Lauterbach GmbH – Elmar Stahleder

Optimization of Embedded Software's Energy Consumption

Motivation

Reducing energy consumption is becoming more and more important for the development of battery powered devices. Characteristics such as stand-by or operating time are crucial for the operation of equipment like mobile phones. Hence comprehensive measures for reducing energy consumption are an integral part of designing battery-powered devices.

Typical measures are:

- The use of power-saving components with power-saving features
- High level of integration
- Variable CPU frequencies
- Variable power supply
- The use of microcontrollers with power-saving features, cache and on-chip memory

However, an optimal reduction of energy consumption can succeed only if the software that controls the equipment consistently exploits all possible power-saving features of the hardware. Energy is the product of current, voltage and time. Each of these parameters can be influenced by the control software. Therefore software developers have to constantly attempt to find the optimal settings for these three parameters in the respective operating mode of the application. For instance you might ask yourself the following questions:

- Is the microcontroller in the right power-saving mode?
- How do program changes affect power consumption?
- Are there any unexpected power peaks?

This kind of checks require a test set-up that measures, records and analyzes the program and data flow of the control software as well as current and voltage gradients. Additionally all these recordings have to be related to each other in a simple way. At the Embedded World 2007, LAUTERBACH as the first manufacturer worldwide presents a powerful and easy-to-use solution addressing these issues.

Requirements

To be able to specify the energy consumption for each line of code, the following data has to be collected:

- The chronological program flow of the control software; typically a debugger with real-time trace capabilities is used for this task nowadays.
- The current and voltage gradients during the program runtime; a logic analyzer with an analog/digital probe is suitable as a measuring device for this purpose.

The main challenge is to correlate the measurement of the program flow with the measurement of current and voltage. If non-integrated equipment from different manufacturers is used, this correlation is enormously complicated if not impossible. Analysis and statistical evaluation is also very difficult and inconvenient. The solution to this problem and the development of suitable, easy-to-use display and analysis methods is at heart of LAUTERBACH’s innovation.

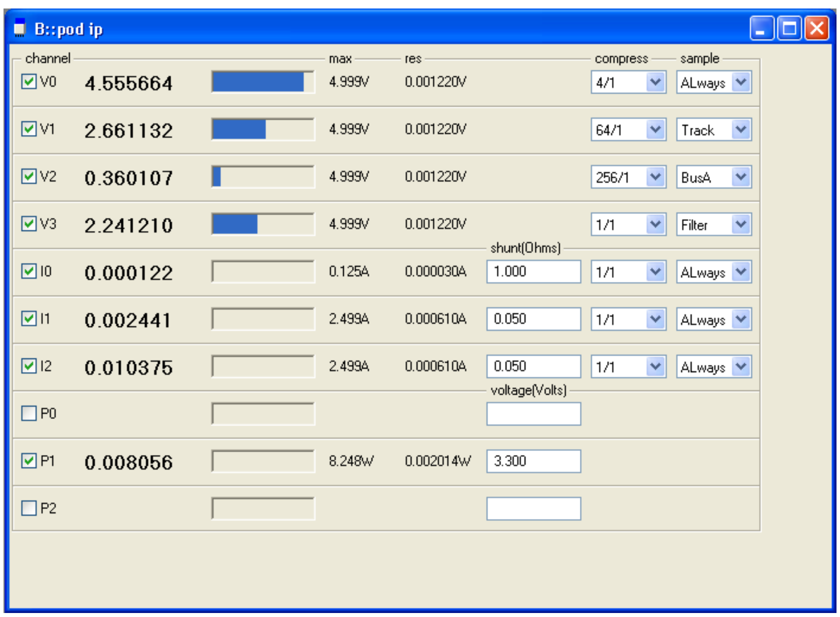
Solution

LAUTERBACH’s integrated power measurement solution, consisting of a debugger, a real-time trace and a logic analyzer provides a precise time stamp counter for each of these devices, which are synchronized at program start. Each recording is time-stamped and can therefore be correlated with the recordings of the other measuring devices, thus allowing determining the power consumption of each line of code. To acquire the data for current or voltage gradients, a TRACE32 PowerTrace II or PowerIntegrator or CombiProbe can be extended with a TRACE32 Analog Probe offering the following features:

