LAUTERBACH
DEVELOPMENT TOOLS

Application Note Debug Cable C166

Release 02.2025

Application Note Debug Cable C166

Application Note Debug Cable C166

Version 13-Feb-2025

Introduction

The debugger communicates with the CPU via its debug interface. Depending on the CPU and the chip, there are different debug interfaces available. For Infineon CPUs the most commonly used interfaces are JTAG and DAP.

This Application Note describes the Lauterbach OCDS Debug Cables for the C166 and XC2000 architecture and the physical debug ports required on the target side. All CPUs support at least the JTAG Interface, and some in addition the DAP interface. All OCDS debug cables support the JTAG interface, and the bi-directional cables also the DAP interface.

JTAG (Joint Test Action Group) is a 5-pin interface, a state machine and a serial communication protocol defined by IEEE Std 1149.1. See the standard for more details.

DAP (Device Access Port) is a 2- or 3-wire interface and a telegram based communication protocol promoted by Infineon. DAP supports two modes:

- DAP2 with a clock line (`DAP0`) and a bidirectional data line (`DAP1`).
- DAP3 with a clock line (`DAP0`), an input (`DAP1`) and an output line (`DAP2`)(only Tricore).

Note that the DAP Interface used by Arm cores is completely different from the Infineon DAP.

The following architectures are supported by the OCDS Debug Cables:

Infineon	C166, C166CBC, XC16x, XC2000, XE16x
ST	SUPER10

Not all of these CPUs have an interface compatible with an OCDS Debug Cable. Also not all CPUs support all debug interfaces. Consult **"XC2000/XC16x/C166CBC Debugger"** (debugger_166cbc.pdf) or the CPU documentation for details.

Debug Cables

OCDS Debug Cables

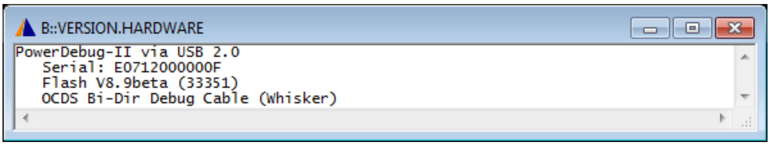
There are two basic types of OCDS Debug Cables: uni-directional and bi-directional

Cable type	Interfaces	Reset line	TriState output
unidirectional	JTAG	set	yes
bidirectional	JTAG, DAP	set, read	no

Cables with reset line read-back are able to detect a reset event on the `nRESET` line.

The TriState output indicates whether the output drivers of the debug cable are currently tristated or not, e.g. when **SYStem.Mode NoDebug** is active. This can be used, e.g. for triggering a Logic Analyzer. For more information, please refer to the description of **SYStem.LOCK** in**"XC2000/XC16x/C166CBC Debugger"** (debugger_166cbc.pdf).

The type and the version of the cable can be obtained by the **VERSION.HARDWARE** command:



```
PowerDebug-II via USB 2.0
   Serial: E0712000000F
   Flash V8.9beta (33351)
   OCDS Bi-Dir Debug Cable (Whisker)
```

The output on the screenshot reports a bidirectional cable.

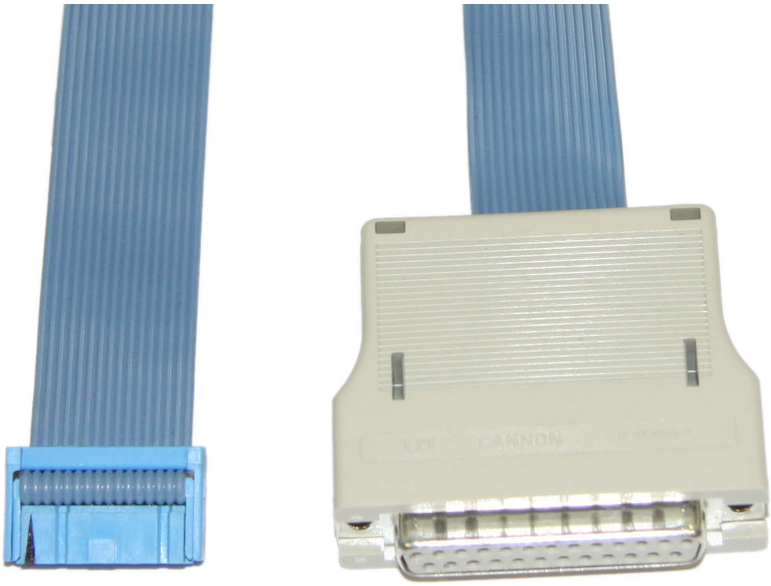
On the back side of the Debug Cable there is a sticker with the Serial Number and the programmed licenses.

