PLIST (List PRACTICE script) PMACRO (PRACTICE macros)	**PRACTICE debugging improved** Supporting macro and internal variable display and modification from PLIST and PMACRO windows.

PMACRO.EXPLICIT

Build 60219 IMP 02417 Script Language PRACTICE PMACRO.EXPLICIT (Enforce explicit PRACTICE macro declaration) PMACRO.IMPLICIT (Implicit PRACTICE macro declaration) PSTEPOUT (Back to caller)	**New PRACTICE commands** PMACRO.EXPLICIT enforces that macros are declared as PRIVAT, LOCAL or GLOBAL PMACRO.IMPLICIT allows to use an undeclared macro, undeclared macros are LOCAL be default PSTEPOUT allows to step out of the current PRACTICE block

PMACRO.IMPLICIT

Build 60219 IMP 02417 Script Language PRACTICE PMACRO.EXPLICIT (Enforce explicit PRACTICE macro declaration) PMACRO.IMPLICIT (Implicit PRACTICE macro declaration) PSTEPOUT (Back to caller)	**New PRACTICE commands** PMACRO.EXPLICIT enforces that macros are declared as PRIVAT, LOCAL or GLOBAL PMACRO.IMPLICIT allows to use an undeclared macro, undeclared macros are LOCAL be default PSTEPOUT allows to step out of the current PRACTICE block

PRINT

Build 50104 IMP 02355 Script Language PRACTICE PRINT (Write text and data to an AREA window (without format decoration))	**PRACTICE command PRINT extended** expression types <address> and <addressrange> full supported now
Build 8248 IMP 02026 PRINT (Write text and data to an AREA window (without format decoration))	**PRINT command** new expression types TIMERANGE and RANGE added

PRINTF

Build 88288 IMP 02487 Script Language PRACTICE ECHO (Write text and data to an AREA window (with format decoration)) PRINTF (Write formatted data to an AREA window)	**Streamlined output to AREA window from PRACTICE** The ECHO command keeps type and radix information of data and PRACTICE macros. This improves the readablity of the output. The PRINTF command allows to format outputs in the style of the printf() function in C/C++.

PRIVATE

Build 7181 IMP 02002 Script Language PRACTICE PRIVATE (Create private PRACTICE macro)	**New PRACTICE command PRIVATE** Creates only local accessible macro.

PRinTer.EXPORT

Build 76594 IMP 02454 Integrated Development Environment PRinTer.EXPORT (Export formatted printer output to file) PRinTer.FILE (Re-route printer output to a file in specified file format)	**Behavior of PRinTer dialog changed** The file extension matches the output format now. As part of this change, the commands PRinTer.EXPORT and PRinTer.FILE have been updated. PRinTer.EXPORT additionally allows to export the output in ASCII and XHTML format. Now PRinTer.FILE allows to set the file name, the output format and the append parameter.
Build 1198 IMP 01710 PRinTer.EXPORT (Export formatted printer output to file)	**new command PRT.EXPORT** Same as PRT.FILE command, but outputs data in CSV (Comma Separated Value) format.

PRinTer.FILE

Build 130863 IMP 02611 Integrated Development Environment PRinTer.FILE (Re-route printer output to a file in specified file format) WinPrint (Print address or record range of a window)	**new PRinTer file format PCL** The PCL file format has been added to the PRinTer.FILE command.
Build 76594 IMP 02454 Integrated Development Environment PRinTer.EXPORT (Export formatted printer output to file) PRinTer.FILE (Re-route printer output to a file in specified file format)	**Behavior of PRinTer dialog changed** The file extension matches the output format now. As part of this change, the commands PRinTer.EXPORT and PRinTer.FILE have been updated. PRinTer.EXPORT additionally allows to export the output in ASCII and XHTML format. Now PRinTer.FILE allows to set the file name, the output format and the append parameter.

PRinTer.FileType

Build 1198 IMP 01709 PRinTer.FileType (Select file format)	**New file and clipboard format CSV** The "Comma Separated Value" format is useful to export windows (e.g. dumps, variables, statistic results) to other applications.

PSTEP

Build 1145 IMP 01561 PSTEP (Execute single line)	PSTEP with same parameters as "DO" command will start PRACTICE file for debugging

PSTEPOUT

Build 60219 IMP 02417 Script Language PRACTICE PMACRO.EXPLICIT (Enforce explicit PRACTICE macro declaration) PMACRO.IMPLICIT (Implicit PRACTICE macro declaration) PSTEPOUT (Back to caller)	**New PRACTICE commands** PMACRO.EXPLICIT enforces that macros are declared as PRIVAT, LOCAL or GLOBAL PMACRO.IMPLICIT allows to use an undeclared macro, undeclared macros are LOCAL be default PSTEPOUT allows to step out of the current PRACTICE block

Probe.Program

Build 105499 IMP 02556 Integrated Development Environment DIALOG.Program (Interactive programming) EDIT (TRACE32 editor) MENU.Program (Interactive programming) PEDIT (Edit PRACTICE script) Analyzer.Program (Program trigger unit) Data.PROGRAM (Editor for writing assembler program) Integrator.Program (Program trigger unit) Probe.Program (Program trigger unit) PER.Program (Interactive programming)	**new internal text editor** New text editor version with syntax highlighting, nesting awareness, undo/redo and other features activated.

Probe.Timing

Build 1161 FIX 01625 <trace>.Timing (Waveform of trace buffer) Probe.Timing (Display trace contents as timing diagram)	**Unexpected zoom out in timing windows** Scrolling timing windows out of the visible area may cause the window to zoom out.

RETURNVALUES

Build 65657 IMP 02430 Script Language PRACTICE ENTRY (Parameter passing) PARAMETERS (Parameter fetching) RETURNVALUES (Take return values)	**New commands PARAMETERS and RETURNVALUES** The PARAMETER command first creates PRIVATE macros for all macro names not found in the current scope and then assigns the passed values to the macros. RETURNVALUES takes the return values of a PRACTICE script/subroutine. An error message is generated, if a macro name is used, that cannot be found in the current scope. Both commands fix known issues of the ENTRY command.

RTS

Build 80996 IMP 02477 ARM/Cortex Trace (parallel) RTS (Real-time profiling (RTS))	**RTS command group supports ETMv4** RTS command group supports ETMv4.
Build 26464 IMP 02223 CombiProbe RTS (Real-time profiling (RTS))	**RTS supported with CombiProbe** Can be used for long time traces and coverage analysis for Cortex-M series.

RTS.state

Build 88288 IMP 02494 TriCore AGBT with PowerTrace Serial TriCore AGBT with Serial Preprocessor RTS.state (Open status and control window)	**RTS implemented for TriCore** RTS implemented for TriCore architecture.
Build 80996 IMP 02483 MPC57xx NEXUS High Speed Serial Trace Port QorIQ PowerPC NEXUS Aurora Trace Port MPC55xx/MPC56xx NEXUS Debugger and Trace RTS.state (Open status and control window)	**RTS implemented for PowerPC (all E200zx cores) and QorIQ** RTS implemented for PowerPC (all E200zx cores) and QorIQ.
Build 80996 CHG 02479 RTS.state (Open status and control window)	**Features of RTS command group were reduced** No full trace, no nesting processing, no database save. Command group is under redesign.

RePeaT

Build 76594 CHG 02459 Script Language PRACTICE RePeaT (Loop with check at end of loop)	**Syntax of PRACTICE command RePeaT changed** The complex syntax of the RePeaT command: RePeaT <count> <block> WHILE <condition> is no longer supported.
Build 50104 CHG 02366 Script Language PRACTICE IF (Conditional script execution) RePeaT (Loop with check at end of loop) WHILE (Loop with check at start of loop)	**Empty lines in PRACTICE allowed** Empty lines are now ignored in PRACTICE. This may change the execution flow of existing programs (e.g. when an IF, REPEAT or WHILE statement is followed by an empty line).

Register

Build 19417 IMP 02158 Register (Processor registers)	**New options for Register, Frame and List** Allows to select the register context. "REGSET" option selects a registerset for architectures with multiple sets, "TASK" selects the registers of a certain task (when the task awareness supports it) and "CORE" selects a certain core (in SMP configurations).
Build 1077 IMP 01177 MPC74XX Debugger Register (Processor registers)	ALTIVEC registers and variables implememted

Register.RELOAD

Build 2940 IMP 01916 Register.RELOAD (Reload the compiler register settings)	**New command Register.RELOAD** Reloads registers as done with last Data.LOAD command.

Build 51144 IMP 02396 Register.Set (Modify register contents)	**New pseudo registers for x86** The registers (CSR,DSR,ESR,FSR,GSR,SSR,TRR,LDTRR) allow to modify a segment register without updating the related base, limit and attribute registers.
Build 4728 FIX 01946 PPC440 Debugger Register.view (Display registers) Register.Set (Modify register contents)	**MCSRR0/1 register write with r.s** MCSRR0/1 register write from the register window (r.s mcsrr0 xxxx) do not take any effect.
Build 4728 FIX 01941 C5500 Debugger Register.Set (Modify register contents)	**Register modification failed** Modification of CFCT, RETA and XARn was not working.

Build 88288 IMP 02497 Data.LOAD.Elf (Load ELF file) EXTension.LOAD (Load extension) Frame.view (Display stack frame) MMU.FORMAT (Define MMU table structure) Register.view (Display registers) TASK.List.MACHINES (List machines) TASK.List.TREE (Display tasks in a tree structure) TRANSlation.List (List MMU translation table) TRANSlation.TableWalk (Automatic MMU page table walk)	**Hypervisor-aware debugging feature introduced (Part 2-2)** - the debugger address translation (MMU and TRANS command groups) can be individually configured for each virtual machine by specifying a machineid in the address parameters. Where appropriate, MMU commands accept the new option /MACHINE <machineid>. Page tables of all machines can be viewed. Access class I: is used to specify intermediate addresses. - individual symbol sets can be loaded for each virtual machine by loading them onto the appropriate extended address: Data.LOAD.<type> <file> <accessclass>:<machineid>:::0 <options> Data.LOAD.<type> <file> <accessclass>:<machineid>:::<spaceID>::0 <options> - multiple awarenesses can be loaded to individual machines using the EXTension.LOAD command with the /machine option: EXTension.LOAD <awarenessfile> /MACHINE <machineid> [/NAME <extname>] - extension commands may use an infix to specify uniquely a command of a specific extension: TASK.[<extname>.]<extcommand> EXTension.[<extname>.]<extcommand> - new commands TASK.List.MACHINES and TASK.List.TREE are available to display the target's hypervisor/guest/process/thread structure - task names may be extended by the machine name, separated by three colons, to uniquely identify them: <machinename>:::<taskname> - register sets and stack frames of machines and tasks of arbitrary guests can be viewed, e.g.: Register.view /MACHINE <machineid> Frame.view /TASK "<machinename>:::<taskname>"
Build 51144 CHG 02390 Register.view (Display registers)	**Stack not shown by default in register window** The command line option Stack enables the stack display. The display can also be toggled by the button in the register window.
Build 4728 FIX 01946 PPC440 Debugger Register.view (Display registers) Register.Set (Modify register contents)	**MCSRR0/1 register write with r.s** MCSRR0/1 register write from the register window (r.s mcsrr0 xxxx) do not take any effect.
Build 1077 FIX 01113 Register.view (Display registers)	**Spotlight display** Spotlight display in register window may not mark changes when stack changes

RunTime

Build 13751 IMP 02102 RunTime (Runtime measurement)	**Tolerance information added to RunTime** Tolerance displayed in tooltip of last execution time.

RunTime.state

Build 2940 IMP 01889 TriCore Debugger RunTime.state (Display results)	**RunTime measurement** Runtime Measurement is more accurate now.
Build 1224 FIX 01746 PPC400 Debugger PPC440 Debugger Go.direct (Start the program execution) RunTime.state (Display results)	**Runtime counter stop randomly** Internal driver error. Wrong mask for JDSR[PSP] bit.

SCREEN.WAIT

Build 80996 CHG 02480 Script Language PRACTICE SCREEN.WAIT (Update screen while waiting)	**Extended functionality of SCREEN.WAIT command** SCREEN.WAIT now waits and updates the screen while waiting. It behaves similar to WAIT command. The old behavior of waiting until all processing windows are completed remained unchanged.

SETUP.ASCIITEXT

Build 1258 IMP 01853 SETUP.ASCIITEXT (Configure ASCII text display) Data.dump (Memory dump) Var.View (Display variables)	**UTF-8 support for ASCII strings/dumps** UTF-8 characters is enabled with SETUP.ASCIITEXT UTF-8.

SETUP.DropCoMmanD

Build 100486 IMP 02543 Integrated Development Environment SETUP.DropCoMmanD (Set command for files dropped into command line)	**Change default for file drop to command line** SETUP.DropCoMmanD allows changing the default command which is used when a file of certain type is dropped into the TRACE32 command line.

SETUP.EDITEXT

Build 51144 IMP 02383 Script Language PRACTICE SETUP.EDITEXT (Define an external editor)	**Editor plug-in for PRACTICE syntax highlighting** Syntax highlighting files for TextPad, UltraEdit, Kate and Notepad++ are provided under demo\practice\syntaxhighlighting

SETUP.IMASKASM

Build 1077 FIX 01123 XScale Debugger SETUP.IMASKASM (Mask interrupts during assembler step) SETUP.IMASKHLL (Mask interrupts during HLL step)	**not working** XScale: IMaskAsm and IMaskHll not working

SETUP.IMASKHLL

Build 1077 FIX 01123 XScale Debugger SETUP.IMASKASM (Mask interrupts during assembler step) SETUP.IMASKHLL (Mask interrupts during HLL step)	**not working** XScale: IMaskAsm and IMaskHll not working

SETUP.PDFViewer

Build 69655 IMP 02438 Installation Guide SETUP.PDFViewer (Context-sensitive help via your favorite PDF viewer)	**New command SETUP.PDFViewer** The new command SETUP.PDFViewer allows to configure a PDF viewer for the TRACE32 manuals. Adobe Acrobat Reader stays the default viewer.

SETUP.StepAutoAsm

Build 1196 IMP 01697 SETUP.StepAutoAsm (HLL steps stops at assembler code)	**New command SETUP.StepAutoAsm** Stops HLL stepping when assembler code is reached.

SHA1SUM

Build 100486 IMP 02535 Uniform Look-And-Feel for all Processors SHA1SUM (Calculate SHA1 checksum of a file)	**New command to calculate SHA-1 checksum for a file** The SHA1SUM command calculates a 160 bit checksum for the given files using the Secure Hash Algorithm.

SIM.CACHE

Build 19417 IMP 02185 TRACE32 Instruction Set Simulator for CORTEX / ARM / XSCALE SIM.CACHE (Cache/MMU simulation and more)	**MMU and Cache support in ARM simulator** MMUs and caches can also be simulated now.

SIM.LOAD

Build 4790 FIX 01978 SIM.LOAD (Load simulator module)	**Simulator port callbacks may fail** Callbacks for port changes may be called with wrong value value (=0) for certain 32bit ports.

SMMU

Build 65657 02432 ARM Debugger Cortex-A/-R Armv8 Debugger SMMU (Hardware system MMU (SMMU))	**New command group for ARM System MMU Analysis** Allows viewing the configuration and status of the ARM System MMU and associated page tables.

SNOOPer

Build 38589 IMP 02318 Intel~ x86/x64 JTAG Debugger BMC.state (Display BMC configuration window) SNOOPer (Sample-based trace)	**Intel StopAndGo Snooping** For Intel Atom/x86 CPUs it is possible to snoop PC and BMC using StopAndGo. The snoop results can be used for profiling, charting and statistics.
Build 1133 IMP 01490 List (Display modes for programs) SNOOPer (Sample-based trace)	SNOOPer/Logger now always 32bit

SNOOPer.CORE

Build 100486 02536 Snooper SNOOPer.CORE (Select cores for PC snooping)	**Core-specific PC snooping for SMP systems** Allows to specify core(s) for SNOOPer.Mode PC in SMP systems.

SNOOPer.Mode

Build 7181 IMP 01992 SNOOPer.Mode (Set operation mode of SNOOPer trace)	**New modes for SNOOPer** StopAndGo mode allows all snooping modes to stop the core temporarily.
Build 4928 IMP 01989 SNOOPer.Mode (Set operation mode of SNOOPer trace)	**SNOOPER can trace MMU spaces** New mode PC+MMU will trace PC and memory space.
Build 2940 IMP 01895 SNOOPer.Mode (Set operation mode of SNOOPer trace)	**New SNOOP.Mode PC for all cores** Allows to make statistics or time charts based on PC snoops.

SPRINTF

Build 88288 IMP 02488 Script Language PRACTICE SPRINTF (Write formatted data to a PRACTICE macro)	**Use sprintf() style formatting for PRACTICE macros** The command SPRINTF allows to assign text and formatted data to a PRACTICE macro in the style of the sprintf() function of C/C++.

SSE.view

Build 51144 IMP 02385 AVX.view (Display AVX registers) AVX512.view (Display AVX512 registers) FPU.view (Display FPU registers) MMX.view (Open MMX register window) SSE.view (Display SSE registers) VPU.view (Display ALTIVEC register window)	**Larger than 64 bit SIMD register sets** FPU, VPE, SPE, MMX, SSE, AVX, AVX512 - Parser for wide registers (>64 bit) unified, values wider than 64 bit possible.

STM

Build 13751 IMP 02088 ICD Solutions for ARM STM (Configure STM component on target)	**STM** New command STM to support configuration of STM and SDTI trace ports.

STOre

Build 125398 IMP 02590 STOre (Store settings as PRACTICE script)	**new command option SYStemLOG for STOre <file>** New command option STOre <file> SYStemLOG to save the settings of the SYStem.LOG window.
Build 112182 IMP 02565 Integrated Development Environment STOre (Store settings as PRACTICE script)	**new option AREA added** Option AREA added to command STOre to save the actual AREA settings to a PRACTICE script.
Build 105499 IMP 02555 ICD In-Circuit Debugger FLASH.Create (Declare FLASH) FLASH.CFI (Generate FLASH declaration by CFI) FLASH.CHANGEtype (Changes the FLASH type) STOre (Store settings as PRACTICE script)	**new option /DISableBulkErase added** For commands FLASH.Create and FLASH.CHANGEtype option /DISableBulkErase added. Now it is also possible to declare it explicitly for non-CFI flash devices to avoid unintended chip erase for truncated flash address range.
Build 1190 FIX 01674 ICE In-Circuit Emulator STOre (Store settings as PRACTICE script)	**STORE Break BreakAll BreakANH** STORE <filename> BREAK, BREAKALL, BREAKANH Inside the generated file the old syntax was used for storing Break.SELect FOREGROUND and Break.SELect BACKGROUND (since 2003.12.15) and old syntax was locked.

SUBTITLE

Build 19417 IMP 02178 SUBTITLE (Define a window subtitle for AMP debugging)	**New command SUBTITLE** Allows to add text in the header of each window. Can be use to distinguish different GUIs of a multicore target.

SYStem

Build 19417 CHG 02181 MPC5xxx and SPC5xx Debugger MPC55xx/MPC56xx NEXUS Debugger and Trace SYStem (System configuration) TrOnchip (Onchip triggers)	**SYStem.Up with CPU selection MPC55XX** The behavior of the MPC55XX (auto-selection type) has changed. When this type is selected, the debugger will reset all SYStem, TrOnchip and trace settings to default upon SYStem mode change. If settings are needed to be set before SYStem.Up, e.g. SYStem.Option.WATCHDOG, SYStem.Option.ETK, first select proper CPU type before, or use SYStem.DETect CPU before setting SYStem options.
Build 1228 FIX 01766 M-Core Debugger SYStem (System configuration)	**SYStem.JtagClock was missing in window** Executing the command from the commandline is possible.

SYStem.CONFIG

Build 19417 IMP 02134 SYStem.CONFIG (Configure debugger according to target topology)	**New command SYStem.CONFIG ListCORE** Allows to show running platforms for CADI and MCD frontend debuggers.

SYStem.CONFIG XCP

Build 100486 IMP 02524 Debugging via XCP SYStem.CONFIG XCP (XCP specific settings)	**Support for ASAM XCP** Support for debugging over XCP as specified in ASAM MCD-1 (XCP) standard, version 1.5 by ASAM e.V for GTM, Qorivva, RH850 and Tricore.

SYStem.CPU

Build 19417 IMP 02118 PPC400 Debugger SYStem.CPU (Select CPU)	**Add PPC405EX/405EXr**
Build 7181 IMP 01994 PPC400 Debugger SYStem.CPU (Select CPU)	**support XILINX designs with JTAGPPC ctrl** support for PPC405 core(s) in chain with FPGA (JTAGPPC controller) Please also see application note in FAQs.

SYStem.JtagClock

Build 1241 IMP 01811 ARM11 JTAG Debugger ARM7 JTAG Debugger ARM9 JTAG Debugger SYStem.JtagClock (Define JTAG frequency)	**New JTAG clock modes CTCK and CRTCK** New JTAG clock modes CTCK and CRTCK (besides RTCK and ARTCK) and improved download speed.

SYStem.LOG.List

Build 88288 IMP 02501 MMU Support SYStem.LOG.List (Log the accesses made by TRACE32) SYStem.LOG.Set (Select the TRACE32 accesses to be logged)	**SYStem.LOG.List now shows address translation failures** SYStem.LOG.List window now shows address translation failures as TRANS errors. If SYStem.LOG.Set TRANSlation is ON all address translation calls are shown in the SYStem.LOG.List window (logical addresses only).

SYStem.LOG.Set

Build 88288 IMP 02501 MMU Support SYStem.LOG.List (Log the accesses made by TRACE32) SYStem.LOG.Set (Select the TRACE32 accesses to be logged)	**SYStem.LOG.List now shows address translation failures** SYStem.LOG.List window now shows address translation failures as TRANS errors. If SYStem.LOG.Set TRANSlation is ON all address translation calls are shown in the SYStem.LOG.List window (logical addresses only).

Build 1190 IMP 01663 SYStem.MemAccess (Run-time memory access (non-intrusive))	**ARM RealMonitor supported** The new memory access mode "RealMON" supports communication with the ARM RealMonitor.

SYStem.Mode

Build 1241 FIX 01802 MPC5xx/8xx Debugger SYStem.Mode (Select mode)	**Sys.M.NoDebug reliable deact. DebugMode** SYS.Mode.NoDebug will not work. Even though if sys.m.nodebug is selected and an external reset will occur. The reason is a permanent running DSCK clock signal.
Build 1224 IMP 01753 TriCore Debugger SYStem.Mode (Select mode)	**Tricore ATTACH mode implemented** Attach allows to connect to running target. New feature is available on all "Power" tools.
Build 1081 IMP 01405 In-Circuit Emulator for MC68020/30 SYStem.Mode (Select mode)	speed of sys.mode standby for ICE68020 improved
Build 1077 FIX 01289 MPC56x NEXUS Debugger and Trace SYStem.Mode (Select mode)	Change of DTM setting for MPC5xx nexus was not affecting next sys.mode standby

Build 13751 FIX 02067 SYStem.Option (Special setup)	**SYStem.Option SOFTLONG fails**
Build 9640 FIX 02027 MPC5xx/8xx Debugger SYStem.Option (Special setup)	**MPC500/800:SYStem.Option.SCRATCH** independent from the SYStem.Option.SCRATCH setting the AUTO mode was used.
Build 7181 CHG 01999 TriCore Debugger SYStem.Option (Special setup)	**TriCore SYStem.Option change** After a transition period, the following TriCore SYStem.Options will become obsolete: SYStem.Option TC1130FIX SYStem.Option TC1796FIX SYStem.Option TC1900FIX These options were used for disabling the internal watchdog on SYStem.Mode [Up|Go|Attach] as a bugfix. As replacement was introduced: SYStem.Option WATCHDOGFIX The functionality will exactly be the same, except that it has only effect on TriCore Chips that have the corresponding bug. For disabling the watchdog (e.g. for flash programming) please use Data.Set 0xF0000024 %Long 0x00000008 after the SYStem.Up command. See the demo scripts for an example.
Build 4728 CHG 01958 NIOS II Debugger SYStem.Option (Special setup)	**New sytem options BTM,DTM,SYNC** Control behavior of off-chip and on-chip trace via system.option.
Build 4728 CHG 01956 NIOS II Debugger SYStem.Option (Special setup)	**New system option FSS** Support for different trace clock timings.
Build 4728 CHG 01904 MPC5200 Debugger PowerQUICC II/Pro Debugger SYStem.Option (Special setup)	**new SYStem.Option.IP.BOTH** new MSR[IP] handling for Onchip-BP's on processor with two Onchip-BP's. (G2_LE /ZEPPO core) Default is SYS.O.IP.AUTO where only the Onchip-BP control the active (MSR[IP]) exception handler block.
Build 1246 FIX 01819 PowerQUICC II/Pro Debugger SYStem.Option (Special setup)	**SYStem.Option.IP AUTO do not work at all** SYStem.Option.IP.AUTO may fail in conjunction with SYStem.Option.BASE.AUTO
Build 1241 FIX 01795 XScale Debugger SYStem.Option (Special setup)	**System.Option DynVector does not work** Non working system option may cause problems for operating system awareness. Unintended interrupt may occur because modified vector table is not read during runtime.

Build 1190 01685 DSP56K Debugger ONCE Debugger for DSP56300 SYStem.Option (Special setup)	**DSP56xxx IMASKASM/IMASKHLL locked** SYStem.Option IMASKASM/IMASKHLL was locked unintentionally.
Build 1133 IMP 01495 ARM Debugger SYStem.Option (Special setup)	SYStem.Option StepVector and ICEbreaker for single stepping changed for ARM
Build 1133 FIX 01461 SYStem.Option (Special setup)	SYStem.Option IP now taken as operation value for breakpoints no PPC6xx, PPC7xx and PPC8xxx

SYStem.Option MACHINESPACES

Build 88288 IMP 02498 ARM Debugger Cortex-A/-R Armv8 Debugger Intel~ x86/x64 JTAG Debugger SYStem.Option MACHINESPACES (Address extension for guest OSes) SYStem.Option MACHINESPACES (Address extension for guest OSes) SYStem.Option MACHINESPACES (Address extension for guest OSes)	**Hypervisor-aware debugging feature introduced (Part 1-2)** The TRACE32 machinespaces framework was added for architectures ARM and Intel x86 to support Hypervisor environments. This framework allows the debugger to perform stop-mode debugging of targets running a hypervisor and one or multiple guest machines. In conjunction with a hypervisor awareness it is possible to debug all guest machines concurrently, whether they are currently active or currently inactive. TRACE32 allows to load an individual OS awareness for each virtual machine on the target. The machinespaces framework is enabled with the command SYStem.Option.MACHINESPACES ON. A new identifier <machineid> is introduced which allows to identify virtual machines on the target. TRACE32 addresses are augmented with the machineid to specify which machine the address belongs to. The TRACE32 debugger address translation (enabled with TRANSlation.ON) is extended to fully support the two-stage address translation which is usually applied for hardware virtualized guest machines. The first translation stage translates guest virtual addresses to intermediate physical addresses. The second translation stage translates intermediate physical addresses to absolute physical addresses. The MMU registers needed for the page table walk steps are automatically fetched from the hardware registers or, for inactive guests, from hypervisor data structures. If SYStem.Option.MACHINESPACES is enabled, - addresses are extended with a machine ID identifier, separating the machine ID with three colons from the (extended) virtual address: <accessclass>:<machineid>:::<spaceid>::<virtaddress> The machine ID clearly specifies which host or guest machine the address belongs to. The host machine always uses machineid 0. Guests have a machineid larger than 0. TRACE32 currently supports up to 30 machines.

SYStem.Option MEMoryMODEL

Build 51144 ODD 02384 Intel~ x86/x64 JTAG Debugger SYStem.Option MEMoryMODEL (Define memory model)	**LARGE memory model for Intel x86** Default memory model for Intel x86 is LARGE now.

SYStem.Option PARITY

Build 1224 IMP 01761 PowerQUICC II/Pro Debugger SYStem.Option PARITY (Generate parity on memory access)	**write parity support for PQ2 (groucho)**

SYStem.Option STEPSOFT

Build 9640 IMP 02034 TriCore Debugger SYStem.Option STEPSOFT (Step with software breakpoints)	**TriCore soft single-stepping added** TriCore single-stepping can now also be performed by using software breakpoints. Use SYStem.Option STEPSOFT ON to enable this feature. The default is using on-chip breakpoints.

SYStem.Option ZoneSPACES

Build 60219 IMP 02422 Cortex-A/-R Armv8 Debugger SYStem.Option ZoneSPACES (Enable symbol management for Arm zones)	**Onchip Breakpoints ARMv8 are now mode aware** Onchip breakpoints are now secure/nonsecure/hypervisor aware when SYStem.Option ZoneSPACES is enabled.

SYStem.POLLING

Build 9640 IMP 02046 SYStem.POLLING (Polling mode of CPU)	**New command to control JTAG polling** Allows to increase or decrease polling of target via JTAG when target is running.

SYStem.RESet

Build 1161 FIX 01619 S12 Debugger SYStem.RESet (Reset configuration)	**SYStem.RESet clears option of MCS12** Executing "SYStem.RESet" clears the "SYStem.Option ROMHM" when a MCS12 cpu is selected. On default this option should be set.

SYStem.RESetOut

Build 9640 IMP 02023 TriCore Debugger SYStem.RESetOut (Reset peripherals)	**Target reset feature added** SYStem.RESetOut triggers a 2 ms low pulse on the nRESET line on the JTAG connector. This will reset the CPU. Formerly this command was only available in the Simulator.

SYStem.RESetTarget

Build 9640 IMP 02048 SYStem.RESetTarget (Release target reset)	**New command SYStem.RESetTarget** Resets target system. On most targets simmilar to system.up and register.reset. On virtual platforms usually activates a target platform reset.

SYnch.Connect

Build 100486 IMP 02537 Multicore Debugging SYnch.Connect (Connect to other TRACE32 PowerView instances)	**Keywords and names simplify use of SYnch.Connect command** The SYnch.Connect command allow using the keywords SELF, ALL and OTHERS to address TRACE32 instance(s), as well as the name assigned to a TRACE32 instance and a name pattern.

Step

Build 1077 CHG 01094 Step (Steps through the program)	**Step.XXX restores context** Step.XXX restores context (from R.UP etc.) before executing

Step.BackChange

Build 1264 CHG 01860 ICE In-Circuit Emulator Step.BackChange (Step back till expression changes)	**commandname changed** longname changed from Step.BackChang to Step.BackChange
Build 1077 FIX 01320 Step.BackChange (Step back till expression changes)	StepBackChange was stepping forward (CTS)

Step.BackOver

Build 1264 IMP 01861 ICE In-Circuit Emulator Go.BackEntry (Go back in program to function entry (CTS)) Step.BackOver (Step back)	**new commands for CTS introduced** Go.BackEntry Go.BackTillViolation Go.TillViolation Step.BackOver
Build 1264 IMP 01857 Go.BackEntry (Go back in program to function entry (CTS)) Step.BackOver (Step back)	**New CTS backward execution features** Step-Back-Over HLL or assembler and Go-Back-Till_Entry supported.

Step.Diverge

Build 51144 IMP 02391 Step.Diverge (Step till next unreached line)	**New command Step.Diverge** Command allow to step still the target branches to new locations that have not yet been stepped.

Step.Hll

Build 7181 FIX 02007 MPC5xxx and SPC5xx Debugger Step.Hll (HLL single-stepping)	**HLL stepping may ignore BPs on MPC55xx**

Step.Over

Build 1077 FIX 01077 Step.Over (Step over call)	**indirect jump** Step.Over HLL made regular Step when target function was called by indirect jump instruction

SystemTrace

Build 13751 IMP 02091 ICD Solutions for ARM CAnalyzer (Trace features of Compact Analyzer) SystemTrace (MIPI STP and CoreSight ITM)	**CORTEX-M3 Trace Support** Support for Cortex-M3 tracing via SWV and TPIU for ETM and ITM added.

TAR

Build 93173 IMP 02508 Integrated Development Environment TAR (Pack files into an archive)	**Pack files into an archive** Packs the specified files without compression into a tape archive.

TASK.CONFIG

Build 88288 IMP 02493 OS-aware Debugging TASK.select (Display context of specified task) TASK.CONFIG (Configure OS Awareness)	**TASK.select is now default command for TASK command group** Now TASK.select is default command for TASK command group. It was TASK.CONFIG before.

TASK.CrashDump

Build 93173 IMP 02507 RTOS Debugger for Windows Data.LOAD.CrashDump (Load MS Windows Crash Dump file) TASK.CrashDump (Windows crash dump analysis)	**Support for loading Windows crash dump** After loading the full Crash Dump file into the simulator or the target memory with the command Data.LOAD.CrashDump, the Windows awareness command TASK.CrashDump.LOADNT is used to auto-load the kernel debug symbols. After that the command TASK.CrashDump.LOADREG could be used to load the context of all the available cores.

TASK.DMESG

Build 42354 IMP 02349 RTOS Debugger for Linux TASK.DMESG (Display the kernel ring buffer) TASK.DTB (Display the device tree blob)	**new commands for Linux awareness** TASK.DMESG to display kernel log TASK.DTB to display device tree blob

TASK.DTB

Build 42354 IMP 02349 RTOS Debugger for Linux TASK.DMESG (Display the kernel ring buffer) TASK.DTB (Display the device tree blob)	**new commands for Linux awareness** TASK.DMESG to display kernel log TASK.DTB to display device tree blob

Build 88288 IMP 02497 Data.LOAD.Elf (Load ELF file) EXTension.LOAD (Load extension) Frame.view (Display stack frame) MMU.FORMAT (Define MMU table structure) Register.view (Display registers) TASK.List.MACHINES (List machines) TASK.List.TREE (Display tasks in a tree structure) TRANSlation.List (List MMU translation table) TRANSlation.TableWalk (Automatic MMU page table walk)	**Hypervisor-aware debugging feature introduced (Part 2-2)** - the debugger address translation (MMU and TRANS command groups) can be individually configured for each virtual machine by specifying a machineid in the address parameters. Where appropriate, MMU commands accept the new option /MACHINE <machineid>. Page tables of all machines can be viewed. Access class I: is used to specify intermediate addresses. - individual symbol sets can be loaded for each virtual machine by loading them onto the appropriate extended address: Data.LOAD.<type> <file> <accessclass>:<machineid>:::0 <options> Data.LOAD.<type> <file> <accessclass>:<machineid>:::<spaceID>::0 <options> - multiple awarenesses can be loaded to individual machines using the EXTension.LOAD command with the /machine option: EXTension.LOAD <awarenessfile> /MACHINE <machineid> [/NAME <extname>] - extension commands may use an infix to specify uniquely a command of a specific extension: TASK.[<extname>.]<extcommand> EXTension.[<extname>.]<extcommand> - new commands TASK.List.MACHINES and TASK.List.TREE are available to display the target's hypervisor/guest/process/thread structure - task names may be extended by the machine name, separated by three colons, to uniquely identify them: <machinename>:::<taskname> - register sets and stack frames of machines and tasks of arbitrary guests can be viewed, e.g.: Register.view /MACHINE <machineid> Frame.view /TASK "<machinename>:::<taskname>"

TASK.List.TREE

Build 130863 IMP 02606 TASK.List.TREE (Display tasks in a tree structure)	**new option /Machine for command TASK.List.TREE**
Build 88288 IMP 02497 Data.LOAD.Elf (Load ELF file) EXTension.LOAD (Load extension) Frame.view (Display stack frame) MMU.FORMAT (Define MMU table structure) Register.view (Display registers) TASK.List.MACHINES (List machines) TASK.List.TREE (Display tasks in a tree structure) TRANSlation.List (List MMU translation table) TRANSlation.TableWalk (Automatic MMU page table walk)	**Hypervisor-aware debugging feature introduced (Part 2-2)** - the debugger address translation (MMU and TRANS command groups) can be individually configured for each virtual machine by specifying a machineid in the address parameters. Where appropriate, MMU commands accept the new option /MACHINE <machineid>. Page tables of all machines can be viewed. Access class I: is used to specify intermediate addresses. - individual symbol sets can be loaded for each virtual machine by loading them onto the appropriate extended address: Data.LOAD.<type> <file> <accessclass>:<machineid>:::0 <options> Data.LOAD.<type> <file> <accessclass>:<machineid>:::<spaceID>::0 <options> - multiple awarenesses can be loaded to individual machines using the EXTension.LOAD command with the /machine option: EXTension.LOAD <awarenessfile> /MACHINE <machineid> [/NAME <extname>] - extension commands may use an infix to specify uniquely a command of a specific extension: TASK.[<extname>.]<extcommand> EXTension.[<extname>.]<extcommand> - new commands TASK.List.MACHINES and TASK.List.TREE are available to display the target's hypervisor/guest/process/thread structure - task names may be extended by the machine name, separated by three colons, to uniquely identify them: <machinename>:::<taskname> - register sets and stack frames of machines and tasks of arbitrary guests can be viewed, e.g.: Register.view /MACHINE <machineid> Frame.view /TASK "<machinename>:::<taskname>"

TASK.ORTI

Build 42354 IMP 02324 RTOS Debugger for OSEK/ORTI TASK.ORTI (AUTOSAR/OSEK support)	**TASK.ORTI generates ORTI menu** Depending on the contents of the loaded ORTI file, the TASK.ORTI command automatically generates additional menu entries for displaying OS objects and for performance analysis on selected OS attributes.

TASK.select

Build 88288 IMP 02493 OS-aware Debugging TASK.select (Display context of specified task) TASK.CONFIG (Configure OS Awareness)	**TASK.select is now default command for TASK command group** Now TASK.select is default command for TASK command group. It was TASK.CONFIG before.

TERM

Build 1096 IMP 01447 TERM (Terminal emulation)	TERM window with CPU-stop protocol changed to fully stop when nothing is ready

TERM.GATE

Build 19417 IMP 02184 ARC Debugger TERM.GATE (Terminal with virtual hosting)	**Support for Metaware Hostlink library**
Build 13751 IMP 02115 TERM.GATE (Terminal with virtual hosting) TERM.WRITE (Write terminal output to file)	**TERM.WRITE works also for TERM.GATE** Only terminal output is now saved into the file.
Build 1258 FIX 01852 ICD Solutions for ARM TRACE32 Instruction Set Simulator for CORTEX / ARM / XSCALE TERM.GATE (Terminal with virtual hosting)	**ARM semihosting file append not working** Append to existing files (e.g. fopen with "a+" mode) was not working as expected.

TERM.METHOD

Build 100486 IMP 02532 Uniform Look-And-Feel for all Processors TERM.view (Terminal display) TERM.METHOD (Select terminal protocol)	**Multiple open TERMinal windows are now possible** A channel number was introduced to allow the concurrent use of several TERMinal windows.
Build 34458 IMP 02272 TERM.METHOD (Select terminal protocol)	**TCP method added** New method TCP for TERM.METHOD command to support transfers via TCP. Can be used with TERM.TELNET to open telnet terminals.

TERM.Mode

Build 1077 IMP 01064 TERM.Mode (Define terminal type)	TERM.MODE HEX implemented

TERM.SIZE

Build 42354 IMP 02337 TERM.SIZE (Define size of terminal window)	**Terminal window can keep backlog** The third parameter of the TERM.SIZE command defines the size of the backlog. The backlog is updated whenever a line scrolls out of the "real" part of the TERM window.

TERM.TELNET

Build 34458 IMP 02271 TERM.TELNET (Open TELNET terminal window)	**New command TERM.TELNET to support telne**

TERM.WRITE

Build 13751 IMP 02115 TERM.GATE (Terminal with virtual hosting) TERM.WRITE (Write terminal output to file)	**TERM.WRITE works also for TERM.GATE** Only terminal output is now saved into the file.

TERM.view

Build 100486 IMP 02532 Uniform Look-And-Feel for all Processors TERM.view (Terminal display) TERM.METHOD (Select terminal protocol)	**Multiple open TERMinal windows are now possible** A channel number was introduced to allow the concurrent use of several TERMinal windows.
Build 4728 IMP 01964 TERM.view (Terminal display)	**VT100 emulation extended**
Build 1241 IMP 01801 TERM.view (Terminal display)	**TERM window accepts ^C, ^X and DEL**

TPU.List

Build 1077 FIX 01111 TPU Debugger (68332, MPC55x/56x) TPU.List (View microcode)	**TPU Disassembler** TPU Disassembler bug fixed for constant +1/-1 add

TPU.Register.Set

Build 2940 IMP 01907 TPU.Register.Set (Register modification)	**TPU debugger allows FLx setting**

TPU.Step

Build 1145 FIX 01569 TPU Debugger (68332, MPC55x/56x) TPU.Step (Single step TPU)	TPU command was locked on CPU32 BDM debuggers

TRANSlation

Build 19417 CHG 02163 MMU (Memory management unit) TRANSlation (Debugger address translation)	**Debugger MMU commands renamed to TRANS** All commands that just effect the debuggers address translation are moved to the new command "TRANSlation".