- Real-time measurement of up to 4 voltage channels and 3 current channels
- Real-time triggers on limits and measurement ranges for voltage, current and power

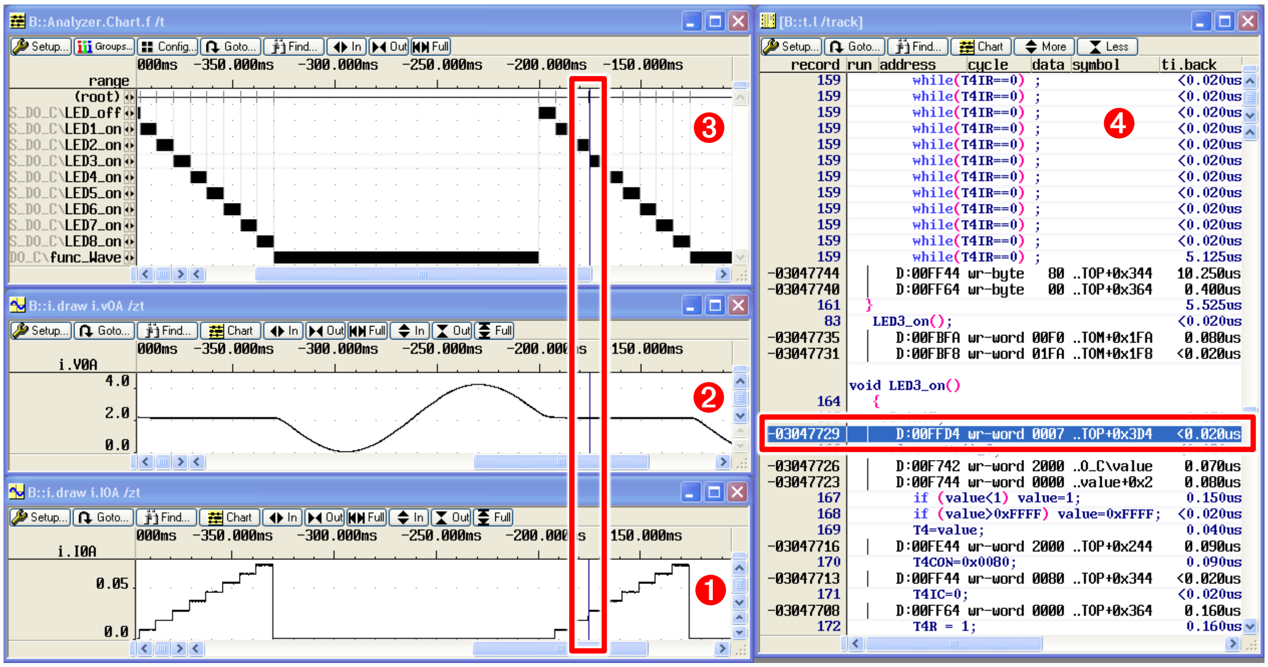
For photos of the hardware configuration options, please refer to the appendix of this whitepaper.

To measure the current, a shunt resistor is needed in the application’s supply cable. The voltage drop at this resistor depends on its value and is proportional to the current through the resistor. The voltage drop at the shunt resistor is measured by the TRACE32 Analog Probe. This method of measuring is commonly used and supported by many evaluation platforms. Current and voltage channels and the shunt resistor values are controlled through a graphical user interface as shown below. The same applies to the power channels, which are calculated from the current and voltage measurements. Alternatively, a fixed voltage can be specified for the calculation.