Unidirectional Cables

The unidirectional OCDS debug cables can operate each line in one direction only, as input or output. This is a hardware restriction, so only JTAG mode is supported.

The following sections are named after the output in the **VERSION.HARDWARE** window.

OCDS Uni-Dir Debug Cable V0



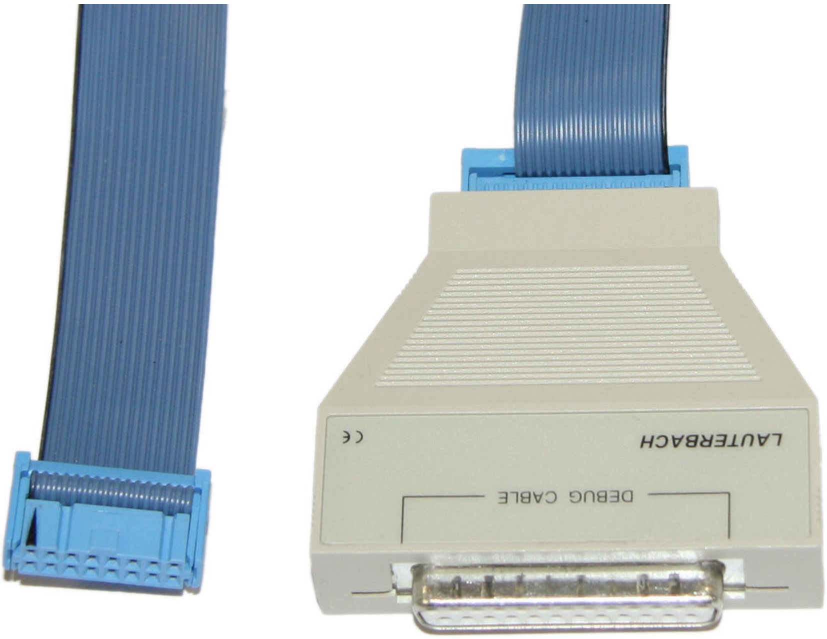
Version 0 debug cables were shipped between 1998 and 2006.

The debug cable consists of a plastic housing with a ribbon cable attached. The total ribbon cable length is 40 cm (15.75 inch), including the part covered by the housing.

Some cables have a connector on their side, indicating whether the cable is in tristate.

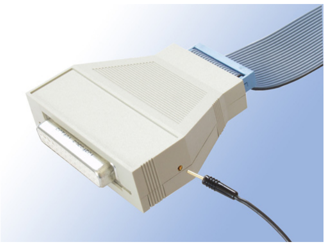
The debug cable type can be identified as follows:

CABLE.NAME()	OCDS Uni-Dir Debug Cable V0
ID.CABLE()	0x0000
VERSION.CABLE()	0x0



Version 1 debug cables completely replaced the version 0 cables in 2006. The last cables were shipped in September 2008.

The cable is RoHS compliant and has a plastic housing. The ribbon cable is removable from the housing, its total length is 40 cm (15.75 inch). For pin assignment, please see **“OCDS-L1 Connector”**, page 13.