TRANSlation.COMMON

Build 65657 02437 MMU Support TRANSlation.COMMON (Common address ranges for kernel and tasks)	**No logical operators in TRANSlation.COMMON command** Old way of specifiying multiple COMMON address ranges with OR operation has been removed (TRANS.COMMON 0x80000000--0x8FFFFFFF||0xA0000000--0xBFFFFFFF). Please use spaces or the command TRANSlation.COMMON.ADD to specify additional address ranges.

Build 88288 IMP 02497 Data.LOAD.Elf (Load ELF file) EXTension.LOAD (Load extension) Frame.view (Display stack frame) MMU.FORMAT (Define MMU table structure) Register.view (Display registers) TASK.List.MACHINES (List machines) TASK.List.TREE (Display tasks in a tree structure) TRANSlation.List (List MMU translation table) TRANSlation.TableWalk (Automatic MMU page table walk)	**Hypervisor-aware debugging feature introduced (Part 2-2)** - the debugger address translation (MMU and TRANS command groups) can be individually configured for each virtual machine by specifying a machineid in the address parameters. Where appropriate, MMU commands accept the new option /MACHINE <machineid>. Page tables of all machines can be viewed. Access class I: is used to specify intermediate addresses. - individual symbol sets can be loaded for each virtual machine by loading them onto the appropriate extended address: Data.LOAD.<type> <file> <accessclass>:<machineid>:::0 <options> Data.LOAD.<type> <file> <accessclass>:<machineid>:::<spaceID>::0 <options> - multiple awarenesses can be loaded to individual machines using the EXTension.LOAD command with the /machine option: EXTension.LOAD <awarenessfile> /MACHINE <machineid> [/NAME <extname>] - extension commands may use an infix to specify uniquely a command of a specific extension: TASK.[<extname>.]<extcommand> EXTension.[<extname>.]<extcommand> - new commands TASK.List.MACHINES and TASK.List.TREE are available to display the target's hypervisor/guest/process/thread structure - task names may be extended by the machine name, separated by three colons, to uniquely identify them: <machinename>:::<taskname> - register sets and stack frames of machines and tasks of arbitrary guests can be viewed, e.g.: Register.view /MACHINE <machineid> Frame.view /TASK "<machinename>:::<taskname>"

Build 88288 IMP 02497 Data.LOAD.Elf (Load ELF file) EXTension.LOAD (Load extension) Frame.view (Display stack frame) MMU.FORMAT (Define MMU table structure) Register.view (Display registers) TASK.List.MACHINES (List machines) TASK.List.TREE (Display tasks in a tree structure) TRANSlation.List (List MMU translation table) TRANSlation.TableWalk (Automatic MMU page table walk)	**Hypervisor-aware debugging feature introduced (Part 2-2)** - the debugger address translation (MMU and TRANS command groups) can be individually configured for each virtual machine by specifying a machineid in the address parameters. Where appropriate, MMU commands accept the new option /MACHINE <machineid>. Page tables of all machines can be viewed. Access class I: is used to specify intermediate addresses. - individual symbol sets can be loaded for each virtual machine by loading them onto the appropriate extended address: Data.LOAD.<type> <file> <accessclass>:<machineid>:::0 <options> Data.LOAD.<type> <file> <accessclass>:<machineid>:::<spaceID>::0 <options> - multiple awarenesses can be loaded to individual machines using the EXTension.LOAD command with the /machine option: EXTension.LOAD <awarenessfile> /MACHINE <machineid> [/NAME <extname>] - extension commands may use an infix to specify uniquely a command of a specific extension: TASK.[<extname>.]<extcommand> EXTension.[<extname>.]<extcommand> - new commands TASK.List.MACHINES and TASK.List.TREE are available to display the target's hypervisor/guest/process/thread structure - task names may be extended by the machine name, separated by three colons, to uniquely identify them: <machinename>:::<taskname> - register sets and stack frames of machines and tasks of arbitrary guests can be viewed, e.g.: Register.view /MACHINE <machineid> Frame.view /TASK "<machinename>:::<taskname>"

TargetSystem.NewInstance

Build 105499 IMP 02554 Integrated Development Environment TargetSystem.NewInstance (Start new TRACE32 PowerView instance)	**TargetSystem.NewInstance supports new port types** TargetSystem.NewInstance supports GDB, TCF port as well as multiple API ports now. Added new options: /API<n>.PORT <port>, /GDB.PORT <port>, /GDB.PPROTocolTCP|UDP, /TCF.PORT <port>
Build 100486 IMP 02540 Multicore Debugging TargetSystem.NewInstance (Start new TRACE32 PowerView instance)	**New command TargetSystem.NewInstance** The command TargetSystem.NewInstance creates the config file for the new TRACE32 instance and then starts the TRACE32 instance for the specified architecture (e.g. t32marm.exe). Each started TRACE32 instance gets a name, which simplifies the handling of multiple instances. The new instance is automatically linked to the same PowerDebug module or the same MCI Server as the instance that accepted the command.

TrBus.Out

Build 4728 FIX 01960 TriCore Debugger TrBus.Out (Define source for the external trigger pulse)	**Trigger on target break** Added option to generate a PODBUS trigger pulse when target breaks.
Build 2940 CHG 01891 TrBus.Out (Define source for the external trigger pulse)	**TrBus.Out now off by default** Was enabled by default till now.

Build 19417 CHG 02181 MPC5xxx and SPC5xx Debugger MPC55xx/MPC56xx NEXUS Debugger and Trace SYStem (System configuration) TrOnchip (Onchip triggers)	**SYStem.Up with CPU selection MPC55XX** The behavior of the MPC55XX (auto-selection type) has changed. When this type is selected, the debugger will reset all SYStem, TrOnchip and trace settings to default upon SYStem mode change. If settings are needed to be set before SYStem.Up, e.g. SYStem.Option.WATCHDOG, SYStem.Option.ETK, first select proper CPU type before, or use SYStem.DETect CPU before setting SYStem options.
Build 13751 CHG 02080 PPC400 Debugger PPC440 Debugger TrOnchip (Onchip triggers)	**change mode of r/w data breakpoints** Mode for data BP will be ASYNCHRONOUS instead of SYNCHRONOUS by default from now on. With 'TrOnchip.SYNCHRONOUS ON' could be switched to old mode.
Build 13751 IMP 02060 TriCore Debugger TrOnchip (Onchip triggers)	**MCBS feature added for TriCore and PCP** Newer TriCore chips (e.g. the AUDO-NG or TC11xx family) have two internal break buses and a suspend bus which can be used to distribute break signals beween different modules (MLI, DMA, ...) and cores (TriCore, PCP). Configuration of this feature can be done via the TrOnchip window or the TrOnchip commands.
Build 1190 IMP 01668 TrOnchip (Onchip triggers)	**Benchmarking counters on TI ARM cores** The benchmarking counters no TI ARM cores (ARM925T and ARM926EJ) are now supported by the TrOnchip window.

TrOnchip.state

Build 1077 CHG 01225 C166 Family Real-Time Trace TrOnchip.state (Display onchip trigger window)	C166 Risctrace trace control breakpoints implemented default settings of TrOnchip changed

TrPOD.state

Build 80996 CHG 02476 Trigger Probe for PODBUS TrPOD.state (State display)	**Command group TrPod renamed to TrPOD** Command group TrPod renamed to TrPOD.

Trace.METHOD

Build 19417 IMP 02148 Trace.METHOD (Select trace method)	**New trace methods for SMP traces** MergedAnalyzer and MergedOnchip show the trace for all cores together. Same as supplying the /MergeCORE option to the trace commands.

UNARchive

Build 80996 IMP 02482 Integrated Development Environment UNARchive (Linux and Microsoft libraries)	**Extract Linux/Microsoft libraries to directory** A new command UNARchive is available to extract files from Linux libraries (.a) and Microsoft libraries (.lib) to a directory. The two additional subcommands UNARchive.Table and UNARchive.Show help to determine the contents of the archive and to check the result of the unarchive operation.

VE

Build 88288 CHG 02489 VE (Virtual execution mode)	**Command group VM.* renamed** Command group VM.* renamed to VE.* (VirtualExecution).

VPU.view

Build 51144 IMP 02385 AVX.view (Display AVX registers) AVX512.view (Display AVX512 registers) FPU.view (Display FPU registers) MMX.view (Open MMX register window) SSE.view (Display SSE registers) VPU.view (Display ALTIVEC register window)	**Larger than 64 bit SIMD register sets** FPU, VPE, SPE, MMX, SSE, AVX, AVX512 - Parser for wide registers (>64 bit) unified, values wider than 64 bit possible.

Build 1228 IMP 01775 Var (HLL variables and expressions)	**MAP.VOLATILE affects now var display** Memory access width rules for arrays and structures in volatile areas are now handled more strict, i.e. an array of 16bit values will be read with 16bit memory accesses instead of 32bit.
Build 1228 FIX 01769 Var (HLL variables and expressions)	**Variable window with inherited may crash** Displaying classes with inherited members and long inherited class names ay crash.
Build 1228 IMP 01765 Var (HLL variables and expressions)	**untagged unions in structs supported** Access to untagged unions in HLL expressions is now possible.

Build 88288 IMP 02491 High-Level-Language Debugging Var.AddWatch (Add variable to Var.Watch window) Var.Watch (Open Var.Watch window)	**Var.Watch keeps formatting if opened without variable** Var.Watch window applies formatting options to all added variables, if opened with format parameters but without variable names.
Build 38589 IMP 02302 Var.set, Var.Assign, Var.AddWatch, Var.Break.direct, Var.Break.Delete, Var.Break.Pass, Var.Break.Set, Var.Call, Var.CHAIN, Var.DRAW, Var.DUMP, Var.DelWatch, Var.FixedCHAIN, Var.FixedTABle, Var.Go, Var.Go.Back, Var.Go.Change, Var.Go.Till, Var.IF, Var.INFO, Var.Local, Var.LOG, Var.NEW, Var.NEWGLOBAL, Var.NEWLOCAL, Var.PRINT, Var.Ref, Var.Step.BackChange, Var.Step.BackTill, Var.Step.Change, Var.Step.Till, Var.TABle, Var.TYPE, Var.View, Var.Watch, Var.WHILE, Var.WRITE	**HLL expression parser extended** Pointers and address values can be casted to certain segments or memory access classes.
Build 1151 FIX 01613 Var.AddWatch (Add variable to Var.Watch window)	**AddWatch** problem fixed in AddWatch pulldown menu item - may crash debugger

Var.Assign

Build 38589 IMP 02302 Var.set, Var.Assign, Var.AddWatch, Var.Break.direct, Var.Break.Delete, Var.Break.Pass, Var.Break.Set, Var.Call, Var.CHAIN, Var.DRAW, Var.DUMP, Var.DelWatch, Var.FixedCHAIN, Var.FixedTABle, Var.Go, Var.Go.Back, Var.Go.Change, Var.Go.Till, Var.IF, Var.INFO, Var.Local, Var.LOG, Var.NEW, Var.NEWGLOBAL, Var.NEWLOCAL, Var.PRINT, Var.Ref, Var.Step.BackChange, Var.Step.BackTill, Var.Step.Change, Var.Step.Till, Var.TABle, Var.TYPE, Var.View, Var.Watch, Var.WHILE, Var.WRITE	**HLL expression parser extended** Pointers and address values can be casted to certain segments or memory access classes.

Var.Break.Delete

Build 38589 IMP 02302 Var.set, Var.Assign, Var.AddWatch, Var.Break.direct, Var.Break.Delete, Var.Break.Pass, Var.Break.Set, Var.Call, Var.CHAIN, Var.DRAW, Var.DUMP, Var.DelWatch, Var.FixedCHAIN, Var.FixedTABle, Var.Go, Var.Go.Back, Var.Go.Change, Var.Go.Till, Var.IF, Var.INFO, Var.Local, Var.LOG, Var.NEW, Var.NEWGLOBAL, Var.NEWLOCAL, Var.PRINT, Var.Ref, Var.Step.BackChange, Var.Step.BackTill, Var.Step.Change, Var.Step.Till, Var.TABle, Var.TYPE, Var.View, Var.Watch, Var.WHILE, Var.WRITE	**HLL expression parser extended** Pointers and address values can be casted to certain segments or memory access classes.

Build 38589 IMP 02302 Var.set, Var.Assign, Var.AddWatch, Var.Break.direct, Var.Break.Delete, Var.Break.Pass, Var.Break.Set, Var.Call, Var.CHAIN, Var.DRAW, Var.DUMP, Var.DelWatch, Var.FixedCHAIN, Var.FixedTABle, Var.Go, Var.Go.Back, Var.Go.Change, Var.Go.Till, Var.IF, Var.INFO, Var.Local, Var.LOG, Var.NEW, Var.NEWGLOBAL, Var.NEWLOCAL, Var.PRINT, Var.Ref, Var.Step.BackChange, Var.Step.BackTill, Var.Step.Change, Var.Step.Till, Var.TABle, Var.TYPE, Var.View, Var.Watch, Var.WHILE, Var.WRITE	**HLL expression parser extended** Pointers and address values can be casted to certain segments or memory access classes.

Build 38589 IMP 02302 Var.set, Var.Assign, Var.AddWatch, Var.Break.direct, Var.Break.Delete, Var.Break.Pass, Var.Break.Set, Var.Call, Var.CHAIN, Var.DRAW, Var.DUMP, Var.DelWatch, Var.FixedCHAIN, Var.FixedTABle, Var.Go, Var.Go.Back, Var.Go.Change, Var.Go.Till, Var.IF, Var.INFO, Var.Local, Var.LOG, Var.NEW, Var.NEWGLOBAL, Var.NEWLOCAL, Var.PRINT, Var.Ref, Var.Step.BackChange, Var.Step.BackTill, Var.Step.Change, Var.Step.Till, Var.TABle, Var.TYPE, Var.View, Var.Watch, Var.WHILE, Var.WRITE	**HLL expression parser extended** Pointers and address values can be casted to certain segments or memory access classes.
Build 22490 IMP 02203 Var.Break.Set (Set breakpoint to HLL expression)	**Var.Break.Set data capabilities extended** The data expression can now also be a bitmask or range.
Build 19417 IMP 02149 Break.Set (Set breakpoints) Var.Break.Set (Set breakpoint to HLL expression)	**Data value breakpoint support** Data value breakpoints are now supported via stop and go when the target has no data comparision capabilities.
Build 13751 IMP 02085 ICD Solutions for ARM Break.Set (Set breakpoints) Var.Break.Set (Set breakpoint to HLL expression)	**ARM ETM Breakpoints Extended** ETM Breakpoints now support also combined program/data breakpoints and breakpoints with condition pass/fail.
Build 4943 FIX 01990 C5500 Debugger Break.Set (Set breakpoints) Var.Break.Set (Set breakpoint to HLL expression)	**C55x data breakpoint problems fixed** Breakpoints with data value and long access breakpoints failed sometimes.

Build 1253 FIX 01847 Break.Set (Set breakpoints) Var.Break.Set (Set breakpoint to HLL expression)	**Onchip program exclude breakpoint failed** Breakpoint refused when no ETM breakpoints usable and range does not fit into single address mask.
Build 1133 IMP 01477 Var.Break.Set (Set breakpoint to HLL expression)	New command TrOnchip.VarCONVert to convert Var.Break.Set commands into single address when possible
Build 1081 FIX 01404 Var.Break.Set (Set breakpoint to HLL expression)	Default data width for Var.Break.Set /DATA option not used for /MemoryRead and /MemoryWrite breakpoints

Var.Break.direct

Build 38589 IMP 02302 Var.set, Var.Assign, Var.AddWatch, Var.Break.direct, Var.Break.Delete, Var.Break.Pass, Var.Break.Set, Var.Call, Var.CHAIN, Var.DRAW, Var.DUMP, Var.DelWatch, Var.FixedCHAIN, Var.FixedTABle, Var.Go, Var.Go.Back, Var.Go.Change, Var.Go.Till, Var.IF, Var.INFO, Var.Local, Var.LOG, Var.NEW, Var.NEWGLOBAL, Var.NEWLOCAL, Var.PRINT, Var.Ref, Var.Step.BackChange, Var.Step.BackTill, Var.Step.Change, Var.Step.Till, Var.TABle, Var.TYPE, Var.View, Var.Watch, Var.WHILE, Var.WRITE	**HLL expression parser extended** Pointers and address values can be casted to certain segments or memory access classes.

Build 38589 IMP 02302 Var.set, Var.Assign, Var.AddWatch, Var.Break.direct, Var.Break.Delete, Var.Break.Pass, Var.Break.Set, Var.Call, Var.CHAIN, Var.DRAW, Var.DUMP, Var.DelWatch, Var.FixedCHAIN, Var.FixedTABle, Var.Go, Var.Go.Back, Var.Go.Change, Var.Go.Till, Var.IF, Var.INFO, Var.Local, Var.LOG, Var.NEW, Var.NEWGLOBAL, Var.NEWLOCAL, Var.PRINT, Var.Ref, Var.Step.BackChange, Var.Step.BackTill, Var.Step.Change, Var.Step.Till, Var.TABle, Var.TYPE, Var.View, Var.Watch, Var.WHILE, Var.WRITE	**HLL expression parser extended** Pointers and address values can be casted to certain segments or memory access classes.

Build 38589 IMP 02302 Var.set, Var.Assign, Var.AddWatch, Var.Break.direct, Var.Break.Delete, Var.Break.Pass, Var.Break.Set, Var.Call, Var.CHAIN, Var.DRAW, Var.DUMP, Var.DelWatch, Var.FixedCHAIN, Var.FixedTABle, Var.Go, Var.Go.Back, Var.Go.Change, Var.Go.Till, Var.IF, Var.INFO, Var.Local, Var.LOG, Var.NEW, Var.NEWGLOBAL, Var.NEWLOCAL, Var.PRINT, Var.Ref, Var.Step.BackChange, Var.Step.BackTill, Var.Step.Change, Var.Step.Till, Var.TABle, Var.TYPE, Var.View, Var.Watch, Var.WHILE, Var.WRITE	**HLL expression parser extended** Pointers and address values can be casted to certain segments or memory access classes.
Build 1077 FIX 01258 C166CBC Debugger FIRE Emulator for C166 Family In-Circuit Emulator for C166/ST10 - Out of Production C166 Monitor Var.Call (Call a new procedure)	bugs fixed in Var.Call for C166 architecture

Build 130863 IMP 02615 Var.DRAW (Graphical variable display)	**Var.DRAW extended to display two-dimensional arrays** The command Var.DRAW has been extended to show contents of two-dimensional arrays.
Build 38589 IMP 02302 Var.set, Var.Assign, Var.AddWatch, Var.Break.direct, Var.Break.Delete, Var.Break.Pass, Var.Break.Set, Var.Call, Var.CHAIN, Var.DRAW, Var.DUMP, Var.DelWatch, Var.FixedCHAIN, Var.Fixed-TABle, Var.Go, Var.Go.Back, Var.Go.Change, Var.Go.Till, Var.IF, Var.INFO, Var.Local, Var.LOG, Var.NEW, Var.NEW-GLOBAL, Var.NEWLOCAL, Var.PRINT, Var.Ref, Var.Step.BackChange, Var.Step.BackTill, Var.Step.Change, Var.Step.Till, Var.TABle, Var.TYPE, Var.View, Var.Watch, Var.WHILE, Var.WRITE	**HLL expression parser extended** Pointers and address values can be casted to certain segments or memory access classes.
Build 26464 IMP 02228 Data.DRAW (Graphical memory display of arrays) Var.DRAW (Graphical variable display)	**Graphical data display enhanced** Can now show up to six channels in one window. New option "Element" can select certain element of structure in an array of structures.
Build 4728 IMP 01967 Data.DRAW (Graphical memory display of arrays) Var.DRAW (Graphical variable display)	**Data.DRAW and Var.DRAW extended** Can now display up to six channels.
Build 1224 IMP 01752 Var.DRAW (Graphical variable display)	**Data.DRAW and Var.DRAW with new options** Option /Alternate can be used to interpret one-dimensional array as two channels. Options /XY and /YX display X-Y graphics.
Build 1079 FIX 01388 Var.DRAW (Graphical variable display)	Var.DRAW without parameters caused software to crash

Build 38589 IMP 02302 Var.set, Var.Assign, Var.AddWatch, Var.Break.direct, Var.Break.Delete, Var.Break.Pass, Var.Break.Set, Var.Call, Var.CHAIN, Var.DRAW, Var.DUMP, Var.DelWatch, Var.FixedCHAIN, Var.FixedTABle, Var.Go, Var.Go.Back, Var.Go.Change, Var.Go.Till, Var.IF, Var.INFO, Var.Local, Var.LOG, Var.NEW, Var.NEWGLOBAL, Var.NEWLOCAL, Var.PRINT, Var.Ref, Var.Step.BackChange, Var.Step.BackTill, Var.Step.Change, Var.Step.Till, Var.TABle, Var.TYPE, Var.View, Var.Watch, Var.WHILE, Var.WRITE	**HLL expression parser extended** Pointers and address values can be casted to certain segments or memory access classes.

Var.DelWatch

Build 38589 IMP 02302 Var.set, Var.Assign, Var.AddWatch, Var.Break.direct, Var.Break.Delete, Var.Break.Pass, Var.Break.Set, Var.Call, Var.CHAIN, Var.DRAW, Var.DUMP, Var.DelWatch, Var.FixedCHAIN, Var.FixedTABle, Var.Go, Var.Go.Back, Var.Go.Change, Var.Go.Till, Var.IF, Var.INFO, Var.Local, Var.LOG, Var.NEW, Var.NEWGLOBAL, Var.NEWLOCAL, Var.PRINT, Var.Ref, Var.Step.BackChange, Var.Step.BackTill, Var.Step.Change, Var.Step.Till, Var.TABle, Var.TYPE, Var.View, Var.Watch, Var.WHILE, Var.WRITE	**HLL expression parser extended** Pointers and address values can be casted to certain segments or memory access classes.

Var.FixedCHAIN

Build 38589 IMP 02302 Var.set, Var.Assign, Var.AddWatch, Var.Break.direct, Var.Break.Delete, Var.Break.Pass, Var.Break.Set, Var.Call, Var.CHAIN, Var.DRAW, Var.DUMP, Var.DelWatch, Var.FixedCHAIN, Var.FixedTABle, Var.Go, Var.Go.Back, Var.Go.Change, Var.Go.Till, Var.IF, Var.INFO, Var.Local, Var.LOG, Var.NEW, Var.NEWGLOBAL, Var.NEWLOCAL, Var.PRINT, Var.Ref, Var.Step.BackChange, Var.Step.BackTill, Var.Step.Change, Var.Step.Till, Var.TABle, Var.TYPE, Var.View, Var.Watch, Var.WHILE, Var.WRITE	**HLL expression parser extended** Pointers and address values can be casted to certain segments or memory access classes.

Var.FixedTABle

Build 38589 IMP 02302 Var.set, Var.Assign, Var.AddWatch, Var.Break.direct, Var.Break.Delete, Var.Break.Pass, Var.Break.Set, Var.Call, Var.CHAIN, Var.DRAW, Var.DUMP, Var.DelWatch, Var.FixedCHAIN, Var.FixedTABle, Var.Go, Var.Go.Back, Var.Go.Change, Var.Go.Till, Var.IF, Var.INFO, Var.Local, Var.LOG, Var.NEW, Var.NEWGLOBAL, Var.NEWLOCAL, Var.PRINT, Var.Ref, Var.Step.BackChange, Var.Step.BackTill, Var.Step.Change, Var.Step.Till, Var.TABle, Var.TYPE, Var.View, Var.Watch, Var.WHILE, Var.WRITE	**HLL expression parser extended** Pointers and address values can be casted to certain segments or memory access classes.

Var.Go

Build 38589 IMP 02302 Var.set, Var.Assign, Var.AddWatch, Var.Break.direct, Var.Break.Delete, Var.Break.Pass, Var.Break.Set, Var.Call, Var.CHAIN, Var.DRAW, Var.DUMP, Var.DelWatch, Var.FixedCHAIN, Var.FixedTABle, Var.Go, Var.Go.Back, Var.Go.Change, Var.Go.Till, Var.IF, Var.INFO, Var.Local, Var.LOG, Var.NEW, Var.NEWGLOBAL, Var.NEWLOCAL, Var.PRINT, Var.Ref, Var.Step.BackChange, Var.Step.BackTill, Var.Step.Change, Var.Step.Till, Var.TABle, Var.TYPE, Var.View, Var.Watch, Var.WHILE, Var.WRITE	**HLL expression parser extended** Pointers and address values can be casted to certain segments or memory access classes.

Var.Go.Back

Build 38589 IMP 02302 Var.set, Var.Assign, Var.AddWatch, Var.Break.direct, Var.Break.Delete, Var.Break.Pass, Var.Break.Set, Var.Call, Var.CHAIN, Var.DRAW, Var.DUMP, Var.DelWatch, Var.FixedCHAIN, Var.FixedTABle, Var.Go, Var.Go.Back, Var.Go.Change, Var.Go.Till, Var.IF, Var.INFO, Var.Local, Var.LOG, Var.NEW, Var.NEWGLOBAL, Var.NEWLOCAL, Var.PRINT, Var.Ref, Var.Step.BackChange, Var.Step.BackTill, Var.Step.Change, Var.Step.Till, Var.TABle, Var.TYPE, Var.View, Var.Watch, Var.WHILE, Var.WRITE	**HLL expression parser extended** Pointers and address values can be casted to certain segments or memory access classes.

Var.Go.Change

Build 38589 IMP 02302 Var.set, Var.Assign, Var.AddWatch, Var.Break.direct, Var.Break.Delete, Var.Break.Pass, Var.Break.Set, Var.Call, Var.CHAIN, Var.DRAW, Var.DUMP, Var.DelWatch, Var.FixedCHAIN, Var.FixedTABle, Var.Go, Var.Go.Back, Var.Go.Change, Var.Go.Till, Var.IF, Var.INFO, Var.Local, Var.LOG, Var.NEW, Var.NEWGLOBAL, Var.NEWLOCAL, Var.PRINT, Var.Ref, Var.Step.BackChange, Var.Step.BackTill, Var.Step.Change, Var.Step.Till, Var.TABle, Var.TYPE, Var.View, Var.Watch, Var.WHILE, Var.WRITE	**HLL expression parser extended** Pointers and address values can be casted to certain segments or memory access classes.

Var.Go.Till

Build 38589 IMP 02302 Var.set, Var.Assign, Var.AddWatch, Var.Break.direct, Var.Break.Delete, Var.Break.Pass, Var.Break.Set, Var.Call, Var.CHAIN, Var.DRAW, Var.DUMP, Var.DelWatch, Var.FixedCHAIN, Var.FixedTABle, Var.Go, Var.Go.Back, Var.Go.Change, Var.Go.Till, Var.IF, Var.INFO, Var.Local, Var.LOG, Var.NEW, Var.NEWGLOBAL, Var.NEWLOCAL, Var.PRINT, Var.Ref, Var.Step.BackChange, Var.Step.BackTill, Var.Step.Change, Var.Step.Till, Var.TABle, Var.TYPE, Var.View, Var.Watch, Var.WHILE, Var.WRITE	**HLL expression parser extended** Pointers and address values can be casted to certain segments or memory access classes.

Build 38589 IMP 02302 Var.set, Var.Assign, Var.AddWatch, Var.Break.direct, Var.Break.Delete, Var.Break.Pass, Var.Break.Set, Var.Call, Var.CHAIN, Var.DRAW, Var.DUMP, Var.DelWatch, Var.FixedCHAIN, Var.FixedTABle, Var.Go, Var.Go.Back, Var.Go.Change, Var.Go.Till, Var.IF, Var.INFO, Var.Local, Var.LOG, Var.NEW, Var.NEWGLOBAL, Var.NEWLOCAL, Var.PRINT, Var.Ref, Var.Step.BackChange, Var.Step.BackTill, Var.Step.Change, Var.Step.Till, Var.TABle, Var.TYPE, Var.View, Var.Watch, Var.WHILE, Var.WRITE	**HLL expression parser extended** Pointers and address values can be casted to certain segments or memory access classes.

Build 38589 IMP 02302 Var.set, Var.Assign, Var.AddWatch, Var.Break.direct, Var.Break.Delete, Var.Break.Pass, Var.Break.Set, Var.Call, Var.CHAIN, Var.DRAW, Var.DUMP, Var.DelWatch, Var.FixedCHAIN, Var.FixedTABle, Var.Go, Var.Go.Back, Var.Go.Change, Var.Go.Till, Var.IF, Var.INFO, Var.Local, Var.LOG, Var.NEW, Var.NEWGLOBAL, Var.NEWLOCAL, Var.PRINT, Var.Ref, Var.Step.BackChange, Var.Step.BackTill, Var.Step.Change, Var.Step.Till, Var.TABle, Var.TYPE, Var.View, Var.Watch, Var.WHILE, Var.WRITE	**HLL expression parser extended** Pointers and address values can be casted to certain segments or memory access classes.
Build 1077 FIX 01068 Var.INFO (View information about HLL variable or HLL expression)	window creation size of Var.INFO may be wrong

Build 38589 IMP 02302 Var.set, Var.Assign, Var.AddWatch, Var.Break.direct, Var.Break.Delete, Var.Break.Pass, Var.Break.Set, Var.Call, Var.CHAIN, Var.DRAW, Var.DUMP, Var.DelWatch, Var.FixedCHAIN, Var.FixedTABle, Var.Go, Var.Go.Back, Var.Go.Change, Var.Go.Till, Var.IF, Var.INFO, Var.Local, Var.LOG, Var.NEW, Var.NEWGLOBAL, Var.NEWLOCAL, Var.PRINT, Var.Ref, Var.Step.BackChange, Var.Step.BackTill, Var.Step.Change, Var.Step.Till, Var.TABle, Var.TYPE, Var.View, Var.Watch, Var.WHILE, Var.WRITE	**HLL expression parser extended** Pointers and address values can be casted to certain segments or memory access classes.
Build 1077 IMP 01065 Var.LOG (Log variables)	**Extended** Var.LOG max line length extended to 4K

Var.Local

Build 38589 IMP 02302 Var.set, Var.Assign, Var.AddWatch, Var.Break.direct, Var.Break.Delete, Var.Break.Pass, Var.Break.Set, Var.Call, Var.CHAIN, Var.DRAW, Var.DUMP, Var.DelWatch, Var.FixedCHAIN, Var.FixedTABle, Var.Go, Var.Go.Back, Var.Go.Change, Var.Go.Till, Var.IF, Var.INFO, Var.Local, Var.LOG, Var.NEW, Var.NEWGLOBAL, Var.NEWLOCAL, Var.PRINT, Var.Ref, Var.Step.BackChange, Var.Step.BackTill, Var.Step.Change, Var.Step.Till, Var.TABle, Var.TYPE, Var.View, Var.Watch, Var.WHILE, Var.WRITE	**HLL expression parser extended** Pointers and address values can be casted to certain segments or memory access classes.

Var.NEW

Build 38589 IMP 02302 Var.set, Var.Assign, Var.AddWatch, Var.Break.direct, Var.Break.Delete, Var.Break.Pass, Var.Break.Set, Var.Call, Var.CHAIN, Var.DRAW, Var.DUMP, Var.DelWatch, Var.FixedCHAIN, Var.FixedTABle, Var.Go, Var.Go.Back, Var.Go.Change, Var.Go.Till, Var.IF, Var.INFO, Var.Local, Var.LOG, Var.NEW, Var.NEWGLOBAL, Var.NEWLOCAL, Var.PRINT, Var.Ref, Var.Step.BackChange, Var.Step.BackTill, Var.Step.Change, Var.Step.Till, Var.TABle, Var.TYPE, Var.View, Var.Watch, Var.WHILE, Var.WRITE	**HLL expression parser extended** Pointers and address values can be casted to certain segments or memory access classes.

Var.NEWGLOBAL

Build 38589 IMP 02302 Var.set, Var.Assign, Var.AddWatch, Var.Break.direct, Var.Break.Delete, Var.Break.Pass, Var.Break.Set, Var.Call, Var.CHAIN, Var.DRAW, Var.DUMP, Var.DelWatch, Var.FixedCHAIN, Var.FixedTABle, Var.Go, Var.Go.Back, Var.Go.Change, Var.Go.Till, Var.IF, Var.INFO, Var.Local, Var.LOG, Var.NEW, Var.NEWGLOBAL, Var.NEWLOCAL, Var.PRINT, Var.Ref, Var.Step.BackChange, Var.Step.BackTill, Var.Step.Change, Var.Step.Till, Var.TABle, Var.TYPE, Var.View, Var.Watch, Var.WHILE, Var.WRITE	**HLL expression parser extended** Pointers and address values can be casted to certain segments or memory access classes.

Var.NEWLOCAL

Build 38589 IMP 02302 Var.set, Var.Assign, Var.AddWatch, Var.Break.direct, Var.Break.Delete, Var.Break.Pass, Var.Break.Set, Var.Call, Var.CHAIN, Var.DRAW, Var.DUMP, Var.DelWatch, Var.FixedCHAIN, Var.FixedTABle, Var.Go, Var.Go.Back, Var.Go.Change, Var.Go.Till, Var.IF, Var.INFO, Var.Local, Var.LOG, Var.NEW, Var.NEWGLOBAL, Var.NEWLOCAL, Var.PRINT, Var.Ref, Var.Step.BackChange, Var.Step.BackTill, Var.Step.Change, Var.Step.Till, Var.TABle, Var.TYPE, Var.View, Var.Watch, Var.WHILE, Var.WRITE	**HLL expression parser extended** Pointers and address values can be casted to certain segments or memory access classes.

Build 4728 IMP 01971 Var.PATtern (Display variables allowing wildcards for symbol name and type)	**New command to show variables by name**

Var.PRINT

Build 38589 IMP 02302 Var.set, Var.Assign, Var.AddWatch, Var.Break.direct, Var.Break.Delete, Var.Break.Pass, Var.Break.Set, Var.Call, Var.CHAIN, Var.DRAW, Var.DUMP, Var.DelWatch, Var.FixedCHAIN, Var.FixedTABle, Var.Go, Var.Go.Back, Var.Go.Change, Var.Go.Till, Var.IF, Var.INFO, Var.Local, Var.LOG, Var.NEW, Var.NEWGLOBAL, Var.NEWLOCAL, Var.PRINT, Var.Ref, Var.Step.BackChange, Var.Step.BackTill, Var.Step.Change, Var.Step.Till, Var.TABle, Var.TYPE, Var.View, Var.Watch, Var.WHILE, Var.WRITE	**HLL expression parser extended** Pointers and address values can be casted to certain segments or memory access classes.

Build 69655 IMP 02442 High-Level-Language Debugging Trace Features <trace>.DRAW.channel (Plot no-data values against time) <trace>.DRAW.Data (Plot data values against time) <trace>.DRAW.Var (Plot variable values against time) Data.DRAW (Graphical memory display of arrays) Data.PROfile (Graphical display of data value) Var.PROfile (Graphical display of variable)	**Multiple variables of different type can be drawn** TRACE32 allows to draw multiple variables/channels with different formats e.g. integers and floats for all listed commands.
Build 50104 IMP 02376 Data.PROfile (Graphical display of data value) Var.PROfile (Graphical display of variable)	**New command Var.PROfile** HLL counterpart to Data.PROfile to show variables over time.

Var.Ref

Build 38589 IMP 02302 Var.set, Var.Assign, Var.AddWatch, Var.Break.direct, Var.Break.Delete, Var.Break.Pass, Var.Break.Set, Var.Call, Var.CHAIN, Var.DRAW, Var.DUMP, Var.DelWatch, Var.FixedCHAIN, Var.FixedTABle, Var.Go, Var.Go.Back, Var.Go.Change, Var.Go.Till, Var.IF, Var.INFO, Var.Local, Var.LOG, Var.NEW, Var.NEWGLOBAL, Var.NEWLOCAL, Var.PRINT, Var.Ref, Var.Step.BackChange, Var.Step.BackTill, Var.Step.Change, Var.Step.Till, Var.TABle, Var.TYPE, Var.View, Var.Watch, Var.WHILE, Var.WRITE	**HLL expression parser extended** Pointers and address values can be casted to certain segments or memory access classes.

Var.Step.BackChange

Build 38589 IMP 02302 Var.set, Var.Assign, Var.AddWatch, Var.Break.direct, Var.Break.Delete, Var.Break.Pass, Var.Break.Set, Var.Call, Var.CHAIN, Var.DRAW, Var.DUMP, Var.DelWatch, Var.FixedCHAIN, Var.FixedTABle, Var.Go, Var.Go.Back, Var.Go.Change, Var.Go.Till, Var.IF, Var.INFO, Var.Local, Var.LOG, Var.NEW, Var.NEWGLOBAL, Var.NEWLOCAL, Var.PRINT, Var.Ref, Var.Step.BackChange, Var.Step.BackTill, Var.Step.Change, Var.Step.Till, Var.TABle, Var.TYPE, Var.View, Var.Watch, Var.WHILE, Var.WRITE	**HLL expression parser extended** Pointers and address values can be casted to certain segments or memory access classes.

Var.Step.BackTill

Build 38589 IMP 02302 Var.set, Var.Assign, Var.AddWatch, Var.Break.direct, Var.Break.Delete, Var.Break.Pass, Var.Break.Set, Var.Call, Var.CHAIN, Var.DRAW, Var.DUMP, Var.DelWatch, Var.FixedCHAIN, Var.FixedTABle, Var.Go, Var.Go.Back, Var.Go.Change, Var.Go.Till, Var.IF, Var.INFO, Var.Local, Var.LOG, Var.NEW, Var.NEWGLOBAL, Var.NEWLOCAL, Var.PRINT, Var.Ref, Var.Step.BackChange, Var.Step.BackTill, Var.Step.Change, Var.Step.Till, Var.TABle, Var.TYPE, Var.View, Var.Watch, Var.WHILE, Var.WRITE	**HLL expression parser extended** Pointers and address values can be casted to certain segments or memory access classes.

Var.Step.Change

Build 38589 IMP 02302 Var.set, Var.Assign, Var.AddWatch, Var.Break.direct, Var.Break.Delete, Var.Break.Pass, Var.Break.Set, Var.Call, Var.CHAIN, Var.DRAW, Var.DUMP, Var.DelWatch, Var.FixedCHAIN, Var.FixedTABle, Var.Go, Var.Go.Back, Var.Go.Change, Var.Go.Till, Var.IF, Var.INFO, Var.Local, Var.LOG, Var.NEW, Var.NEWGLOBAL, Var.NEWLOCAL, Var.PRINT, Var.Ref, Var.Step.BackChange, Var.Step.BackTill, Var.Step.Change, Var.Step.Till, Var.TABle, Var.TYPE, Var.View, Var.Watch, Var.WHILE, Var.WRITE	**HLL expression parser extended** Pointers and address values can be casted to certain segments or memory access classes.

Var.Step.Till

Build 38589 IMP 02302 Var.set, Var.Assign, Var.AddWatch, Var.Break.direct, Var.Break.Delete, Var.Break.Pass, Var.Break.Set, Var.Call, Var.CHAIN, Var.DRAW, Var.DUMP, Var.DelWatch, Var.FixedCHAIN, Var.FixedTABle, Var.Go, Var.Go.Back, Var.Go.Change, Var.Go.Till, Var.IF, Var.INFO, Var.Local, Var.LOG, Var.NEW, Var.NEWGLOBAL, Var.NEWLOCAL, Var.PRINT, Var.Ref, Var.Step.BackChange, Var.Step.BackTill, Var.Step.Change, Var.Step.Till, Var.TABle, Var.TYPE, Var.View, Var.Watch, Var.WHILE, Var.WRITE	**HLL expression parser extended** Pointers and address values can be casted to certain segments or memory access classes.

Var.TABle

Build 38589 IMP 02302 Var.set, Var.Assign, Var.AddWatch, Var.Break.direct, Var.Break.Delete, Var.Break.Pass, Var.Break.Set, Var.Call, Var.CHAIN, Var.DRAW, Var.DUMP, Var.DelWatch, Var.FixedCHAIN, Var.FixedTABle, Var.Go, Var.Go.Back, Var.Go.Change, Var.Go.Till, Var.IF, Var.INFO, Var.Local, Var.LOG, Var.NEW, Var.NEWGLOBAL, Var.NEWLOCAL, Var.PRINT, Var.Ref, Var.Step.BackChange, Var.Step.BackTill, Var.Step.Change, Var.Step.Till, Var.TABle, Var.TYPE, Var.View, Var.Watch, Var.WHILE, Var.WRITE	**HLL expression parser extended** Pointers and address values can be casted to certain segments or memory access classes.

Var.TREE

Build 4728 IMP 01972 Var.TREE (Display variables in the form of a tree structure)	**New command to show variables in tree**

Var.TYPE

Build 38589 IMP 02302 Var.set, Var.Assign, Var.AddWatch, Var.Break.direct, Var.Break.Delete, Var.Break.Pass, Var.Break.Set, Var.Call, Var.CHAIN, Var.DRAW, Var.DUMP, Var.DelWatch, Var.FixedCHAIN, Var.FixedTABle, Var.Go, Var.Go.Back, Var.Go.Change, Var.Go.Till, Var.IF, Var.INFO, Var.Local, Var.LOG, Var.NEW, Var.NEWGLOBAL, Var.NEWLOCAL, Var.PRINT, Var.Ref, Var.Step.BackChange, Var.Step.BackTill, Var.Step.Change, Var.Step.Till, Var.TABle, Var.TYPE, Var.View, Var.Watch, Var.WHILE, Var.WRITE	**HLL expression parser extended** Pointers and address values can be casted to certain segments or memory access classes.