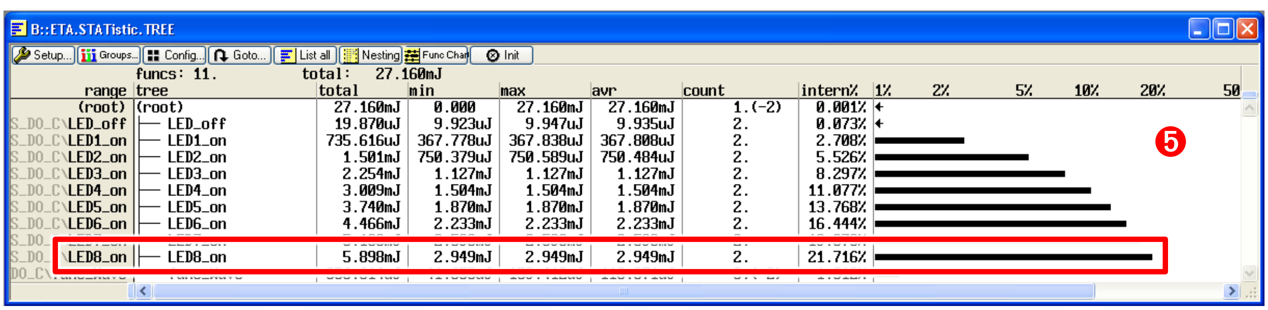


Current, voltage and power can be displayed as tables and time-dependent graphs. All time dependent displays (program flow, chart, current, voltage, power) are synchronized by the track capability. If a position is marked in one of the windows, the other windows are automatically refreshed and the corresponding position is also marked.

In the example shown below, 8 LEDs are switched on in succession and then a sine voltage is generated. The current-time graph (❶) shows the stepped shape of the current energy increase. The sine voltage can be seen in the voltage-time graph (❷). In the current (❶), voltage (❷) and program chart (❸) display, the **LED3_on** function is marked by the black vertical track line. In the **Trace.List** window (❹), the corresponding recording is marked by a blue horizontal bar.