There is a single cable contact on the casing of the debug cable which can be used to detect if the JTAG connector of the debugger is tristated e.g. when **SYStem.Mode NoDebug** is active. If so, also this signal is tristated, otherwise it is pulled low. This can be used, e.g. for triggering a Logic Analyzer. See the **SYStem.LOCK** command for additional information.

The debug cable type can be identified as follows:

CABLE.NAME()	OCDS Uni-Dir Debug Cable V1
ID.CABLE()	0x0000 or 0x0028
VERSION.CABLE()	0x1

Electrical Characteristics

All unidirectional cables have the same electrical characteristics.

- Input Pins

Input voltage range	2.5…5.2 V.
VTREF voltage range	2.25 … 5.5 V, about 2 mA required.
Maximum level for low detection	V_{IL_max} = 0.8 V
Minimum level for high detection	V_{IH_min} = 2.0 V

- Output Pins

  Output voltage depends on VTREF.

  All outputs have a 47 Ω serial termination.

- TDO has a 47 kΩ pull-up resistor.

- Target power detection

  The level of VTREF where target power down is detected depends on the used POWER DEBUG or POWER TRACE device. Please contact Lauterbach for more information. Please choose **Help** menu > **Support** > **System Info** and provide the generated support information.

Bidirectional OCDS Debug Cables

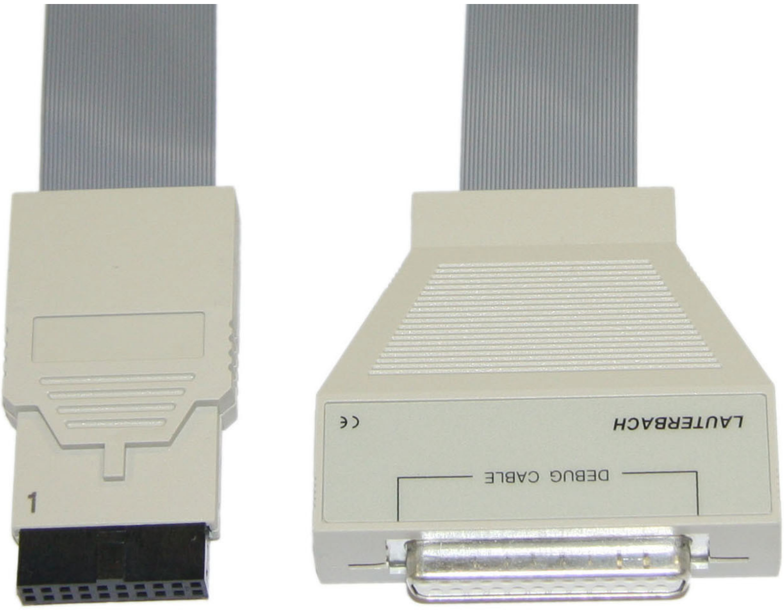
The bidirectional OCDS debug cable was developed for supporting JTAG and DAP mode. Because the adaption logic has moved from the big housing to the smaller one on the target connector for improving EMC behavior, this kind of cable is also called “Whisker”.

For using the JTAG interface, the bidirectional cable can be directly connected to the target’s debug port. For the DAP2 modes an adapter may be necessary.

The following sections are named after the output in the **VERSION.HARDWARE** window.

OCDS Bi-Dir Debug Cables V1 and V2

Prototypes, never shipped.



Beginning with September 2008, the bidirectional debug cables completely replaced the unidirectional OCDS debug cables. In JTAG mode they are backwards compatible.

The cable is RoHS compliant and consists of two plastic housings. The ribbon cable is attached to the Whisker and the housing, its total length is 45 cm (17.72 inch). For pin assignment, please see OCDS-L1 connector - **"Extended Pinout"**, page 14.

The debug cable type can be identified as follows:

CABLE.NAME()	OCDS Bi-Dir Debug Cable (Whisker)
ID.CABLE()	0x0029
VERSION.CABLE()	0x3 or 0x4

All bidirectional cables have the same electrical characteristics.