Build 93173 IMP 02512 Integrated Development Environment Frame.view (Display stack frame) List.auto (Display program listing) Var.View (Display variables) Var.Watch (Open Var.Watch window)	**Mouse interface changed for all text windows** Free text selection is now possible with the left mouse. Click and drag on a selection allows drag and drop.
Build 88288 IMP 02490 High-Level-Language Debugging Var.AddWatch (Add variable to Var.Watch window) Var.View (Display variables) Var.Watch (Open Var.Watch window)	**Extend arithmetic in Hll expressions to more the 64 bits** Integer arithmetic in HLL expressions is extended to support more than 64-bits.
Build 38589 IMP 02302 Var.set, Var.Assign, Var.AddWatch, Var.Break.direct, Var.Break.Delete, Var.Break.Pass, Var.Break.Set, Var.Call, Var.CHAIN, Var.DRAW, Var.DUMP, Var.DelWatch, Var.FixedCHAIN, Var.FixedTABle, Var.Go, Var.Go.Back, Var.Go.Change, Var.Go.Till, Var.IF, Var.INFO, Var.Local, Var.LOG, Var.NEW, Var.NEWGLOBAL, Var.NEWLOCAL, Var.PRINT, Var.Ref, Var.Step.BackChange, Var.Step.BackTill, Var.Step.Change, Var.Step.Till, Var.TABle, Var.TYPE, Var.View, Var.Watch, Var.WHILE, Var.WRITE	**HLL expression parser extended** Pointers and address values can be casted to certain segments or memory access classes.
Build 4728 FIX 01932 High-Level-Language Debugging Var.View (Display variables)	**Internal functions in HLL parser fail** When used in windows (e.g. Var.View) the functions will cause strange effect on the command line.

Build 1258 IMP 01853 SETUP.ASCIITEXT (Configure ASCII text display) Data.dump (Memory dump) Var.View (Display variables)	**UTF-8 support for ASCII strings/dumps** UTF-8 characters is enabled with SETUP.ASCIITEXT UTF-8.

Var.WHILE

Build 38589 IMP 02302 Var.set, Var.Assign, Var.AddWatch, Var.Break.direct, Var.Break.Delete, Var.Break.Pass, Var.Break.Set, Var.Call, Var.CHAIN, Var.DRAW, Var.DUMP, Var.DelWatch, Var.FixedCHAIN, Var.FixedTABle, Var.Go, Var.Go.Back, Var.Go.Change, Var.Go.Till, Var.IF, Var.INFO, Var.Local, Var.LOG, Var.NEW, Var.NEWGLOBAL, Var.NEWLOCAL, Var.PRINT, Var.Ref, Var.Step.BackChange, Var.Step.BackTill, Var.Step.Change, Var.Step.Till, Var.TABle, Var.TYPE, Var.View, Var.Watch, Var.WHILE, Var.WRITE	**HLL expression parser extended** Pointers and address values can be casted to certain segments or memory access classes.

Build 38589 IMP 02302 Var.set, Var.Assign, Var.AddWatch, Var.Break.direct, Var.Break.Delete, Var.Break.Pass, Var.Break.Set, Var.Call, Var.CHAIN, Var.DRAW, Var.DUMP, Var.DelWatch, Var.FixedCHAIN, Var.FixedTABle, Var.Go, Var.Go.Back, Var.Go.Change, Var.Go.Till, Var.IF, Var.INFO, Var.Local, Var.LOG, Var.NEW, Var.NEWGLOBAL, Var.NEWLOCAL, Var.PRINT, Var.Ref, Var.Step.BackChange, Var.Step.BackTill, Var.Step.Change, Var.Step.Till, Var.TABle, Var.TYPE, Var.View, Var.Watch, Var.WHILE, Var.WRITE	**HLL expression parser extended** Pointers and address values can be casted to certain segments or memory access classes.

Build 93173 IMP 02512 Integrated Development Environment Frame.view (Display stack frame) List.auto (Display program listing) Var.View (Display variables) Var.Watch (Open Var.Watch window)	**Mouse interface changed for all text windows** Free text selection is now possible with the left mouse. Click and drag on a selection allows drag and drop.
Build 88288 IMP 02491 High-Level-Language Debugging Var.AddWatch (Add variable to Var.Watch window) Var.Watch (Open Var.Watch window)	**Var.Watch keeps formatting if opened without variable** Var.Watch window applies formatting options to all added variables, if opened with format parameters but without variable names.
Build 38589 IMP 02302 Var.set, Var.Assign, Var.AddWatch, Var.Break.direct, Var.Break.Delete, Var.Break.Pass, Var.Break.Set, Var.Call, Var.CHAIN, Var.DRAW, Var.DUMP, Var.DelWatch, Var.FixedCHAIN, Var.FixedTABle, Var.Go, Var.Go.Back, Var.Go.Change, Var.Go.Till, Var.IF, Var.INFO, Var.Local, Var.LOG, Var.NEW, Var.NEWGLOBAL, Var.NEWLOCAL, Var.PRINT, Var.Ref, Var.Step.BackChange, Var.Step.BackTill, Var.Step.Change, Var.Step.Till, Var.TABle, Var.TYPE, Var.View, Var.Watch, Var.WHILE, Var.WRITE	**HLL expression parser extended** Pointers and address values can be casted to certain segments or memory access classes.

Build 60219 IMP 02424 High-Level-Language Debugging Var.set (Modify variable)	**C/C++ parser supports scaled assignments** C/C++ expression parser extended to support scaled assignment (e.g. for ASAP file format).
Build 38589 FIX 02313 MPC5xx/8xx Debugger Var.set (Modify variable)	**MPC500-Nexus: dualport access** fix dualport access for Var.set command
Build 38589 IMP 02302 Var.set, Var.Assign, Var.AddWatch, Var.Break.direct, Var.Break.Delete, Var.Break.Pass, Var.Break.Set, Var.Call, Var.CHAIN, Var.DRAW, Var.DUMP, Var.DelWatch, Var.FixedCHAIN, Var.FixedTABle, Var.Go, Var.Go.Back, Var.Go.Change, Var.Go.Till, Var.IF, Var.INFO, Var.Local, Var.LOG, Var.NEW, Var.NEWGLOBAL, Var.NEWLOCAL, Var.PRINT, Var.Ref, Var.Step.BackChange, Var.Step.BackTill, Var.Step.Change, Var.Step.Till, Var.TABle, Var.TYPE, Var.View, Var.Watch, Var.WHILE, Var.WRITE	**HLL expression parser extended** Pointers and address values can be casted to certain segments or memory access classes.
Build 8248 IMP 02017 Var.set (Modify variable)	**Var commands possible without file load** HLL expressions can now also be handled when no HLL file is loaded.
Build 1077 FIX 01074 Var.set (Modify variable)	**Clicking on methods** Clicking on methods in Var windows fails

WELCOME.SCRIPTS

Build 56057 IMP 02411 Uniform Look-And-Feel for all Processors WELCOME.view (Open the welcome window) WELCOME.SCRIPTS (Open the script search window) WELCOME.STARTUP (Open the welcome window if not disabled)	**Welcome dialog and script search** Welcome dialog for new installations and search dialog for scripts (debugger software only)

WELCOME.STARTUP

Build 56057 IMP 02411 Uniform Look-And-Feel for all Processors WELCOME.view (Open the welcome window) WELCOME.SCRIPTS (Open the script search window) WELCOME.STARTUP (Open the welcome window if not disabled)	**Welcome dialog and script search** Welcome dialog for new installations and search dialog for scripts (debugger software only)

WELCOME.view

Build 56057 IMP 02411 Uniform Look-And-Feel for all Processors WELCOME.view (Open the welcome window) WELCOME.SCRIPTS (Open the script search window) WELCOME.STARTUP (Open the welcome window if not disabled)	**Welcome dialog and script search** Welcome dialog for new installations and search dialog for scripts (debugger software only)

WHILE

Build 50104 CHG 02366 Script Language PRACTICE IF (Conditional script execution) RePeaT (Loop with check at end of loop) WHILE (Loop with check at start of loop)	**Empty lines in PRACTICE allowed** Empty lines are now ignored in PRACTICE. This may change the execution flow of existing programs (e.g. when an IF, REPEAT or WHILE statement is followed by an empty line).

WRITE

Build 65657 ODD 02429 Script Language PRACTICE WRITE (Write to data file)	**WRITE %CONTinue behavior changed** The behavior of "WRITE %CONTinue" was changed to harmonize the manner of commands PRINT and WRITE. old: %CONT surpresses the writing of CR/LF and effects the next writing new: %CONT removes the CR/LF from the last writing and concatenates the new data
Build 1077 IMP 01184 WRITE (Write to data file)	WRITE #x %CONT will write to file without line feed

WRITEB

Build 30461 IMP 02246 OPEN (Open data file) WRITEB (Write binary data to file)	**PRACTICE can now handle binary files** New option /Binary for "OPEN" command allows to open or create binary files. New command "WRITEB" allows to write binary data to a file.

WinDEFaultSIZE

Build 38589 IMP 02307 WinDEFaultSIZE (Apply a user-defined default size to windows)	**Default window creating size** The command WinDEFaultSIZE can be used to define the default size for new windows.

WinPOS

Build 100486 IMP 02526 Integrated Development Environment CmdPOS (Controls the position of TRACE32 in MWI window mode) FramePOS (Controls the position of TRACE32 in MDI window mode) WinPOS (Define window dimensions and window name)	**Improved support for multimonitor desktops under MS Windows** The maximum possible window creation size was enlarged to cover the full multimonitor desktop size under MS Windows.

WinPrint

Build 130863 IMP 02611 Integrated Development Environment PRinTer.FILE (Re-route printer output to a file in specified file format) WinPrint (Print address or record range of a window)	**new PRinTer file format PCL** The PCL file format has been added to the PRinTer.FILE command.
Build 38589 CHG 02306 WinPrint (Print address or record range of a window)	**Window pulldown item "Print All"** Now "Print All" or "To Clipboard All" will redirect the whole window content and not only the 1st page for the most window types with finite size.

sYmbol

Build 19417 IMP 02182 Data.LOAD (Load file) sYmbol (Debug symbols)	**Support for code overlays** Supports different concepts of code overlays for multiple architecures.
Build 1133 FIX 01475 sYmbol (Debug symbols)	Clicking on module that has same name as valid variable in symbol browser may select wrong symbol

sYmbol.AddInfo.LOADASAP2

Build 1196 IMP 01703 Data.LOAD.ASAP2 (Load ASAP2 file) sYmbol.AddInfo.LOADASAP2 (Load scaling information from ASAP2 file)	**New command Data.LOAD.ASAP2** Loads information from ASAP2 database file and creates symbols and type information to display the values WITHOUT the need for a debug symbol file.

sYmbol.Browse

Build 2940 IMP 01906 sYmbol.Browse (Browse symbols)	**Symbol browser uses GROUP hide info** Symbols belonging to hidden groups are not displayed.
Build 1096 IMP 01431 sYmbol.Browse (Browse symbols)	sYmbol.Browse window did not show all symbols when symbol number increases while window is open

sYmbol.Browse.Type

Build 4728 IMP 01939 sYmbol.Browse.Type (Browse HLL types)	**New commands for type/class browsing** sYmbol.Browse.Type browses for any types. sYmbol.Browse.ENUM browses for enumeration types. sYmbol.Browse.Struct browses for structs, unions and classes. sYmbol.Browse.Class browses for C++ classes.

sYmbol.CLEANUP

Build 1145 FIX 01574 sYmbol.CLEANUP (Workarounds for redundant symbol information)	y.cleanup removed also "global" module symbols

sYmbol.CREATE

Build 1079 FIX 01393 sYmbol.CREATE (Create and modify user-defined symbols) sYmbol.NEW (Create new symbol)	Accessing user defined functions (Y.CREATE) with Var commands may produce crash

sYmbol.CREATE.ATTRibute

Build 1241 IMP 01807 sYmbol.CREATE.ATTRibute (Create user-defined attribute) sYmbol.NEW.ATTRibute (Create user-defined memory attribute)	**new command sYmbol.NEW.ATTRibute** Creates new attribute entries. Useful when ARM/Thumb awareness is needed and no debug information for it is present.

sYmbol.CREATE.MACRO

Build 19417 FIX 02125 sYmbol.CREATE.MACRO (Create user-defined macro) sYmbol.NEW.MACRO (Create user-defined macro)	**Macro creation not working** The macro name and contents are mixed up.

sYmbol.Delete

Build 1077 FIX 01117 sYmbol.Delete (Delete symbols of one program)	**extended addressing** y.delete may cause crash when removing programs in extended addressing (16:32 format)

sYmbol.DeleteMACRO

Build 130863 IMP 02604 sYmbol.DeleteMACRO (Delete macro information)	**new command sYmbol.DeleteMACRO added**

sYmbol.ECA.LOAD

Build 130863 IMP 02605 Trace-based Code Coverage sYmbol.ECA.LOAD (Load a single ECA file) sYmbol.ECA.LOADALL (Load all ECA files)	**new options /SkipErrors and /LENient for sYmbol.ECA.LOAD** The /SkipErrors option ensures that warnings are issued instead of error messages. For scripts, error messages cause the script to stop. Warnings keep the script running. The /LENient option allows loading of ECA files with minor errors as invalid file version or checksum mismatch.

sYmbol.ECA.LOADALL

Build 130863 IMP 02605 Trace-based Code Coverage sYmbol.ECA.LOAD (Load a single ECA file) sYmbol.ECA.LOADALL (Load all ECA files)	**new options /SkipErrors and /LENient for sYmbol.ECA.LOAD** The /SkipErrors option ensures that warnings are issued instead of error messages. For scripts, error messages cause the script to stop. Warnings keep the script running. The /LENient option allows loading of ECA files with minor errors as invalid file version or checksum mismatch.

sYmbol.EXIT(<symbol>)

Build 1085 FIX 01421 sYmbol.EXIT(<symbol>)	Y.EXIT(<func>) function may fail with error message
Build 1079 IMP 01394 sYmbol.EXIT(<symbol>)	New function Y.EXIT(<function_name>) to get the last statement of a function

sYmbol.List.SOURCE

Build 65657 02434 High-Level-Language Debugging sYmbol.List.SOURCE (Display source file names) sYmbol.SourcePATH.List (List source search paths) sYmbol.SourcePATH.SetBaseDir (Define directory as base for relative paths) sYmbol.SourcePATH.Translate (Replace part of the source path)	**Improved source path correction** Module/source that could not be loaded, because the source path needs to be adapted in the current debug environment are printed in red in the sYmbol.List.SOURCE window. A Resolve Path command was added to the Source context menu, which fixes the paths in the current debug environment after one correct path was provided.
Build 42354 IMP 02340 sYmbol.List.SOURCE (Display source file names)	**Debug info keeps full original source name** The matching algorithm for source names has been extended to match also for files in subdirectories.

sYmbol.MARKER

Build 34458 IMP 02273 sYmbol.MARKER (Fine-tune the nested function run-time analysis)	**new command group sYmbol.MARKER added** New command group sYmbol.MARKER to maintain markers. Markers can be used to handle special cases for nested trace statistics.

sYmbol.MARKER.Create

Build 34458 IMP 02279 sYmbol.MARKER.Create (Marker for nesting function run-time analysis)	**New MARKER capabilities for nested stats** New TASKSWITCH marker defines when a taskswitch should take place (delayed taskswitch). Multiple markers can now be set on a single address.

sYmbol.Modify

Build 19417 IMP 02154 sYmbol.Modify (Modify symbols)	**New sYmbol.MODIFY commands** Allow to convert labels into ranges and vice versa or ranges into functions.

sYmbol.Modify.CutFunction

Build 1077 IMP 01358 sYmbol.Modify.CutFunction (Reduce function address information)	New command Y.Modify.CutFunction to remove the address range from functions

sYmbol.Modify.TYPE

Build 4728 IMP 01966 sYmbol.Modify.TYPE (Modify type of symbols) sYmbol.NEW.Var (Create user-defined variable)	**New variable create and modify commands** Can create new (target) variables or modify the HLL type of an existing variable or label.

sYmbol.NAMESPACES

Build 19417 IMP 02175 sYmbol.NAMESPACES (Search symbol in C++ namespace)	**New command to define namespaces** The command sYmbol.NAMESPACES defines a list of namespace which are searched by the debugger (similar to "using" statement).

sYmbol.NEW

Build 1079 IMP 01395 sYmbol.NEW (Create new symbol)	Y.CREATE/Y.NEW can now also create address range labels
Build 1079 FIX 01393 sYmbol.CREATE (Create and modify user-defined symbols) sYmbol.NEW (Create new symbol)	Accessing user defined functions (Y.CREATE) with Var commands may produce crash

sYmbol.NEW.ATTRibute

Build 1241 IMP 01807 sYmbol.CREATE.ATTRibute (Create user-defined attribute) sYmbol.NEW.ATTRibute (Create user-defined memory attribute)	**new command sYmbol.NEW.ATTRibute** Creates new attribute entries. Useful when ARM/Thumb awareness is needed and no debug information for it is present.

sYmbol.NEW.MACRO

Build 19417 FIX 02125 sYmbol.CREATE.MACRO (Create user-defined macro) sYmbol.NEW.MACRO (Create user-defined macro)	**Macro creation not working** The macro name and contents are mixed up.

sYmbol.NEW.Var

Build 4728 IMP 01966 sYmbol.Modify.TYPE (Modify type of symbols) sYmbol.NEW.Var (Create user-defined variable)	**New variable create and modify commands** Can create new (target) variables or modify the HLL type of an existing variable or label.

sYmbol.RELOCate

Build 1133 FIX 01498 sYmbol.RELOCate (Relocate symbols)	Y.RELOC crashed when attribute or stack table was not empty

sYmbol.RELOCate.shift

Build 1190 FIX 01646 sYmbol.RELOCate.shift (Relocate symbols)	**sYmbol.RELOcate crashes with ext. addrs** sYmbol.RELOCate.shift crashes with extended addresses (16:32 format) when attribute information is present.

sYmbol.SourceCONVert

Build 1190 IMP 01682 sYmbol.SourceCONVert (Conversion for Japanese font)	**EUC-JP characterset support added** Strings in target memory will be displayed in EUC with the setting "SETUP.ASCIITEXT EUC-JP". Sourcetext is converted from EUC-JP by the setting "sYmbol.SourceCONVert EUC-JP".

sYmbol.SourcePATH.List

Build 65657 02434 High-Level-Language Debugging sYmbol.List.SOURCE (Display source file names) sYmbol.SourcePATH.List (List source search paths) sYmbol.SourcePATH.SetBaseDir (Define directory as base for relative paths) sYmbol.SourcePATH.Translate (Replace part of the source path)	**Improved source path correction** Module/source that could not be loaded, because the source path needs to be adapted in the current debug environment are printed in red in the sYmbol.List.SOURCE window. A Resolve Path command was added to the Source context menu, which fixes the paths in the current debug environment after one correct path was provided.

sYmbol.SourcePATH.SetBaseDir

Build 65657 02434 High-Level-Language Debugging sYmbol.List.SOURCE (Display source file names) sYmbol.SourcePATH.List (List source search paths) sYmbol.SourcePATH.SetBaseDir (Define directory as base for relative paths) sYmbol.SourcePATH.Translate (Replace part of the source path)	**Improved source path correction** Module/source that could not be loaded, because the source path needs to be adapted in the current debug environment are printed in red in the sYmbol.List.SOURCE window. A Resolve Path command was added to the Source context menu, which fixes the paths in the current debug environment after one correct path was provided.

sYmbol.SourcePATH.Translate

Build 65657 02434 High-Level-Language Debugging sYmbol.List.SOURCE (Display source file names) sYmbol.SourcePATH.List (List source search paths) sYmbol.SourcePATH.SetBaseDir (Define directory as base for relative paths) sYmbol.SourcePATH.Translate (Replace part of the source path)	**Improved source path correction** Module/source that could not be loaded, because the source path needs to be adapted in the current debug environment are printed in red in the sYmbol.List.SOURCE window. A Resolve Path command was added to the Source context menu, which fixes the paths in the current debug environment after one correct path was provided.
Build 1203 IMP 01718 sYmbol.SourcePATH.Translate (Replace part of the source path)	**New command sYmbol.SourcePATH.Translate** The command translates parts of filenames or directories from object files. This is useful when directory names change names or are mangled by MSDOS.

Sorted by Devices

8 Bit Emulation Compact Controller

Build 30461 FIX 02251 8 Bit Emulation Compact Controller	**commands TM.S X TM.S TO locked** For hardware configuration ICE and ECC8 only the commands {TrMain||Trigger}.Set {eXception||TimeOut} were locked since build 25948 from 06.10.2010.

ARC Debugger

Build 38589 IMP 02312 ARC Debugger	**Added Support for ARC-EM family**
Build 19417 IMP 02188 ARC Debugger <trace>.List (List trace contents)	**Support for ARC onchip trace (SmaRT)**
Build 19417 IMP 02184 ARC Debugger TERM.GATE (Terminal with virtual hosting)	**Support for Metaware Hostlink library**

Build 125398 IMP 02587 ARM Debugger CEVA-X Debugger Trace Analyzer ETM.TraceCORE (Core specific default tracing)	**new command ETM.TraceCORE** New command ETM.TraceCORE to allow core specific default tracing.
Build 88288 IMP 02498 ARM Debugger Cortex-A/-R Armv8 Debugger Intel~ x86/x64 JTAG Debugger SYStem.Option MACHINESPACES (Address extension for guest OSes) SYStem.Option MACHINESPACES (Address extension for guest OSes) SYStem.Option MACHINESPACES (Address extension for guest OSes)	**Hypervisor-aware debugging feature introduced (Part 1-2)** The TRACE32 machinespaces framework was added for architectures ARM and Intel x86 to support Hypervisor environments. This framework allows the debugger to perform stop-mode debugging of targets running a hypervisor and one or multiple guest machines. In conjunction with a hypervisor awareness it is possible to debug all guest machines concurrently, whether they are currently active or currently inactive. TRACE32 allows to load an individual OS awareness for each virtual machine on the target. The machinespaces framework is enabled with the command SYStem.Option.MACHINESPACES ON. A new identifier <machineid> is introduced which allows to identify virtual machines on the target. TRACE32 addresses are augmented with the machineid to specify which machine the address belongs to. The TRACE32 debugger address translation (enabled with TRANSlation.ON) is extended to fully support the two-stage address translation which is usually applied for hardware virtualized guest machines. The first translation stage translates guest virtual addresses to intermediate physical addresses. The second translation stage translates intermediate physical addresses to absolute physical addresses. The MMU registers needed for the page table walk steps are automatically fetched from the hardware registers or, for inactive guests, from hypervisor data structures. If SYStem.Option.MACHINESPACES is enabled, - addresses are extended with a machine ID identifier, separating the machine ID with three colons from the (extended) virtual address: <accessclass>:<machineid>:::<spaceid>::<virtaddress> The machine ID clearly specifies which host or guest machine the address belongs to. The host machine always uses machineid 0. Guests have a machineid larger than 0. TRACE32 currently supports up to 30 machines.

Build 80996 IMP 02475 Android Debugging ARM Debugger Cortex-A/-R Armv8 Debugger Data.LOAD.OAT (Load OAT file)	**Android Debugging** The TRACE32 Android debug support has been extended for Android versions based on the Android RunTime (ART). This includes Android versions L, M and N. The new Android support allows the debugging of the ahead-of-time compiled Android framework and apps. TRACE32 automatically detects ahead-of-time compiled objects and and loads the DWARF/ELF info if available. Otherwise the debugger can parse the OAT data to extract the debug info. Additionally, TRACE32 supports the hybrid compilation introduced in Android N. For interpreted code, it is possible to display the stack frame with native to Java and Java to native transitions. A double click on a Java method displays the high level code together with the Dalvik disassembly. In case the code is just-in-time compiled, TRACE32 uses the symbols of the Android libart.so library to parse the JIT cache in order to get the names and ranges of the so-called hot methods. The new command Data.LOAD.OAT allows to load files generated by Android RunTime (ART).
Build 80996 IMP 02471 ARM Debugger	**New secure/nonsecure/hypervisor virtual address classes** New access classes ZVM:, NVM:, HVM: allow zone specific MMU translations of xVM: accesses to AVM:
Build 76594 IMP 02451 ARM Debugger	**Enables ASID specific onchip breakpoint** Enable ASID specific onchip breakpoint, when target is using the context ID register in this way.
Build 65657 02432 ARM Debugger Cortex-A/-R Armv8 Debugger SMMU (Hardware system MMU (SMMU))	**New command group for ARM System MMU Analysis** Allows viewing the configuration and status of the ARM System MMU and associated page tables.
Build 38589 IMP 02301 ARM Debugger	**TrOnchip.Set vector catch capabilites ex** Allows to set secure/nonsecure/monitor vector catches for ARM cores with Trustzone support.
Build 19417 IMP 02160 ARM Debugger MIPS32/MIPS64 Debugger MPC5xx/8xx Debugger Data.LOAD (Load file)	**New DIFFLOAD options and targets** New options allow further compression or compressed load (/ZIPLOAD) and quick download checks (/CHECKLOAD and /CHECKONLY). Target agents for MIPS32 and MIPS64 architectures added.
Build 19417 IMP 02136 ARM Debugger	**MMU.TableWalk support for WinCE 5.x**
Build 19417 CHG 02133 ARM Debugger FDX.METHOD (Select communication channel)	**FDX method DCC wrong** Is named "DCC4" now. "DCC3" method introduced.

Build 13751 IMP 02111 ARM Debugger MMX (MMX registers (MultiMedia eXtension))	**XScale MMX register support**
Build 13751 IMP 02108 ARM Debugger ETM.DataTrace (Configure data-trace)	**New data trace modes**
Build 9640 CHG 02055 ARM Debugger C5500 Debugger SH Debugger BMC (Benchmark counters)	**Benchmark counter commands renamed** All benchmark counter related onchip trigger commands have been moved to the new command "BMC".
Build 4728 IMP 01974 ARM Debugger TRACE32 Instruction Set Simulator for CORTEX / ARM / XSCALE	**New command to fix literal attibutes** The command sYmbol.CLEANUP.CodeLiterals fixes literal attributes that really contain code (instructions synthesized by a define word).
Build 4728 IMP 01963 ARM Debugger TRACE32 Instruction Set Simulator for CORTEX / ARM / XSCALE	**Unified Assembly Language supported** SETUP.DIS command extended to choose between traditional disassembly and unified assembly language mode.
Build 4728 FIX 01925 ARM Debugger Data.LOAD.COFF (Load COFF file)	**Function info missing for ARM TI-COFF** Only symbol name and type information is there.
Build 4728 FIX 01923 ARM Debugger Data.LOAD.COFF (Load COFF file)	**TI ARM compiler ARM/Thumb attributes** Attributes may be wrong after literal constants in code.
Build 2940 CHG 01915 ARM Debugger	**access modes** additional access modes: ATE: EATE: ETE: TE: access modes deleted: ASJ: AUJ: EASJ: EAUJ: ESJ: EUJ: SJ: UJ:
Build 1190 IMP 01684 ARM Debugger ARM9 JTAG Debugger FIRE Emulator for ARM7 In-Circuit Emulator for ARM7 TRACE32 Instruction Set Simulator for CORTEX / ARM / XSCALE	**new accessmodes for ARM family** EC0:..EC15: EDEBG: EETB: EETM: EICE: EINI: added
Build 1151 IMP 01608 ARM Debugger	ARM SWI compatible semihosting supported (via breakpoint at SWI or via DCC channel)
Build 1151 FIX 01605 ARM Debugger Go (Debug control, program execution, and real-time emulation)	**OMAP1510** bug fixed in OMAP1510 multicore synchronous GO (caused ARM to step only)

Build 1145 IMP 01573 ARM Debugger	First implementation of background monitor for ARM
Build 1133 IMP 01495 ARM Debugger SYStem.Option (Special setup)	SYStem.Option StepVector and ICEbreaker for single stepping changed for ARM
Build 1133 FIX 01474 ARM Debugger	A.COV.List display on ARM with flowtrace was not conforming to the coverage conventions for ARM
Build 1096 FIX 01429 ARM Debugger	ARM MMU translation wrong for 64K pages sizes
Build 1085 FIX 01425 ARM Debugger	Stackframe for ARM frames was wrong for framesizes >= 256
Build 1079 FIX 01391 ARM Debugger	Simple range breakpoints on ARM that need to be converted into two masks may fail
Build 1077 FIX 01383 ARM Debugger	**Linux** ARM on linux host may crash when analyzing certain stack frames
Build 1077 IMP 01331 ARM Debugger	Onchip address ranges use both ICEbreaker registers to better convert the range
Build 1077 FIX 01328 ARM Debugger	ETM based breakpoints where not working without trace hardware
Build 1077 IMP 01324 ARM Debugger Break.SetFunc (Mark HLL functions)	B.SF on ARM was causing wrong memory access class (no other effects)
Build 1077 FIX 01306 ARM Debugger FIRE Emulator for ARM7	Target function call from Var commands for ARM fixed
Build 1077 FIX 01293 ARM Debugger C5500 Debugger	**Bugfix implemented for Multicore** Bugfix implemented for Multicore start/stop ARM+C55x
Build 1077 FIX 01290 ARM Debugger	**ELF/DWARF1** Stack backtrace on ARM changed for ELF/DWARF1 format and generally
Build 1077 FIX 01157 ARM Debugger MPC56x NEXUS Debugger and Trace Break.Set (Set breakpoints)	**B.Set dialog** B.Set dialog may crash on Sun for ARM or PPC-Nexus
Build 1077 FIX 01090 ARM Debugger FIRE Emulator for ARM7 In-Circuit Emulator for ARM7	**Disassembler** condition code in disassembler was missing on some instructions for ARM

Build 100486 IMP 02523 Hypervisor-aware Debugging ARM/Cortex Trace (parallel) Intel~ Processor Trace QorIQ PowerPC NEXUS Aurora Trace Port <trace>.List (List trace contents)	**Decoding of machine id supported** The machine ID within the Context-ID or ownership packet is now decoded.
Build 88288 IMP 02504 ARM/Cortex Trace (parallel) ETM.DataTrace (Configure data-trace) ETM.TimeStampsTrace (Specify data trace correlation method (ETMv4))	**ETMv4 data trace correlation extended to use timestamps** ETM.TimeStampTrace ON advises TRACE32 to assigns the data address/data value information of the data trace stream to the appropriate load/store instruction by using the timestamps. This requires that a timestamp unit is available.
Build 80996 IMP 02477 ARM/Cortex Trace (parallel) RTS (Real-time profiling (RTS))	**RTS command group supports ETMv4** RTS command group supports ETMv4.
Build 65657 02435 ARM/Cortex Trace (parallel) MPC55xx/MPC56xx NEXUS Debugger and Trace <trace>.List (List trace contents) <trace>.STATistic.PAddress (Which instructions accessed data address) <trace>.STATistic.PsYmbol (Shows which functions accessed data address)	**Prestore address information for ARM-ETM/NEXUS** The columns PAddress/PsYmbol in the trace listing show the address of the instruction that was executed before a read/write access was performed. The statistic function Trace.STATistic.PsYmbol analyses which functions performed read/write accesses. The statistic function Trace.STATistic.PAddress analyses which assembler instruction performed the read/write accesses.
Build 51144 ODD 02388 ARM/Cortex Trace (parallel)	**Analyzer.TSelect EXT** External trigger input on ETM connector is now turned off by default. The command Analyzer.TSelect EXT can enable or disable the trigger function.
Build 30461 IMP 02238 ARM/Cortex Trace (parallel)	**ETM+HTM trace correlation simplified** New "correlated" modes of "HTM.DataTrace" allow correlation of HTM trace to ETM. "Trace" commands shows merged traces.
Build 30461 IMP 02234 ARM/Cortex Trace (parallel)	**STM traces via Analyzer possible**
Build 26464 IMP 02225 ARM/Cortex Trace (parallel) ETM.ReturnStack (Enable return stack tracing mode)	**New command EMU.ETM.ReturnStack** Enables return stack compression in ETM (PFT) trace.

Build 22490 IMP 02201 ARM/Cortex Trace (parallel) ETM.PseudoDataTrace (Enable pseudo data trace detection)	**Pseudo data trace for ARM ETMv3** Pseudo data trace allows to transmit data information via a pure program trace by target code instrumentation. This brings limited data trace to Cortex-A8 core.
Build 19417 IMP 02179 ARM/Cortex Trace (parallel) LA.IMPORT.TracePort (Import off-chip trace data)	**New LA.IMPORT.TracePort command** Allows to import pure binary files from ETBs.
Build 19417 IMP 02173 ARM/Cortex Trace (parallel) <trace>.AutoFocus (Calibrate AUTOFOCUS preprocessor)	**Autofocus algorithm changed** For high speed traceports and AutoFocus-II preprocessor the old algorithm has choosen the wrong eye (too high delay).
Build 19417 IMP 02161 ARM/Cortex Trace (parallel) <trace>.TestFocus (Test trace port recording)	**Utilisation test for ETM traceports** "Analyzer.TestFocus /Utilisation" can test the utilisation of the traceport and uncover CoreSight trace bus configuration problems.
Build 19417 IMP 02156 ARM/Cortex Trace (parallel) DTM (DTM trace sources (Data Trace Module))	**Custom trace source support** The DTM command provides support for "simple" custom CoreSight trace sources.
Build 19417 IMP 02146 ARM/Cortex Trace (parallel) ETM.Register (Display the ETM registers)	**ETM.Register window enhanced** Contains now also related CoreSight components.
Build 19417 IMP 02143 ARM/Cortex Trace (parallel) Analyzer (Trace method Analyzer, recording, and analysis commands) ETM.PortSize (Define trace port width)	**Trace buffer usage for 8bit ETM changed** Doubles available trace records and increases frequency limit of older trace modules for ETM ports with 8 or less bits.
Build 19417 CHG 02140 ARM/Cortex Trace (parallel) <trace>.LOAD (Load trace file for offline processing) <trace>.SAVE (Save trace for postprocessing in TRACE32)	**Trace save/load for cycle accurate trace** Information in file is now using clock information. The display of timestamps in the loaded trace will depend on the Trace.CLOCK command.
Build 19417 IMP 02131 ARM/Cortex Trace (parallel) ETM.PortSize (Define trace port width)	**ETM support for special port sizes** Supporting also 3,5,6,7 bits (in normal mode) and unaligned trace data.

Build 19417 FIX 02124 ARM/Cortex Trace (parallel) Break.Set (Set breakpoints) ETM (Embedded Trace Macrocell (ETM))	**ETM selective data trace** Selective data trace may miss cycles in ETMv3.
Build 13751 IMP 02092 ARM/Cortex Trace (parallel)	**External Timestamping improved** New command Analyzer.ExportCLOCK to define export trace clock for CoreSight targets. Command Analyzer.ExportCLOCKDelay can fix additional delay on some trace ports.
Build 9640 IMP 02056 ARM/Cortex Trace (parallel)	**ETM programming dialog extended** Supports now new data, external input, external output and trace controls.
Build 4790 IMP 01982 ARM/Cortex Trace (parallel)	**FCSE (WinCE) supported for ARM trace**
Build 4728 IMP 01919 ARM/Cortex Trace (parallel)	**Smarttrace for ARM improved** Can now also generate interrupts in gaps. Several new algorithms added.
Build 2940 FIX 01881 ARM/Cortex Trace (parallel)	**ETM V3 possible flow errors with BLX** Flow error when BLX <Rn> call does not change operation mode.
Build 2940 FIX 01877 Embedded Trace Buffer Real-Time Trace ARM/Cortex Trace (parallel)	**ETMV3 cycle accurate trace timestamp** Timestamp wrong with selective tracing.
Build 1246 CHG 01812 ARM/Cortex Trace (parallel) Analyzer.Mode (Set the trace operation mode)	**Analyzer.Mode Prestore for ARM ETM** Selective data traces, setup with /TraceEnable breakpoints, will now display only the data cycles without code information. Enabling "A.Mode Prestore" will display (and trace) also the program information.
Build 1151 FIX 01593 ARM/Cortex Trace (parallel)	SWP instructions caused data alignment problems on ARM ETM
Build 1133 FIX 01514 ARM/Cortex Trace (parallel)	Bug fixed in ARM ETM preprocessor may work in wrong mode or with wrong threshold level
Build 1133 FIX 01494 ARM/Cortex Trace (parallel)	Bug fixed in ETM disassembly of interrupts and exceptions
Build 1133 IMP 01478 ARM/Cortex Trace (parallel)	onchip breakpoints for ETM make use of A0 and A1 address ignore bits to avoid full range resource when possible
Build 1133 FIX 01476 ARM/Cortex Trace (parallel)	ETM trigger dialog fails for TraceEnable (Global) with two addresses or ranges
Build 1079 IMP 01399 ARM/Cortex Trace (parallel) SmartTrace	SmartTrace for ARM improved
Build 1079 IMP 01389 ARM/Cortex Trace (parallel)	ETM.ON also clears all ETM registers
Build 1077 CHG 01369 ARM/Cortex Trace (parallel)	SYS.O ETMSWAP ON is now default (only 8&16 bit demux mode configuration)

Build 1077 FIX 01281 ARM/Cortex Trace (parallel)	**SYSTEM.Option ETM** SYSTEM.Option ETM... was not setting configuration on PowerTrace immediately
Build 1077 FIX 01220 ARM/Cortex Trace (parallel)	ARM ETM data messages after long trace gaps may cause flow errors (only in noncompressed trace mode)
Build 1077 FIX 01088 ARM/Cortex Trace (parallel)	**ETM counter programming failed** ETM counter programming failed when no RESTART condition was specified

ARM11 JTAG Debugger

Build 7181 ODD 01993 ARM11 JTAG Debugger	**MMU.SCANALL on ARM11 modified common** Common range is set by accident to 0xa0000000--0xffffffff.
Build 4728 IMP 01945 ARM11 JTAG Debugger Break.Set (Set breakpoints)	**ContextID supported by breakpoints+ETM** When TrOnchip.ContextID is enabled task-related breakpoints will use the ContextID comparators of breakpoint unit and ETM.
Build 1241 IMP 01811 ARM11 JTAG Debugger ARM7 JTAG Debugger ARM9 JTAG Debugger SYStem.JtagClock (Define JTAG frequency)	**New JTAG clock modes CTCK and CRTCK** New JTAG clock modes CTCK and CRTCK (besides RTCK and ARTCK) and improved download speed.
Build 1224 IMP 01754 ARM11 JTAG Debugger	**DFAR, DFSR, IFAR, IFSR saved/restored** DFAR, DFSR, IFAR, IFSR registers will be saved when entering debug mode and restored when restarting the user application. This avoids that memory accesses in debug mode, which would cause an abort, will modify these registers.

ARM7 Family Real-Time Trace

Build 1077 FIX 01260 ARM7 Family Real-Time Trace	**ARM bus trace disassembly** serveral fixes for ARM bus trace disassembly and CTS, new option EXEC to analyze EXEC signal

ARM7 JTAG Debugger

Build 1241 IMP 01811 ARM11 JTAG Debugger ARM7 JTAG Debugger ARM9 JTAG Debugger SYStem.JtagClock (Define JTAG frequency)	**New JTAG clock modes CTCK and CRTCK** New JTAG clock modes CTCK and CRTCK (besides RTCK and ARTCK) and improved download speed.
Build 1224 IMP 01758 ARM7 JTAG Debugger	**Flash prog. for PCD8070x implemented**

ARM9 JTAG Debugger

Build 1241 IMP 01811 ARM11 JTAG Debugger ARM7 JTAG Debugger ARM9 JTAG Debugger SYStem.JtagClock (Define JTAG frequency)	**New JTAG clock modes CTCK and CRTCK** New JTAG clock modes CTCK and CRTCK (besides RTCK and ARTCK) and improved download speed.
Build 1228 IMP 01778 ARM9 JTAG Debugger	**MOVE Coprocessor Register Access** The peripheral file and the required target routine is available in the system directory of the TRACE32 installation or on the CDROM at demo\arm\etc\coprocessor\move. Read the commends in "move.cmm".
Build 1190 IMP 01684 ARM Debugger ARM9 JTAG Debugger FIRE Emulator for ARM7 In-Circuit Emulator for ARM7 TRACE32 Instruction Set Simulator for CORTEX / ARM / XSCALE	**new accessmodes for ARM family** EC0:..EC15: EDEBG: EETB: EETM: EICE: EINI: added
Build 1190 CHG 01647 ARM9 JTAG Debugger	**Software breakpoints on ARM966E** Software breakpoints on ARM966E Rev.1 will fail sometimes. Setting of "SYStem.Option BUGFIX ON" is required to activate the workarround in the software. ARM966E Rev.2 has this buf already fixed.
Build 1145 FIX 01577 ARM9 JTAG Debugger	TrOnchip.Set StepVector was always active on ARM9
Build 1077 FIX 01265 ARM9 JTAG Debugger	**ICEbreaker** FCSE handing for ICEbreaker changed
Build 1077 FIX 01129 ARM9 JTAG Debugger	**ICEbreaker setting** ICEbreaker setting for FCSE on ARM926EJ was wrong

ASM Debugger

Build 100486 IMP 02531 ASM Debugger Data.USRACCESS (Prepare USR access)	**Set-up of Data.USRACCESS is reset on SYStem.Mode change** Set-up of Data.USRACCESS is reset on every change of the SYStem.Mode. If Data.USRACCESS is not configured, data accesses to access class USR cause an error.
Build 88288 CHG 02503 ASM Debugger Data.CLEARVM (Clear the TRACE32 virtual memory (VM:))	**New behavior for TRACE32 Virtual Memory** The TRACE32 Virtual Memory (VM:) is no longer initialized by 64KB blocks upon a write access. Instead, the data are just written to the specified addess/address range. The command Data.CLEARVM clears now exactly the address/address range specified.
Build 69655 CHG 02444 ASM Debugger Data.LOAD.Binary (Load binary file)	**Data.LOAD.Binary requires now start address or range** The command Data.LOAD.Binary now requires a start address or a range. Before a load to address zero was performed when no start address or range was provided.