Statistical analyses are run automatically after each program stop. They provide information about minimum, maximum and mean values of the energy consumption of the executed functions. Similarly, the absolute and percentage share of the total energy consumption is calculated for each function. This makes it easy to locate the program parts that use the most energy.



The statistical analysis shows the energy consumption of the individual functions. The bar chart (❺) shows that the **LED8_on** function (all LEDs on) has the highest energy consumption.

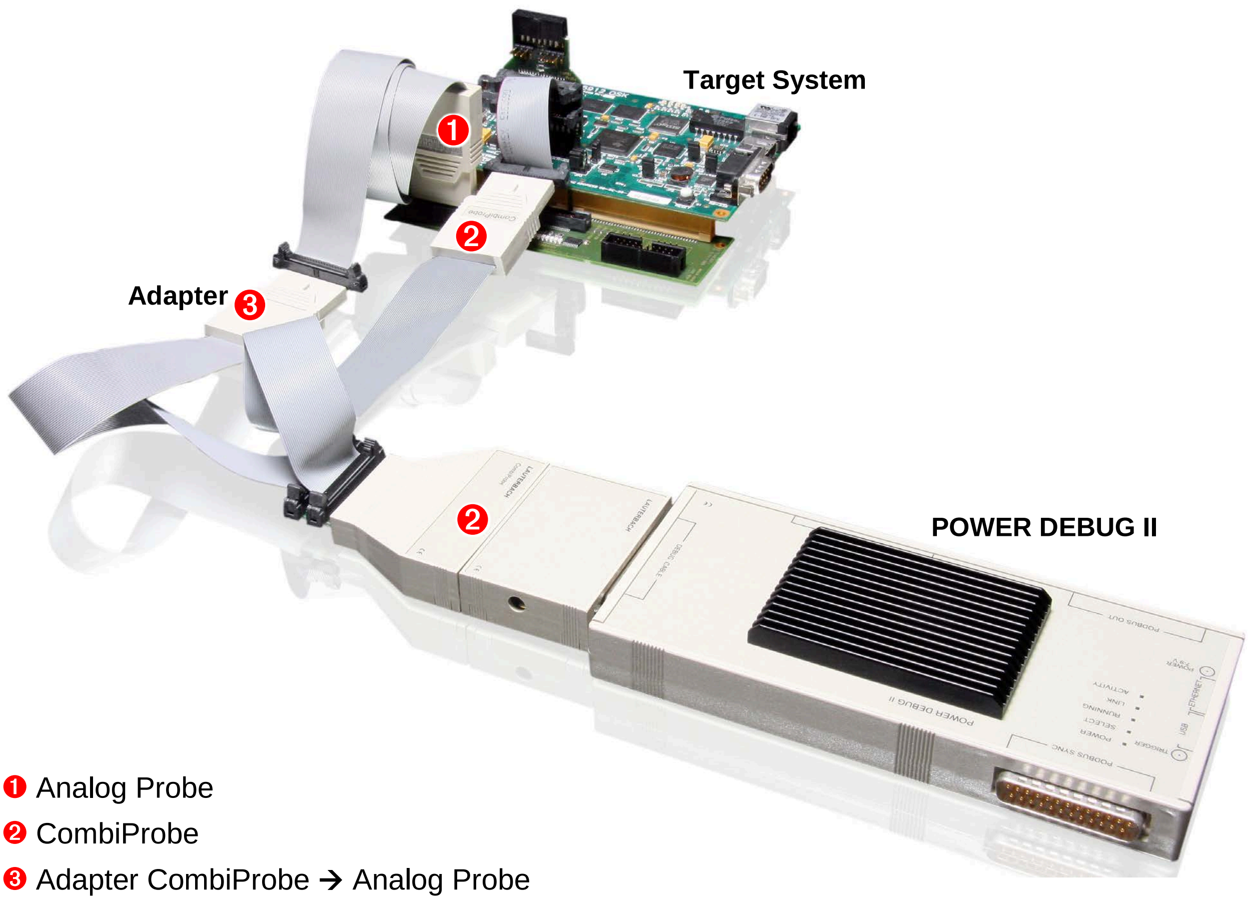
Voltage and power trigger events will be defined. Their appearance can be used by the trigger unit to enable or disable the recording (selective trace). Alternatively, a trigger signal can stop the program in real-time. This helps the user to quickly determine the causes of current peaks.