- Input pins

Input voltage range	-0.3…5.3 V.
VTREF voltage range	-0.3…5.3 V, about 2 mA required.
High/ low detection	VTREF / 2, Hysteresis = 50 mV
Low level detection range	-0.5…0.99 V
High level detection range	2.31…5.5 V

- All output pins have a 47 Ω serial termination.

  Output voltage depends on VTREF.

- TDO has a 100 kΩ pull-down resistor.

- Target power detection

  Target power down is detected for VTREF ≤ 0.27 V.

Debug Interface Description

JTAG Interface

The JTAG interface can only be used with the standard 16-pin debug connector, also called JTAG Connector. The 10-pin DAP connector does not support JTAG.

For directly connecting the Debug Cable to the target board, a **Standard 2 x 8 Pin Connector** is required on the target. In case of limited space a **Half-size 2x8 Pin Connector** and the **Half-Size Adapter** can be used instead.

Pin Assignments

OCDS Debug Port for 166CBC, Super10 and XC2000

Signal	Pin	Pin	Signal
TMS	1	2	VCCS
TDO	3	4	GND
CPUCLOCK	5	6	GND
TDI	7	8	RESET-
TRST-	9	10	BRKOUT-
TCLK	11	12	GND
BRKIN-	13	14	N/C
N/C	15	16	GND

Name	I/O	Description	Recommendations
`TMS`	O	Test Mode Select	None.
`TCLK`	O	Test clock	None.
`TDI`	O	Test Data In	No other devices in the JTAG chain are allowed between the Debug Cable and the CPU (only TriCore, 166CBC and XC2000).
`TDO`	I	Test Data Out	If there are multiple devices on the JTAG chain, connect `TDO` to the `TDI` signal of the next device in the chain. Connected to VCC via a 47 kΩ pull-up resistor (Uni-directional Debug Cable) or to GND via a 100 kΩ pull-down resistor (Bi-directional Debug Cable).
$\overline{\mathtt{TRST}}$	O	Test Reset	Connect to $\overline{\mathtt{TRST}}$. Never connect $\overline{\mathtt{TRST}}$ to $\overline{\mathtt{PORST}}$.
`VCCS`	I	VCC Sense	`VCCS` is the processor power supply voltage. It is used to detect if target power is on and it is used to supply the output buffers of the debugger. That means the output voltage of the debugger signals depends directly on `VCCS`.
`GND`	-	System Ground Plan	Connect to digital ground.
$\overline{\mathtt{RESET}}$	O	Target Reset Power-on Reset	Connect to $\overline{\mathtt{PORST}}$. Connect $\overline{\mathtt{PORST}}$ to `VCC` via a 10 kΩ pull-up resistor. Never connect to $\overline{\mathtt{PORST}}$ to $\overline{\mathtt{TRST}}$.
$\overline{\mathtt{BRKIN}}$	(I)/O	Break input	Optional. Connected via a 47 Ω serial resistor.
$\overline{\mathtt{BRKOUT}}$	I/(O)	Break output	Optional. Connected to VCC via a 47 kΩ pull-up resistor.
`CPUCLOCK`	I	CPU Clock	Optional.
`N/C`	-	Not Connected	Connect to Ground.
`KEY`	-	Keying pin 16.	Can be removed from connector. If the corresponding plug hole is removed, this also avoids incorrect polarization. If not removed from connector, connect to `GND` for shielding purposes.

General recommendations:

- Do not connect the “reserved” pin.
- Do connect all GND and N/C for shielding purpose, though they are connected on the debugger.
- $\overline{\text{RESET}}$ is controlled by an open drain driver. An external watchdog must be switched off if the In-Circuit Debugger is used.
- The Bidirectional Debug Cables have a 47 kΩ pull-up resistor connected to $\overline{\text{RESET}}$ to guarantee a proper **SYStem.Mode Up**.
- $\overline{\text{BRKIN}}$ and $\overline{\text{BRKOUT}}$ must be configured in Port configuration before they can be used.
- CPUCLOCK is optional. The CPU clock output has to be enabled by setting the appropriate register.