AUTOFOCUS Self Calibration

Build 100486 IMP 02541 AUTOFOCUS Self Calibration CAnalyzer.SAMPLE (Set sample time offset) CAnalyzer.ShowFocus (Display data eye) CAnalyzer.ShowFocusClockEye (Show clock eye) CAnalyzer.TERMination (Configure parallel trace termination)	**Support for Whisker MIPI20T-HS CombiProbe/uTrace Arm** New commands to use new capability of Whisker MIPI20T-HS CombiProbe/uTrace Arm.

Adaptation for MIPS

Build 1241 IMP 01786 Adaption for Embedded Trace Macrocell Adaptation for MIPS MPC55xx/MPC56xx NEXUS Debugger and Trace <trace>.AutoFocus (Calibrate AUTOFOCUS preprocessor) <trace>.TestFocus (Test trace port recording)	**New commands to verify trace connection** The Analyzer.AutoTest command verifys the integrity of the trace connection. For adapters with adjustable threshold the command Analyzer.AutoFocus will also choose the threshold in the middle of the data eye.

Adaptation for OCDS Level 2 TriCore

Build 1268 IMP 01871 Adaptation for OCDS Level 2 TriCore Break.Set (Set breakpoints)	**onchip breakpoints mixable** Mixing onchip breakpoints with trace action and other onchip breakpoints was not allowed. When setting them anyway, they did not work. Mixing them is now allowed and works.

Adaption for Embedded Trace Macrocell

Build 1241 IMP 01786 Adaption for Embedded Trace Macrocell Adaptation for MIPS MPC55xx/MPC56xx NEXUS Debugger and Trace <trace>.AutoFocus (Calibrate AUTOFOCUS preprocessor) <trace>.TestFocus (Test trace port recording)	**New commands to verify trace connection** The Analyzer.AutoTest command verifys the integrity of the trace connection. For adapters with adjustable threshold the command Analyzer.AutoFocus will also choose the threshold in the middle of the data eye.

Android Debugging

Build 80996 IMP 02475 Android Debugging ARM Debugger Cortex-A/-R Armv8 Debugger Data.LOAD.OAT (Load OAT file)	**Android Debugging** The TRACE32 Android debug support has been extended for Android versions based on the Android RunTime (ART). This includes Android versions L, M and N. The new Android support allows the debugging of the ahead-of-time compiled Android framework and apps. TRACE32 automatically detects ahead-of-time compiled objects and and loads the DWARF/ELF info if available. Otherwise the debugger can parse the OAT data to extract the debug info. Additionally, TRACE32 supports the hybrid compilation introduced in Android N. For interpreted code, it is possible to display the stack frame with native to Java and Java to native transitions. A double click on a Java method displays the high level code together with the Dalvik disassembly. In case the code is just-in-time compiled, TRACE32 uses the symbols of the Android libart.so library to parse the JIT cache in order to get the names and ranges of the so-called hot methods. The new command Data.LOAD.OAT allows to load files generated by Android RunTime (ART).

C166 Family Real-Time Trace

Build 1096 BUG 01427 C166CBC Debugger C166 Family Real-Time Trace Break.SetFunc (Mark HLL functions)	breakpoints set with Break.SetFunc command may not be stored on CBC166 Risctrace
Build 1077 FIX 01318 FIRE Emulator for C166 Family C166 Family Real-Time Trace	CTS for C166 was tagging byte writes to registers wrong
Build 1077 IMP 01244 FIRE Emulator for C166 Family FIRE Emulator for ST10 In-Circuit Emulator for C166/ST10 - Out of Production C166 Family Real-Time Trace	**CTS** handling of volatile memory and atomic instructions improved for CTS with C166
Build 1077 CHG 01225 C166 Family Real-Time Trace TrOnchip.state (Display onchip trigger window)	C166 Risctrace trace control breakpoints implemented default settings of TrOnchip changed
Build 1077 FIX 01089 C166 Family Real-Time Trace	**Disassembler** Disassembler was not working for C166 RiscTrace

Build 1077 FIX 01258 C166CBC Debugger FIRE Emulator for C166 Family In-Circuit Emulator for C166/ST10 - Out of Production C166 Monitor Var.Call (Call a new procedure)	bugs fixed in Var.Call for C166 architecture
Build 1077 FIX 01232 C166CBC Debugger FIRE Emulator for C166 Family In-Circuit Emulator for C166/ST10 - Out of Production C166 Monitor	**Symbol comment** Symbol comment for C166 family direct addressing was wrong or missing

C166CBC Debugger

Build 1190 IMP 01683 C166CBC Debugger Super10 Debugger FIRE Emulator for C166 Family In-Circuit Emulator for C166/ST10 - Out of Production SIM Instruction Set Simulator for C166/ST10	**new access mode for 80166 family** EDBG: added
Build 1096 BUG 01427 C166CBC Debugger C166 Family Real-Time Trace Break.SetFunc (Mark HLL functions)	breakpoints set with Break.SetFunc command may not be stored on CBC166 Risctrace
Build 1077 FIX 01258 C166CBC Debugger FIRE Emulator for C166 Family In-Circuit Emulator for C166/ST10 - Out of Production C166 Monitor Var.Call (Call a new procedure)	bugs fixed in Var.Call for C166 architecture
Build 1077 FIX 01232 C166CBC Debugger FIRE Emulator for C166 Family In-Circuit Emulator for C166/ST10 - Out of Production C166 Monitor	**Symbol comment** Symbol comment for C166 family direct addressing was wrong or missing

Build 38589 CHG 02309 C2000 JTAG Debugger C5500 Debugger C6000 Debugger Data.LOAD.COFF (Load COFF file)	**Code loaded to logical address** Data.LOAD.COFF for TI architectures now loads the code at the logical address of the COFF file. The option /PHYSLOAD can force the load to the physical address.

Build 38589 CHG 02309 C2000 JTAG Debugger C5500 Debugger C6000 Debugger Data.LOAD.COFF (Load COFF file)	**Code loaded to logical address** Data.LOAD.COFF for TI architectures now loads the code at the logical address of the COFF file. The option /PHYSLOAD can force the load to the physical address.
Build 13751 IMP 02112 C5500 Debugger	**Correlated program & data trace for C55x**
Build 9640 CHG 02055 ARM Debugger C5500 Debugger SH Debugger BMC (Benchmark counters)	**Benchmark counter commands renamed** All benchmark counter related onchip trigger commands have been moved to the new command "BMC".
Build 4943 FIX 01990 C5500 Debugger Break.Set (Set breakpoints) Var.Break.Set (Set breakpoint to HLL expression)	**C55x data breakpoint problems fixed** Breakpoints with data value and long access breakpoints failed sometimes.
Build 4728 FIX 01941 C5500 Debugger Register.Set (Modify register contents)	**Register modification failed** Modification of CFCT, RETA and XARn was not working.
Build 1228 CHG 01780 TMS320C54XX Debugger C5500 Debugger	**accessmodes** additional access modes ETB: EETB:
Build 1224 IMP 01755 C5500 Debugger Onchip (Trace method Onchip, recording, and analysis commands)	**ETB support implemented** ETB (Embedded Trace Buffer) is an on-chip trace memory module from ARM, but can also be used on some derivatives to store the trace data produced by the DSP. The trace data can be read out by the debugger using the JTAG interface.
Build 1151 FIX 01604 C5500 Debugger Go (Debug control, program execution, and real-time emulation)	bug fixed in OMAP1510 multicore synchronous GO (caused ARM to step only)
Build 1145 FIX 01567 C5500 Debugger Data.LOAD.COFF (Load COFF file)	**COFF loader** bug fixed in C55x COFF loader (support of large pointers)
Build 1133 IMP 01509 C5500 Debugger	y.autoload.checkdll implemented for automatic loading of DLLs for C55x
Build 1133 FIX 01496 C5500 Debugger	Software breakpoints on C55x may not hit when realtime memory access is in use

Build 1133 IMP 01491 C5500 Debugger	New command TrOnchip.PROfile for time and event profiling on C55x
Build 1133 IMP 01469 C5500 Debugger	Advanced onchip breakpoints and benchmark counters on TMS320C55x supported
Build 1133 FIX 01459 C5500 Debugger PowerProbe - Logic and Protocol Analyzer	PowerProbe was not working as "Probe" command in TMS320C55x debugger
Build 1096 FIX 01443 C5500 Debugger	RCL/InterCom handling not suspended during long commands (caused subcore debug time on C55x/ARM debugger)
Build 1077 FIX 01376 C5500 Debugger	Nonaligned memory writes without TargetServer write wrong data on TMS320C55x
Build 1077 FIX 01343 C5500 Debugger	C55x fast stack model implemented
Build 1077 FIX 01341 C5500 Debugger	C55x bug fixed in multicore stepping when DSP was triggered before start
Build 1077 FIX 01293 ARM Debugger C5500 Debugger	**Bugfix implemented for Multicore** Bugfix implemented for Multicore start/stop ARM+C55x
Build 1077 IMP 01142 C5500 Debugger Data.LOAD.COFF (Load COFF file)	**Reset PC** Reset PC taken from COFF file for C55x
Build 1077 FIX 01140 C5500 Debugger	**C5xx disassembler** Bug fixed in C5xx disassembler, may crash on xcc and xccpart constructs

C6000 Debugger

Build 38589 CHG 02309 C2000 JTAG Debugger C5500 Debugger C6000 Debugger Data.LOAD.COFF (Load COFF file)	**Code loaded to logical address** Data.LOAD.COFF for TI architectures now loads the code at the logical address of the COFF file. The option /PHYSLOAD can force the load to the physical address.

CEVA-X Debugger

Build 125398 IMP 02587 ARM Debugger CEVA-X Debugger Trace Analyzer ETM.TraceCORE (Core specific default tracing)	**new command ETM.TraceCORE** New command ETM.TraceCORE to allow core specific default tracing.

CPU32 Debugger

Build 2940 FIX 01897 CPU32 Debugger	**CpuAccess Dualport fails** Will crash host software on all passive debuggers (eg. 68k,HC16).
Build 1190 FIX 01693 CPU32 Debugger	**Data.List mixed mode may fail** Mixed mode Data.List window display may caused a debug port fail in very rare cases.

ColdFire BDM Debugger

Build 1241 FIX 01804 ColdFire BDM Debugger	**Onchip breakpoints not working** Onchip breakpoints are not working on passive debug modules.
Build 1133 FIX 01503 ColdFire BDM Debugger	Coldfire 5606e was not recognized as MAC coldfire
Build 1096 FIX 01454 ColdFire BDM Debugger	Condition/Command/Spot not working for read/write breakpoints on Coldfire
Build 1096 FIX 01451 ColdFire BDM Debugger PERF.Mode (Specify sampling object)	PERF with StopAndGo method not working on Coldfire with PowerDebug
Build 1077 FIX 01385 ColdFire BDM Debugger	several bugs fixed in ColdfireV4 onchip breakpoints, watch and exclude breakpoints implemented
Build 1077 FIX 01197 ColdFire BDM Debugger	**Coldfire** R.SWAP and R.UNDO may fail for A7 on Coldfire BDM

ColdFire Trace

Build 13751 IMP 02106 ColdFire Trace	**Trace & Trigger on Coldfire trace** WDDATA and PULSE instructions can be used to control the trace.
Build 1077 IMP 01368 ColdFire Trace PowerTrace	Coldfire supported on PowerTrace

CombiProbe

Build 26464 IMP 02223 CombiProbe RTS (Real-time profiling (RTS))	**RTS supported with CombiProbe** Can be used for long time traces and coverage analysis for Cortex-M series.

Combiprobe

Build 30461 IMP 02237 Combiprobe ARM/Cortex Trace (parallel) ITM (CoreSight ITM (Instrumentation Trace Macrocell))	**ETM+ITM trace correlation simplified** "ITM.DataTrace DataCorrelated" selects data trace function with correlation to ETM trace. "Trace" commands will then show merged trace data.

Build 130863 CHG 02619 Cortex-A/-R Armv8 Debugger	**discontinue announcement** The usage of the TRACE32 executable named t32marm64 is now deprecated. t32marm64 was merged into t32marm, which now supports both 32-bit and 64-bit Arm CPUs. Please modify the executable name accordingly in your preferred start method(s) e.g. shortcut properties, shell scripts, ... To ease the migration of existing installations, we will provide a redirection executable or shell script in place of t32marm64 for a limited time.
Build 88288 IMP 02498 ARM Debugger Cortex-A/-R Armv8 Debugger Intel~ x86/x64 JTAG Debugger SYStem.Option MACHINESPACES (Address extension for guest OSes) SYStem.Option MACHINESPACES (Address extension for guest OSes) SYStem.Option MACHINESPACES (Address extension for guest OSes)	**Hypervisor-aware debugging feature introduced (Part 1-2)** The TRACE32 machinespaces framework was added for architectures ARM and Intel x86 to support Hypervisor environments. This framework allows the debugger to perform stop-mode debugging of targets running a hypervisor and one or multiple guest machines. In conjunction with a hypervisor awareness it is possible to debug all guest machines concurrently, whether they are currently active or currently inactive. TRACE32 allows to load an individual OS awareness for each virtual machine on the target. The machinespaces framework is enabled with the command SYStem.Option.MACHINESPACES ON. A new identifier <machineid> is introduced which allows to identify virtual machines on the target. TRACE32 addresses are augmented with the machineid to specify which machine the address belongs to. The TRACE32 debugger address translation (enabled with TRANSlation.ON) is extended to fully support the two-stage address translation which is usually applied for hardware virtualized guest machines. The first translation stage translates guest virtual addresses to intermediate physical addresses. The second translation stage translates intermediate physical addresses to absolute physical addresses. The MMU registers needed for the page table walk steps are automatically fetched from the hardware registers or, for inactive guests, from hypervisor data structures. If SYStem.Option.MACHINESPACES is enabled, - addresses are extended with a machine ID identifier, separating the machine ID with three colons from the (extended) virtual address: <accessclass>:<machineid>:::<spaceid>::<virtaddress> The machine ID clearly specifies which host or guest machine the address belongs to. The host machine always uses machineid 0. Guests have a machineid larger than 0. TRACE32 currently supports up to 30 machines.

Build 80996 IMP 02475 Android Debugging ARM Debugger Cortex-A/-R Armv8 Debugger Data.LOAD.OAT (Load OAT file)	**Android Debugging** The TRACE32 Android debug support has been extended for Android versions based on the Android RunTime (ART). This includes Android versions L, M and N. The new Android support allows the debugging of the ahead-of-time compiled Android framework and apps. TRACE32 automatically detects ahead-of-time compiled objects and and loads the DWARF/ELF info if available. Otherwise the debugger can parse the OAT data to extract the debug info. Additionally, TRACE32 supports the hybrid compilation introduced in Android N. For interpreted code, it is possible to display the stack frame with native to Java and Java to native transitions. A double click on a Java method displays the high level code together with the Dalvik disassembly. In case the code is just-in-time compiled, TRACE32 uses the symbols of the Android libart.so library to parse the JIT cache in order to get the names and ranges of the so-called hot methods. The new command Data.LOAD.OAT allows to load files generated by Android RunTime (ART).
Build 65657 02432 ARM Debugger Cortex-A/-R Armv8 Debugger SMMU (Hardware system MMU (SMMU))	**New command group for ARM System MMU Analysis** Allows viewing the configuration and status of the ARM System MMU and associated page tables.
Build 60219 IMP 02422 Cortex-A/-R Armv8 Debugger SYStem.Option ZoneSPACES (Enable symbol management for Arm zones)	**Onchip Breakpoints ARMv8 are now mode aware** Onchip breakpoints are now secure/nonsecure/hypervisor aware when SYStem.Option ZoneSPACES is enabled.

Cortex-M (ARMv6/7/8 32-bit) Debugger

Build 19417 IMP 02167 Cortex-M (ARMv6/7/8 32-bit) Debugger BMC (Benchmark counters) ITM (CoreSight ITM (Instrumentation Trace Macrocell))	**Cortex-M benchmark counter support** Benchmark counters can be traced by ITM and displayed graphically.

Build 1253 FIX 01841 DSP56800 and DSP56800E Debugger	**SYStem.Up with SYS.Option.DE off fixed**
Build 1084 FIX 01418 DSP56K Debugger ONCE Debugger for DSP56300 DSP56800 and DSP56800E Debugger	DSP56xxx IEEE loaded produced wrong function address ranges
Build 1084 FIX 01415 DSP56K Debugger ONCE Debugger for DSP56300 DSP56800 and DSP56800E Debugger	DSP56xxx bug fixed in IEEE loader for OLD compiler formats
Build 1077 CHG 01254 DSP56800 and DSP56800E Debugger	56800: EPLD support 32bit register, TLM (56800E) and no dummy shifts necessary any more. drvbdm59.c adjust.
Build 1077 IMP 01121 DSP56800 and DSP56800E Debugger	**Second onchip breakpoint unit** DSP56800 Debugger supports second onchip breakpoint unit
Build 1077 IMP 01119 DSP56800 and DSP56800E Debugger	DSP56800 JTAG access optimized for better performance

Build 1190 01685 DSP56K Debugger ONCE Debugger for DSP56300 SYStem.Option (Special setup)	**DSP56xxx IMASKASM/IMASKHLL locked** SYStem.Option IMASKASM/IMASKHLL was locked unintentionally.
Build 1096 IMP 01442 DSP56K Debugger ONCE Debugger for DSP56300	DSP5600x/DSP563xx strings displayed as "WideString"
Build 1085 FIX 01426 DSP56K Debugger MPC5xx/8xx Debugger SH Debugger	missing files after software installation - install database corrected files are on the CD, but weren't not copied FIRE/ICD-C166/ST10: men7860t.men ICD-DSP56K : per56802.per per56807.per per56809.per per56367.per ICD-PPC : per8280.per ICD-SH2 : per7294.per
Build 1084 FIX 01418 DSP56K Debugger ONCE Debugger for DSP56300 DSP56800 and DSP56800E Debugger	DSP56xxx IEEE loaded produced wrong function address ranges
Build 1084 FIX 01416 DSP56K Debugger	DSP56xxx bug fixed in stack backtrace for internal stack
Build 1084 FIX 01410 DSP56K Debugger ONCE Debugger for DSP56300	**DSP563xx disassembler error** DSP563xx and DSP5600x jump targets calculated wrong (HLL Step into fails)

Debug Modules

Build 125398 CHG 02599 Debug Modules PowerTrace	**discontinued hardware support** This is the last TRACE32 software release which supports TRACE32 Debug Interface LA-7701 TRACE32 PowerTrace 64MB LA-7706 TRACE32 PODBUS Ethernet Controller

Debug Support for Unified EFI Bootloader

Build 76594 IMP 02455 Debug Support for Unified EFI Bootloader	**UEFI Debugging for release builds** Debugging of UEFI drivers in release builds (no debug symbols, optimized code) is now supported.

Debugger Features

Build 60219 IMP 02423	**Full SMP support for auto sequences**
Debugger Features Data.EPILOG.state (Display data epilogs) Data.PROLOG.state (Display data prologs) Data.SOFTEPILOG.state (Display data softepilogs) Data.SOFTPROLOG.state (Display data softprologs) Data.STARTUP.state (Startup data state display) Data.TIMER.state (Timer state display) FLASH.EPILOG.state (Display FLASH epilogs) FLASH.PROLOG.state (Display FLASH prologs)	Auto-sequences can be supplied for each core in an SMP system.

Debugger for 80C196EA

Build 125398 CHG 02597	**restricted TRACE32 ICD software development**
JTAG Debugger Debugger for 80C196EA MMDSP Debugger Host Driver Software Integrated Development Environment x186 Monitor x196 Monitor XA51 Monitor TRACE32 Software	This is the last TRACE32 software release which contains for certain ICD/debugger software derived from the standard TRACE32 development trunk. A separate branch from the TRACE32 software will be created for supporting legacy ICD/debugger hardware only. Only bugfixes and host OS adaptations are planned for the future. effected CPU architectures: XA51, h85, i196, 186, MMDSP+

Debugging via XCP

Build 100486 IMP 02524	**Support for ASAM XCP**
Debugging via XCP SYStem.CONFIG XCP (XCP specific settings)	Support for debugging over XCP as specified in ASAM MCD-1 (XCP) standard, version 1.5 by ASAM e.V for GTM, Qorivva, RH850 and Tricore.

EPROM/FLASH Simulator

Build 125398 CHG 02581 EPROM/FLASH Simulator PowerIntegrator - Logic and Bus Analyzer PowerProbe - Logic and Protocol Analyzer TRACE32 Software Stimuli Generator	**discontinued software and hardware support** The t32pbi executables aren't supported anymore.

Embedded Trace Buffer Real-Time Trace

Build 2940 FIX 01877 Embedded Trace Buffer Real-Time Trace ARM/Cortex Trace (parallel)	**ETMV3 cycle accurate trace timestamp** Timestamp wrong with selective tracing.

FIRE Emulator MPC8XX

Build 1206 CHG 01735 MPC5xx/8xx Debugger FIRE Emulator MPC8XX In-Circuit Emulator for Power Architecture SIM Instruction Set Simulator for PowerPC	**accessmodes** additional access modes PMR: F: AF: EF: EAF: FP: AFP: EFP: EAFP: FD: AFD: EFD: EAFD: SF: ASF: ESF: EASF: UF: AUF: EUF: EAUF: V: AV: EV: EAV: VP: AVP: EVP: EAVP: VD: AVD: EVD: EAVD: SV: ASV: ESV: EASV: UV: AUV: EUV: EAUV:
Build 1161 IMP 01621 FIRE Emulator MPC8XX ICD Solutions for PowerPC and Power Architecture SIM Instruction Set Simulator for PowerPC	**Access modes** additional access mode DBG: and EDBG:
Build 1096 FIX 01434 FIRE Emulator MPC8XX MPC5xx/MPC8xx Trace	MPC5xx/MPC8xx flow trace display may hang when PTR cycles are missing

Build 1190 IMP 01684 ARM Debugger ARM9 JTAG Debugger FIRE Emulator for ARM7 In-Circuit Emulator for ARM7 TRACE32 Instruction Set Simulator for CORTEX / ARM / XSCALE	**new accessmodes for ARM family** EC0:..EC15: EDEBG: EETB: EETM: EICE: EINI: added
Build 1161 IMP 01620 FIRE Emulator for ARM7 ICD Solutions for ARM TRACE32 Instruction Set Simulator for CORTEX / ARM / XSCALE	**Access modes** additional access mode ETB:
Build 1077 FIX 01306 ARM Debugger FIRE Emulator for ARM7	Target function call from Var commands for ARM fixed
Build 1077 FIX 01090 ARM Debugger FIRE Emulator for ARM7 In-Circuit Emulator for ARM7	**Disassembler** condition code in disassembler was missing on some instructions for ARM

FIRE Emulator for C166 Cell-Based-Core

Build 4728 FIX 01929 FIRE Emulator for C166 Family FIRE Emulator for C166 Cell-Based-Core FIRE Emulator for H8S and H8/300H FIRE Emulator for SH2 FIRE Emulator for NEC V850 FIRE Emulator for C166S V2 Family	**RESET command crashes**

Build 4728 FIX 01929 FIRE Emulator for C166 Family FIRE Emulator for C166 Cell-Based-Core FIRE Emulator for H8S and H8/300H FIRE Emulator for SH2 FIRE Emulator for NEC V850 FIRE Emulator for C166S V2 Family	**RESET command crashes**
Build 1190 IMP 01683 C166CBC Debugger Super10 Debugger FIRE Emulator for C166 Family In-Circuit Emulator for C166/ST10 - Out of Production SIM Instruction Set Simulator for C166/ST10	**new access mode for 80166 family** EDBG: added
Build 1085 IMP 01423 FIRE Emulator for C166 Family FIRE Emulator for ST10 FIRE Emulator for NEC V850 Break.SetFunc (Mark HLL functions)	C167/ST10 and V850 fire Break.SetFunc improved to save breakpoint resources
Build 1084 FIX 01419 FIRE Emulator for C166 Family FIRE Emulator for ST10	C167/ST10 Fire second block of bondout flag memories did not work
Build 1077 FIX 01318 FIRE Emulator for C166 Family C166 Family Real-Time Trace	CTS for C166 was tagging byte writes to registers wrong
Build 1077 FIX 01258 C166CBC Debugger FIRE Emulator for C166 Family In-Circuit Emulator for C166/ST10 - Out of Production C166 Monitor Var.Call (Call a new procedure)	bugs fixed in Var.Call for C166 architecture
Build 1077 IMP 01244 FIRE Emulator for C166 Family FIRE Emulator for ST10 In-Circuit Emulator for C166/ST10 - Out of Production C166 Family Real-Time Trace	**CTS** handling of volatile memory and atomic instructions improved for CTS with C166
Build 1077 FIX 01232 C166CBC Debugger FIRE Emulator for C166 Family In-Circuit Emulator for C166/ST10 - Out of Production C166 Monitor	**Symbol comment** Symbol comment for C166 family direct addressing was wrong or missing

FIRE Emulator for C166S V2 Family

Build 4728 FIX 01929 FIRE Emulator for C166 Family FIRE Emulator for C166 Cell-Based-Core FIRE Emulator for H8S and H8/300H FIRE Emulator for SH2 FIRE Emulator for NEC V850 FIRE Emulator for C166S V2 Family	**RESET command crashes**

FIRE Emulator for Freescale 68HC12 / MCS12 / S12X

Build 1206 CHG 01730 S12 Debugger FIRE Emulator for Freescale 68HC12 / MCS12 / S12X In-Circuit Emulator for 68HC11 In-Circuit Emulator for 68HC12 SIM Instruction Set Simulator for HC12/MCS12	**accessmodes** additional access modes X: XD: XP: AX: AXD: AXP: EX: EXD: EXP: EAX: EAXD: EAXP:
Build 1204 CHG 01724 S12 Debugger FIRE Emulator for Freescale 68HC12 / MCS12 / S12X In-Circuit Emulator for 68HC11 In-Circuit Emulator for 68HC12 SIM Instruction Set Simulator for HC12/MCS12	**access modes** additional access modes AG: EAG: GD: AGD: EGD: GP: AGP: EGP:
Build 1198 CHG 01713 S12 Debugger FIRE Emulator for Freescale 68HC12 / MCS12 / S12X In-Circuit Emulator for 68HC11 In-Circuit Emulator for 68HC12 SIM Instruction Set Simulator for HC12/MCS12	**access modes** additional access modes G: EG:

Build 4728 CHG 01953 S08 Debugger FIRE Emulator for Freescale HC08 / HCS08 In-Circuit Emulator for 68HC05 and 68HC08	**accessmodes** additional access modes: DBG: EDBG: INI:

FIRE Emulator for H8S and H8/300H

Build 4728 FIX 01929 FIRE Emulator for C166 Family FIRE Emulator for C166 Cell-Based-Core FIRE Emulator for H8S and H8/300H FIRE Emulator for SH2 FIRE Emulator for NEC V850 FIRE Emulator for C166S V2 Family	**RESET command crashes**
Build 1228 FIX 01768 FIRE Emulator for H8S and H8/300H	**Fire H8S system window store fails** Processor mode may be stored wrong. Some settings missing.
Build 1096 FIX 01449 FIRE Emulator for H8S and H8/300H	H8 internal memory is only shadow memory on Fire
Build 1096 FIX 01441 FIRE Emulator for H8S and H8/300H	Dualport access problem to bit variables denied for H8S/H8300H in 64K and 1M addressing modes
Build 1077 FIX 01105 FIRE Emulator for H8S and H8/300H	**USB interface during boot** Fire H8S may cause link error or fail on USB interface during boot

FIRE Emulator for NEC V850

Build 4728 FIX 01929 FIRE Emulator for C166 Family FIRE Emulator for C166 Cell-Based-Core FIRE Emulator for H8S and H8/300H FIRE Emulator for SH2 FIRE Emulator for NEC V850 FIRE Emulator for C166S V2 Family	**RESET command crashes**
Build 1085 IMP 01423 FIRE Emulator for C166 Family FIRE Emulator for ST10 FIRE Emulator for NEC V850 Break.SetFunc (Mark HLL functions)	C167/ST10 and V850 fire Break.SetFunc improved to save breakpoint resources

FIRE Emulator for SH2

Build 4728 FIX 01929 FIRE Emulator for C166 Family FIRE Emulator for C166 Cell-Based-Core FIRE Emulator for H8S and H8/300H FIRE Emulator for SH2 FIRE Emulator for NEC V850 FIRE Emulator for C166S V2 Family	**RESET command crashes**
Build 1096 FIX 01456 FIRE Emulator for SH2	Single step on SH2 Fire was not working

Build 1085 IMP 01423 FIRE Emulator for C166 Family FIRE Emulator for ST10 FIRE Emulator for NEC V850 Break.SetFunc (Mark HLL functions)	C167/ST10 and V850 fire Break.SetFunc improved to save breakpoint resources
Build 1084 FIX 01419 FIRE Emulator for C166 Family FIRE Emulator for ST10	C167/ST10 Fire second block of bondout flag memories did not work
Build 1077 IMP 01249 FIRE Emulator for ST10	**Onchip program breakpoints** Onchip program breakpoints implemented for ST10 bondout, also used for stepping
Build 1077 IMP 01244 FIRE Emulator for C166 Family FIRE Emulator for ST10 In-Circuit Emulator for C166/ST10 - Out of Production C166 Family Real-Time Trace	**CTS** handling of volatile memory and atomic instructions improved for CTS with C166
Build 1077 FIX 01082 FIRE Portanalyzer FIRE Emulator for ST10	FIRE Port analyzer channels for ST10F27x where those of ST10F280

FIRE Emulator for TriCore

Build 1161 CHG 01627 FIRE Emulator for TriCore ICD Solutions for TriCore SIM Instruction Set Simulator for TriCore	**Access modes** access modes BBB: and EBBB: renamed to EEC: and EEEC:
Build 1161 IMP 01623 FIRE Emulator for TriCore ICD Solutions for TriCore SIM Instruction Set Simulator for TriCore	**Access modes** additional access modes BBB: and EBBB:

Build 125398 CHG 02596 In-Circuit Emulator FIRE Fully Integrated RISC Emulator Host Driver Software Integrated Development Environment TRACE32 Software	**restricted TRACE32 FIRE software development** This is the last TRACE32 software release which contains FIRE software derived from the standard TRACE32 development trunk. A separate branch from the TRACE32 software will be created for supporting FIRE hardware only. Only bugfixes and host OS adaptations are planned for the future.
Build 13751 FIX 02066 FIRE Fully Integrated RISC Emulator Out (Output control)	**fire analyser trigger unit - OUT.A** output signal from fire analyser trigger unit didn't work
Build 9640 FIX 02031 FIRE Fully Integrated RISC Emulator Host Driver Software	**named USB devices on FIRE** storing of devices name caused crash
Build 2940 IMP 01911 JTAG Debugger FIRE Fully Integrated RISC Emulator ICE In-Circuit Emulator PowerTrace for NEXUS TRACE32 Instruction Set Simulators	**Practice functions** synonym D.SUM() for DATA.SUM() added
Build 1246 CHG 01830 FIRE Fully Integrated RISC Emulator Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**programming of analyzer triggerunit** keyword for addressevent definition ADDR replaced by ADDRESS
Build 1246 CHG 01813 JTAG Debugger FIRE Fully Integrated RISC Emulator ICE In-Circuit Emulator ROM Monitor	**Address range expressions** Behavior of general parameter parser changed The checking of accessmode equality inside address range expressions like address1--address2 is switched off now. Now it's allowed to use different accessmodes for begin and end address of address ranges. The accessmode of the end address will be ignored and will not cause an error message anymore. e.g. UP:1000--SD:2000 will be handled as UP:1000--UP:2000
Build 1241 FIX 01794 FIRE Fully Integrated RISC Emulator Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Programming of complex triggerunit** hardware and onchip breakpoints configured wrong under circumstances when breakpoints were defined inside a analyzer trigger program via ADDR declaration

Build 1190 FIX 01665 FIRE Fully Integrated RISC Emulator Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**analyzer trigger unit programming** hostbased software may crash under certain circumstances
Build 1161 CHG 01638 FIRE Fully Integrated RISC Emulator Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**analyzer triggerunit programming** declarationtype synonymes introduced EVCNT for EVENTCOUNTER TICNT for TIMECOUNTER
Build 1151 FIX 01614 FIRE Fully Integrated RISC Emulator Data.LOAD.Binary (Load binary file)	**Download** download of binary files to monitor control based FIREs may fail
Build 1145 FIX 01549 FIRE Fully Integrated RISC Emulator Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Analyzer trigger unit programming** output events BUS.A..BUS.D programmed with inverse level (only FEC hardware release > 5 effected)
Build 1077 FIX 01363 FIRE Fully Integrated RISC Emulator	MAP.PROTECT was locked on FIREs which have memory protection
Build 1077 IMP 01315 FIRE Fully Integrated RISC Emulator PowerTrace	New analyzer mode Leach for PowerTrace and FIRE
Build 1077 FIX 01279 FIRE Fully Integrated RISC Emulator	Problem on FCC may not produce coredump on USB but instead stall USB
Build 1077 FIX 01153 FIRE Fully Integrated RISC Emulator ICE In-Circuit Emulator PowerIntegrator - Logic and Bus Analyzer PowerTrace PowerProbe - Logic and Protocol Analyzer <trace>.AutoInit (Automatic initialization)	Automatic init of analyzers (Integrator,PowerTrace,FIRE,ICE) changed
Build 1077 CHG 01107 FIRE Fully Integrated RISC Emulator Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Renamed** FIRE: analyzer triggerunit programming: Trigger.Puls and Trigger.PULSHIGH renamed to Trigger.Pulse and Trigger.PULSEHIGH

Build 1077 FIX 01082 FIRE Portanalyzer FIRE Emulator for ST10	FIRE Port analyzer channels for ST10F27x where those of ST10F280

FLASH Programming (Memory-Mapped)

Build 93173 IMP 02519 FLASH Programming (Memory-Mapped) NAND FLASH Programming	**Allows TRACE32 to send SPI FLASH commands** Allows TRACE32 to send valid FLASH command sequences to a SPI FLASH device.
Build 93173 IMP 02506 FLASH Programming (Memory-Mapped) FLASH.HOOKSCRIPT (PRACTICE script based FLASH programming prolog)	**PRACTICE script based FLASH programming prolog** The <hook_script> can perform checks, set-ups etc. to guarantee that the FLASH programming works properly afterwards, e.g. to avoid fatal problems that might occur when the FLASH programming erases or modifies FLASH sectors that contain information that is necessary to operate the debug interface or the chip. First use case: TriCore TC3xx.
Build 69655 IMP 02441 FLASH Programming (Memory-Mapped) FLASH.CFI (Generate FLASH declaration by CFI) FLASH.TARGET2 (Define second target controlled algorithm)	**Simultaneous programming of multiple NOR flash devices** The new command FLASH.TARGET2 allows the simultaneous programming of multiple flash devices with two different flash algorithms. This is needed for example to program processor internal flash and processor external NOR flash/HyperFlash/QSPI flash together. The new option /TARGET2 for the FLASH.CFI command allows to link a second flash algorithm to a CFI conform flash device. The new function FLASH.TARGET2.FILE() is provided to get the filename of the second flash algorithm.
Build 65657 02433 FLASH Programming (Memory-Mapped) FLASH.state (FLASH programming dialog)	**FLASH.state command** New dialog to create, check and modify the setup for NOR and onchip flash programming.
Build 38589 IMP 02298 TriCore Debugger FLASH Programming (Memory-Mapped)	**protection for TriCore Boot Mode Header** Mechanism implemented for preventing unintended deletion or invalidation of a BMI Header to avoid a lock-out of your AURIX device. See the corresponding flash scripts in ~~/demo/tricore/flash/ for details.
Build 5151 ERR 01996 FLASH Programming (Memory-Mapped) FLASH.Create (Declare FLASH)	**Flash family codes AM29x256 not working** Flash programming with family codes AM29LV256, AM29LV256B, AM29M256, AM29M256B, AM29N256 and AM29M2562 is failing after a few words (since 09 March 2006).

GTM Debugger (TriCore)

Build 105499 IMP 02560 XC2000/C166SV2 Debugger GTM Debugger (TriCore) PCP Debugger (TriCore) TriCore Debugger	**improved MCDS trace decoder speed**
Build 38589 IMP 02299 GTM Debugger (TriCore) MPC5xxx and SPC5xx Debugger TriCore Debugger	**initinal support for BOSCH GTM added** Initinal support for the BOSCH GTM (Generic Timer Module) added. Available in TriCore and PowerPC devices.
Build 38589 IMP 02297 GTM Debugger (TriCore) PCP Debugger (TriCore) TriCore Debugger CLOCK (Display date and time)	**on-chip clock display and settings** The CLOCK command group simplifies the setup for timestamp decoding of an on-chip trace. Can also be used to verify PLL and clock setup.

H8S and H8/300H Monitor

Build 1133 FIX 01499 H8S and H8/300H Monitor	standard H8S rom monitor was not working correctly with extended interrupt mode for CPU-type = 7

Help System

Build 22490 ODD 02213 Help System HELP.FILTER.Delete (Delete filter from filter list)	**online help** command HELP.FILTER.Delete syntax change - parameter is filter name instead of number now

Build 88288 IMP 02491 High-Level-Language Debugging Var.AddWatch (Add variable to Var.Watch window) Var.Watch (Open Var.Watch window)	**Var.Watch keeps formatting if opened without variable** Var.Watch window applies formatting options to all added variables, if opened with format parameters but without variable names.
Build 76594 ODD 02453 High-Level-Language Debugging Go.Return (Complete HLL function)	**Behavior of Go.Return command was changed** First Go.Return stops now at function epilog, here local variables are still valid. Second Go.Return stops at the return of the function.
Build 69655 CHG 02443 High-Level-Language Debugging	**Data.LOAD.auto generates error for unknown file format** Data.LOAD.auto throws error if file format is unknown. Before the file was loaded as binary.
Build 65657 02434 High-Level-Language Debugging sYmbol.List.SOURCE (Display source file names) sYmbol.SourcePATH.List (List source search paths) sYmbol.SourcePATH.SetBaseDir (Define directory as base for relative paths) sYmbol.SourcePATH.Translate (Replace part of the source path)	**Improved source path correction** Module/source that could not be loaded, because the source path needs to be adapted in the current debug environment are printed in red in the sYmbol.List.SOURCE window. A Resolve Path command was added to the Source context menu, which fixes the paths in the current debug environment after one correct path was provided.
Build 60219 IMP 02424 High-Level-Language Debugging Var.set (Modify variable)	**C/C++ parser supports scaled assignments** C/C++ expression parser extended to support scaled assignment (e.g. for ASAP file format).
Build 60219 CHG 02421 Intel~ x86/x64 JTAG Debugger High-Level-Language Debugging Data.LOAD.Elf (Load ELF file)	**Logical memory access for Data.LOAD.Elf** Data.LOAD.Elf use logical memory access when load is not done to a physical address (x86/x64)
Build 4728 FIX 01932 High-Level-Language Debugging Var.View (Display variables)	**Internal functions in HLL parser fail** When used in windows (e.g. Var.View) the functions will cause strange effect on the command line.