Summary

The integrated measurement solution for energy analysis gives developers a powerful, easy-to-use tool for detecting the relationship between program code and its current/power consumption. The new features are integrated in the existing user interface. Analysis hardware that is already deployed in the field can be easily extended. Close cooperation with important mobile phone manufacturers has ensured a practical implementation. Again LAUTERBACH has proved its innovative skills in the field of development tools for the embedded processor market. By integrating new, innovative solutions such as the TRACE32 Analog Probe in the existing modular concept, LAUTERBACH is constantly adding value to existing tool sets that are already deployed at customer sites.

Appendix – Hardware Configuration Options

Option 1

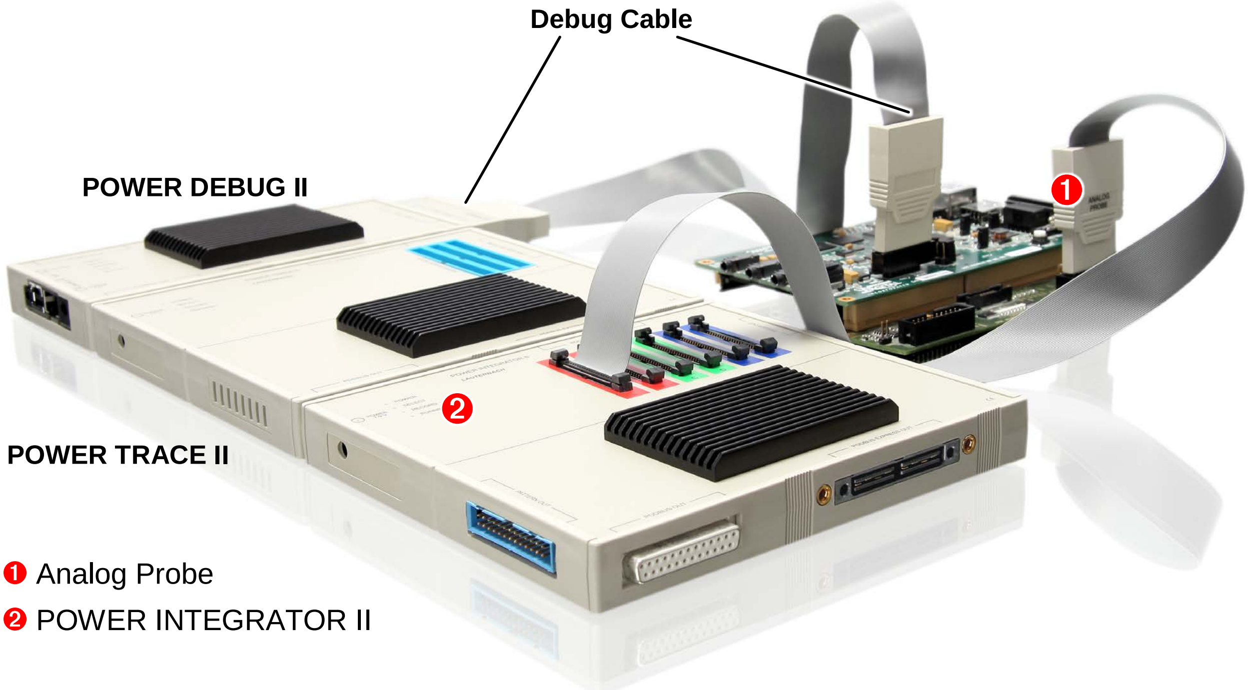


❶ Analog Probe

❷ CombiProbe

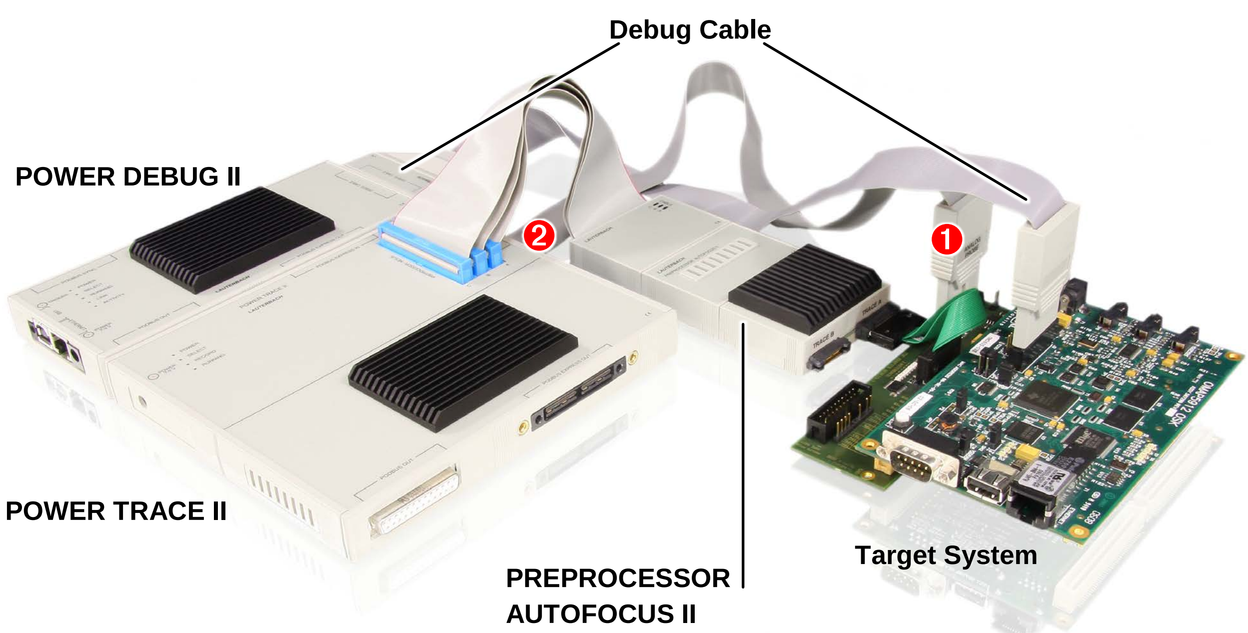
❸ Adapter CombiProbe → Analog Probe

Option 2



❶ Analog Probe
❷ POWER INTEGRATOR II

Option 3



❶ Analog Probe
❷ IProbe connector