If available, see the Evaluation Board Schematics for an example.

The Infineon devices of the TriCore, 166CBC, XC16x, XC2000 and XE16x chip families have a so-called Cerberus IO Client. The protocol of this IO Client does not allow daisy-chaining of more than one device with a Cerberus Module. If combined with other non-Cerberus devices, the Cerberus device must be the first one in the chain.

Enabling the On-chip Debug System OCDS

For security reasons some chips have the OCDS disabled by default. When a Power-on Reset is performed, the chip samples the state of the $\overline{\text{OCDSE}}$ signal at the falling edge of the $\overline{\text{PORST}}$ or $\overline{\text{RESET}}$ signal. When it is low, OCDS is enabled, otherwise disabled.

The debugger drives the $\overline{\text{OCDSE}}$ signal low when performing a reset in case of a **SYStem.Mode Up** or **SYStem.Mode Go** thus enabling OCDS. In case of a disabled OCDS, debugging after a Hot Attach using **SYStem.Mode Attach** is not possible. To make enabling of OCDS independent of the debugger, $\overline{\text{OCDSE}}$ can be connected permanently or switchable to GND via a pull-down.

Some CPUs do not have an $\overline{\text{OCDSE}}$ signal and are able to detect the debugger presence by another mechanism. For these CPUs the Hot Attach is possible without restrictions.

DAP Interface

The DAP Interface can only be used with the 10-pin debug connector, also called DAP connector.

For connecting the Debug Cable to the **Half-size 2x5 Pin DAP Connector** the **DAP Converter** is required.

Pin Assignment

DAP Debug Port for XC2000/XE16x (10-pin)

Signal	Pin	Pin	Signal
VREF	1	2	DAP1
GND	3	4	DAP0
GND	5	6	USER0
GND	7	8	DAPEN- (TRST-)
GND	9	10	RESET- (PORST-)

Signal Description

Name	I/O	Description	Recommendations
VREF	I	Reference Voltage	VREF is the processor power supply voltage. It is used to detect if target power is on and it is used to supply the output buffers of the debugger. That means the output voltage of the debugger signals depends directly on VREF.
GND	-	Ground	Connect to Ground.
DAP0	O	DAP Clock	None.
DAP1	(I)/O	DAP Data In/ Out	None.
DAPEN	O	DAP Port Enable	Optional. If high by Debugger at Power-on Reset, the DAP Port is enabled by the CPU. In case DAPEN is handled by the target hardware, this signal can be used as the USER1 signal.
USER0	I/O	User signal 0.	Optional. Can be used as input or output. Can not be used in DAP3 Mode.

USER1	I/O	User signal 1.	Optional. Can be used as input or output. Can not be used when DAP Port Enabling is handled by the debugger.
$\overline{\text{RESET}}$	O	Target Reset Power-on Reset	Connect to $\overline{\text{RESET}}$ or $\overline{\text{PORST}}$. Never connect to $\overline{\text{TRST}}$.

General recommendations:

- Do connect all GND and N/C for shielding purpose, though they are connected on the debugger.
- $\overline{\text{RESET}}$ is controlled by an open drain driver. An external watchdog must be switched off if the In-Circuit Debugger is used.
- The Bidirectional Debug Cables have a 47 kΩ pull-up resistor connected to $\overline{\text{RESET}}$ to guarantee a proper **SYStem.Mode Up**.

If available, see the Evaluation Board Schematics for an example.

Enabling the DAP Interface on the Chip

For safety reasons the DAP Interface on the Chip is normally disabled. On a Power-on Reset, a configuration pin of the chip (normally $\overline{\text{TRST}}$) is sampled by the CPU on the falling edge of $\overline{\text{PORST}}$. Depending on the level of the configuration pin the DAP Interface is enabled.