Build 125398 CHG 02597 JTAG Debugger Debugger for 80C196EA MMDSP Debugger Host Driver Software Integrated Development Environment x186 Monitor x196 Monitor XA51 Monitor TRACE32 Software	**restricted TRACE32 ICD software development** This is the last TRACE32 software release which contains for certain ICD/debugger software derived from the standard TRACE32 development trunk. A separate branch from the TRACE32 software will be created for supporting legacy ICD/debugger hardware only. Only bugfixes and host OS adaptations are planned for the future. effected CPU architectures: XA51, h85, i196, 186, MMDSP+
Build 125398 CHG 02595 Host Driver Software	**discontinued host OS version** This is the last TRACE32 software release which supports Windows XP, Windows Vista and Linux 32Bit.
Build 125398 CHG 02579 In-Circuit Emulator Host Driver Software ICE In-Circuit Emulator Integrated Development Environment TRACE32 Software	**restricted TRACE32 ICE software development** This is the last TRACE32 software release which contains ICE software derived from the standard TRACE32 development trunk. A separate branch from the TRACE32 software will be created for supporting ICE hardware only. Only bugfixes and host OS adaptations are planned for the future.
Build 112182 ODD 02563 Host Driver Software	**new TRACE32 screen driver model** Lauterbach implemented a new screen driver model to support better different GUI frameworks. The GUI specific software parts were split out into shared libraries (t32screen*). Detailed information can be found under https://www.lauterbach.com/3737.
Build 105499 IMP 02551 Host Driver Software	**config.t32: new keyword DNSRETRIES=** There is a new option DNSRETRIES= for the NET related settings under PBI= and LINK= in config.t32. DNSRETRIES= sets the number of retries when trying to resolve a DNS name on startup. DNSRETRIES= has a backwards compatible default of 0. This can be helpful when the DNS entry for a TRACE32 debugger is dynamically created or changed by DHCP and needs time to propagate through the DNS hierarchy.
Build 88288 CHG 02561 Host Driver Software	**discontinued host OS version** This is the last TRACE32 software release which supports Solaris 8, RedHat RHEL5, Suse SLES10/SLED10 and CentOS5.
Build 88288 CHG 02486 Host Driver Software	**Minimum requirements for Linux/Motif executables changed** After the DVD 2017/09 the minimum requirements for the Linux/Motif executables will be raised to: glibc >= 2.12, kernel >= 2.6.32 This corresponds to the same requirements as the Linux/Qt builds.
Build 34458 FIX 02253 Host Driver Software	**online help linking** online help linking fixed for window prefix commands like Win, WinFreeze, WinBack, WinResist, WinPrint, WinSmall, WinMid, WinLarge, WinTrans, WinExt

Build 26464 CHG 02230 Host Driver Software	**discontinued windows versions** This is the last TRACE32 software release which supports Windows 9x, ME, and NT 4.0.
Build 22490 IMP 02218 Host Driver Software	**font installation under LINUX simplified**
Build 22490 IMP 02216 Host Driver Software	**modified USB driver file for LINUX** The UDEV driver file for USB interface was modified to avoid troubles with newer kernel releases.
Build 13751 IMP 02110 Host Driver Software	**Graphics on Windows improved** Support for ClearType fonts and additional colors for bitmaps. Adapted to Windows Vista.
Build 13751 FIX 02076 Host Driver Software	**t32rem** t32rem crashed with too long command line or return message strings
Build 13751 IMP 02072 Host Driver Software	**new PRACTICE function EOF()** Function returns a boolean whether the last READ command reached the file end or not.
Build 13751 IMP 02068 Host Driver Software	**general command parameter parser** comparisions == and != added for boolean values
Build 9640 FIX 02033 Host Driver Software	**Systeminfo usage under UNIX failed** menuitem Help -> Support -> Systeminfo... didn't worked properly under UNIX file support.cmm won't be found
Build 9640 IMP 02030 Host Driver Software	**support of named USB devices added** Multiple named USB devices can be used simultaneously on Windows and Linux (x86). This feature requires a debugger firmware update (at least V8.0) and for Windows use of the current USB driver (V5.5).
Build 7181 IMP 02011 Host Driver Software	**general command parameter parser** division and multiplication of single time values added
Build 7181 IMP 01997 Host Driver Software	**ethernet interface** hostbased software changed to increase the tolerance concerning wrong package order (avoiding sw hang)
Build 4728 FIX 01936 Host Driver Software OS.screen (Call up the shell or execute host command)	**command OS crashes under circumstances** The command OS.screen crashed if the length of <commandline> exceeded 252 characters.
Build 2940 CHG 01910 Host Driver Software AREA.CLEAR (Clear area)	**area.clear behavior changed** without a given area name the default area 'A000' is cleared now
Build 1246 FIX 01836 Host Driver Software	**PowerDebug USB2 on USB1 transfer errors** Sporadic transfer errors appear on Windows XP with USB1 interface and USB2 debug module.

Build 1224 FIX 01854 Host Driver Software	**USB under WIN98SE** installation of TRACE32 usb driver failed under Windows 98SE
Build 1204 IMP 01848 Host Driver Software	**USB under LINUX** USB interface supported under LINUX (hostbased driver only)
Build 1190 IMP 01689 Host Driver Software	**Support for Windows 2003 Server** USB drivers are available for x86, ia64 and amd64 versions.
Build 1133 FIX 01486 Host Driver Software	**USB Win98** USB disconnect while operating was causing bluescreen on Win98

Hypervisor-aware Debugging

Build 105499 IMP 02559 Hypervisor-aware Debugging OS-aware Debugging	**option /ACCESS added for TASK.CONFIG and EXTension.LOAD** The defined access class will be used always from the OS and Hypervisor awareness.
Build 100486 IMP 02523 Hypervisor-aware Debugging ARM/Cortex Trace (parallel) Intel~ Processor Trace QorIQ PowerPC NEXUS Aurora Trace Port <trace>.List (List trace contents)	**Decoding of machine id supported** The machine ID within the Context-ID or ownership packet is now decoded.

Build 125398 CHG 02580 JTAG Debugger PODBUS Ethernet Controller ICD In-Circuit Debugger TRACE32 Software	**discontinued software and hardware support** The TRACE32 controller based software (like t32win.exe, t32cde, t32qt, ... + mcpm*.t32) doesn't support PodbusEthernetController and ICD debug modul usage anymore. Please use in this case host based TRACE32 software executables (like t32marm.exe, t32marm, t32marm-qt, ...) instead.
Build 105499 IMP 02555 ICD In-Circuit Debugger FLASH.Create (Declare FLASH) FLASH.CFI (Generate FLASH declaration by CFI) FLASH.CHANGEtype (Changes the FLASH type) STOre (Store settings as PRACTICE script)	**new option /DISableBulkErase added** For commands FLASH.Create and FLASH.CHANGEtype option /DISableBulkErase added. Now it is also possible to declare it explicitly for non-CFI flash devices to avoid unintended chip erase for truncated flash address range.
Build 7181 IMP 01995 Host Driver Software ICD In-Circuit Debugger	**ethernet interface** new firmware version V6.9 to avoid port remapping problems in NAT configuration
Build 1161 CHG 01639 ICD In-Circuit Debugger Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Programming of complex triggerunit** declarationtype synonymes introduced EVCNT for EVENTCOUNTER TICNT for TIMECOUNTER
Build 1096 FIX 01437 ICD In-Circuit Debugger Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Trigger Programming** ANAICD triggerprogramming: program crash fixed

Build 13751 IMP 02091 ICD Solutions for ARM CAnalyzer (Trace features of Compact Analyzer) SystemTrace (MIPI STP and CoreSight ITM)	**CORTEX-M3 Trace Support** Support for Cortex-M3 tracing via SWV and TPIU for ETM and ITM added.
Build 13751 IMP 02088 ICD Solutions for ARM STM (Configure STM component on target)	**STM** New command STM to support configuration of STM and SDTI trace ports.
Build 13751 IMP 02085 ICD Solutions for ARM Break.Set (Set breakpoints) Var.Break.Set (Set breakpoint to HLL expression)	**ARM ETM Breakpoints Extended** ETM Breakpoints now support also combined program/data breakpoints and breakpoints with condition pass/fail.
Build 2940 IMP 01918 ICD Solutions for ARM	**TLB DUMP functions implemented for ARM11** Allows to dump TLB contents during debug.
Build 1258 FIX 01852 ICD Solutions for ARM TRACE32 Instruction Set Simulator for CORTEX / ARM / XSCALE TERM.GATE (Terminal with virtual hosting)	**ARM semihosting file append not working** Append to existing files (e.g. fopen with "a+" mode) was not working as expected.
Build 1161 IMP 01620 FIRE Emulator for ARM7 ICD Solutions for ARM TRACE32 Instruction Set Simulator for CORTEX / ARM / XSCALE	**Access modes** additional access mode ETB:

ICD Solutions for PowerPC and Power Architecture

Build 1161 IMP 01621 FIRE Emulator MPC8XX ICD Solutions for PowerPC and Power Architecture SIM Instruction Set Simulator for PowerPC	**Access modes** additional access mode DBG: and EDBG:

ICD Solutions for PowerPC400

Build 1241 FIX 01781 ICD Solutions for PowerPC400 <trace>.THreshold (Optimize threshold for trace lines)	**Threshold voltage set wrong on PPC4xx** The analyzer window settings where not set correctly. Threshold may be wrong.

ICD Solutions for TriCore

Build 1161 CHG 01627 FIRE Emulator for TriCore ICD Solutions for TriCore SIM Instruction Set Simulator for TriCore	**Access modes** access modes BBB: and EBBB: renamed to EEC: and EEEC:
Build 1161 IMP 01623 FIRE Emulator for TriCore ICD Solutions for TriCore SIM Instruction Set Simulator for TriCore	**Access modes** additional access modes BBB: and EBBB:

Build 125398 CHG 02579 In-Circuit Emulator Host Driver Software ICE In-Circuit Emulator Integrated Development Environment TRACE32 Software	**restricted TRACE32 ICE software development** This is the last TRACE32 software release which contains ICE software derived from the standard TRACE32 development trunk. A separate branch from the TRACE32 software will be created for supporting ICE hardware only. Only bugfixes and host OS adaptations are planned for the future.
Build 2940 IMP 01911 JTAG Debugger FIRE Fully Integrated RISC Emulator ICE In-Circuit Emulator PowerTrace for NEXUS TRACE32 Instruction Set Simulators	**Practice functions** synonym D.SUM() for DATA.SUM() added
Build 1264 IMP 01861 ICE In-Circuit Emulator Go.BackEntry (Go back in program to function entry (CTS)) Step.BackOver (Step back)	**new commands for CTS introduced** Go.BackEntry Go.BackTillViolation Go.TillViolation Step.BackOver
Build 1246 CHG 01834 ICE In-Circuit Emulator PowerTrace for NEXUS Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**programming of complex triggerunit** commands renamed: Flag.OFF -> Flag.FALSE Flag.ON -> Flag.TRUE
Build 1246 CHG 01831 ICE In-Circuit Emulator Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**programming of analyzer triggerunit** keyword for addressevent definition ADDR replaced by ADDRESS
Build 1246 CHG 01813 JTAG Debugger FIRE Fully Integrated RISC Emulator ICE In-Circuit Emulator ROM Monitor	**Address range expressions** Behavior of general parameter parser changed The checking of accessmode equality inside address range expressions like address1--address2 is switched off now. Now it's allowed to use different accessmodes for begin and end address of address ranges. The accessmode of the end address will be ignored and will not cause an error message anymore. e.g. UP:1000--SD:2000 will be handled as UP:1000--UP:2000
Build 1190 FIX 01674 ICE In-Circuit Emulator STOre (Store settings as PRACTICE script)	**STORE Break BreakAll BreakANH** STORE <filename> BREAK, BREAKALL, BREAKANH Inside the generated file the old syntax was used for storing Break.SELect FOREGROUND and Break.SELect BACKGROUND (since 2003.12.15) and old syntax was locked.

Build 1161 CHG 01635 ICE In-Circuit Emulator Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**analyzer triggerunit programming** declarationtype synonymes introduced EVCNT for EVENTCOUNTER EXCNT for EXTERNCOUNTER TICNT for TIMECOUNTER
Build 1161 CHG 01622 ICE In-Circuit Emulator Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Analyzer trigger unit programming** command renames: Count.* -> Counter.* Count.Enable -> Counter.Increment declaration renames: EVENT -> EVENTCOUNTER EXTERN -> EXTERNCOUNTER TIME -> TIMECOUNTER
Build 1151 FIX 01611 ICE In-Circuit Emulator	old fashioned command B.B for B.BG was locked (since 15.12.2003)
Build 1151 CHG 01588 ICE In-Circuit Emulator Go.Back (Go back in program (CTS))	**commands renamed** Break.SELect FORE -> Break.SELect FOREGROUND Break.SELect BACK -> Break.SELect BACKGROUND Break.Back -> Break.BackGround Go.Back -> Go.BackGround Go.Back is used for CTS now
Build 1077 FIX 01153 FIRE Fully Integrated RISC Emulator ICE In-Circuit Emulator PowerIntegrator - Logic and Bus Analyzer PowerTrace PowerProbe - Logic and Protocol Analyzer <trace>.AutoInit (Automatic initialization)	Automatic init of analyzers (Integrator,PowerTrace,FIRE,ICE) changed

Build 125398 CHG 02596 In-Circuit Emulator FIRE Fully Integrated RISC Emulator Host Driver Software Integrated Development Environment TRACE32 Software	**restricted TRACE32 FIRE software development** This is the last TRACE32 software release which contains FIRE software derived from the standard TRACE32 development trunk. A separate branch from the TRACE32 software will be created for supporting FIRE hardware only. Only bugfixes and host OS adaptations are planned for the future.
Build 125398 CHG 02579 In-Circuit Emulator Host Driver Software ICE In-Circuit Emulator Integrated Development Environment TRACE32 Software	**restricted TRACE32 ICE software development** This is the last TRACE32 software release which contains ICE software derived from the standard TRACE32 development trunk. A separate branch from the TRACE32 software will be created for supporting ICE hardware only. Only bugfixes and host OS adaptations are planned for the future.

In-Circuit Emulator for 386

Build 19417 CHG 02162 In-Circuit Emulator for 386 MMU.view (View MMU registers)	**80386 MMU window changed** Attribute registers shown as 16bit values now.

In-Circuit Emulator for 68HC05 and 68HC08

Build 4728 CHG 01953 S08 Debugger FIRE Emulator for Freescale HC08 / HCS08 In-Circuit Emulator for 68HC05 and 68HC08	**accessmodes** additional access modes: DBG: EDBG: INI:

Build 1206 CHG 01730 S12 Debugger FIRE Emulator for Freescale 68HC12 / MCS12 / S12X In-Circuit Emulator for 68HC11 In-Circuit Emulator for 68HC12 SIM Instruction Set Simulator for HC12/MCS12	**accessmodes** additional access modes X: XD: XP: AX: AXD: AXP: EX: EXD: EXP: EAX: EAXD: EAXP:
Build 1204 CHG 01724 S12 Debugger FIRE Emulator for Freescale 68HC12 / MCS12 / S12X In-Circuit Emulator for 68HC11 In-Circuit Emulator for 68HC12 SIM Instruction Set Simulator for HC12/MCS12	**access modes** additional access modes AG: EAG: GD: AGD: EGD: GP: AGP: EGP:
Build 1198 CHG 01713 S12 Debugger FIRE Emulator for Freescale 68HC12 / MCS12 / S12X In-Circuit Emulator for 68HC11 In-Circuit Emulator for 68HC12 SIM Instruction Set Simulator for HC12/MCS12	**access modes** additional access modes G: EG:
Build 1077 FIX 01147 In-Circuit Emulator for 68HC11	**dualported RAM** ICE 68HC11: flag information of full dualported RAM may be wrong

Build 1206 CHG 01730 S12 Debugger FIRE Emulator for Freescale 68HC12 / MCS12 / S12X In-Circuit Emulator for 68HC11 In-Circuit Emulator for 68HC12 SIM Instruction Set Simulator for HC12/MCS12	**accessmodes** additional access modes X: XD: XP: AX: AXD: AXP: EX: EXD: EXP: EAX: EAXD: EAXP:
Build 1204 CHG 01724 S12 Debugger FIRE Emulator for Freescale 68HC12 / MCS12 / S12X In-Circuit Emulator for 68HC11 In-Circuit Emulator for 68HC12 SIM Instruction Set Simulator for HC12/MCS12	**access modes** additional access modes AG: EAG: GD: AGD: EGD: GP: AGP: EGP:
Build 1198 CHG 01713 S12 Debugger FIRE Emulator for Freescale 68HC12 / MCS12 / S12X In-Circuit Emulator for 68HC11 In-Circuit Emulator for 68HC12 SIM Instruction Set Simulator for HC12/MCS12	**access modes** additional access modes G: EG:

In-Circuit Emulator for 8051

Build 13751 IMP 02093 XC800 Debugger In-Circuit Emulator for 8051	**OMF2 format supported**

In-Circuit Emulator for ARM7

Build 1190 IMP 01684 ARM Debugger ARM9 JTAG Debugger FIRE Emulator for ARM7 In-Circuit Emulator for ARM7 TRACE32 Instruction Set Simulator for CORTEX / ARM / XSCALE	**new accessmodes for ARM family** EC0:..EC15: EDEBG: EETB: EETM: EICE: EINI: added
Build 1077 FIX 01090 ARM Debugger FIRE Emulator for ARM7 In-Circuit Emulator for ARM7	**Disassembler** condition code in disassembler was missing on some instructions for ARM

In-Circuit Emulator for C166/ST10 - Out of Production

Build 1203 FIX 01720 In-Circuit Emulator for C166/ST10 - Out of Production Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**analyzer trigger programming** address events OD, ODX, ODL, ODLX, ODLH: programming of breakpoints partly wrong
Build 1190 IMP 01683 C166CBC Debugger Super10 Debugger FIRE Emulator for C166 Family In-Circuit Emulator for C166/ST10 - Out of Production SIM Instruction Set Simulator for C166/ST10	**new access mode for 80166 family** EDBG: added
Build 1077 FIX 01258 C166CBC Debugger FIRE Emulator for C166 Family In-Circuit Emulator for C166/ST10 - Out of Production C166 Monitor Var.Call (Call a new procedure)	bugs fixed in Var.Call for C166 architecture
Build 1077 IMP 01244 FIRE Emulator for C166 Family FIRE Emulator for ST10 In-Circuit Emulator for C166/ST10 - Out of Production C166 Family Real-Time Trace	**CTS** handling of volatile memory and atomic instructions improved for CTS with C166
Build 1077 FIX 01232 C166CBC Debugger FIRE Emulator for C166 Family In-Circuit Emulator for C166/ST10 - Out of Production C166 Monitor	**Symbol comment** Symbol comment for C166 family direct addressing was wrong or missing

In-Circuit Emulator for MC68020/30

Build 1081 IMP 01405 In-Circuit Emulator for MC68020/30 SYStem.Mode (Select mode)	speed of sys.mode standby for ICE68020 improved

In-Circuit Emulator for Power Architecture

Build 1206 CHG 01735 MPC5xx/8xx Debugger FIRE Emulator MPC8XX In-Circuit Emulator for Power Architecture SIM Instruction Set Simulator for PowerPC	**accessmodes** additional access modes PMR: F: AF: EF: EAF: FP: AFP: EFP: EAFP: FD: AFD: EFD: EAFD: SF: ASF: ESF: EASF: UF: AUF: EUF: EAUF: V: AV: EV: EAV: VP: AVP: EVP: EAVP: VD: AVD: EVD: EAVD: SV: ASV: ESV: EASV: UV: AUV: EUV: EAUV:

In-Circuit Emulator for the 80186XL

Build 7181 FIX 02006 In-Circuit Emulator for the 80186XL Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Analyzer trigger unit programming** DATA event declaration: the modes B0, B1, W0 were locked depending on the CPU derivate (since build 4789)
Build 1077 FIX 01357 In-Circuit Emulator for the 80186XL x186 Monitor	**MMU** Handling of MMU translation for banked 80186 targets fixed

Build 69655 IMP 02438 Installation Guide SETUP.PDFViewer (Context-sensitive help via your favorite PDF viewer)	**New command SETUP.PDFViewer** The new command SETUP.PDFViewer allows to configure a PDF viewer for the TRACE32 manuals. Adobe Acrobat Reader stays the default viewer.
Build 13751 FIX 02075 Installation Guide	**Silent mode of TRACE32 installer** Due to an error inside the TRACE32 installer software from InstallShield the silent mode didn't work correctly. Please don't use the default path and filename C:\windows\setup.iss!

Build 130863 IMP 02611 Integrated Development Environment PRinTer.FILE (Re-route printer output to a file in specified file format) WinPrint (Print address or record range of a window)	**new PRinTer file format PCL** The PCL file format has been added to the PRinTer.FILE command.
Build 125398 CHG 02597 JTAG Debugger Debugger for 80C196EA MMDSP Debugger Host Driver Software Integrated Development Environment x186 Monitor x196 Monitor XA51 Monitor TRACE32 Software	**restricted TRACE32 ICD software development** This is the last TRACE32 software release which contains for certain ICD/debugger software derived from the standard TRACE32 development trunk. A separate branch from the TRACE32 software will be created for supporting legacy ICD/debugger hardware only. Only bugfixes and host OS adaptations are planned for the future. effected CPU architectures: XA51, h85, i196, 186, MMDSP+
Build 125398 CHG 02579 In-Circuit Emulator Host Driver Software ICE In-Circuit Emulator Integrated Development Environment TRACE32 Software	**restricted TRACE32 ICE software development** This is the last TRACE32 software release which contains ICE software derived from the standard TRACE32 development trunk. A separate branch from the TRACE32 software will be created for supporting ICE hardware only. Only bugfixes and host OS adaptations are planned for the future.
Build 117056 IMP 02576 Integrated Development Environment List.auto (Display program listing)	**new mouse feature** New functionality on middle mouse button for List and Var windows: single click performs an add to watch window and double click a view in a new window.
Build 112182 IMP 02565 Integrated Development Environment STOre (Store settings as PRACTICE script)	**new option AREA added** Option AREA added to command STOre to save the actual AREA settings to a PRACTICE script.

Build 105499 IMP 02556 Integrated Development Environment DIALOG.Program (Interactive programming) EDIT (TRACE32 editor) MENU.Program (Interactive programming) PEDIT (Edit PRACTICE script) Analyzer.Program (Program trigger unit) Data.PROGRAM (Editor for writing assembler program) Integrator.Program (Program trigger unit) Probe.Program (Program trigger unit) PER.Program (Interactive programming)	**new internal text editor** New text editor version with syntax highlighting, nesting awareness, undo/redo and other features activated.
Build 105499 IMP 02554 Integrated Development Environment TargetSystem.NewInstance (Start new TRACE32 PowerView instance)	**TargetSystem.NewInstance supports new port types** TargetSystem.NewInstance supports GDB, TCF port as well as multiple API ports now. Added new options: /API<n>.PORT <port>, /GDB.PORT <port>, /GDB.PPROTocolTCP|UDP, /TCF.PORT <port>
Build 100486 CHG 02545 Integrated Development Environment ChDir (Change directory)	**The prefix ChDir. can now be used for all commands** The ChDir. command prefix can now be used for all commands. Previously it was limited to IDE commands.
Build 100486 IMP 02543 Integrated Development Environment SETUP.DropCoMmanD (Set command for files dropped into command line)	**Change default for file drop to command line** SETUP.DropCoMmanD allows changing the default command which is used when a file of certain type is dropped into the TRACE32 command line.
Build 100486 CHG 02528 Integrated Development Environment DO (Start PRACTICE script)	**autostart.cmm replaces t32.cmm** autostart.cmm is located in the system directory and always executed when TRACE32 instance (e.g. t32marm.exe) is started. If no autostart.cmm exists, TRACE32 falls back to the outdated behavior: t32.cmm is executed from either the working or the system directory, but only if not start-up script was explicitly specified as command line option. For details refer to "Automatic Start-up Scripts" in practice_user.pdf

Build 100486 IMP 02526 Integrated Development Environment CmdPOS (Controls the position of TRACE32 in MWI window mode) FramePOS (Controls the position of TRACE32 in MDI window mode) WinPOS (Define window dimensions and window name)	**Improved support for multimonitor desktops under MS Windows** The maximum possible window creation size was enlarged to cover the full multimonitor desktop size under MS Windows.
Build 93173 IMP 02514 Integrated Development Environment <trace>.DRAW.channel (Plot no-data values against time) <trace>.DRAW.Data (Plot data values against time) <trace>.DRAW.Var (Plot variable values against time)	**Mouse interface changed for Trace.Draw command group** Zooming into the selected area is now done via double-click with the left mouse button. Free selection is now possible in the draw area. Value range or time/record range selection is possible via the scale area. Drag and drop of the selected area to another trace window with graphical display aligns this window to the same area.
Build 93173 IMP 02512 Integrated Development Environment Frame.view (Display stack frame) List.auto (Display program listing) Var.View (Display variables) Var.Watch (Open Var.Watch window)	**Mouse interface changed for all text windows** Free text selection is now possible with the left mouse. Click and drag on a selection allows drag and drop.
Build 93173 IMP 02508 Integrated Development Environment TAR (Pack files into an archive)	**Pack files into an archive** Packs the specified files without compression into a tape archive.
Build 80996 IMP 02482 Integrated Development Environment UNARchive (Linux and Microsoft libraries)	**Extract Linux/Microsoft libraries to directory** A new command UNARchive is available to extract files from Linux libraries (.a) and Microsoft libraries (.lib) to a directory. The two additional subcommands UNARchive.Table and UNARchive.Show help to determine the contents of the archive and to check the result of the unarchive operation.
Build 76594 IMP 02454 Integrated Development Environment PRinTer.EXPORT (Export formatted printer output to file) PRinTer.FILE (Re-route printer output to a file in specified file format)	**Behavior of PRinTer dialog changed** The file extension matches the output format now. As part of this change, the commands PRinTer.EXPORT and PRinTer.FILE have been updated. PRinTer.EXPORT additionally allows to export the output in ASCII and XHTML format. Now PRinTer.FILE allows to set the file name, the output format and the append parameter.

Build 60219 IMP 02420 Integrated Development Environment	**XML format supported for WinPrint** The results of TRACE32 commands can now be saved to a file in XML format.
Build 56057 IMP 02413 Integrated Development Environment HELP (Online help)	**support of additional PDF viewer types added** The TRACE32 online help system officially supports the usage of alternative PDF viewer different from Adobe Reader. More details are described inside the IDE User's Guide.
Build 13751 IMP 02082 Integrated Development Environment	**new IDE functions** the following PRACTICE functions were added: CLOCK.DAY(), CLOCK.MONTH(), CLOCK.YEAR(), OS.PCF(), OS.PHELPD(), OS.PLF(), OS.PPD(), OS.PPF()
Build 13751 ODD 02064 Integrated Development Environment Eval (Evaluate expression)	**EVAL EVAL() EVAL.TYPE()** behavior of command EVAL changed: now an empty expression parameter sets value to 0 and value type to EMPTY instead of using the old values PRACTICE functions EVAL() and EVAL.TYPE(): expression type EMPTY added

Intel~ Processor Trace

Build 100486 IMP 02523 Hypervisor-aware Debugging ARM/Cortex Trace (parallel) Intel~ Processor Trace QorIQ PowerPC NEXUS Aurora Trace Port <trace>.List (List trace contents)	**Decoding of machine id supported** The machine ID within the Context-ID or ownership packet is now decoded.

Build 88288 IMP 02498 ARM Debugger Cortex-A/-R Armv8 Debugger Intel~ x86/x64 JTAG Debugger SYStem.Option MACHINESPACES (Address extension for guest OSes) SYStem.Option MACHINESPACES (Address extension for guest OSes) SYStem.Option MACHINESPACES (Address extension for guest OSes)	**Hypervisor-aware debugging feature introduced (Part 1-2)** The TRACE32 machinespaces framework was added for architectures ARM and Intel x86 to support Hypervisor environments. This framework allows the debugger to perform stop-mode debugging of targets running a hypervisor and one or multiple guest machines. In conjunction with a hypervisor awareness it is possible to debug all guest machines concurrently, whether they are currently active or currently inactive. TRACE32 allows to load an individual OS awareness for each virtual machine on the target. The machinespaces framework is enabled with the command SYStem.Option.MACHINESPACES ON. A new identifier <machineid> is introduced which allows to identify virtual machines on the target. TRACE32 addresses are augmented with the machineid to specify which machine the address belongs to. The TRACE32 debugger address translation (enabled with TRANSlation.ON) is extended to fully support the two-stage address translation which is usually applied for hardware virtualized guest machines. The first translation stage translates guest virtual addresses to intermediate physical addresses. The second translation stage translates intermediate physical addresses to absolute physical addresses. The MMU registers needed for the page table walk steps are automatically fetched from the hardware registers or, for inactive guests, from hypervisor data structures. If SYStem.Option.MACHINESPACES is enabled, - addresses are extended with a machine ID identifier, separating the machine ID with three colons from the (extended) virtual address: <accessclass>:<machineid>:::<spaceid>::<virtaddress> The machine ID clearly specifies which host or guest machine the address belongs to. The host machine always uses machineid 0. Guests have a machineid larger than 0. TRACE32 currently supports up to 30 machines.
Build 76594 CHG 02458 Intel~ x86/x64 JTAG Debugger	**Renaming of segment register aliases** To avoid confusion with access classes ES:, ESR:, GS: and GSR: all six segment selector address prefixes have been renamed from CS:/DS:/ESR:/FS:/GS:/SS: to CSS:/DSS:/ESS:/FSS:/GSS:/SSS:.
Build 76594 IMP 02450 Intel~ x86/x64 JTAG Debugger	**Keep protection level bits in PC segment** In protected mode (large memory model), TRACE32 now preserves the protection level bits from the segment descriptors (bits RPL = bits [1:0] of segment descriptor). This prevents that the protection level is lost if the PC is modified in debug mode, which may lead in consequence to a privilege violation in the application.
Build 60219 CHG 02421 Intel~ x86/x64 JTAG Debugger High-Level-Language Debugging Data.LOAD.Elf (Load ELF file)	**Logical memory access for Data.LOAD.Elf** Data.LOAD.Elf use logical memory access when load is not done to a physical address (x86/x64)
Build 60219 IMP 02419 Intel~ x86/x64 JTAG Debugger RTOS Debugger for Windows	**WinSTD Awareness for x86/x64** Task-aware debugging for Win7/Win8

Build 51144 ODD 02384 Intel~ x86/x64 JTAG Debugger SYStem.Option MEMory-MODEL (Define memory model)	**LARGE memory model for Intel x86** Default memory model for Intel x86 is LARGE now.
Build 38589 IMP 02318 Intel~ x86/x64 JTAG Debugger BMC.state (Display BMC configuration window) SNOOPer (Sample-based trace)	**Intel StopAndGo Snooping** For Intel Atom/x86 CPUs it is possible to snoop PC and BMC using StopAndGo. The snoop results can be used for profiling, charting and statistics.
Build 34458 IMP 02264 Intel~ x86/x64 JTAG Debugger	**Add option "/core" to MMU window for x86**

Build 125398 CHG 02597 JTAG Debugger Debugger for 80C196EA MMDSP Debugger Host Driver Software Integrated Development Environment x186 Monitor x196 Monitor XA51 Monitor TRACE32 Software	**restricted TRACE32 ICD software development** This is the last TRACE32 software release which contains for certain ICD/debugger software derived from the standard TRACE32 development trunk. A separate branch from the TRACE32 software will be created for supporting legacy ICD/debugger hardware only. Only bugfixes and host OS adaptations are planned for the future. effected CPU architectures: XA51, h85, i196, 186, MMDSP+
Build 125398 CHG 02592 JTAG Debugger FLASH.SPI.CMD (Send data to SPI FLASH device) FLASHFILE.SPI.CMD (Send data to SPI FLASH device)	**renamed commands FLASH.SPICMD and FLASHFILE.SPICMD** The commands FLASH.SPICMD and FLASHFILE.SPICMD have been renamed to FLASH.SPI.CMD and FLASHFILE.SPI.CMD.
Build 125398 IMP 02588 JTAG Debugger FLASH.SPI.GETSFDP (Read FLASH discovery parameters) FLASHFILE.SPI.GETSFDP (Read FLASH discovery parameters)	**new commands FLASH.SPI.GETSFDP and FLASHFILE.SPI.GETSFDP** New commands introduced to read Serial Flash Discovery Parameters from SPI FLASH. The several parameter values can be utilized afterwards using the FLASH.SPI.SFDP() and FLASHFILE.SPI.SFDP() functions.
Build 125398 IMP 02585 JTAG Debugger FLASH.SPI.CFI (Generate SPI FLASH sector declaration by CFI) FLASHFILE.SPI.CFI (Generate SPI FLASH sector declaration by CFI)	**new commands FLASH.SPI.CFI and FLASHFILE.SPI.CFI** New commands introduced to generate the FLASH declaration by using CFI information.
Build 125398 CHG 02580 JTAG Debugger PODBUS Ethernet Controller ICD In-Circuit Debugger TRACE32 Software	**discontinued software and hardware support** The TRACE32 controller based software (like t32win.exe, t32cde, t32qt, ... + mcpm*.t32) doesn't support PodbusEthernetController and ICD debug modul usage anymore. Please use in this case host based TRACE32 software executables (like t32marm.exe, t32marm, t32marm-qt, ...) instead.
Build 117056 IMP 02575 JTAG Debugger Data.VECTOR (Display memory as vectors)	**new command Data.VECTOR** The new Data.VECTOR command allows to display memory contents as vectors side by side.

Build 112182 CHG 02571 JTAG Debugger Break.CONFIG.InexactAddress Break.CONFIG.MatchASID Break.CONFIG.MatchMachine Break.CONFIG.MatchZone Break.CONFIG.UseContextID Break.CONFIG.UseMachineID Break.CONFIG.VarConvert	**trigger onchip command group and dialog changed** The trigger onchip dialog and the command parser was changed for all architectures, because commands were renamed and moved to the global Break.CONFIG.state dialog and parser. The Break.CONFIG.state dialog was adapted to show the new settings made by the commands. New Break.CONFIG commands were introduced. The following list gives an overview of the deprecated TrOnchip commands which were replaced by new Break.CONFIG commands: TrOnchip.ContextID was renamed to Break.CONFIG.UseContextID, TrOnchip.CONVert was renamed to Break.CONFIG.InexactAddress, TrOnchip.CONVert0cycle was renamed to Break.CONFIG.InexactAddress0cycle, TrOnchip.MachineID was renamed to Break.CONFIG.UseMachineID, TrOnchip.MatchASID was renamed to Break.CONFIG.MatchASID, TrOnchip.MatchMachine was renamed to Break.CONFIG.MatchMachine, TrOnchip.MatchZone was renamed to Break.CONFIG.MatchZone, TrOnchip.VarCONVert was renamed to Break.CONFIG.VarConvert. The new commands introduced for selected architectures are Break.InexactData, Break.InexactTrigger and Break.InexactResume.
Build 112182 IMP 02569 JTAG Debugger JTAG.XUSEFUSE (Program Xilinx UltraScale eFUSEs)	**new command JTAG.XUSEFUSE added** new command introduced to burn eFuses of Xilinx UltraScale FPGAs
Build 22490 IMP 02214 JTAG Debugger	**New command SYStem.DETECT.JtagClock**
Build 2940 IMP 01911 JTAG Debugger FIRE Fully Integrated RISC Emulator ICE In-Circuit Emulator PowerTrace for NEXUS TRACE32 Instruction Set Simulators	**Practice functions** synonym D.SUM() for DATA.SUM() added
Build 1246 CHG 01813 JTAG Debugger FIRE Fully Integrated RISC Emulator ICE In-Circuit Emulator ROM Monitor	**Address range expressions** Behavior of general parameter parser changed The checking of accessmode equality inside address range expressions like address1--address2 is switched off now. Now it's allowed to use different accessmodes for begin and end address of address ranges. The accessmode of the end address will be ignored and will not cause an error message anymore. e.g. UP:1000--SD:2000 will be handled as UP:1000--UP:2000
Build 1151 FIX 01595 JTAG Debugger	JTAG sequence improved to reach 20MHz from the beginning
Build 1145 FIX 01568 JTAG Debugger PowerTrace	PowerDebug/Ethernet was responding to ping commands on multicast addresses

Build 1096 FIX 01452 JTAG Debugger	Problem fixed in JTAG PLD for Super10, Tricore, 166SV2, 166CBC, H8S, PPC400, MCORE, DSP56XXX, may cause instable JTAG connection
Build 1077 IMP 01237 JTAG Debugger PowerTrace	**Ethernet** Lost packets on PowerDebug Ethernet handled better
Build 1077 IMP 01179 JTAG Debugger	**Multiple serial numbers** Supporting multiple serial numbers in one debug cable

Logical Display of Peripherals

Build 117056 ODD 02578 Logical Display of Peripherals Script Language PRACTICE	**PRACTICE and PER macro naming rules changed** Macro names must not start with a numerical digit anymore.
Build 100486 IMP 02530 Logical Display of Peripherals AUTOINDENT (Indent content of peripheral file automatically)	**Automatic indentation for PER files** The AUTOIDENT command allows activating various indentation styles.
Build 80996 IMP 02481 Logical Display of Peripherals PER.STOre (Generate PRACTICE script from PER settings)	**Generate PRACTICE script for PER settings** New command PER.STOre and extended local pulldown menu in PER.View window to generated script/command sequence for current PER settings.

M-Core Debugger

Build 1228 FIX 01766 M-Core Debugger SYStem (System configuration)	**SYStem.JtagClock was missing in window** Executing the command from the commandline is possible.
Build 1206 CHG 01740 M-Core Debugger M-Core NEXUS Debugger and Trace	**accessmodes** additional access modes DBG: EDBG:

M-Core NEXUS Debugger and Trace

Build 1206 CHG 01740 M-Core Debugger M-Core NEXUS Debugger and Trace	**accessmodes** additional access modes DBG: EDBG:

MAC71xx/72xx NEXUS Debugger and Trace

Build 1246 FIX 01835 MAC71xx/72xx NEXUS Debugger and Trace <trace>.List (List trace contents)	**Traced flow incorrect on MAC7xxx** One instruction too much displayed before interrupt entry.

MIPS32 Trace

Build 7181 CHG 02008 MIPS32/MIPS64 Debugger MIPS32 Trace SIM Instruction Set Simulator for MIPS SIM Instruction Set Simulator for MIPS5K	**accessmodes** additional access modes: ADC: AIC: ANC:

Build 19417 IMP 02160 ARM Debugger MIPS32/MIPS64 Debugger MPC5xx/8xx Debugger Data.LOAD (Load file)	**New DIFFLOAD options and targets** New options allow further compression or compressed load (/ZIPLOAD) and quick download checks (/CHECKLOAD and /CHECKONLY). Target agents for MIPS32 and MIPS64 architectures added.
Build 19417 IMP 02127 MIPS32/MIPS64 Debugger	**SPR overlay handling** Implement read and write access to instruction Scratch Pad RAM (only for ksegs).
Build 7181 CHG 02008 MIPS32/MIPS64 Debugger MIPS32 Trace SIM Instruction Set Simulator for MIPS SIM Instruction Set Simulator for MIPS5K	**accessmodes** additional access modes: ADC: AIC: ANC:
Build 1228 CHG 01779 MIPS32/MIPS64 Debugger	**accessmodes** additional access modes CBU: ECBU:
Build 1228 FIX 01771 MIPS32/MIPS64 Debugger StarCore Debugger	**Some ICDs fail on USB2** MIPS5k and StarCore fail at system.up or single step.
Build 1224 FIX 01760 MIPS32/MIPS64 Debugger PERF (Sample-based profiling)	**Performance Analysis for BCMxx corrected** Broadcom specific DERET handling in BDM_GetPC added. Changes necessary for functionality of performance analysis.
Build 1151 IMP 01612 MIPS32/MIPS64 Debugger SH Debugger Data.LOAD (Load file)	**DWARF1** Changes made to DWARF1 loader for Hitachi-SH compiler and Greenhills-MIPS
Build 1077 FIX 01100 MIPS32/MIPS64 Debugger	**FlowError with Mips Preprocessor** SETUP.IMASKASM / SETUP.IMASKHLL not working on MIPS64
Build 1077 FIX 01092 MIPS32/MIPS64 Debugger	Register.RESet not working on MIPS32/MIPS64
Build 1077 FIX 01078 MIPS32/MIPS64 Debugger	**IEEE** MIPS5K: 32bit addresses of IEEE files where not sign-extended

MMDSP Debugger

Build 125398 CHG 02597 JTAG Debugger Debugger for 80C196EA MMDSP Debugger Host Driver Software Integrated Development Environment x186 Monitor x196 Monitor XA51 Monitor TRACE32 Software	**restricted TRACE32 ICD software development** This is the last TRACE32 software release which contains for certain ICD/debugger software derived from the standard TRACE32 development trunk. A separate branch from the TRACE32 software will be created for supporting legacy ICD/debugger hardware only. Only bugfixes and host OS adaptations are planned for the future. effected CPU architectures: XA51, h85, i196, 186, MMDSP+
Build 1206 CHG 01728 MMDSP Debugger	**access modes** deleted accessmodes: P2: Y2: EP2: EY2: AP2: AY2: EAP2: EAY2: additional access modes : DC: IC: L2: NC:

MMU Support

Build 88288 IMP 02501 MMU Support SYStem.LOG.List (Log the accesses made by TRACE32) SYStem.LOG.Set (Select the TRACE32 accesses to be logged)	**SYStem.LOG.List now shows address translation failures** SYStem.LOG.List window now shows address translation failures as TRANS errors. If SYStem.LOG.Set TRANSlation is ON all address translation calls are shown in the SYStem.LOG.List window (logical addresses only).
Build 65657 02437 MMU Support TRANSlation.COMMON (Common address ranges for kernel and tasks)	**No logical operators in TRANSlation.COMMON command** Old way of specifiying multiple COMMON address ranges with OR operation has been removed (TRANS.COMMON 0x80000000--0x8FFFFFFF||0xA0000000--0xBFFFFFFF). Please use spaces or the command TRANSlation.COMMON.ADD to specify additional address ranges.

Build 4728 CHG 01904 MPC5200 Debugger PowerQUICC II/Pro Debugger SYStem.Option (Special setup)	**new SYStem.Option.IP.BOTH** new MSR[IP] handling for Onchip-BP's on processor with two Onchip-BP's. (G2_LE /ZEPPO core) Default is SYS.O.IP.AUTO where only the Onchip-BP control the active (MSR[IP]) exception handler block.
Build 1077 IMP 01346 MPC5200 Debugger PowerQUICC II/Pro Debugger	MPC82XX, MGT5100/5200 ICache decoder added
Build 1077 IMP 01298 MPC5200 Debugger	**Boot Low/High Auto detection** MGT5100, 5200, MPC8280: Boot Low/High Auto detection
Build 1077 FIX 01294 MPC5200 Debugger PowerQUICC II/Pro Debugger	**SystemUp failed when Boot** MGT5200/MPC8280: SystemUp failed when Boot from 0x00000100
Build 1077 FIX 01283 MPC5200 Debugger	**Fixed single step over rfci instructions** MGT5100/5200 Fixed single step over rfci instructions

Build 100486 CHG 02522 ARM/Cortex Trace (parallel) Intel~ Processor Trace MPC57xx NEXUS High Speed Serial Trace Port QorIQ PowerPC NEXUS Aurora Trace Port MPC55xx/MPC56xx NEXUS Debugger and Trace <trace>.List (List trace contents)	**Cycle type for Context-ID and Ownership packets changed** If a Context-ID or ownership packet is decoded and if it is assignable to a task, the "task" cycle type and the task name is displayed. The displayed data value is not longer the packet content, but a TRACE32 internal value. If a Context-ID or ownership packet is decoded and if it can not be assigned to a task or any other protocol-specific content, the cycle type "traceid" and the packet content is displayed.
Build 80996 IMP 02483 MPC57xx NEXUS High Speed Serial Trace Port QorIQ PowerPC NEXUS Aurora Trace Port MPC55xx/MPC56xx NEXUS Debugger and Trace RTS.state (Open status and control window)	**RTS implemented for PowerPC (all E200zx cores) and QorIQ** RTS implemented for PowerPC (all E200zx cores) and QorIQ.
Build 65657 02435 ARM/Cortex Trace (parallel) MPC55xx/MPC56xx NEXUS Debugger and Trace <trace>.List (List trace contents) <trace>.STATistic.PAddress (Which instructions accessed data address) <trace>.STATistic.PsYmbol (Shows which functions accessed data address)	**Prestore address information for ARM-ETM/NEXUS** The columns PAddress/PsYmbol in the trace listing show the address of the instruction that was executed before a read/write access was performed. The statistic function Trace.STATistic.PsYmbol analyses which functions performed read/write accesses. The statistic function Trace.STATistic.PAddress analyses which assembler instruction performed the read/write accesses.
Build 42354 IMP 02354 MPC5xxx and SPC5xx Debugger MPC57xx NEXUS High Speed Serial Trace Port MPC55xx/MPC56xx NEXUS Debugger and Trace	**depletion recovery for affected on-chip flash modules added**
Build 22490 IMP 02189 MPC5xxx and SPC5xx Debugger MPC55xx/MPC56xx NEXUS Debugger and Trace FLASH.ReProgram (Re-program FLASH)	**MPC55XX/56XX** FLASH.ReProgram now supports programming Flash with ECC errors without prior FLASH.Erase call

Build 19417 CHG 02181 MPC5xxx and SPC5xx Debugger MPC55xx/MPC56xx NEXUS Debugger and Trace SYStem (System configuration) TrOnchip (Onchip triggers)	**SYStem.Up with CPU selection MPC55XX** The behavior of the MPC55XX (auto-selection type) has changed. When this type is selected, the debugger will reset all SYStem, TrOnchip and trace settings to default upon SYStem mode change. If settings are needed to be set before SYStem.Up, e.g. SYStem.Option.WATCHDOG, SYStem.Option.ETK, first select proper CPU type before, or use SYStem.DETect CPU before setting SYStem options.
Build 2940 FIX 01876 MPC55xx/MPC56xx NEXUS Debugger and Trace <trace>.List (List trace contents)	**eTPU trace may produce flow errors** Occurs when long traces from other cores are also in the trace or at certain slot instruction combinations during the end of a channel program.
Build 1241 IMP 01786 Adaption for Embedded Trace Macrocell Adaptation for MIPS MPC55xx/MPC56xx NEXUS Debugger and Trace <trace>.AutoFocus (Calibrate AUTOFOCUS preprocessor) <trace>.TestFocus (Test trace port recording)	**New commands to verify trace connection** The Analyzer.AutoTest command verifys the integrity of the trace connection. For adapters with adjustable threshold the command Analyzer.AutoFocus will also choose the threshold in the middle of the data eye.
Build 1206 CHG 01741 MPC55xx/MPC56xx NEXUS Debugger and Trace Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Programming of complex triggerunit** supports Copperhead now particularly supports DTM, WTM, OTME with unitnumbers

Build 19417 IMP 02166 MPC56x NEXUS Debugger and Trace CTS (Context tracking system (CTS))	**CTS on PowerPC extended to support FPU** CTS reconstructs now also the FPU registers.
Build 9640 FIX 02028 MPC56x NEXUS Debugger and Trace	**MPC565 MaskD:SIUMCR display error** 8-bit burst bit of the SIUMCR register was cleared after step/break. This will only take effect on the MPC565 MaskD!
Build 4728 FIX 01948 MPC56x NEXUS Debugger and Trace	**runtime counter overflow fixed**
Build 1196 IMP 01734 MPC56x NEXUS Debugger and Trace	**MPC56x 8BitBurst workaround f. RevD ext.** Additional MPC56x 8bit burst workaround for Rev.D for sys.m.standby.
Build 1196 IMP 01705 MPC56x NEXUS Debugger and Trace	**MPC56x 8bit burst workaround for RevD** MPC56x 8bit burst workaround for Rev.D
Build 1151 FIX 01609 MPC56x NEXUS Debugger and Trace	MPC500: 55x&56x bits in SCCR and PLPRCR of peripherie corrected
Build 1151 IMP 01597 MPC56x NEXUS Debugger and Trace	MPC56x Nexus relocated exception table addresses from trace automatically converted when sys.option vectors is set to the table address.
Build 1085 IMP 01422 MPC56x NEXUS Debugger and Trace	Selective program trace of single locations supported on MPC56x Nexus
Build 1084 IMP 01420 MPC56x NEXUS Debugger and Trace	SmartTrace for MPC56x Nexus improved to better deal with very long fifo full gaps
Build 1077 FIX 01374 MPC56x NEXUS Debugger and Trace Super10 NEXUS Debugger and Trace	A.VIEW may crash on Super10 Nexus or PPC Nexus
Build 1077 CHG 01347 MPC5xx/MPC8xx Trace MPC56x NEXUS Debugger and Trace	MPC55x/56x: DMBR/DBOR improved display.
Build 1077 IMP 01321 MPC56x NEXUS Debugger and Trace	Frequency Counter selection changed for PPC Nexus
Build 1077 FIX 01308 MPC56x NEXUS Debugger and Trace	PPC Nexus data breakpoint reason display fixed

Build 1077 FIX 01289 MPC56x NEXUS Debugger and Trace SYStem.Mode (Select mode)	Change of DTM setting for MPC5xx nexus was not affecting next sys.mode standby
Build 1077 FIX 01286 MPC56x NEXUS Debugger and Trace	**StandBy mode** PPC Nexus was not tracing in StandBy mode (trace switched to stack mode)
Build 1077 FIX 01280 MPC56x NEXUS Debugger and Trace	563xx: per Files. group address of PLL was wrong. PLL DSCR register missed. ESSIx hand over parameter was wrong.
Build 1077 FIX 01251 MPC5xx/8xx Debugger MPC56x NEXUS Debugger and Trace	**RSTCONF** MPC500: MEN55x/MEN56x update for RSTCONF programming. Old routine do not work if ISB base is != 0x0.
Build 1077 FIX 01172 MPC56x NEXUS Debugger and Trace	Bug fixed in data cycle alignment on PPC Nexus when many pipelined cycles span across multiple jumps
Build 1077 FIX 01170 MPC56x NEXUS Debugger and Trace	Assignment of halfword and byte pipelined memory accesses failed on PowerPC nexus
Build 1077 FIX 01165 Trace-based Debugging (CTS) MPC56x NEXUS Debugger and Trace SmartTrace	**cts** Several fixed in CTS/Smarttrace for PowerPC Nexus
Build 1077 FIX 01157 ARM Debugger MPC56x NEXUS Debugger and Trace Break.Set (Set breakpoints)	**B.Set dialog** B.Set dialog may crash on Sun for ARM or PPC-Nexus
Build 1077 IMP 01126 MPC56x NEXUS Debugger and Trace	**MPC56x Rev.D** MPC56x Rev.D supported
Build 1077 FIX 01124 MPC56x NEXUS Debugger and Trace	**ISYNC** ISYNC was not considered as an indirect branch for NexusPPC565
Build 1077 IMP 01104 MPC56x NEXUS Debugger and Trace	**High memory accesses** new option SYStem.Option HighMemory for Nexus-56x support high memory accesses

Build 100486 CHG 02522 ARM/Cortex Trace (parallel) Intel~ Processor Trace MPC57xx NEXUS High Speed Serial Trace Port QorIQ PowerPC NEXUS Aurora Trace Port MPC55xx/MPC56xx NEXUS Debugger and Trace <trace>.List (List trace contents)	**Cycle type for Context-ID and Ownership packets changed** If a Context-ID or ownership packet is decoded and if it is assignable to a task, the "task" cycle type and the task name is displayed. The displayed data value is not longer the packet content, but a TRACE32 internal value. If a Context-ID or ownership packet is decoded and if it can not be assigned to a task or any other protocol-specific content, the cycle type "traceid" and the packet content is displayed.
Build 80996 IMP 02483 MPC57xx NEXUS High Speed Serial Trace Port QorIQ PowerPC NEXUS Aurora Trace Port MPC55xx/MPC56xx NEXUS Debugger and Trace RTS.state (Open status and control window)	**RTS implemented for PowerPC (all E200zx cores) and QorIQ** RTS implemented for PowerPC (all E200zx cores) and QorIQ.
Build 42354 IMP 02354 MPC5xxx and SPC5xx Debugger MPC57xx NEXUS High Speed Serial Trace Port MPC55xx/MPC56xx NEXUS Debugger and Trace	**depletion recovery for affected on-chip flash modules added**
Build 42354 IMP 02352 MPC5xxx and SPC5xx Debugger MPC57xx NEXUS High Speed Serial Trace Port	**run-time performance monitor register access (SNOOPer) added**

Build 38589 FIX 02313 MPC5xx/8xx Debugger Var.set (Modify variable)	**MPC500-Nexus: dualport access** fix dualport access for Var.set command
Build 19417 IMP 02160 ARM Debugger MIPS32/MIPS64 Debugger MPC5xx/8xx Debugger Data.LOAD (Load file)	**New DIFFLOAD options and targets** New options allow further compression or compressed load (/ZIPLOAD) and quick download checks (/CHECKLOAD and /CHECKONLY). Target agents for MIPS32 and MIPS64 architectures added.
Build 13751 FIX 02077 MPC5xx/8xx Debugger	**Bus error on byte/word write access** Each write to byte or word wide accesses result in a bus error.
Build 9640 FIX 02029 MPC5xx/8xx Debugger	**MPC500/800 hang on old PowerDebugModules**
Build 9640 FIX 02027 MPC5xx/8xx Debugger SYStem.Option (Special setup)	**MPC500/800:SYStem.Option.SCRATCH** independent from the SYStem.Option.SCRATCH setting the AUTO mode was used.
Build 7181 CHG 02004 MPC5xx/8xx Debugger FPU.view (Display FPU registers)	**FPU support for PPC440EP** preliminary 'diag 3201 138 <RAM addr> command is replaced by a official TRACE32 command 'SYStem.Option.SCRATCH <RAM_addr>'
Build 4728 FIX 01926 MPC5xx/8xx Debugger	**runtime counter overflow fixed**
Build 1253 IMP 01840 MPC5xx/8xx Debugger	**SEC Lite Module 885/875 implemented** per885.per and per875.per will also support the "SEC Lite" module register
Build 1246 FIX 01806 MPC5xx/8xx Debugger	**Debugging in compressed mode may fail** The debugger may access wrong locations for stackframe analysis.
Build 1241 FIX 01802 MPC5xx/8xx Debugger SYStem.Mode (Select mode)	**Sys.M.NoDebug reliable deact. DebugMode** SYS.Mode.NoDebug will not work. Even though if sys.m.nodebug is selected and an external reset will occur. The reason is a permanent running DSCK clock signal.
Build 1206 CHG 01735 MPC5xx/8xx Debugger FIRE Emulator MPC8XX In-Circuit Emulator for Power Architecture SIM Instruction Set Simulator for PowerPC	**accessmodes** additional access modes PMR: F: AF: EF: EAF: FP: AFP: EFP: EAFP: FD: AFD: EFD: EAFD: SF: ASF: ESF: EASF: UF: AUF: EUF: EAUF: V: AV: EV: EAV: VP: AVP: EVP: EAVP: VD: AVD: EVD: EAVD: SV: ASV: ESV: EASV: UV: AUV: EUV: EAUV:
Build 1151 FIX 01610 MPC5xx/8xx Debugger	MPC500: 55x&56x bits in SCCR and PLPRCR of peripherie corrected

Build 1085 FIX 01426 DSP56K Debugger MPC5xx/8xx Debugger SH Debugger	missing files after software installation - install database corrected files are on the CD, but weren't not copied FIRE/ICD-C166/ST10: men7860t.men ICD-DSP56K : per56802.per per56807.per per56809.per per56367.per ICD-PPC : per8280.per ICD-SH2 : per7294.per
Build 1081 FIX 01406 MPC5xx/8xx Debugger	MPC8xx/MPC5xx small data breakpoint range bugfix was not working correctly
Build 1081 FIX 01403 MPC5xx/8xx Debugger	MPC8xx/MPC5xx combined /MemoryRead and /MemoryWrite breakpoint was not allowed
Build 1079 IMP 01390 MPC5xx/8xx Debugger	MPC8xx and MPC5xx flowtrace correlation of PTR cycles improved
Build 1077 FIX 01339 MPC5xx/8xx Debugger	MPC500/800: TrB trigger for AICD (Proj.97) and ICD (8254/8282) changed and tested. m?800f.tdf necessary.
Build 1077 IMP 01325 MPC5xx/8xx Debugger	Advanced breakpoints for MPC5xx/MPC8xx cover now all onchip capabilities
Build 1077 FIX 01251 MPC5xx/8xx Debugger MPC56x NEXUS Debugger and Trace	**RSTCONF** MPC500: MEN55x/MEN56x update for RSTCONF programming. Old routine do not work if ISB base is != 0x0.
Build 1077 FIX 01203 MPC5xx/8xx Debugger	PPC500/800 data write breakpoints with data value disallowed (chip bug)
Build 1077 FIX 01194 MPC5xx/8xx Debugger	passive PPC500/800 debugger was ignoring powerfail while running

MPC5xx/MPC8xx Trace

Build 1096 FIX 01434 FIRE Emulator MPC8XX MPC5xx/MPC8xx Trace	MPC5xx/MPC8xx flow trace display may hang when PTR cycles are missing
Build 1077 CHG 01347 MPC5xx/MPC8xx Trace MPC56x NEXUS Debugger and Trace	MPC55x/56x: DMBR/DBOR improved display.
Build 1077 FIX 01332 MPC5xx/MPC8xx Trace	MPC5xx flow trace error fixed (burst size was 4 instead of 8)
Build 1077 FIX 01288 MPC5xx/MPC8xx Trace	**Flowtrace error** Flowtrace error on MPC8xx/MPC5xx mtmsr was not considered as indirect branch

Build 42354 IMP 02354 MPC5xxx and SPC5xx Debugger MPC57xx NEXUS High Speed Serial Trace Port MPC55xx/MPC56xx NEXUS Debugger and Trace	**depletion recovery for affected on-chip flash modules added**
Build 42354 IMP 02352 MPC5xxx and SPC5xx Debugger MPC57xx NEXUS High Speed Serial Trace Port	**run-time performance monitor register access (SNOOPer) added**
Build 38589 IMP 02299 GTM Debugger (TriCore) MPC5xxx and SPC5xx Debugger TriCore Debugger	**initinal support for BOSCH GTM added** Initinal support for the BOSCH GTM (Generic Timer Module) added. Available in TriCore and PowerPC devices.
Build 38589 IMP 02296 MPC5xxx and SPC5xx Debugger TriCore Debugger	**Automotive Debug Cable** Support for a universal automotive debug cable added. Currently TriCore and PowerPC are supported.
Build 22490 IMP 02189 MPC5xxx and SPC5xx Debugger MPC55xx/MPC56xx NEXUS Debugger and Trace FLASH.ReProgram (Re-program FLASH)	**MPC55XX/56XX** FLASH.ReProgram now supports programming Flash with ECC errors without prior FLASH.Erase call
Build 19417 CHG 02181 MPC5xxx and SPC5xx Debugger MPC55xx/MPC56xx NEXUS Debugger and Trace SYStem (System configuration) TrOnchip (Onchip triggers)	**SYStem.Up with CPU selection MPC55XX** The behavior of the MPC55XX (auto-selection type) has changed. When this type is selected, the debugger will reset all SYStem, TrOnchip and trace settings to default upon SYStem mode change. If settings are needed to be set before SYStem.Up, e.g. SYStem.Option.WATCHDOG, SYStem.Option.ETK, first select proper CPU type before, or use SYStem.DETect CPU before setting SYStem options.
Build 7181 FIX 02007 MPC5xxx and SPC5xx Debugger Step.Hll (HLL single-stepping)	**HLL stepping may ignore BPs on MPC55xx**
Build 4728 FIX 01924 MPC5xxx and SPC5xx Debugger Data.LOAD.Elf (Load ELF file)	**PowerPC e500 floating point returns** Floating point function return values are not displayed.

MPC74XX Debugger

Build 1077 FIX 01299 MPC74XX Debugger	**Software Breakpoints did not work** MPC74XX: Software Breakpoints did not work properly with MSR_VEC = 1
Build 1077 FIX 01282 PPC600/750 Debugger MPC74XX Debugger	**SRR/CSRR Bug** Fixed SRR/CSRR Bug (wrong values displayed) (750,755,74XX,MGT)
Build 1077 IMP 01256 PPC600/750 Debugger MPC74XX Debugger	**Multicore debugging** PPC75X, MPC740X Multicore debugging supported
Build 1077 IMP 01177 MPC74XX Debugger Register (Processor registers)	ALTIVEC registers and variables implememted

MicroBlaze Debugger

Build 19417 IMP 02187 MicroBlaze Debugger	**Chipscope supported for Microblaze**

Multicore Debugging

Build 100486 IMP 02540 Multicore Debugging TargetSystem.NewInstance (Start new TRACE32 PowerView instance)	**New command TargetSystem.NewInstance** The command TargetSystem.NewInstance creates the config file for the new TRACE32 instance and then starts the TRACE32 instance for the specified architecture (e.g. t32marm.exe). Each started TRACE32 instance gets a name, which simplifies the handling of multiple instances. The new instance is automatically linked to the same PowerDebug module or the same MCI Server as the instance that accepted the command.
Build 100486 IMP 02538 Multicore Debugging InterCom.execute (Execute command via InterCom system) InterCom.Evaluate (Evaluate function via InterCom system) InterCom.executeNoWait (Execute command via InterCom system) InterCom.WAIT (Wait for remote InterCom system)	**Keywords and names simplify use of InterCom commands** All listed InterCom commands allow using the keywords SELF, ALL and OTHERS to address TRACE32 instance(s), as well as the name assigned to a TRACE32 instance and a name pattern.

NAND FLASH Programming

Build 93173 IMP 02519 FLASH Programming (Memory-Mapped) NAND FLASH Programming	**Allows TRACE32 to send SPI FLASH commands** Allows TRACE32 to send valid FLASH command sequences to a SPI FLASH device.

NIOS II Debugger

Build 4728 CHG 01958 NIOS II Debugger SYStem.Option (Special setup)	**New sytem options BTM,DTM,SYNC** Control behavior of off-chip and on-chip trace via system.option.
Build 4728 CHG 01956 NIOS II Debugger SYStem.Option (Special setup)	**New system option FSS** Support for different trace clock timings.
Build 4728 IMP 01954 NIOS II Debugger	**new command JTAG.LOADRBF**
Build 1198 CHG 01714 NIOS II Debugger SIM Instruction Set Simulator for NIOS II	**access modes** additional access modes DC: IC: NC: EDC: EIC: ENC:

Build 1190 01685 DSP56K Debugger ONCE Debugger for DSP56300 SYStem.Option (Special setup)	**DSP56xxx IMASKASM/IMASKHLL locked** SYStem.Option IMASKASM/IMASKHLL was locked unintentionally.
Build 1096 IMP 01442 DSP56K Debugger ONCE Debugger for DSP56300	DSP5600x/DSP563xx strings displayed as "WideString"
Build 1084 FIX 01418 DSP56K Debugger ONCE Debugger for DSP56300 DSP56800 and DSP56800E Debugger	DSP56xxx IEEE loaded produced wrong function address ranges
Build 1084 FIX 01415 DSP56K Debugger ONCE Debugger for DSP56300 DSP56800 and DSP56800E Debugger	DSP56xxx bug fixed in IEEE loader for OLD compiler formats
Build 1084 FIX 01410 DSP56K Debugger ONCE Debugger for DSP56300	**DSP563xx disassembler error** DSP563xx and DSP5600x jump targets calculated wrong (HLL Step into fails)
Build 1077 FIX 01295 ONCE Debugger for DSP56300 PER.view (Display peripherals)	**All per Files reworked** 563xx All per Files reworked. Last changes has wrong include parameter.
Build 1077 CHG 01173 ONCE Debugger for DSP56300 Data.LOAD (Load file)	DSP56300: speed up download speed. changed status read in drvbdm58.c
Build 1077 FIX 01137 ONCE Debugger for DSP56300	DSP56300: BIT SCR[REIE] changed. [ena,dis] -> [dis,ena]

OS-aware Debugging

Build 105499 IMP 02559 Hypervisor-aware Debugging OS-aware Debugging	**option /ACCESS added for TASK.CONFIG and EXTension.LOAD** The defined access class will be used always from the OS and Hypervisor awareness.
Build 88288 IMP 02493 OS-aware Debugging TASK.select (Display context of specified task) TASK.CONFIG (Configure OS Awareness)	**TASK.select is now default command for TASK command group** Now TASK.select is default command for TASK command group. It was TASK.CONFIG before.

PCP Debugger (TriCore)

Build 105499 IMP 02560 XC2000/C166SV2 Debugger GTM Debugger (TriCore) PCP Debugger (TriCore) TriCore Debugger	**improved MCDS trace decoder speed**
Build 38589 IMP 02297 GTM Debugger (TriCore) PCP Debugger (TriCore) TriCore Debugger CLOCK (Display date and time)	**on-chip clock display and settings** The CLOCK command group simplifies the setup for timestamp decoding of an on-chip trace. Can also be used to verify PLL and clock setup.

PODBUS Ethernet Controller

Build 125398 CHG 02580 JTAG Debugger PODBUS Ethernet Controller ICD In-Circuit Debugger TRACE32 Software	**discontinued software and hardware support** The TRACE32 controller based software (like t32win.exe, t32cde, t32qt, ... + mcpm*.t32) doesn't support PodbusEthernetController and ICD debug modul usage anymore. Please use in this case host based TRACE32 software executables (like t32marm.exe, t32marm, t32marm-qt, ...) instead.
Build 2940 FIX 01900 PODBUS Ethernet Controller	**DHCP problem** boothang of podbus ethernet controllers (old revisions only) solved

Build 19417 CHG 02119 PPC400 Debugger PPC440 Debugger	**improve bus error handler for PPC4xx**
Build 19417 IMP 02117 PPC400 Debugger PPC440 Debugger	**new command SYS.DETECT.CPU** Add CPU autotedection for PowerPC 440/40x. (command: SYStem.DETECT.CPU)
Build 13751 CHG 02080 PPC400 Debugger PPC440 Debugger TrOnchip (Onchip triggers)	**change mode of r/w data breakpoints** Mode for data BP will be ASYNCHRONOUS instead of SYNCHRONOUS by default from now on. With 'TrOnchip.SYNCHRONOUS ON' could be switched to old mode.
Build 8248 FIX 02019 PPC400 Debugger	**PPC in Xilinx FPGA** SW expect a wrong peripheral file for VirtexPPC, VirtexPPC1st and VirexPPC2nd. Debugger do not work at all with VirtexPPC, VirtexPPC1st and VirexPPC2nd.
Build 7181 IMP 01994 PPC400 Debugger SYStem.CPU (Select CPU)	**support XILINX designs with JTAGPPC ctrl** support for PPC405 core(s) in chain with FPGA (JTAGPPC controller) Please also see application note in FAQs.
Build 1264 FIX 01863 PPC400 Debugger PER.view (Display peripherals)	**TBL/TBH address in perfile corrected** Correct TBL/TBH read/write address implemented in per405xx.per files
Build 1241 FIX 01782 PPC400 Debugger PPC440 Debugger	**1st ICD start get fatal error (timeout)** 1st ICD start after a power cycle may get a fatal error because of a timeout.
Build 1224 FIX 01762 PPC400 Debugger	**support for new dongle on passiv modules** The new redesigned PPC4xx dongle (gray ribbon cable) will be also supported by the standard "Debug Interface" (passiv ICD, LA-7700, LA-7701).
Build 1224 IMP 01748 PPC400 Debugger	**PPC40x core and Xilinx V2P support** certain CPU cores like 405D3 will be supported as well as in the multicore environment.
Build 1224 FIX 01746 PPC400 Debugger PPC440 Debugger Go.direct (Start the program execution) RunTime.state (Display results)	**Runtime counter stop randomly** Internal driver error. Wrong mask for JDSR[PSP] bit.
Build 1196 FIX 01706 PPC400 Debugger PPC440 Debugger	**sync- or async-break fail after GO**
Build 1133 CHG 01500 PPC400 Debugger	PPC400: the DBSR[MRR] bit will be mask out and not be cleared by the debugger anymore.

Build 1077 FIX 01205 PPC400 Debugger PPC440 Debugger	**Bus error handling** Bus error handling for OPB for PPC405 and PPC440 improved
Build 1077 FIX 01183 PPC400 Debugger	**D-Cache** Bug fixed in fast download for PPC405 when D-Cache was dirty
Build 1077 IMP 01136 PPC400 Debugger	**Powerdebug** PPC405 SLER!=0 supported on powerdebug
Build 1077 CHG 01116 PPC400 Debugger	**debug cable version check** debug cable version check for PPC4xx only done in PowerDebug III

Build 38589 FIX 02293 PPC440 Debugger	**corrupt TLB (index=5)** TLB (index=5) could have been corrupted after STEP or STOP/GO
Build 26464 FIX 02219 PPC440 Debugger	**Fix for Virtex5 dual core designs** two PPC440 core in one JTAG chain with FPGA is working now.
Build 22490 IMP 02194 PPC440 Debugger	**add PPC460GTX, APM821x1, APM83290**
Build 19417 CHG 02119 PPC400 Debugger PPC440 Debugger	**improve bus error handler for PPC4xx**
Build 19417 IMP 02117 PPC400 Debugger PPC440 Debugger	**new command SYS.DETECT.CPU** Add CPU autotedection for PowerPC 440/40x. (command: SYStem.DETECT.CPU)
Build 13751 CHG 02080 PPC400 Debugger PPC440 Debugger TrOnchip (Onchip triggers)	**change mode of r/w data breakpoints** Mode for data BP will be ASYNCHRONOUS instead of SYNCHRONOUS by default from now on. With 'TrOnchip.SYNCHRONOUS ON' could be switched to old mode.
Build 7451 FIX 02015 PPC440 Debugger	**440EPx support**
Build 4728 FIX 01946 PPC440 Debugger Register.view (Display registers) Register.Set (Modify register contents)	**MCSRR0/1 register write with r.s** MCSRR0/1 register write from the register window (r.s mcsrr0 xxxx) do not take any effect.
Build 4728 IMP 01927 PPC440 Debugger	**per440ep: EBC mod., USB2.0 added**
Build 2940 FIX 01899 PPC440 Debugger	**new bus error handling for PPC440EP** support for crossbar Arbiter on PPC440EP-Pass2 instead of Core PLB4 Arbiter from PPC440EP-Pass1
Build 2940 FIX 01885 PPC440 Debugger	**PPC440 not working**
Build 1241 FIX 01803 PPC440 Debugger	**PPC440GX CPU (re)support** Bug fix for PPC440GX. Software change has had disabled the PPC440GX derivative.
Build 1241 FIX 01782 PPC400 Debugger PPC440 Debugger	**1st ICD start get fatal error (timeout)** 1st ICD start after a power cycle may get a fatal error because of a timeout.
Build 1224 IMP 01749 PPC440 Debugger	**PPC44x core support**
Build 1224 IMP 01747 PPC400 Debugger PPC440 Debugger	**Multicore support for PPC4xx**

Build 1196 FIX 01706 PPC400 Debugger PPC440 Debugger	**sync- or async-break fail after GO**
Build 1077 FIX 01242 PPC440 Debugger	**TLB** TLB replacement caused debug TLBs to overwrite target TLBs for PPC440
Build 1077 FIX 01213 PPC440 Debugger	**ICache** ICache recoding for PPC440 implemented
Build 1077 IMP 01208 PPC440 Debugger	**Peripherial file** Peripherial file for PPC440 rewriten
Build 1077 FIX 01206 PPC440 Debugger	**physical memory access** Bug fixed in physical memory access for PPC440
Build 1077 IMP 01134 PPC440 Debugger	**Full memory model** PPC440 full memory model supported

Build 1246 IMP 01816 PPC600/750 Debugger	**Additional 740/750 derivative supported** 750A2/740A2 Support for PVR==0x30082202 and 0x10082202 (740P/750P:Conan/Doyle)
Build 1190 FIX 01648 PPC600/750 Debugger	**Wrong IP value after Register.RESet** PPC6xx: PC was initialized with 0x00 instead of 0xfff00100
Build 1151 FIX 01590 PPC600/750 Debugger	MPC603e: some 603e derivatives (Projekt99) do not work with PowerDebugUSB/PowerDebugEthernet. Or at least with with low JTAG frequency.
Build 1077 FIX 01297 PPC600/750 Debugger	**L2: DEC value partially inverted** PPC603E, PPC750A2, A3, L2: DEC value partially inverted
Build 1077 FIX 01284 PPC600/750 Debugger	**Updated Memory access functions** MPC755 Updated Memory access functions (for MPC107 companion chip access)
Build 1077 FIX 01282 PPC600/750 Debugger MPC74XX Debugger	**SRR/CSRR Bug** Fixed SRR/CSRR Bug (wrong values displayed) (750,755,74XX,MGT)
Build 1077 CHG 01268 PPC600/750 Debugger	**HW/SW-BP handling changed** HW/SW-BP handling changed. Will be set with BDM_start.
Build 1077 FIX 01263 PPC600/750 Debugger	**ICache+DCache** MPC755 L1 ICache+DCache r/w and *BAT4..7 write fixed
Build 1077 IMP 01256 PPC600/750 Debugger MPC74XX Debugger	**Multicore debugging** PPC75X, MPC740X Multicore debugging supported
Build 1077 IMP 01182 PPC600/750 Debugger PER.view (Display peripherals)	MPC755: own per file.
Build 1077 IMP 01146 PPC600/750 Debugger	**PPC750CXdd2** PPC750CX: Added Support for PPC750CXdd2
Build 1077 FIX 01132 PPC600/750 Debugger	**RUNN counter enlarged** PPC750L3, MPC755: RUNN counter enlarged for slow MEM accesses + wait after RUNN (PDE).

Power Architecture PQII Trace

Build 1096 IMP 01453 Power Architecture PQII Trace	New option NOTC for RiscTrace for for MPC82xx to support tracing without TC lines

Build 125398 CHG 02581 EPROM/FLASH Simulator PowerIntegrator - Logic and Bus Analyzer PowerProbe - Logic and Protocol Analyzer TRACE32 Software Stimuli Generator	**discontinued software and hardware support** The t32pbi executables aren't supported anymore.
Build 9640 FIX 02037 PowerIntegrator - Logic and Bus Analyzer <trace>.Arm (Arm the trace) <trace>.Init (Initialize trace)	**no sampling at all** under circumstances the PowerIntergrator sampled nothing at all
Build 7181 CHG 02001 PowerIntegrator - Logic and Bus Analyzer Integrator.Program (Program trigger unit) Integrator.ReProgram (Program trigger unit)	**complex trigger unit** SELECTOR definition: mode RAISING renamed to RISING
Build 4728 FIX 01980 PowerIntegrator - Logic and Bus Analyzer Integrator.Program (Program trigger unit) Integrator.ReProgram (Program trigger unit) Integrator.TSYNC (Select trigger line and mode)	**transient recognition changed** for simple and complex trigger unit the transient recognition for the dataselectors was changed and some bugs fixed
Build 2940 FIX 01883 PowerIntegrator - Logic and Bus Analyzer Integrator.ABCDEF (Sampling configuration for probes ABCDEF) Integrator.JKLMNO (Sampling configuration for probes JKLMNO)	**Transient detection - sample control** transient detection for sample control was working partly wrong under circumstances affected modes: State, StatePLL, StatePLLBoth, DDR
Build 1253 FIX 01880 PowerIntegrator - Logic and Bus Analyzer Integrator.Program (Program trigger unit) Integrator.ReProgram (Program trigger unit)	**analyzer complex trigger unit** hostbased software may crash under certain circumstances
Build 1246 FIX 01814 PowerIntegrator - Logic and Bus Analyzer Integrator.ABCDEF (Sampling configuration for probes ABCDEF) Integrator.JKLMNO (Sampling configuration for probes JKLMNO)	**Transient detection - sample control** transient detection for sample control was working partly wrong under circumstances affected modes: State, StatePLL, StatePLLBoth, DDR

Build 1204 IMP 01726 PowerIntegrator - Logic and Bus Analyzer	**programming of complex triggerunit** design of selector realization changed (more flexible, higher internal number) several supervising modes for inputchannel available
Build 1190 FIX 01664 PowerTrace for NEXUS PowerIntegrator - Logic and Bus Analyzer PowerProbe - Logic and Protocol Analyzer Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**programming of complex triggerunit** hostbased software version only: trigger unit programmed wrong under certain circumstances
Build 1190 FIX 01660 PowerIntegrator - Logic and Bus Analyzer PowerProbe - Logic and Protocol Analyzer Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**programming of complex triggerunit** since V1.12 from 2004.02.23 the usage of several single channels was blocked inside a SELECTOR definition
Build 1161 CHG 01636 PowerIntegrator - Logic and Bus Analyzer PowerProbe - Logic and Protocol Analyzer Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**programming of complex triggerunit** declarationtype synonymes introduced EVCNT for EVENTCOUNTER EXSYCNT for EXTERNSYNCCOUNTER TICNT for TIMECOUNTER
Build 1151 CHG 01596 PowerIntegrator - Logic and Bus Analyzer PowerProbe - Logic and Protocol Analyzer	**Programming of complex triggerunit** command Counter.Enable renamed to Counter.Increment
Build 1151 IMP 01589 PowerIntegrator - Logic and Bus Analyzer PowerProbe - Logic and Protocol Analyzer Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	size of all counters increased from 44 to 45 bits
Build 1146 FIX 01579 PowerIntegrator - Logic and Bus Analyzer PowerProbe - Logic and Protocol Analyzer Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Programming of complex triggerunit** usage of levelname START may crash hostbased software

Build 1077 FIX 01153 FIRE Fully Integrated RISC Emulator ICE In-Circuit Emulator PowerIntegrator - Logic and Bus Analyzer PowerTrace PowerProbe - Logic and Protocol Analyzer <trace>.AutoInit (Automatic initialization)	Automatic init of analyzers (Integrator,PowerTrace,FIRE,ICE) changed

PowerPC 400 Trace

Build 1096 IMP 01448 PowerPC 400 Trace	Debug cycles in RiscTrace for PPC440 now suppressed

Build 125398 CHG 02581 EPROM/FLASH Simulator PowerIntegrator - Logic and Bus Analyzer PowerProbe - Logic and Protocol Analyzer TRACE32 Software Stimuli Generator	**discontinued software and hardware support** The t32pbi executables aren't supported anymore.
Build 9640 CHG 02040 PowerProbe - Logic and Protocol Analyzer	**PowerProbe prompt removed** PowerProbe can now only be controlled with "Probe" command of the debugger.
Build 7181 IMP 02005 PowerProbe - Logic and Protocol Analyzer	**List defaults for PowerProbe changed** List shows raw data when no disassembler configured. Grouped or named channels shown first.
Build 1190 FIX 01664 PowerTrace for NEXUS PowerIntegrator - Logic and Bus Analyzer PowerProbe - Logic and Protocol Analyzer Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**programming of complex triggerunit** hostbased software version only: trigger unit programmed wrong under certain circumstances
Build 1190 FIX 01660 PowerIntegrator - Logic and Bus Analyzer PowerProbe - Logic and Protocol Analyzer Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**programming of complex triggerunit** since V1.12 from 2004.02.23 the usage of several single channels was blocked inside a SELECTOR definition
Build 1161 CHG 01636 PowerIntegrator - Logic and Bus Analyzer PowerProbe - Logic and Protocol Analyzer Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**programming of complex triggerunit** declarationtype synonymes introduced EVCNT for EVENTCOUNTER EXSYCNT for EXTERNSYNCCOUNTER TICNT for TIMECOUNTER
Build 1151 CHG 01596 PowerIntegrator - Logic and Bus Analyzer PowerProbe - Logic and Protocol Analyzer	**Programming of complex triggerunit** command Counter.Enable renamed to Counter.Increment

Build 1151 IMP 01589 PowerIntegrator - Logic and Bus Analyzer PowerProbe - Logic and Protocol Analyzer Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	size of all counters increased from 44 to 45 bits
Build 1146 FIX 01579 PowerIntegrator - Logic and Bus Analyzer PowerProbe - Logic and Protocol Analyzer Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Programming of complex triggerunit** usage of levelname START may crash hostbased software
Build 1133 IMP 01487 PowerProbe - Logic and Protocol Analyzer	PowerProbe complex trigger polarities selectable with Probe.TOUT.A..D
Build 1133 FIX 01459 C5500 Debugger PowerProbe - Logic and Protocol Analyzer	PowerProbe was not working as "Probe" command in TMS320C55x debugger
Build 1096 IMP 01455 PowerProbe - Logic and Protocol Analyzer Count (Universal counter)	frequency meassurement extended beyond 20MHz on PowerProbe, new counter window
Build 1096 FIX 01436 PowerProbe - Logic and Protocol Analyzer	PowerProbe timestamps in 50MHz trace mode wrong
Build 1077 FIX 01375 PowerProbe - Logic and Protocol Analyzer	PowerProbe timestamps where wrong in 50MHz mode
Build 1077 FIX 01223 PowerProbe - Logic and Protocol Analyzer	CPU selection caused channel modes of PROBE command to be wrong
Build 1077 FIX 01153 FIRE Fully Integrated RISC Emulator ICE In-Circuit Emulator PowerIntegrator - Logic and Bus Analyzer PowerTrace PowerProbe - Logic and Protocol Analyzer <trace>.AutoInit (Automatic initialization)	Automatic init of analyzers (Integrator,PowerTrace,FIRE,ICE) changed
Build 1077 FIX 01093 PowerProbe - Logic and Protocol Analyzer	PowerProbe displays wrong records in Stack mode, time correlation lost
Build 1077 IMP 01085 PowerProbe - Logic and Protocol Analyzer	A.STATE() and A.RECORDS() implemented for PowerProbe

Build 4728 CHG 01904 MPC5200 Debugger PowerQUICC II/Pro Debugger SYStem.Option (Special setup)	**new SYStem.Option.IP.BOTH** new MSR[IP] handling for Onchip-BP's on processor with two Onchip-BP's. (G2_LE /ZEPPO core) Default is SYS.O.IP.AUTO where only the Onchip-BP control the active (MSR[IP]) exception handler block.
Build 1246 FIX 01819 PowerQUICC II/Pro Debugger SYStem.Option (Special setup)	**SYStem.Option.IP AUTO do not work at all** SYStem.Option.IP.AUTO may fail in conjunction with SYStem.Option.BASE.AUTO
Build 1224 IMP 01761 PowerQUICC II/Pro Debugger SYStem.Option PARITY (Generate parity on memory access)	**write parity support for PQ2 (groucho)**
Build 1077 FIX 01377 PowerQUICC II/Pro Debugger	MPC82XX: enabled watchdog servicing via debugger
Build 1077 FIX 01350 PowerQUICC II/Pro Debugger	MPC82XX manual setting of base address. IOBASE() does not worked.
Build 1077 IMP 01346 MPC5200 Debugger PowerQUICC II/Pro Debugger	MPC82XX, MGT5100/5200 ICache decoder added
Build 1077 IMP 01335 PowerQUICC II/Pro Debugger	MPC82xx step/onchip breakpoint pipeline fix changed
Build 1077 FIX 01294 MPC5200 Debugger PowerQUICC II/Pro Debugger	**SystemUp failed when Boot** MGT5200/MPC8280: SystemUp failed when Boot from 0x00000100

Build 125398 CHG 02599 Debug Modules PowerTrace	**discontinued hardware support** This is the last TRACE32 software release which supports TRACE32 Debug Interface LA-7701 TRACE32 PowerTrace 64MB LA-7706 TRACE32 PODBUS Ethernet Controller
Build 4728 FIX 01962 PowerTrace Analyzer.List (List trace contents)	**sporadic ghost records fixed** Under certain circumstances the trace contained "ghost" records (records which actually were never recorded by the hardware).
Build 1145 FIX 01568 JTAG Debugger PowerTrace	PowerDebug/Ethernet was responding to ping commands on multicast addresses
Build 1077 IMP 01368 ColdFire Trace PowerTrace	Coldfire supported on PowerTrace
Build 1077 FIX 01352 PowerTrace	Analyzer upload of PowerTrace on Ethernet sporadically decompressed wrong data 1:100000000
Build 1077 IMP 01348 Real-time Trace PowerTrace	Trigger record now always marked in RiscTrace/PowerTrace
Build 1077 IMP 01315 FIRE Fully Integrated RISC Emulator PowerTrace	New analyzer mode Leach for PowerTrace and FIRE
Build 1077 IMP 01278 PowerTrace <trace>.SAVE (Save trace for postprocessing in TRACE32)	**A.SAVE changed** A.SAVE changed to a compressed format for PowerTrace
Build 1077 IMP 01237 JTAG Debugger PowerTrace	**Ethernet** Lost packets on PowerDebug Ethernet handled better
Build 1077 FIX 01153 FIRE Fully Integrated RISC Emulator ICE In-Circuit Emulator PowerIntegrator - Logic and Bus Analyzer PowerTrace PowerProbe - Logic and Protocol Analyzer <trace>.AutoInit (Automatic initialization)	Automatic init of analyzers (Integrator,PowerTrace,FIRE,ICE) changed

Build 2940 IMP 01911 JTAG Debugger FIRE Fully Integrated RISC Emulator ICE In-Circuit Emulator PowerTrace for NEXUS TRACE32 Instruction Set Simulators	**Practice functions** synonym D.SUM() for DATA.SUM() added
Build 1246 CHG 01834 ICE In-Circuit Emulator PowerTrace for NEXUS Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**programming of complex triggerunit** commands renamed: Flag.OFF -> Flag.FALSE Flag.ON -> Flag.TRUE
Build 1246 CHG 01832 PowerTrace for NEXUS Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**programming of complex triggerunit** keyword for addressevent definition ADDR replaced by ADDRESS
Build 1203 IMP 01719 PowerTrace for NEXUS	**PRACTICE functions** A.COUNTER.TIME(), A.COUNTER.EVENT() implemented for NEXUS now
Build 1190 FIX 01664 PowerTrace for NEXUS PowerIntegrator - Logic and Bus Analyzer PowerProbe - Logic and Protocol Analyzer Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**programming of complex triggerunit** hostbased software version only: trigger unit programmed wrong under certain circumstances
Build 1133 FIX 01493 PowerTrace for NEXUS	Frequency counter for Nexus messages was multiplied by 8
Build 1079 FIX 01387 PowerTrace for NEXUS <trace>.List (List trace contents)	A.L NEXUS may not show last nexus message of trace window
Build 1077 IMP 01125 PowerTrace for NEXUS	**data trace modes** New nexus data trace modes
Build 1077 FIX 01103 PowerTrace for NEXUS	NEXUS data message addresses sign extended

QorIQ PowerPC 32/64 Bit Debugger

Build 38589 IMP 02320 QorIQ PowerPC 32/64 Bit Debugger	**Improved synchronous break switch in AMP** For further descriptions and details of restrictions of the synchronous hardware based break please refer to the debugger_ppcqoriq.pdf manual.
Build 38589 IMP 02317 QorIQ PowerPC 32/64 Bit Debugger	**DDR trace support added** The NEXUS window gives easy access to the options.