The configuration pin can either be programmed by the target hardware, e.g. by a permanent or switchable pull-up to VCC, or it can be handled through the debugger by connecting to the DAPEN or ALTEN pin of the DAP debug connector. The debugger has to be configured which method is applied on the target hardware by using the **SYStem.CONFIG DAP.DAPENable** command (see the particular Processor Architecture Manual for details). Note that the **SYStem.CONFIG DAP** option only has an effect on the DAPEN signal.

Although DAP can be disabled, JTAG can be activated at any time. A change of the Debug Interface Mode (JTAG or DAP) is only possible when the debugger is not in System. Mode Up.

More information can also be found in chapter **Enabling the DAP Interface on the Chip**.

Configuring PowerView

The Interface Mode required by the target cannot be detected automatically. Additionally it might be necessary to configure and access the `DAPEN` and DAP User Pins. This chapter describes how to do so.

Selecting the Interface Mode

At the time this document is written, all known CPUs and all available Debug Cables support the JTAG Interface, so this Interface Mode is selected by default.

If another Interface Mode is desired or required, e.g. by target configuration, it has to be selected by the **SYStem.CONFIG DEBUGPORTTYPE** command (see the particular Processor Architecture Manual for details):

```
SYStem.Mode Down                      ; switch to a System Mode where the
                                      Interface Mode can be changed

SYStem.CPU XC2287M-104F               ; select a CPU supporting DAP

SYStem.CONFIG DEBUGPORTTYPE DAP2      ; choose DAP2 Mode for debug
                                      connection

SYStem.Mode Up                        ; establish debug connection in
                                      DAP2 Mode
```

Interface modes not supported either by the cable or the CPU are locked. Note that the Interface mode can only be changed before target communication is established.

Enabling the DAP Interface on the Chip

If the target hardware is designed in a way that the debugger enables the DAP Mode on the CPU, this has to be configured by the **SYStem.CONFIG DAP.DAPENable** command (see the particular Processor Architecture Manual for details):

```
SYStem.Mode Down                      ; switch to a System Mode where the
                                      Interface Mode can be changed

SYStem.CPU TC1797ED                   ; select a CPU supporting DAP

SYStem.CONFIG DEBUGPORTTYPE DAP2      ; choose DAP2 Mode for debug
                                      connection

SYStem.CONFIG DAP.DAPENable ON        ; enable DAP Mode on the CPU

SYStem.Mode Up                        ; establish debug connection in
                                      DAP2 Mode
```

The DAP Interface Mode has to be selected before configuring **SYStem.CONFIG DAP.DAPENable**.

The CPU only enables it's DAP Interface on a Power-on Reset event. The debugger does not perform this reset automatically, instead it is up to the user, e.g. by issuing a **SYStem.Mode Up** command.

If the target is designed to enable the DAP Mode by Hardware, the default setting should be used

```
SYStem.CONFIG DAP.DAPENable TARGET    ; DAP port enabling is done by
                                      target hardware
```

Do not confuse:

- **SYStem.CONFIG DAP.DAPENable** instructs the debugger whether the enabling of the CPU's DAP Port is done by the target hardware (`TARGET`) or by the debugger. The debugger can enable (`ON`) or disable (`OFF`) the port. The CPU samples the debugger's setting only on a Power-on Reset event.

- **SYStem.CONFIG DEBUGPORTTYPE** configures the debugger software which Interface Mode to use: JTAG or DAP2.

DAP User Pins

The DAP User Pins `USER0` and `USER1` can be used independently either as input, e.g. for target status feedback to the debugger, or as output, e.g. for additional target configuration. Although their purpose is not defined by the specification, their availability is dependent:

- `USER0` and `USER1` can not be used in JTAG Interface Mode.

- `USER0` can not be used in DAP3 Interface Mode.

- `USER1` can only be used when the enabling of the CPU's DAP Port is completely handled by the target hardware.

The DAP User Pins can be configured by using the commands **SYStem.CONFIG DAP.USER0** and **SYStem.CONFIG DAP.USER0** (see the particular Processor Architecture Manual for details). If configured as input, the PRACTICE functions **DAP.USER0()** and **DAP.USER1()** can be used to read the status.

The following example configures the USER0 pin as output:

```
SYStem.CONFIG DAP.USER0 OUT           ; USER0 pin is output

SYStem.CONFIG DAP.USER0 Set LOW       ; USER0 pin is low

PRINT DAP.USER0()                     ; print status of USER0 pin

SYStem.CONFIG DAP.USER0 Set HIGH      ; USER0 pin is high
```

The following example configures the USER1 pin as input:

```
SYStem.CONFIG DAP.USER1 IN            ; USER1 pin is input
IF (DAP.USER1()==1)
  PRINT "external event occurred"     ; perform some action...
```

A possible use case for the User Pins could be disabling an external watchdog circuit, e.g. for Flash programming or checking an important hardware status.

Adapters, Converters and Extensions

An adapter or converter is necessary to connect a debug cable and/or a trace preprocessor to a target board in case different connector types are used or the pin assignment is different.

- An adapter just adapts from one connector type to another, the signal mapping and pin numbers are identical. An example is the half-size adapter which allows to connect a debug cable with a standard 2.54 inch (100 mil) plug to a target using a standard 1.27 inch (50 mil) connector.

  An adapter is always passive so no power supply is required.

- A converter allows connecting devices with different signal mappings and connector types. An example is the converter for the 26-pin automotive debug cable to a target using a 16-pin OCDS JTAG connector. Unused signals are connected to GND, VCC or are not connected at all. There may also be a jumper or soldering option for a user-specific solution.

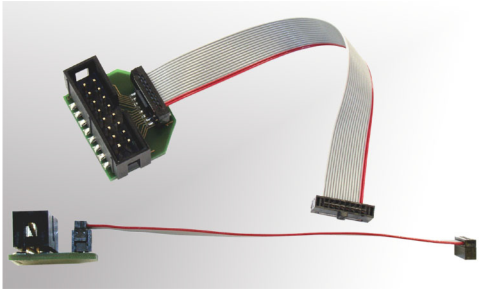
  Converters can connect two or more devices with each other, e.g. they connect a debug cable and a trace preprocessor to the target board using a single target connector.

  Converters may be passive or active. An active converter needs an additional power supply, e.g. for signal level adaption or glue logic. Most active converters get this additional power from the target (VCC, VCCS or VTREF).

  Almost all converters are related to debuggers and trace preprocessors.

Adapter 16-pin 100 mil to 50 mil

The 16-pin half-size adapter can be used to connect a 100 mil 16-pin plug to a 50 mil connector, e.g. for using a small-size 16-pin debug connector. The signals are mapped 1:1, so this is a general purpose adapter.

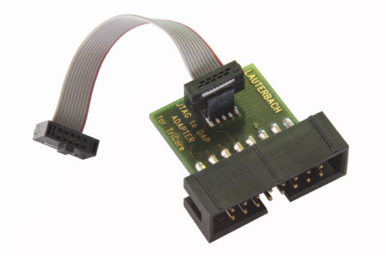


Order-Code	Description
LA-2102	Adapter 16-pin 100 mil to 50 mil.

Recommended target connector: **Half-size 2x8 connector**.

Converter 16-pin JTAG to DAP for TriCore/XC2000/XC800

The DAP converter is required for converting the DAP signals of the 16-pin OCDS bidirectional debug cable to the 10-pin Infineon DAP connector. This converter is mandatory when using DAP with the XC2000, XC800 or XE16x chip family, it is optional for the TriCore family.



Order-Code	Description
LA-3815	Converter 16-pin JTAG to DAP for TriCore/XC2000/XC800

Recommended target connector: **Half-size 2x5 connector with keying pin 7**.