QorIQ PowerPC NEXUS Aurora Trace Port

Build 100486 IMP 02523 Hypervisor-aware Debugging ARM/Cortex Trace (parallel) Intel~ Processor Trace QorIQ PowerPC NEXUS Aurora Trace Port <trace>.List (List trace contents)	**Decoding of machine id supported** The machine ID within the Context-ID or ownership packet is now decoded.
Build 80996 IMP 02483 MPC57xx NEXUS High Speed Serial Trace Port QorIQ PowerPC NEXUS Aurora Trace Port MPC55xx/MPC56xx NEXUS Debugger and Trace RTS.state (Open status and control window)	**RTS implemented for PowerPC (all E200zx cores) and QorIQ** RTS implemented for PowerPC (all E200zx cores) and QorIQ.

ROM Monitor

Build 1246 CHG 01813 JTAG Debugger FIRE Fully Integrated RISC Emulator ICE In-Circuit Emulator ROM Monitor	**Address range expressions** Behavior of general parameter parser changed The checking of accessmode equality inside address range expressions like address1--address2 is switched off now. Now it's allowed to use different accessmodes for begin and end address of address ranges. The accessmode of the end address will be ignored and will not cause an error message anymore. e.g. UP:1000--SD:2000 will be handled as UP:1000--UP:2000

RTOS Debugger for Linux

Build 42354 IMP 02351 RTOS Debugger for Linux	**support for physical addresses larger than 32bit improved** only 32bit architectures are affected
Build 42354 IMP 02349 RTOS Debugger for Linux TASK.DMESG (Display the kernel ring buffer) TASK.DTB (Display the device tree blob)	**new commands for Linux awareness** TASK.DMESG to display kernel log TASK.DTB to display device tree blob

RTOS Debugger for OSEK/ORTI

Build 42354 IMP 02324 RTOS Debugger for OSEK/ORTI TASK.ORTI (AUTOSAR/OSEK support)	**TASK.ORTI generates ORTI menu** Depending on the contents of the loaded ORTI file, the TASK.ORTI command automatically generates additional menu entries for displaying OS objects and for performance analysis on selected OS attributes.

RTOS Debugger for Windows

Build 93173 IMP 02507 RTOS Debugger for Windows Data.LOAD.CrashDump (Load MS Windows Crash Dump file) TASK.CrashDump (Windows crash dump analysis)	**Support for loading Windows crash dump** After loading the full Crash Dump file into the simulator or the target memory with the command Data.LOAD.CrashDump, the Windows awareness command TASK.CrashDump.LOADNT is used to auto-load the kernel debug symbols. After that the command TASK.CrashDump.LOADREG could be used to load the context of all the available cores.
Build 60219 IMP 02419 Intel~ x86/x64 JTAG Debugger RTOS Debugger for Windows	**WinSTD Awareness for x86/x64** Task-aware debugging for Win7/Win8

Real-time Trace

Build 1077 IMP 01348 Real-time Trace PowerTrace	Trigger record now always marked in RiscTrace/PowerTrace

Build 4728 CHG 01953 S08 Debugger FIRE Emulator for Freescale HC08 / HCS08 In-Circuit Emulator for 68HC05 and 68HC08	**accessmodes** additional access modes: DBG: EDBG: INI:

Build 7181 ERR 02000 S12 Debugger SIM Instruction Set Simulator for HC12/MCS12	**Star12X Simulator: tstx and tsty swapped** The instructions tstx and tsty were simulated swapped.
Build 4728 FIX 01920 S12 Debugger Data.LOAD.Ubrof (Load UBROF file)	**UBROF10 register variables wrong** Register variables may be shown wrong when UBROF10 format is used.
Build 1224 FIX 01759 S12 Debugger	**Disassembler Mnemonics swapped** Mnemonics PSHA und PSHB were swapped
Build 1206 CHG 01730 S12 Debugger FIRE Emulator for Freescale 68HC12 / MCS12 / S12X In-Circuit Emulator for 68HC11 In-Circuit Emulator for 68HC12 SIM Instruction Set Simulator for HC12/MCS12	**accessmodes** additional access modes X: XD: XP: AX: AXD: AXP: EX: EXD: EXP: EAX: EAXD: EAXP:
Build 1204 CHG 01724 S12 Debugger FIRE Emulator for Freescale 68HC12 / MCS12 / S12X In-Circuit Emulator for 68HC11 In-Circuit Emulator for 68HC12 SIM Instruction Set Simulator for HC12/MCS12	**access modes** additional access modes AG: EAG: GD: AGD: EGD: GP: AGP: EGP:
Build 1198 CHG 01713 S12 Debugger FIRE Emulator for Freescale 68HC12 / MCS12 / S12X In-Circuit Emulator for 68HC11 In-Circuit Emulator for 68HC12 SIM Instruction Set Simulator for HC12/MCS12	**access modes** additional access modes G: EG:
Build 1161 FIX 01619 S12 Debugger SYStem.RESet (Reset configuration)	**SYStem.RESet clears option of MCS12** Executing "SYStem.RESet" clears the "SYStem.Option ROMHM" when a MCS12 cpu is selected. On default this option should be set.
Build 1133 FIX 01497 S12 Debugger	Star12C breakpoint to location 0 with trap action active after breakpoints removed
Build 1133 FIX 01485 S12 Debugger	Star12 C32 onchip breakpoints may be set wrong when starting on a breakpoint

SH Debugger

Build 9640 CHG 02055 ARM Debugger C5500 Debugger SH Debugger BMC (Benchmark counters)	**Benchmark counter commands renamed** All benchmark counter related onchip trigger commands have been moved to the new command "BMC".
Build 1151 IMP 01612 MIPS32/MIPS64 Debugger SH Debugger Data.LOAD (Load file)	**DWARF1** Changes made to DWARF1 loader for Hitachi-SH compiler and Greenhills-MIPS
Build 1085 FIX 01426 DSP56K Debugger MPC5xx/8xx Debugger SH Debugger	missing files after software installation - install database corrected files are on the CD, but weren't not copied FIRE/ICD-C166/ST10: men7860t.men ICD-DSP56K : per56802.per per56807.per per56809.per per56367.per ICD-PPC : per8280.per ICD-SH2 : per7294.per
Build 1077 FIX 01186 SH Debugger	**SH bondout fetch decoding** SH bondout fetch decoding bug fixed

SH Trace

Build 1096 FIX 01457 SH Trace	tracebased performance on SH3/4 flowtrace not working correctly

SIM Instruction Set Simulator for 68K/ColdFire

Build 1151 FIX 01583 SIM Instruction Set Simulator for 68K/ColdFire	Simulator for 68k indexed pc-rel addressing wrong for disp >= 0x40
Build 1077 FIX 01163 SIM Instruction Set Simulator for 68K/ColdFire	**Exception handling** Exception handling (stackframes) fixed in 68k simulator

Build 1190 IMP 01683 C166CBC Debugger Super10 Debugger FIRE Emulator for C166 Family In-Circuit Emulator for C166/ST10 - Out of Production SIM Instruction Set Simulator for C166/ST10	**new access mode for 80166 family** EDBG: added
Build 1077 IMP 01211 SIM Instruction Set Simulator for C166/ST10	**C166** C166 Simulator supports interrupts
Build 1077 FIX 01155 SIM Instruction Set Simulator for C166/ST10	C166 simulator handled short esfr addressing to some registers wrong

Build 8248 FIX 02024 SIM Instruction Set Simulator for HC12/MCS12	**EORX instruction bug fixed** Instruction EORX wrote result to register IY instead of IX.
Build 7181 ERR 02000 S12 Debugger SIM Instruction Set Simulator for HC12/MCS12	**Star12X Simulator: tstx and tsty swapped** The instructions tstx and tsty were simulated swapped.
Build 4728 FIX 01959 SIM Instruction Set Simulator for HC12/MCS12	**EMULS instruction wrong results** EMUL instruction behaved like EMUL (no signed multiplys).
Build 4728 FIX 01942 SIM Instruction Set Simulator for HC12/MCS12	**Flags on load/store operations set wrong** Flag register was set wrong on some load/ store operations (e.g. STD instruction).
Build 4728 FIX 01937 SIM Instruction Set Simulator for HC12/MCS12	**cpex instruction incorrect zero flag set** Zero Flag was alwas set after execution. Should only be set when set before.
Build 1206 CHG 01730 S12 Debugger FIRE Emulator for Freescale 68HC12 / MCS12 / S12X In-Circuit Emulator for 68HC11 In-Circuit Emulator for 68HC12 SIM Instruction Set Simulator for HC12/MCS12	**accessmodes** additional access modes X: XD: XP: AX: AXD: AXP: EX: EXD: EXP: EAX: EAXD: EAXP:
Build 1204 CHG 01724 S12 Debugger FIRE Emulator for Freescale 68HC12 / MCS12 / S12X In-Circuit Emulator for 68HC11 In-Circuit Emulator for 68HC12 SIM Instruction Set Simulator for HC12/MCS12	**access modes** additional access modes AG: EAG: GD: AGD: EGD: GP: AGP: EGP:
Build 1198 CHG 01713 S12 Debugger FIRE Emulator for Freescale 68HC12 / MCS12 / S12X In-Circuit Emulator for 68HC11 In-Circuit Emulator for 68HC12 SIM Instruction Set Simulator for HC12/MCS12	**access modes** additional access modes G: EG:
Build 1151 FIX 01584 SIM Instruction Set Simulator for HC12/MCS12	Simulator for HC12 executed min and max commands wrong

SIM Instruction Set Simulator for MIPS

Build 7181 CHG 02008 MIPS32/MIPS64 Debugger MIPS32 Trace SIM Instruction Set Simulator for MIPS SIM Instruction Set Simulator for MIPS5K	**accessmodes** additional access modes: ADC: AIC: ANC:

SIM Instruction Set Simulator for MIPS5K

Build 7181 CHG 02008 MIPS32/MIPS64 Debugger MIPS32 Trace SIM Instruction Set Simulator for MIPS SIM Instruction Set Simulator for MIPS5K	**accessmodes** additional access modes: ADC: AIC: ANC:

SIM Instruction Set Simulator for NIOS II

Build 1198 CHG 01714 NIOS II Debugger SIM Instruction Set Simulator for NIOS II	**access modes** additional access modes DC: IC: NC: EDC: EIC: ENC:

Build 4728 FIX 01933 SIM Instruction Set Simulator for PowerPC	**PowerPC simulator DEC register problems** Manual set of DEC register fails. DEC register shown incorrect in register window.
Build 2940 FIX 01912 SIM Instruction Set Simulator for PowerPC	**stswi and lswi instructions wrong result** stswi and lswi instructions delivered wrong result when written to an address not modulo 4 aligned.
Build 1248 FIX 01837 SIM Instruction Set Simulator for PowerPC	**isel implemented** simulator stopped when reaching isel instruction
Build 1246 FIX 01833 SIM Instruction Set Simulator for PowerPC	**divw special case handled** When dividing 0x80000000 / -1, on Windows (and probably other systems) an exception occured (Integer Overflow) or TRACE32 quitted without an error message. This is now fixed. Note: Due to PPC spec, divw returns an undefined value when doing this operation or a division by 0.
Build 1246 FIX 01825 SIM Instruction Set Simulator for PowerPC	**mulhwu now takes care of signs** internal handling of signs corrected
Build 1246 FIX 01823 SIM Instruction Set Simulator for PowerPC	**mtpmr command now executed** The Simulator now does not stop any more when reaching the mtpmr command; command can also be stepped now
Build 1246 FIX 01818 SIM Instruction Set Simulator for PowerPC	**sraw[.] right shift with 0 corrected** shifting 0x80000000 >> 0 resulted in 0xFFFFFFFF, expected value is 0x80000000
Build 1206 CHG 01735 MPC5xx/8xx Debugger FIRE Emulator MPC8XX In-Circuit Emulator for Power Architecture SIM Instruction Set Simulator for PowerPC	**accessmodes** additional access modes PMR: F: AF: EF: EAF: FP: AFP: EFP: EAFP: FD: AFD: EFD: EAFD: SF: ASF: ESF: EASF: UF: AUF: EUF: EAUF: V: AV: EV: EAV: VP: AVP: EVP: EAVP: VD: AVD: EVD: EAVD: SV: ASV: ESV: EASV: UV: AUV: EUV: EAUV:
Build 1198 FIX 01717 SIM Instruction Set Simulator for PowerPC	**PPC Simulator LWSI/LWSX may fail** LWSI/LWSX give wrong results when bytecount is not aligned to 4.
Build 1161 IMP 01621 FIRE Emulator MPC8XX ICD Solutions for PowerPC and Power Architecture SIM Instruction Set Simulator for PowerPC	**Access modes** additional access mode DBG: and EDBG:

SIM Instruction Set Simulator for SuperH

Build 1246 FIX 01821 SIM Instruction Set Simulator for SuperH	**SH simulator immediate not sign extended**

SIM Instruction Set Simulator for TriCore

Build 38589 FIX 02289 SIM Instruction Set Simulator for TriCore	**dvinit.h simulation issue fixed**
Build 7181 FIX 02013 SIM Instruction Set Simulator for TriCore	**SHA16 instruction fixed** Shift direction flag of SHA16 instruction shifted left instead of right.
Build 4728 CHG 01979 TriCore Debugger TriCore Parallel Trace SIM Instruction Set Simulator for TriCore	**accessmodes** additional access modes: PCD: PCP: EPCD: EPCP: deleted accessmode: PCP:
Build 1161 CHG 01627 FIRE Emulator for TriCore ICD Solutions for TriCore SIM Instruction Set Simulator for TriCore	**Access modes** access modes BBB: and EBBB: renamed to EEC: and EEEC:
Build 1161 IMP 01623 FIRE Emulator for TriCore ICD Solutions for TriCore SIM Instruction Set Simulator for TriCore	**Access modes** additional access modes BBB: and EBBB:

SIM Instruction Set Simulator for x86

Build 1077 FIX 01150 SIM Instruction Set Simulator for x86	**80x86 simulator** 80x86 simulator scas was taking SI register instead of DI

Build 1145 FIX 01572 Sample-based Profiling PERF (Sample-based profiling)	SH/ARM monitors failed with the PERF command

Build 130863 CHG
02602
Script Language PRACTICE
	obsolete PRACTICE function
	SYStem.CONFIG.PCH() replaced by SYStem.PCH()
	VERSION.LICENSE() replaced by LICENSE.SERIAL()

Build 130863 IMP 02600 Script Language PRACTICE	**new PRACTICE function synonyms** additional PRACTICE function synonyms introduced: SYStem.PCH() for SYStem.CONFIG.PCH()
Build 125398 IMP 02583 Script Language PRACTICE	**new PRACTICE function synonyms** additional PRACTICE function synonyms introduced: hardware.POWERTRACE2LITE() for Analyzer.CONFIG.POWERTRACE2LITE(), hardware.POWERTRACE3 for Analyzer.CONFIG.POWERTRACE3(), TASK.MACHINE.ID() for TASK.MACHINEID()
Build 117056 ODD 02578 Logical Display of Peripherals Script Language PRACTICE	**PRACTICE and PER macro naming rules changed** Macro names must not start with a numerical digit anymore.

Build 117056 IMP
02573
Script Language PRACTICE
	new PRACTICE function synonyms
	additional PRACTICE function synonyms introduced:
	OS.FILE.EXIST() for FILE.EXIST()
	long forms added: TRANSlation.* for TRANS.*()
	TRANSlation.ENABLE() for TRANS.ENABLE()
	TRANSlation.INTERMEDIATE() for TRANS.INTERMEDIATE()
	TRANSlation.INTERMEDIATE.VALID() for TRANS.INTERMEDIATE.VALID()
	TRANSlation.INTERMEDIATEEX() for TRANS.INTERMEDIATEEX()
	TRANSlation.INTERMEDIATEEX.VALID() for TRANS.INTERMEDIATEEX.VALID()
	TRANSlation.LINEAR() for TRANS.LINEAR()
	TRANSlation.LINEAR.VALID() for TRANS.LINEAR.VALID()
	TRANSlation.LINEAREX() for TRANS.LINEAREX()
	TRANSlation.LINEAREX.VALID() for TRANS.LINEAREX.VALID()
	TRANSlation.LIST.LOGRANGE() for TRANS.LIST.LOGRANGE()
	TRANSlation.LIST.LOGRANGE.ZONE() for TRANS.LIST.LOGRANGE.ZONE()
	TRANSlation.LIST.NUMBER() for TRANS.LIST.NUMBER()
	TRANSlation.LIST.NUMBER.ZONE() for TRANS.LIST.NUMBER.ZONE()
	TRANSlation.LIST.PHYSADDR() for TRANS.LIST.PHYSADDR()
	TRANSlation.LIST.PHYSADDR.ZONE() for TRANS.LIST.PHYSADDR.ZONE()
	TRANSlation.LIST.TYPE() for TRANS.LIST.TYPE()
	TRANSlation.LIST.TYPE.ZONE() for TRANS.LIST.TYPE.ZONE()
	TRANSlation.LOGICAL() for TRANS.LOGICAL()
	TRANSlation.LOGICAL.VALID() for TRANS.LOGICAL.VALID()
	TRANSlation.PHYSICAL() for TRANS.PHYSICAL()
	TRANSlation.PHYSICAL.VALID() for TRANS.PHYSICAL.VALID()
	TRANSlation.PHYSICALEX() for TRANS.PHYSICALEX()
	TRANSlation.PHYSICALEX.VALID() for TRANS.PHYSICALEX.VALID()
	TRANSlation.TABLEWALK() for TRANS.TABLEWALK()

Build 112182 IMP 02567 Script Language PRACTICE MENU.Program (Interactive programming) MENU.ReProgram (Menu programming)	**new command PERMENU for menu files added** The new menu programming command PERMENU converts the tree structure of per files to men files.
Build 112182 IMP 02564 Script Language PRACTICE OS.OPEN (Open file in default application)	**new command OS.OPEN added**
Build 112182 IMP 02562 Script Language PRACTICE	**new PRACTICE functions** additional PRACTICE functions introduced: ADDRESS.ACCESS.CMP(), ADDRESS.ACCESS.CMPSTRICT(), CHIP.GTM.MCSModule(), CHIP.GTM.ATOMModule(), CHIP.GTM.TOMModule(), CHIP.GTM.TIMModule(), CIProbe(), CONVert.TIMERAWTOHEX(), CPU.ADDRESS(), ERROR.CMDLINE(), ERROR.MESSAGE(), ERROR.POSITION(), EVAL.TIme(), FLASH.SECTOR.OPTION(), hardware.POWERTRACESERIAL.ADAPTER.NAME(), hardware.POWERTRACESERIAL.ADAPTER.REV(), JTAG.XUSEFUSE.CNTL(), JTAG.XUSEFUSE.DNA(), JTAG.XUSEFUSE.KEY(), JTAG.XUSEFUSE.RESULT(), JTAG.XUSEFUSE.RSA(), JTAG.XUSEFUSE.SEC(), JTAG.XUSEFUSE.USER(), JTAG.XUSEFUSE.USER128(), MMU.DEFAULTPT2(), MMU.DEFAULTPT.ZONE(), MMU.DEFAULTTRANS.LOGRANGE.ZONE(), MMU.DEFAULTTRANS.PHYSADDR.ZONE(), MMU.FORMAT.DETECTED(), MMU.FORMAT.DETECTED.ZONE(), MMU.FORMAT.ZONE(), SYStem.INSTANCECOUNT(), SYStem.IRISconfig.RemoteServer(), SYStem.Option.DUALPORT(), SYStem.CONFIG.PCH(), SYStem.Option.FASTACCESS(), SYStem.Option.PARTitionconfig(), TIME.ZERO(), TRANS.LIST.LOGRANGE(), TRANS.LIST.LOGRANGE.ZONE(), TRANS.LIST.NUMBER(), TRANS.LIST.NUMBER.ZONE(), TRANS.LIST.PHYSADDR(), TRANS.LIST.PHYSADDR.ZONE(), TRANS.LIST.TYPE(), TRANS.LIST.TYPE.ZONE()
Build 105499 CHG 02549 Script Language PRACTICE	**PRACTICE function deleted** XLEN()
Build 100486 IMP 02546 Script Language PRACTICE LOG.DO (Log calls of PRACTICE scripts)	**Log file for PRACTICE scripts** The command LOG.DO allows logging all PRACTICE script calls. The command was mainly introduced for support purposes.

Build 100486 IMP 02533 Script Language PRACTICE	**new PRACTICE functions** additional PRACTICE functions introduced: Analyzer.PCIE.CONFIG(), Analyzer.PCIE.ISCONFIGURED(), Analyzer.PCIE.Register(), COMPonent.NAME(), COMPonent.TYPE(), COVerage.SourceMetric(), ELA.VERSION(), InterCom.NAME(), InterCom.PODPORTNAME(), JTAG.SEQuence.RESULT(), MMU.DEFAULTTRANS.LOGRANGE(), MMU.DEFAULTTRANS.PHYSADDR(), PATH.NUMBER(), PATH.PATH(), STRing.SCANBack(), sYmbol.ISREGISTER(), sYmbol.ISSTACK(), sYmbol.ISSTATIC(), sYmbol.REGISTER(), sYmbol.SECNAME(), SYStem.ADAPTER.FW.OUTDATED(), SYStem.Option.TOPOlogy(), SYStem.Option.TOPOlogy.SOCKETS(), TERM.TRIGGERED(), Var.EXIST(), Var.ISREGISTER(), Var.ISSTACK(), Var.ISSTATIC(), Var.REGISTER(), XLEN()
Build 93173 ODD 02511 Script Language PRACTICE	**modified PRACTICE function behaviour** function hardware.ICD() returns FALSE() in case of PBI=MCILIB or PBI=MCISERVER now
Build 93173 IMP 02509 Script Language PRACTICE	**new PRACTICE function synonyms** PRACTICE functions synonyms introduced: OS.PCF() ==> OS.PresentConfigurationFile(), OS.PDD() ==> OS.PresentDemoDirectory(), OS.PED() ==> OS.PresentExecutableDirectory(), OS.PEF() ==> OS.PresentExecutableFile(), OS.PHD() ==> OS.PresentHomeDirectory(), OS.PHELPD() ==> OS.PresentHELPDirectory(), OS.PLF() ==> OS.PresentLicenseFile(), OS.PPD() ==> OS.PresentPracticeDirectory(), OS.PPF() ==> OS.PresentPracticeFile(), OS.PSD() ==> OS.PresentSystemDirectory(), OS.PTD() ==> OS.PresentTemporaryDirectory(), OS.PWD() ==> OS.PresentWorkingDirectory(), PID() ==> ProcessID()
Build 93173 ODD 02505 Script Language PRACTICE DO (Start PRACTICE script)	**PRACTICE error handling** PRACTICE scripts containing syntax errors are automatically popped (ENDDO) from PRACTICE stack during loadtime (DO <scriptname>) now.
Build 88288 IMP 02488 Script Language PRACTICE SPRINTF (Write formatted data to a PRACTICE macro)	**Use sprintf() style formatting for PRACTICE macros** The command SPRINTF allows to assign text and formatted data to a PRACTICE macro in the style of the sprintf() function of C/C++.

Build 88288 ODD
02485
Script Language PRACTICE
	renamed PRACTICE functions
	ADDRESS.WIDTH() replaced by ADDRESS.MAU()
	Data.MAU() replaced by ADDRESS.MAU()

Build 80996 CHG 02480 Script Language PRACTICE SCREEN.WAIT (Update screen while waiting)	**Extended functionality of SCREEN.WAIT command** SCREEN.WAIT now waits and updates the screen while waiting. It behaves similar to WAIT command. The old behavior of waiting until all processing windows are completed remained unchanged.

Build 80996 CHG
02468
Script Language PRACTICE
	renamed PRACTICE functions
	ADDRESS.SPACE() replaced by ADDRESS.ACCESS()
	HEADID() replaced by ID.PREPRO()

Build 76594 ODD
02462
Script Language PRACTICE
	renamed PRACTICE functions
	ADDRESS.DATA() replaced by ADDRESS.isDATA()
	ADDRESS.INTERMEDIATE() replaced by ADDRESS.isINTERMEDIATE()
	ADDRESS.ONCHIP() replaced by ADDRESS.isONCHIP()
	ADDRESS.PHYSICAL() replaced by ADDRESS.isPHYSICAL()
	ADDRESS.PROGRAM() replaced by ADDRESS.isPROGRAM()
	CPU.FAMILY() replaced by CPU.BASEFAMILY()

Build 76594 CHG 02459 Script Language PRACTICE RePeaT (Loop with check at end of loop)	**Syntax of PRACTICE command RePeaT changed** The complex syntax of the RePeaT command: RePeaT <count> <block> WHILE <condition> is no longer supported.

Build 76594 IMP 02449 Script Language PRACTICE	**new PRACTICE functions** additional PRACTICE functions introduced: ADDRESS.EXPANDACCESS(), ADDRESS.isNONSECURE(), ADDRESS.isNONSECUREEX(), ADDRESS.isSECURE(), ADDRESS.isSECUREEX(), Analyzer.FIRST(), ART.FIRST(), ATrace.RECORDS(), ATrace.REF(), ATrace.SIZE(), ATrace.STATE(), BMC.CLOCK(), BMC.OVERFLOW(), Break.Alpha.EXIST(), Break.Beta.EXIST(), Break.Charly.EXIST(), CAnalyzer.FIRST(), CONVert.ADDRESSTODUALPORT(), CONVert.ADDRESSTONONSECURE(), CONVert.ADDRESSTOSECURE(), CORE.ISACTIVE(), CORE.NAMES(), CPU.PINCOUNT(), Data.WSTRING.BigEndian(), Data.WSTRING.LittleEndian(), FORMAT.Time(), Integrator.FIRST(), INTERFACE.NAME(), IProbe.FIRST(), JTAG.X7EFUSE.CNTL(), JTAG.X7EFUSE.DNA(), JTAG.X7EFUSE.KEY(), JTAG.X7EFUSE.RESULT(), JTAG.X7EFUSE.USER(), LOGGER.FIRST(), MMU.INTERMEDIATE.VALID(), MMU.LINEAR.VALID(), MMU.LOGICAL.VALID(), MMU.PHYSICAL.VALID(), Onchip.FIRST(), OS.FILE.ABSPATH(), OS.FILE.REALPATH(), OS.PORTAVAILABLE.UDP(), OS.PORTAVAILABLE.TCP(), PER.Byte(), PER.HByte(), PER.Long(), PER.Long.BigEndian(), PER.Long.LittleEndian(), PER.LongLong(), PER.LongLong.BigEndian(), PER.LongLong.LittleEndian(), PER.PByte(), PER.Quad(), PER.Quad.BigEndian(), PER.Quad.LittleEndian(), PER.SByte(), PER.SLong(), PER.Short(), PER.Short.BigEndian(), PER.Short.LittleEndian(), PER.TByte(), PER.Word(), PER.Word.BigEndian(), PER.Word.LittleEndian(), PRINTER.FILENAME(), Probe.FIRST(), SNOOPer.FIRST(), sYmbol.AutoLOAD.CHECK(), sYmbol.AutoLOAD.CHECKCMD(), sYmbol.AutoLOAD.CONFIG(), sYmbol.EPILOG(), SYStem.CADIconfig.RemoteServer(), SYStem.CADIconfig.Traceconfig(), SYStem.CONFIG.XCP.INFO(), SYStem.CONFIG.XCP.INFO.STR(), SYStem.MCDconfig.LIBrary(), Trace.FIRST(), Trace.METHOD.ATrace(), TRANS.INTERMEDIATE.VALID(), TRANS.LINEAR.VALID(), TRANS.LOGICAL.VALID(), TRANS.PHYSICAL.VALID(), TrOnchip.IsAvailable(), TrOnchip.IsSet()
Build 69655 ODD 02447 Script Language PRACTICE	**renamed PRACTICE functions** SYStem.Option.MemoryMODEL() replaced by SYStem.Option.MEMoryMODEL()

Build 69655 IMP 02439 Script Language PRACTICE	**new PRACTICE functions** additional PRACTICE functions introduced: CERBERUS.IOINFO(), CERBERUS.IOINFO.IFLCK(), CPU.FAMILY(), CPU.SUBFAMILY(), CONVert.TIMENSTOINT(), Count.Frequency(), Count.Time(), COVerage.TreeWalk(), CTIBASE(), ERROR.ID(), ERROR.OCCURED(), FLASH.TARGET2.FILE(), FOUND.COUNT(), ID.PREPROcessor() as synonym for ID.PREPRO() added, math.FABS(), math.FMAX(), math.FMIN(), math.MAX(), math.MIN(), math.TimeMAX(), math.TimeMIN(), MCDS.Module.NAME(), MCDS.Module.NUMBER(), MCDS.Module.REVision(), MCDS.Module.TYPE(), STATE.OSLK(), SYStem.M() as synonym for SYStem.MODE() added, SYStem.BAUDRATE(), SYStem.CORECLOCK(), SYStem.OSCCLOCK(), TrOnchip.Set.C6Exit(), TrOnchip.Set.ENCLU(), WINdow.COMMAND()
Build 65657 02431 Script Language PRACTICE	**Default precision for parsed floating points increased** Default precision for parsed floating point values increased to 18 digits.
Build 65657 ODD 02429 Script Language PRACTICE WRITE (Write to data file)	**WRITE %CONTinue behavior changed** The behavior of "WRITE %CONTinue" was changed to harmonize the manner of commands PRINT and WRITE. old: %CONT surpresses the writing of CR/LF and effects the next writing new: %CONT removes the CR/LF from the last writing and concatenates the new data

Build 65657 ODD
02427
Script Language PRACTICE
	renamed PRACTICE functions
	CAnalyzer.PINS() replaced by CAnalyzer.PIN()
	CLOCK.DATE() replaced by DATE.DATE()
	CLOCK.DAY() replaced by DATE.DAY()
	CLOCK.MONTH() replaced by DATE.MONTH()
	CLOCK.SECONDS() replaced by DATE.SECONDS()
	CLOCK.TIME() replaced by DATE.TIME()
	CLOCK.UNIX() replaced by DATE.UnixTime()
	CLOCK.YEAR() replaced by DATE.YEAR()
	FCOS() replaced by math.FCOS()
	FEXP() replaced by math.FEXP()
	FEXP10() replaced by math.FEXP10()
	FINF() replaced by math.FINF()
	FLOG() replaced by math.FLOG()
	FLOG10() replaced by math.FLOG10()
	FNAN() replaced by math.FNAN()
	FSIN() replaced by math.FSIN()
	FSQRT() replaced by math.FSQRT()
	LINE.CALLER() replaced by PRACTICE.LINE.CALLER()
	WIN.EXIST() replaced by WINdow.EXIST()

Build 60219 IMP 02417 Script Language PRACTICE PMACRO.EXPLICIT (Enforce explicit PRACTICE macro declaration) PMACRO.IMPLICIT (Implicit PRACTICE macro declaration) PSTEPOUT (Back to caller)	**New PRACTICE commands** PMACRO.EXPLICIT enforces that macros are declared as PRIVAT, LOCAL or GLOBAL PMACRO.IMPLICIT allows to use an undeclared macro, undeclared macros are LOCAL be default PSTEPOUT allows to step out of the current PRACTICE block

Build 60219 CHG
02415
Script Language PRACTICE
	renamed PRACTICE functions
	COMBIPROBE() replaced by hardware.COMBIPROBE()
	DAS() replaced by interface.DAS()
	ESI() replaced by hardware.ESI()
	FIRE() replaced by hardware.FIRE()
	GDI() replaced by interface.GDI()
	HOSTMCI() replaced by interface.HOSTMCI()
	ICD() replaced by hardware.ICD()
	ICE() replaced by hardware.ICE()
	POWERDEBUG() replaced by hardware.POWERDEBUG()
	POWERINTEGRATOR() replaced by hardware.POWERINTEGRATOR()
	POWERINTEGRATOR2() replaced by hardware.POWERINTEGRATOR2()
	POWERNEXUS() replaced by hardware.POWERNEXUS()
	POWERPROBE() replaced by hardware.POWERPROBE()
	POWERTRACE() replaced by hardware.POWERTRACE()
	POWERTRACE2() replaced by hardware.POWERTRACE2()
	RETURNCODE() replaced by TERM.RETURNCODE()
	SCU() replaced by hardware.SCU()
	STG() replaced by hardware.STG()
	SYStem.UP() replaced by SYStem.Up()
	TA32() replaced by hardware.TA32()
	TO.COUNTER() replaced by TrOnchip.COUNTER()
	UTRACE() replaced by hardware.UTRACE()
	VERSION.SERIAL.PREPRO() replaced by VERSION.SERIAL.PREPROcessor()

Build 56057 ODD 02409 Script Language PRACTICE	**changed PRACTICE function** ITMBASE() has function parameter now

Build 56057 IMP 02407 Script Language PRACTICE	**new PRACTICE functions** additional PRACTICE functions introduced: CACHE.DC.LRU(), CACHE.IC.LRU(), CACHE.L2.LRU(), CACHE.L3.LRU(), CAnalyzer.CableTYPE(), CAnalyzer.I2C.DATA(), CHIP.GTMVersion(), Data.HByte(), Data.PByte(), Data.SByte(), MMU.FORMAT(), Onchip(), sYmbol.NAME.AT(), sYmbol.STATE(), SYStem.CONFIG.DEBUGPORT(), SYStem.CONFIG.DEBUGPORTTYPE(), SYStem.CONFIG.ListCORE(), SYStem.CONFIG.ListSIM(), SYStem.GTL.LIBname(), SYStem.GTL.PLUGINVERSION(), SYStem.GTL.VENDORID(), SYStem.GTL.VERSION(), SYStem.Option.MemoryMODEL(), SYStem.ReadPDRL(), SYStem.ReadPDRH(), Trace.METHOD.HAnalyzer(), Trace.STATE(), TrOnchip.Set.SMMINto(), TrOnchip.Set.SMMENtry(), TrOnchip.Set.SMMEXit(), TrOnchip.Set.VMENtry(), TrOnchip.Set.VMEXit(), TrOnchip.Set.VMEXitMask(), TrOnchip.Set.RESet(), TrOnchip.Set.GeneralDetect(), TrOnchip.Set.INIT(), TrOnchip.Set.MachineCheck(), TrOnchip.Set.ShutDown(), TrOnchip.Set.ColdRESet(), TrOnchip.Set.BootStall()
Build 51144 IMP 02383 Script Language PRACTICE SETUP.EDITEXT (Define an external editor)	**Editor plug-in for PRACTICE syntax highlighting** Syntax highlighting files for TextPad, UltraEdit, Kate and Notepad++ are provided under demo\practice\syntaxhighlighting
Build 51144 ODD 02379 Script Language PRACTICE	**removed PRACTICE functions** PRACTICE functions removed: Onchip.GAP() replaced by MCDS.TraceBuffer.SIZE(), MCDS.TraceBuffer.LowerGAP() and MCDS.TraceBuffer.UpperGAP() FLASH.LIST.TYPE() replaced by FLASH.SECTOR.TYPE()
Build 50104 CHG 02366 Script Language PRACTICE IF (Conditional script execution) RePeaT (Loop with check at end of loop) WHILE (Loop with check at start of loop)	**Empty lines in PRACTICE allowed** Empty lines are now ignored in PRACTICE. This may change the execution flow of existing programs (e.g. when an IF, REPEAT or WHILE statement is followed by an empty line).