Recommended Connectors

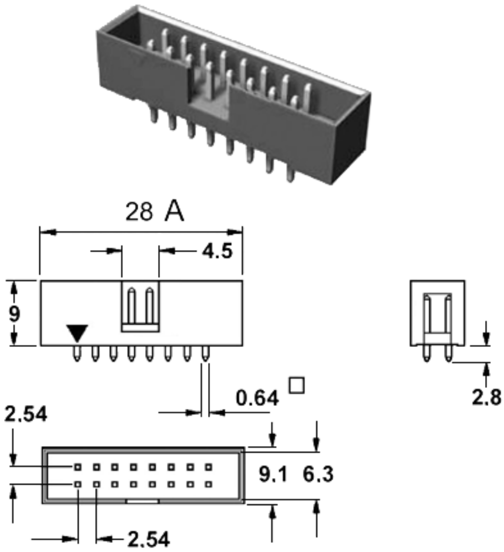
Lauterbach recommends using connectors with shroud for assuring correct polarization. Otherwise the target and/or the debugger might get damaged.

Do not extend the Debug Cable to avoid bad signal quality, e.g. by increased EMC influences.

Standard 2x8 Connector

- Standard 16 pin double row connector (two rows of 8 pins)
- Pin-to-pin spacing: 100 mil = 0.1 inch = 2.54 mm
- Connector example:

Seltronics HC 2532-016-SW	If a terminal strip without shroud is used, the spacing marked with "A" must be a minimum of 25.5mm/1" (see picture).

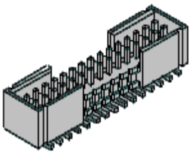


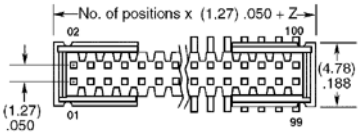
Half-size 2x8 Connector

- Half-size 16 pin double row connector (two rows of 8 pins)
- Pin-to-pin spacing: 50 mil = 0.05 inch = 1.27 mm

- Connector examples:

Samtec FTSH-108-01-L-DV	standard double-row
Samtec FTSH-108-01-L-DV-K	with keying shroud
Samtec FTSH-108-01-L-DV-EJ	wit ejector shroud

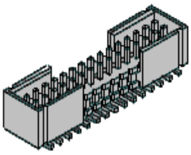


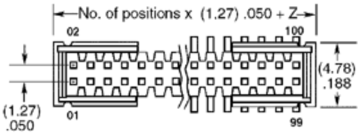


Half-size 2x5 Connector

- Half-size 10 pin double row connector (two rows of 5 pins)
- Pin-to-pin spacing: 50 mil = 0.05 inch = 1.27 mm
- Connector examples:

Samtec FTSH-105-01-L-DV	standard double-row
Samtec FTSH-105-01-L-DV-K	with keying shroud
Samtec FTSH-105-01-L-DV-EJ	with ejector shroud



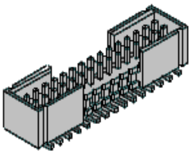


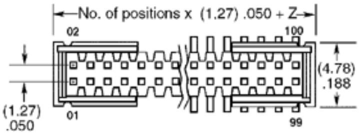
Half-size 2x5 Connector with Keying Pin 7

- Half-size 10 pin double row connector (two rows of 5 pins)
- Pin-to-pin spacing: 50 mil = 0.05 inch = 1.27 mm
- Connector examples:

Samtec FTSH-105-01-L-DV	standard double-row, remove pin 7 manually
Samtec FTSH-105-01-L-DV-K	with keying shroud, remove pin 7 manually
Samtec ASP-168330-03	This is an Application Specific Product code for FTSH-105-01-L-DV--K--P-TR-POL (keying shroud, pick & place pad, tape & reel) with pin 7 already removed. When ordering this version from Samtec, please refer to Lauterbach.
Samtec FTSH-105-01-L-DV-EJ	with ejector shroud, remove pin 7 manually

The picture below shows the version without keying pin.




No. of positions x (1.27) .050 + Z
02
100
(4.78)
.188
(1.27)
.050
01
99