Build 50104 IMP 02358 Script Language PRACTICE	**new PRACTICE functions** additional PRACTICE functions introduced: ADDRESS.INTERMEDIATE(), ADDRESS.PHYSICAL(), CHIP.EmulationDevice(), CONVert.OCTaltoint(), CPUFLASHTYPE() Data.MAU(), Data.STRingN(), DEBUGPORT(), DEBUGPORT.TYPE(), IFTEST.LATENCY(), IFCONFIG.DEVICENAME(), IFCONFIG.IPADDRESS(), PER.ARG(), PER.ARG.ADDRESS(), PORTSHARING(), RTS.RECORDS(), SYStem.CONFIG.DRPRE(), SYStem.CONFIG.DRPOST(), SYStem.CONFIG.IRPRE(), SYStem.CONFIG.IRPOST(), SYStem.CONFIG.Slave(), SYStem.DETECT.CLTapchain(), SYStem.OPBT(), TIMEOUT(), UTRACE(), VPU(), VPUCR()
Build 42354 IMP 02341 Script Language PRACTICE	**new PRACTICE functions** additional PRACTICE functions introduced: ART.RECORD.ADDRESS(), ART.RECORD.OFFSET(), ART.RECORD.TIME(), CAnalyzer.RECORD.ADDRESS(), CAnalyzer.RECORD.DATA(), CAnalyzer.RECORD.OFFSET(), LOGGER.RECORD.ADDRESS(), LOGGER.RECORD.DATA(), LOGGER.RECORD.OFFSET(), LOGGER.RECORD.TIME(), Onchip.RECORD.ADDRESS(), Onchip.RECORD.DATA(), Onchip.RECORD.OFFSET(), Onchip.RECORD.TIME(), SNOOPer.RECORD.ADDRESS(), SNOOPer.RECORD.DATA(), SNOOPer.RECORD.OFFSET(), SNOOPer.RECORD.TIME(), Trace.RECORD.ADDRESS(), Trace.RECORD.DATA(), Trace.RECORD.OFFSET(), Trace.RECORD.TIME(), CACHE.L2.SHARED(), CACHE.L2.SHAREDMASK(), DAP.Available(), Data.MAU(), Data.STRingN(), FLASH.UNIT(), FLASH.UNIT.BEGIN(), FLASH.UNIT.END(), FLASH.UNIT.EXIST(), FLASH.UNIT.NEXT(), FORMAT.CHAR(), FORMAT.STRing(), INTERFACE.CADI(), INTERFACE.MCD(), INTERFACE.VAST(), INTERFACE.VDI(), STATE.TARGET()

Build 38589 IMP 02294 Script Language PRACTICE	**new PRACTICE functions** additional PRACTICE functions introduced: Analyzer.MAXSIZE() ART.MAXSIZE(), ART.MODE(), ART.RECORDS(), ART.REF(), ART.SIZE(), ART.STATE(), ART.TRACK.RECORD(), BSDL.GetDRBit(), BSDL.GetPortLevel(), CIProbe.MAXSIZE(), CIProbe.SIZE(), COMPonent.AVAILABLE(), COMPonent.BASE(), COVerage.BDONE(), COVerage.SCOPE(), DIALOG.STRing2(), ETBCORESIGHT(), LA.MAXSIZE(), LA.SIZE(), LOGGER.SIZE(), NEXUS.PortSize(), Onchip.MAXSIZE(), PORT.MAXSIZE(), PORT.SIZE(), SNOOPer.MAXSIZE(), SNOOPer.SIZE(), STRing.Replace(), sYmbol.LIST.PROGRAM(), sYmbol.LIST.SOURCE(), Trace.MAXSIZE(), Trace.RECORDS(), Trace.SIZE(), VMX(), VMX.Guest(),

Build 34458 IMP 02267
Script Language PRACTICE
	new PRACTICE functions
	additional PRACTICE functions introduced:
	FALSE() and TRUE()
	STRing.TRIM()

Build 30461 FIX 02250 Script Language PRACTICE	**new PRACTICE functions** additional PRACTICE functions introduced: PowerIntegrator: Integrator.COUNTER.EVENT(), Integrator.COUNTER.EXTERN(), Integrator.COUNTER.TIME(), Integrator.FLAG() PowerProbe: Probe.COUNTER.EVENT(), Probe.COUNTER.EXTERN(), Probe.COUNTER.TIME(), Probe.FLAG() IDE: CONVert.TOLOWER(), DIALOG.EXIST(), FILE.EOF(), FILE.EOFLASTREAD(), FILE.EXIST(), FCOS(), FINF(), FNAN(), FSIN(), OS.PED()

Build 26464 IMP 02220
Script Language PRACTICE
	new PRACTICE functions
	additional PRACTICE functions introduced:
	CA.MAXSIZE(), CA.SIZE(), CA.REF(), CA.STATE()
	I.MAXSIZE(), I.SIZE(), IPROBE.SIZE(), PROBE.SIZE()

Build 22490 IMP 02212 Script Language PRACTICE	**new PRACTICE functions** additional PRACTICE functions introduced: ADDRESS.SEGMENT(), ADDRESS.WIDTH(), CA.RECORD.TIME(), CONFIG.SCREEN(), DEBUGGER(), MCDS.GAP(), MCDS.SIZE(), MCDS.STATE(), ONCHIP.GAP(), ONCHIP.RECORDS(), ONCHIP.REF(), ONCHIP.SIZE(), ONCHIP.STATE(), ONCHIP.TRACK.RECORD() PER.BUFFER.BYTE(), PER.BUFFER.LONG(), PER.BUFFER.LONGLONG(), PER.BUFFER.QUAD(), PER.BUFFER.SHORT(), PER.BUFFER.WORD(), TITLE(), VERSION.DATE(), VERSION.SERIAL.CABLE(), VERSION.SOFTWARE()
Build 22490 IMP 02191 Script Language PRACTICE OPEN (Open data file)	**maximum opened files PRACTICE files** The limit of maximal 20 parallel opened files increased to 120. It's independent from the maximal number of opened sourcefiles.
Build 7181 IMP 02002 Script Language PRACTICE PRIVATE (Create private PRACTICE macro)	**New PRACTICE command PRIVATE** Creates only local accessible macro.
Build 1145 IMP 01563 Script Language PRACTICE	new commands DODECRYPT and ENCRYPTDO implemented to handle encrypted PRACTICE files
Build 1077 IMP 01109 Script Language PRACTICE	PRACTICE parameter passing made reentrant

SmartTrace

Build 1079 IMP 01399 ARM/Cortex Trace (parallel) SmartTrace	SmartTrace for ARM improved
Build 1079 CHG 01396 SmartTrace	Saved data for flowtraces changed when FIFOFULLs are saved (to keep smarttrace info)
Build 1077 FIX 01165 Trace-based Debugging (CTS) MPC56x NEXUS Debugger and Trace SmartTrace	**cts** Several fixed in CTS/Smarttrace for PowerPC Nexus

Snooper

Build 100486 02536 Snooper SNOOPer.CORE (Select cores for PC snooping)	**Core-specific PC snooping for SMP systems** Allows to specify core(s) for SNOOPer.Mode PC in SMP systems.

StarCore Debugger

Build 1241 FIX 01798 StarCore Debugger	**SDRAM r/w problem on EB1000 board fixed**
Build 1228 FIX 01771 MIPS32/MIPS64 Debugger StarCore Debugger	**Some ICDs fail on USB2** MIPS5k and StarCore fail at system.up or single step.

Stimuli Generator

Build 125398 CHG 02581 EPROM/FLASH Simulator PowerIntegrator - Logic and Bus Analyzer PowerProbe - Logic and Protocol Analyzer TRACE32 Software Stimuli Generator	**discontinued software and hardware support** The t32pbi executables aren't supported anymore.

Build 1190 IMP 01683 C166CBC Debugger Super10 Debugger FIRE Emulator for C166 Family In-Circuit Emulator for C166/ST10 - Out of Production SIM Instruction Set Simulator for C166/ST10	**new access mode for 80166 family** EDBG: added
Build 1077 IMP 01187 Super10 Debugger Super10 NEXUS Debugger and Trace	Super10 cycle alignment improved
Build 1077 FIX 01144 Super10 Debugger Super10 NEXUS Debugger and Trace	scxt was wrong dequeued in Super10 Nexus

Build 1268 IMP 01869 Super10 NEXUS Debugger and Trace Break.Set (Set breakpoints)	**Onchip breakpoints improved for Super10** NOT bit for improved filtering and exclude breakpoints (R304 cell). Data only breakpoints for stack variables (R305 cell).
Build 1204 FIX 01725 Super10 NEXUS Debugger and Trace	**breakpoint setting** address events deleted under circumstances (/ONCHIP and starting emulation from software breakpoint only) effects analyzer triggerunit programming too
Build 1203 FIX 01722 Super10 NEXUS Debugger and Trace Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**analyzer trigger programming** address events set wrong under circumstances (/HARD only)
Build 1190 FIX 01692 Super10 NEXUS Debugger and Trace Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Programming of complex triggerunit** now a possible necessary second programming of complex triggerunit is superfluous in the case of DTM or WTM usage inside an ADDR declaration
Build 1190 FIX 01659 Super10 NEXUS Debugger and Trace Analyzer.Program (Program trigger unit) Analyzer.ReProgram (Program trigger unit)	**Programming of complex triggerunit** now a possible necessary second programming of complex triggerunit is superfluous in the case of DTM or WTM usage inside an ADDR declaration
Build 1190 FIX 01650 Super10 NEXUS Debugger and Trace	**Analyzer flow processing by loop endless** Analyzer.Find backward (or simmilar commands) may hang when selective tracing (with TraceON/TraceOFF berakpoints) is made.
Build 1133 FIX 01481 Super10 NEXUS Debugger and Trace	Super10 local registersets where wrong in register window, L3 was missing for M345
Build 1133 FIX 01472 Super10 NEXUS Debugger and Trace	Super10 trace disassembly/alignment failed in certain repeat block constructs
Build 1133 CHG 01468 Super10 NEXUS Debugger and Trace PER (Peripheral files)	peripheral files for R303 and super10 renamed to perm340.per
Build 1096 IMP 01458 Super10 NEXUS Debugger and Trace	Data cycle alignment for direct address selective write tracing improved on Super10 Nexus
Build 1096 IMP 01439 Super10 NEXUS Debugger and Trace	Super10 filter assignment improved to have no gaps in filter usage

Build 1096 FIX 01433 Super10 NEXUS Debugger and Trace	Selectively tracing overlayed programs on Super10 fails when last message before SBM was OTR message
Build 1084 FIX 01413 Super10 NEXUS Debugger and Trace	Super10 TraceOFF breakpoint at end of trace may cause long time "tracking" display
Build 1079 IMP 01402 Super10 NEXUS Debugger and Trace	Super10 TraceON / TraceOFF breakpoints can be set on-the-fly
Build 1079 FIX 01392 Super10 NEXUS Debugger and Trace	Super10: new command TrOnchip.BusTriggerDE for on/off triggers on PODBUS
Build 1077 FIX 01374 MPC56x NEXUS Debugger and Trace Super10 NEXUS Debugger and Trace	A.VIEW may crash on Super10 Nexus or PPC Nexus
Build 1077 FIX 01371 Super10 NEXUS Debugger and Trace	Super10 watchpoints and filters where not set during realtime run
Build 1077 IMP 01303 Super10 NEXUS Debugger and Trace	Several enhancements for Super10 filtered program traceing
Build 1077 FIX 01238 Super10 NEXUS Debugger and Trace	Several fixes and workarrounds for Super10 Nexus
Build 1077 FIX 01231 Super10 NEXUS Debugger and Trace	Cycle alignment for Super10 coxxx instructions fixed
Build 1077 IMP 01227 Super10 NEXUS Debugger and Trace	**WAF** Super10 WAF bit supported
Build 1077 IMP 01187 Super10 Debugger Super10 NEXUS Debugger and Trace	Super10 cycle alignment improved
Build 1077 FIX 01185 Super10 NEXUS Debugger and Trace	Super10 tracing of trap #xx and comac with opcode #83 fixed
Build 1077 FIX 01144 Super10 Debugger Super10 NEXUS Debugger and Trace	scxt was wrong dequeued in Super10 Nexus
Build 1077 CHG 01063 Super10 NEXUS Debugger and Trace	Disassembler / cycle info C166 changed to support Super10

System Trace

Build 76594 IMP 02457 System Trace LA.IMPORT.STPByteStream (Import STP recording from file (byte))	**Import STP byte stream for processing in TRACE32** The command allows to import a STP byte stream from a file into TRACE32. The imported STP byte stream can then be displayed and analyzed.

TMS320C54XX Debugger

Build 1228 CHG 01780 TMS320C54XX Debugger C5500 Debugger	**accessmodes** additional access modes ETB: EETB:

TPU Debugger (68332, MPC55x/56x)

Build 1145 FIX 01569 TPU Debugger (68332, MPC55x/56x) TPU.Step (Single step TPU)	TPU command was locked on CPU32 BDM debuggers
Build 1077 FIX 01111 TPU Debugger (68332, MPC55x/56x) TPU.List (View microcode)	**TPU Disassembler** TPU Disassembler bug fixed for constant +1/-1 add

TRACE32 Front-End

Build 125398 CHG 02598 TRACE32 Front-End TRACE32 Software	**discontinued software and hardware support** This is the last TRACE32 software release which supports TRACE32 Frontends for Virtio (VDI) and VaST.
Build 38589 IMP 02323 TRACE32 Front-End	**Multi-Core support (AMP & SMP)** Multi-core virtual targets are support via MCD interface in SMP and AMP cases.

Build 19417 IMP 02185 TRACE32 Instruction Set Simulator for CORTEX / ARM / XSCALE SIM.CACHE (Cache/MMU simulation and more)	**MMU and Cache support in ARM simulator** MMUs and caches can also be simulated now.
Build 4728 IMP 01974 ARM Debugger TRACE32 Instruction Set Simulator for CORTEX / ARM / XSCALE	**New command to fix literal attibutes** The command sYmbol.CLEANUP.CodeLiterals fixes literal attributes that really contain code (instructions synthesized by a define word).
Build 4728 IMP 01963 ARM Debugger TRACE32 Instruction Set Simulator for CORTEX / ARM / XSCALE	**Unified Assembly Language supported** SETUP.DIS command extended to choose between traditional disassembly and unified assembly language mode.
Build 1258 FIX 01852 ICD Solutions for ARM TRACE32 Instruction Set Simulator for CORTEX / ARM / XSCALE TERM.GATE (Terminal with virtual hosting)	**ARM semihosting file append not working** Append to existing files (e.g. fopen with "a+" mode) was not working as expected.
Build 1246 FIX 01820 TRACE32 Instruction Set Simulator for CORTEX / ARM / XSCALE	**ARM Simulator LDR pc,xx did no switch** LDR pc,xx did no ARM/Thumb mode switch on ARM architecture V5e and above.
Build 1190 IMP 01684 ARM Debugger ARM9 JTAG Debugger FIRE Emulator for ARM7 In-Circuit Emulator for ARM7 TRACE32 Instruction Set Simulator for CORTEX / ARM / XSCALE	**new accessmodes for ARM family** EC0:..EC15: EDEBG: EETB: EETM: EICE: EINI: added
Build 1161 IMP 01620 FIRE Emulator for ARM7 ICD Solutions for ARM TRACE32 Instruction Set Simulator for CORTEX / ARM / XSCALE	**Access modes** additional access mode ETB:
Build 1077 FIX 01333 TRACE32 Instruction Set Simulator for CORTEX / ARM / XSCALE	**ARM** ARM simulator was reacting wrong when CPSR_C equal zero
Build 1077 FIX 01245 TRACE32 Instruction Set Simulator for CORTEX / ARM / XSCALE	**ARM** Bug fixed in ARM simulator exception handling (PSR.I and PSR.F where not cleared)

TRACE32 Instruction Set Simulators

Build 69655 CHG 02445 TRACE32 Instruction Set Simulators	**New TRACE32 Simulator License** The TRACE32 Instruction Set Simulator allows to perform 50 PRACTICE script commands/API operations after the first "single-step" or "Go". If you want to perform further PRACTICE script commands/API operations a TRACE32 Simulator License is needed.
Build 2940 IMP 01911 JTAG Debugger FIRE Fully Integrated RISC Emulator ICE In-Circuit Emulator PowerTrace for NEXUS TRACE32 Instruction Set Simulators	**Practice functions** synonym D.SUM() for DATA.SUM() added
Build 1151 FIX 01592 TRACE32 Instruction Set Simulators	PORT analyzer in Simulator with SLAVE mode may cause protection fail

TRACE32 Integration to Eclipse

Build 69655 IMP 02440 TRACE32 Integration to Eclipse TRACE32 Integration to Wind River Workbench	**TRACE32 as TCF Agent** If TCF= is added to the TRACE32 configuration file, TRACE32 is started as a TCF agent. This allows to use the WindRiver Workbench or the Eclipse debugger as IDE and TRACE32 as debugging back end.

TRACE32 Integration to Wind River Workbench

Build 69655 IMP 02440 TRACE32 Integration to Eclipse TRACE32 Integration to Wind River Workbench	**TRACE32 as TCF Agent** If TCF= is added to the TRACE32 configuration file, TRACE32 is started as a TCF agent. This allows to use the WindRiver Workbench or the Eclipse debugger as IDE and TRACE32 as debugging back end.

TRACE32 Maintenance Check

Build 88288 CHG 02499 TRACE32 Maintenance Check	**Demo time for invalid/missing maintenance reduced** A 10 minutes demo time is given when no valid maintenance exists.
Build 13751 IMP 02104 TRACE32 Maintenance Check	**Floating licenses supported** Frontend debuggers (via API) are using a floating license.

TRACE32 Software

Build 125398 CHG 02598 TRACE32 Front-End TRACE32 Software	**discontinued software and hardware support** This is the last TRACE32 software release which supports TRACE32 Frontends for Virtio (VDI) and VaST.
Build 125398 CHG 02596 In-Circuit Emulator FIRE Fully Integrated RISC Emulator Host Driver Software Integrated Development Environment TRACE32 Software	**restricted TRACE32 FIRE software development** This is the last TRACE32 software release which contains FIRE software derived from the standard TRACE32 development trunk. A separate branch from the TRACE32 software will be created for supporting FIRE hardware only. Only bugfixes and host OS adaptations are planned for the future.
Build 125398 CHG 02581 EPROM/FLASH Simulator PowerIntegrator - Logic and Bus Analyzer PowerProbe - Logic and Protocol Analyzer TRACE32 Software Stimuli Generator	**discontinued software and hardware support** The t32pbi executables aren't supported anymore.
Build 125398 CHG 02579 In-Circuit Emulator Host Driver Software ICE In-Circuit Emulator Integrated Development Environment TRACE32 Software	**restricted TRACE32 ICE software development** This is the last TRACE32 software release which contains ICE software derived from the standard TRACE32 development trunk. A separate branch from the TRACE32 software will be created for supporting ICE hardware only. Only bugfixes and host OS adaptations are planned for the future.
Build 42354 IMP 02343 TRACE32 Software	**first official release supporting Power Debug Interface USB3**

Teak/TeakLite Debugger

Build 4790 FIX 01983 Teak/TeakLite Debugger	**Stack backtrace on TEAK not working**
Build 1146 CHG 01578 Teak/TeakLite Debugger	OAK: usage of memory class DBG supported now
Build 1077 IMP 01267 Teak/TeakLite Debugger	**Assembly source level debug** OAK assembly source level debug added

Timing and Protocol Analyzer

Build 1161 CHG 01640 Timing and Protocol Analyzer	**Programming of complex triggerunit** command renames: Count.* -> Counter.* Count.Enable -> Counter.Increment declaration renames: EVENT -> EVENTCOUNTER TIME -> TIMECOUNTER declarationtype synonymes introduced EVCNT for EVENTCOUNTER TICNT for TIMECOUNTER

Build 125398 IMP 02594 Trace Analyzer <trace>.Find (Find specified entry in trace) <trace>.FindAll (Find all specified entries in trace) <trace>.List (List trace contents) <trace>.ListVar (List variable recorded to trace)	**new option /TASK for Trace commands** new option /TASK for Trace.List, Trace.Find, Trace.FindAll and Trace.ListVar to filter for tasks.
Build 125398 CHG 02591 Trace Analyzer <trace>.EXPORT.cycles (Export trace data) LA.IMPORT.cycles (Import bus trace data)	**renamed commands Trace.EXPORT.flow and LA.IMPORT.flow** The commands Trace.EXPORT.flow and LA.IMPORT.flow have been renamed to Trace.EXPORT.cycles and LA.IMPORT.cycles.
Build 125398 IMP 02589 Trace Analyzer ETM.TraceDataPriority (Define data trace priority)	**new command ETM.TraceDataPriority** New command ETM.TraceDataPriority to define data trace priority on ETMv4.
Build 125398 IMP 02587 ARM Debugger CEVA-X Debugger Trace Analyzer ETM.TraceCORE (Core specific default tracing)	**new command ETM.TraceCORE** New command ETM.TraceCORE to allow core specific default tracing.
Build 125398 IMP 02584 Trace Analyzer <trace>.STATistic.TASKLOCK (Analyze lock accesses from tasks)	**new command Trace.STATistic.TASKLOCK** New command introduced to analyze lock accesses from tasks in the trace.
Build 117056 IMP 02577 Trace Analyzer <trace>.STATistic.TASKORINTRState (Task and ISR2 statistic analysis)	**new trace commands** New commands Trace.STATistic.TASKORINTRState and Trace.Chart.TASKORINTRState introduced to analyze tasks and (ORTI) interrupts.
Build 112182 IMP 02570 JTAG Debugger Trace-based Code Coverage Trace Analyzer	**preset filter for coverage command group added** Preset filter introduced to improve tree views.
Build 105499 IMP 02553 Trace Analyzer <trace>.Chart.RUNNABLE (Runnable activity chart)	**New commands to analyze AUTOSAR runnable trace information**

Build 100486 IMP 02527 Trace Analyzer IProbe.state (Display the IProbe configuration window)	**IProbe command group is supported by PowerTrace Serial** As soon as the IProbe is switched from the DISable state to OFF/Arm state, the trace memory is split up. 50% of the trace memory can still be used to record the trace data exported via the Aurora/PCIe trace port. The other 50% are assigned to the IProbe.
Build 93173 IMP 02518 Trace Analyzer GROUP.Create (Create a new group)	**GROUP command group supports access classes and machineIDs** GROUP command group now supports access classes (such as N: Z: H: for the ARM architecture) and MachineIDs.
Build 93173 IMP 02516 Trace Analyzer BTrace.state (Display BTrace configuration window)	**Command group BTrace enables script controlled trace sink** New command group BTrace allows a script controlled trace sink. First use case: apply Trace.STATistic commands to function run-times measured with the benchmark counters of a RH850 debugger.
Build 88288 02496 Trace Analyzer <trace>.Chart.DatasYmbol (Analyze pointer contents graphically) <trace>.Chart.GROUP (Group activity chart) <trace>.Chart.TASK (Task activity chart) <trace>.Chart.sYmbol (Symbol analysis) <trace>.Timing (Waveform of trace buffer)	**Chart and timing windows with Track option keep zoom** The zoom factor of the <trace>.Chart and <trace>.Timing windows is retained, even if the trace content changes, if the /Track option is used.
Build 80996 IMP 02478 Trace Analyzer <trace>.Find (Find specified entry in trace) <trace>.FindAll (Find all specified entries in trace)	**Search features for Trace.Find/Trace.FindAll extended** CHANGE <search_item>: allows searching only for changes in search item. Improved data value search (multiple values, multiple ranges and not matching data).
Build 76594 IMP 02460 Trace Analyzer <trace>.STATistic.Func (Nesting function runtime analysis) <trace>.STATistic.TREE (Tree display of nesting function run-time analysis)	**Option /TASK !<task_name> to exclude task from statistic** The option /TASK !<task_name> allows to exclude the specified task from the nesting statistic, if it causes nesting problems.
Build 76594 IMP 02456 Trace Analyzer <trace>.ListVar (List variable recorded to trace)	**List variables in trace** List all variable recorded to trace.
Build 76594 IMP 02452 Trace Analyzer	**Real-time processing for HTM** Real-time processing for HTM is now possible.

Build 69655 IMP 02448 Trace Analyzer Data.Find (Search in memory) Data.FindCODE (Execute command on specified code type) <trace>.Find (Find specified entry in trace)	**New option /ALL for Trace.Find command** With the option /ALL the TRACE32 Message Line displays how often the searched item was found. The number of occurrences can be processed in a script via the function FOUND.COUNT().
Build 65657 02436 Trace Analyzer <trace>.Find (Find specified entry in trace) <trace>.FindAll (Find all specified entries in trace)	**Improved match for search in trace** Search for addresses finds now also cycles that are close and touch the given address.
Build 56057 IMP 02410 Trace Analyzer <trace>.EXPORT.CSVFunc (Export the function nesting to a CSV file) <trace>.EXPORT.TASKEVENTS (Export task event to CSV)	**New Trace export commands** New commands to export trace information for analysis by a third-party tool.
Build 9640 IMP 02036 Trace Analyzer	**A.MODE()** new PRACTICE function A.MODE() introduced which returns the actual mode of the analyzer

Trace Features

Build 69655 IMP 02442 High-Level-Language Debugging Trace Features <trace>.DRAW.channel (Plot no-data values against time) <trace>.DRAW.Data (Plot data values against time) <trace>.DRAW.Var (Plot variable values against time) Data.DRAW (Graphical memory display of arrays) Data.PROfile (Graphical display of data value) Var.PROfile (Graphical display of variable)	**Multiple variables of different type can be drawn** TRACE32 allows to draw multiple variables/channels with different formats e.g. integers and floats for all listed commands.

Build 130863 IMP 02608 Trace-based Code Coverage COVerage.EXPORT.ListCalleEs (Export the function callees) COVerage.EXPORT.ListCallеRs (Export the function callers) COVerage.ListCalleEs (Display coverage for callees function) COVerage.ListCalleRs (Display coverage for callers function)	**new COVerage commands** New commands COVerage.ListCalleRs, COVerage.ListCalleRs, COVerage.EXPORT.ListCalleEs and COVerage.EXPORT.ListCalleRs. The new commands allow to list/export coverage information for callers and callees of functions.
Build 130863 IMP 02605 Trace-based Code Coverage sYmbol.ECA.LOAD (Load a single ECA file) sYmbol.ECA.LOADALL (Load all ECA files)	**new options /SkipErrors and /LENient for sYmbol.ECA.LOAD** The /SkipErrors option ensures that warnings are issued instead of error messages. For scripts, error messages cause the script to stop. Warnings keep the script running. The /LENient option allows loading of ECA files with minor errors as invalid file version or checksum mismatch.
Build 112182 IMP 02570 JTAG Debugger Trace-based Code Coverage Trace Analyzer	**preset filter for coverage command group added** Preset filter introduced to improve tree views.
Build 93173 IMP 02517 Trace-based Code Coverage COVerage.EXPORT (Export code coverage information to an XML file)	**Improved COVerage.EXPORT for C++ projects** Unique identifiers are provided as additional property for functions. They can be used to identify functions in case of overloading.
Build 93173 IMP 02515 Trace-based Code Coverage COVerage.Option.SourceMetric (Select code coverage metric)	**TRACE32 supports now MCDC coverage** TRACE32 supports now MCDC coverage without code instrumentation. Requires generation of *.eca (Extended Code Analysis) files with preprocessor t32cast.exe. t32cast.exe can be found in ~~/demo/t32cast/bin.
Build 80996 CHG 02473 Trace-based Code Coverage	**COVerage.ACCESS command is deprecated** COVerage.ACCESS is deprecated. Please use Trace.ACCESS instead.

Trace-based Debugging (CTS)

Build 1077 FIX 01165 Trace-based Debugging (CTS) MPC56x NEXUS Debugger and Trace SmartTrace	**cts** Several fixed in CTS/Smarttrace for PowerPC Nexus
Build 1077 FIX 01145 Trace-based Debugging (CTS)	**Function nesting display** Function nesting display in CTS may not be correct

Trace-based Profiling

Build 100486 CHG 02542 Trace-based Profiling ISTATistic.ADD (Add trace contents to ISTAT database)	**PC snooping can be used by ISTAT commands** TRACE32 takes instruction execution information from Analyzer/Onchip trace and PC/timestamp information from the SNOOPer trace in order to improve timestamp accuracy.
Build 80996 ODD 02470 Trace-based Profiling <trace>.STATistic.Func (Nesting function runtime analysis) <trace>.STATistic.TREE (Tree display of nesting function run-time analysis)	**Average time is now calculated for complete function run** Average time values are now calculated only for complete function runs. Previously the average time was calculated by total time divided by count.
Build 69655 IMP 02446 Trace-based Profiling <trace>.Chart.sYmbol (Symbol analysis)	**FIFOFULL is now shown in Trace.Chart.Symbol window**
Build 60219 02418 Trace-based Profiling <trace>.STATistic.INTERRUPT (Interrupt statistic) <trace>.STATistic.TASKORINTERRUPT (Statistic of interrupts and tasks) <trace>.STATistic.TASKVSINTERRUPT (Statistic of interrupts, task-related)	**Improved analysis of interrupt timing** Trace.STATistic.INTERRUPT - analyses timing of interrupts Trace.STATistic.TASKORINTERRUPT - analyses timing of tasks and interrupts Trace.STATistic.TASKVSINTERRUPT - analyses timing of tasks and interrupts, interrupts are analyzed in task context

TriCore AGBT with PowerTrace Serial

Build 88288 IMP 02494 TriCore AGBT with PowerTrace Serial TriCore AGBT with Serial Preprocessor RTS.state (Open status and control window)	**RTS implemented for TriCore** RTS implemented for TriCore architecture.

TriCore AGBT with Serial Preprocessor

Build 88288 IMP 02494 TriCore AGBT with PowerTrace Serial TriCore AGBT with Serial Preprocessor RTS.state (Open status and control window)	**RTS implemented for TriCore** RTS implemented for TriCore architecture.

Build 130863 IMP 02620 TriCore Debugger <trace>.List (List trace contents)	**data-cycle assignment for Aurix implemented** The traced data-cycles are automatically assigned to their refering program-cycles in the Trace.List window now.
Build 112182 CHG 02568 TriCore Debugger	**Support for OCTL removed**
Build 105499 IMP 02560 XC2000/C166SV2 Debugger GTM Debugger (TriCore) PCP Debugger (TriCore) TriCore Debugger	**improved MCDS trace decoder speed**
Build 105499 IMP 02557 TriCore Debugger Data.STANDBY (Standby data-sequences)	**new command Data.STANDBY added**
Build 60219 02425 TriCore Debugger	**Fixed erroneous read out of MCDS onchip trace.**
Build 42354 IMP 02347 TriCore Debugger TriCore Parallel Trace FLASH.Create (Declare FLASH)	**new command option /BootModeHeaDer**
Build 38589 IMP 02299 GTM Debugger (TriCore) MPC5xxx and SPC5xx Debugger TriCore Debugger	**initinal support for BOSCH GTM added** Initinal support for the BOSCH GTM (Generic Timer Module) added. Available in TriCore and PowerPC devices.
Build 38589 IMP 02297 GTM Debugger (TriCore) PCP Debugger (TriCore) TriCore Debugger CLOCK (Display date and time)	**on-chip clock display and settings** The CLOCK command group simplifies the setup for timestamp decoding of an on-chip trace. Can also be used to verify PLL and clock setup.
Build 13751 IMP 02083 TriCore Debugger CACHE.view (Display cache control register)	**Cache Dump added** For the members of the AUDO-FUTURE chip family, the Cache Dump features have been added.
Build 13751 IMP 02062 TriCore Debugger	**Cache Analysis added** For the members of the AUDO-NG and TC116x familiy the Cache Analysis feature has been added.

Build 13751 IMP 02060 TriCore Debugger TrOnchip (Onchip triggers)	**MCBS feature added for TriCore and PCP** Newer TriCore chips (e.g. the AUDO-NG or TC11xx family) have two internal break buses and a suspend bus which can be used to distribute break signals beween different modules (MLI, DMA, ...) and cores (TriCore, PCP). Configuration of this feature can be done via the TrOnchip window or the TrOnchip commands.
Build 9640 IMP 02034 TriCore Debugger SYStem.Option STEPSOFT (Step with software breakpoints)	**TriCore soft single-stepping added** TriCore single-stepping can now also be performed by using software breakpoints. Use SYStem.Option STEPSOFT ON to enable this feature. The default is using on-chip breakpoints.
Build 9640 IMP 02023 TriCore Debugger SYStem.RESetOut (Reset peripherals)	**Target reset feature added** SYStem.RESetOut triggers a 2 ms low pulse on the nRESET line on the JTAG connector. This will reset the CPU. Formerly this command was only available in the Simulator.
Build 8248 FIX 02025 TriCore Debugger Break.Set (Set breakpoints)	**Support for 4 Onchip Breakpoints in Code**
Build 7181 IMP 02012 TriCore Debugger	**TriCore: PCP single stepping improved** The single stepping for the PCP subcore debugger is improved. The channel is no more disabled when a software breakpoint is hit. So no dcr error is generated any more, e.g. when single stepping certain instructions.
Build 7181 FIX 02010 TriCore Debugger	**TC1130 window flickering fixed** Data or peripheral windows were flickering when at least one bus error was displayed in some windows. Affected TC1130, TC1115 and TC1100.
Build 7181 CHG 01999 TriCore Debugger SYStem.Option (Special setup)	**TriCore SYStem.Option change** After a transition period, the following TriCore SYStem.Options will become obsolete: SYStem.Option TC1130FIX SYStem.Option TC1796FIX SYStem.Option TC1900FIX These options were used for disabling the internal watchdog on SYStem.Mode [Up|Go|Attach] as a bugfix. As replacement was introduced: SYStem.Option WATCHDOGFIX The functionality will exactly be the same, except that it has only effect on TriCore Chips that have the corresponding bug. For disabling the watchdog (e.g. for flash programming) please use Data.Set 0xF0000024 %Long 0x00000008 after the SYStem.Up command. See the demo scripts for an example.
Build 7117 ERR 01975 TriCore Debugger	**TriCore Data Cache** For TriCore devices with Data Cache (DCACHE in DMI, e.g. TC10GP, TC11IB, TC1100, TC1115 and TC1130) it is not possible writing data back to memory. Since the debugger can not access the data cache, Data.dump windows with cached memory will contain outdated (wrong) data. Workaround: If possible, disable data cache or link to uncached locations. Note: This does not affect the data cached by the EBU.

Build 4728 CHG 01979 TriCore Debugger TriCore Parallel Trace SIM Instruction Set Simulator for TriCore	**accessmodes** additional access modes: PCD: PCP: EPCD: EPCP: deleted accessmode: PCP:
Build 4728 FIX 01960 TriCore Debugger TrBus.Out (Define source for the external trigger pulse)	**Trigger on target break** Added option to generate a PODBUS trigger pulse when target breaks.
Build 2940 IMP 01889 TriCore Debugger RunTime.state (Display results)	**RunTime measurement** Runtime Measurement is more accurate now.
Build 1256 IMP 01850 TriCore Debugger PER.view (Display peripherals)	**perfile for TC1796 revised** various errors corrected, support for P-Bit protected registers added, improved descriptions and layout
Build 1241 FIX 01791 TriCore Debugger Data.EPILOG (Automatic data modification on program execution halt)	**Data.Epilog did not execute** Data.EPILOG was not executed when breaking
Build 1224 IMP 01753 TriCore Debugger SYStem.Mode (Select mode)	**Tricore ATTACH mode implemented** Attach allows to connect to running target. New feature is available on all "Power" tools.
Build 1133 IMP 01507 TriCore Debugger	fract, sfract, laccum data types supported (for Tricore)
Build 1096 IMP 01450 TriCore Debugger	separate perfile for PCP2 (Tricore)
Build 1077 IMP 01367 TriCore Debugger	Tricore DBGSR register used to detect break reason
Build 1077 IMP 01340 TriCore Debugger	Advanced breakpoints for Tricore implemented (Trace control, MEMORY, STACK)
Build 1077 FIX 01161 TriCore Debugger TriCore Parallel Trace	**Error handling** Error handling in Tricore OCDS-L2 trace may cause software to crash

Build 42354 IMP 02347 TriCore Debugger TriCore Parallel Trace FLASH.Create (Declare FLASH)	**new command option /BootModeHeaDer**
Build 38589 IMP 02295 TriCore Parallel Trace	**TriCore AGBT (Serial Trace)** Support for the TriCore AGBT (Aurora GigaBit Trace) has been added. Requirements for this feature is a Serial Trace v2 preprocessor and a TriCore AURIX Emulation Device. Demonstrator devices are not supported.
Build 4728 CHG 01979 TriCore Debugger TriCore Parallel Trace SIM Instruction Set Simulator for TriCore	**accessmodes** additional access modes: PCD: PCP: EPCD: EPCP: deleted accessmode: PCP:
Build 1077 FIX 01161 TriCore Debugger TriCore Parallel Trace	**Error handling** Error handling in Tricore OCDS-L2 trace may cause software to crash

Trigger Probe for PODBUS

Build 80996 CHG 02476 Trigger Probe for PODBUS TrPOD.state (State display)	**Command group TrPod renamed to TrPOD** Command group TrPod renamed to TrPOD.

Build 100486 IMP 02535 Uniform Look-And-Feel for all Processors SHA1SUM (Calculate SHA1 checksum of a file)	**New command to calculate SHA-1 checksum for a file** The SHA1SUM command calculates a 160 bit checksum for the given files using the Secure Hash Algorithm.
Build 100486 IMP 02532 Uniform Look-And-Feel for all Processors TERM.view (Terminal display) TERM.METHOD (Select terminal protocol)	**Multiple open TERMinal windows are now possible** A channel number was introduced to allow the concurrent use of several TERMinal windows.
Build 100486 02525 Uniform Look-And-Feel for all Processors Break.DeletePATtern (Delete breakpoints allowing wildcards) Break.PATtern (Set temporary breakpoints allowing wildcards) Break.SetPATtern (Set breakpoints allowing wildcards)	**Set/delete breakpoints on symbol pattern** Breakpoints can be set on symbol pattern.
Build 100486 IMP 02521 Uniform Look-And-Feel for all Processors Break.List (Display list of breakpoints)	**Break.List window was revised** Enabling/disabling breakpoints is now possible with a single click to the check mark in the new "icon" column. Change Breakpoint window is opened by pushing the Change button. Double click to a Program breakpoint opens now source list window (List command). Double click to a Read or Write breakpoint opens Data.View window.
Build 56057 IMP 02411 Uniform Look-And-Feel for all Processors WELCOME.view (Open the welcome window) WELCOME.SCRIPTS (Open the script search window) WELCOME.STARTUP (Open the welcome window if not disabled)	**Welcome dialog and script search** Welcome dialog for new installations and search dialog for scripts (debugger software only)
Build 34458 02259 Uniform Look-And-Feel for all Processors PEDIT (Edit PRACTICE script)	**pedit not closing file after close**
Build 1151 IMP 01598 Uniform Look-And-Feel for all Processors Data.IMAGE (Display image data)	new formats added to Data.IMAGE command (TrueColor formats RGB24 and YUV422)

Universal EPROM/Flash Adapter

Build 1084 IMP 01408 Universal EPROM/Flash Adapter	ESICON trace supports two baseaddresses for address reconstruction (sys.ots ...)

Unspecific Worktime Account

Build 9640 IMP 02035 Unspecific Worktime Account APU.GREP (Search for string)	**APU.GREP command implemented** APU.GREP works similar to Data.GREP, but opens APU.* windows instead of Data.* windows.

XA51 Monitor

Build 125398 CHG 02597 JTAG Debugger Debugger for 80C196EA MMDSP Debugger Host Driver Software Integrated Development Environment x186 Monitor x196 Monitor XA51 Monitor TRACE32 Software	**restricted TRACE32 ICD software development** This is the last TRACE32 software release which contains for certain ICD/debugger software derived from the standard TRACE32 development trunk. A separate branch from the TRACE32 software will be created for supporting legacy ICD/debugger hardware only. Only bugfixes and host OS adaptations are planned for the future. effected CPU architectures: XA51, h85, i196, 186, MMDSP+

XC2000/C166SV2 Debugger

Build 105499 IMP 02560 XC2000/C166SV2 Debugger GTM Debugger (TriCore) PCP Debugger (TriCore) TriCore Debugger	**improved MCDS trace decoder speed**

XC800 Debugger

Build 13751 IMP 02093 XC800 Debugger In-Circuit Emulator for 8051	**OMF2 format supported**

XScale Debugger

Build 13751 FIX 02063 XScale Debugger	**SYStem.Up for XScale 80219/80321 fails**
Build 1241 FIX 01795 XScale Debugger SYStem.Option (Special setup)	**System.Option DynVector does not work** Non working system option may cause problems for operating system awareness. Unintended interrupt may occur because modified vector table is not read during runtime.
Build 1077 FIX 01148 XScale Debugger	**Go at 0** XScale Go at 0 makes step first
Build 1077 FIX 01123 XScale Debugger SETUP.IMASKASM (Mask interrupts during assembler step) SETUP.IMASKHLL (Mask interrupts during HLL step)	**not working** XScale: IMaskAsm and IMaskHll not working

eTPU Debugger

Build 1198 CHG 01712 eTPU Debugger	**access modes** additional access modes H: EH:

x186 Monitor

Build 125398 CHG 02597 JTAG Debugger Debugger for 80C196EA MMDSP Debugger Host Driver Software Integrated Development Environment x186 Monitor x196 Monitor XA51 Monitor TRACE32 Software	**restricted TRACE32 ICD software development** This is the last TRACE32 software release which contains for certain ICD/debugger software derived from the standard TRACE32 development trunk. A separate branch from the TRACE32 software will be created for supporting legacy ICD/debugger hardware only. Only bugfixes and host OS adaptations are planned for the future. effected CPU architectures: XA51, h85, i196, 186, MMDSP+
Build 1077 FIX 01357 In-Circuit Emulator for the 80186XL x186 Monitor	**MMU** Handling of MMU translation for banked 80186 targets fixed

x196 Monitor

Build 125398 CHG 02597 JTAG Debugger Debugger for 80C196EA MMDSP Debugger Host Driver Software Integrated Development Environment x186 Monitor x196 Monitor XA51 Monitor TRACE32 Software	**restricted TRACE32 ICD software development** This is the last TRACE32 software release which contains for certain ICD/debugger software derived from the standard TRACE32 development trunk. A separate branch from the TRACE32 software will be created for supporting legacy ICD/debugger hardware only. Only bugfixes and host OS adaptations are planned for the future. effected CPU architectures: XA51, h85, i196, 186, MMDSP+

x386 and x486 Monitor

Build 19417 CHG 02153 x386 and x486 Monitor	**Executable t32m86 renamed** New name is t32mx86. Covers all x86 32bit targets.