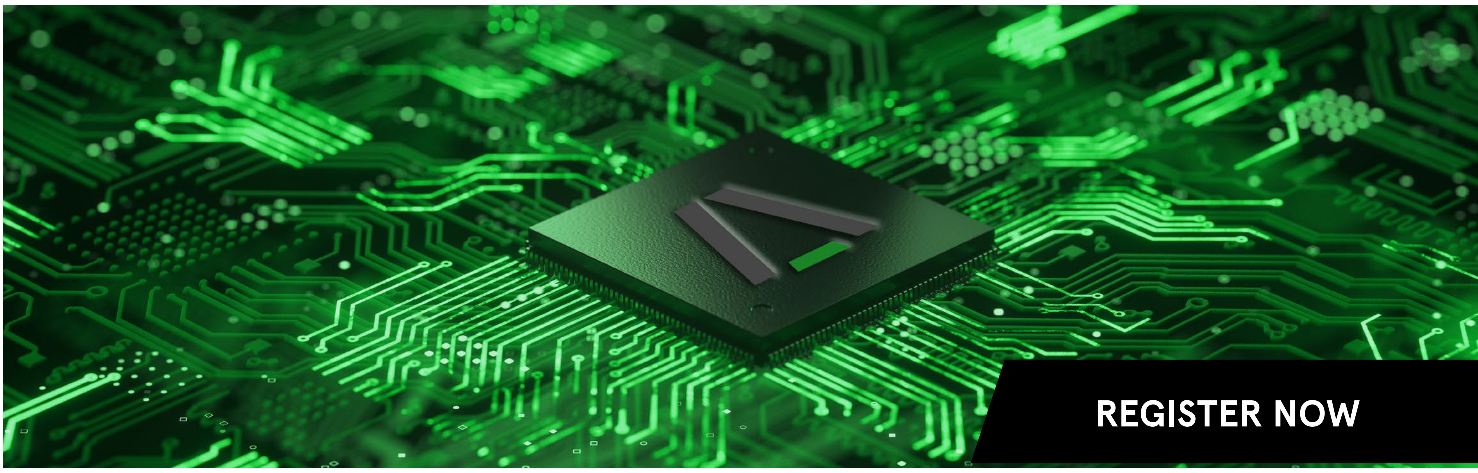
INVITATION


REGISTER NOW

TRENZ SYSTEM-ON-MODULES (SoMs)

WORKSHOP

In this workshop, we offer a comprehensive introduction to the integration of Trenz System-on-Modules (SoMs) using an AMD Xilinx System-on-Chip (SoC) for application in customer projects with custom carrier board.

We start with the criteria for selecting suitable SoMs and then focus on all the design steps involved in developing and bringing up such a custom carrier board.

This includes addressing both hardware and software aspects and providing an overview of all the necessary design steps: hardware design (Altium), software design (Vivado + Petalinux), and debugging techniques (Vivado).

The goal of the workshop is to provide the necessary overview to navigate the integration process, avoid common pitfalls, and develop successful SoM-based solutions.

DATE:

19th September 2023

LOCATION:

SmartFactoryOWL
Campusallee 3,
D-32657 Lemgo,
Germany

FACTS:

Language of talks: German

1-Day-Workshop with Lunch and Coffee-Breaks

AGENDA

- 9:30: Introduction
- 10:00 Hardware (Martin Rohrmüller, Trenz Electronic)
- 12:00 Lunch
- 13:00 Software (Markus Kirberg, Trenz Electronic)
- 15:00 Debugging (Marcus Malitschek, Avnet Silica)
- 16:00 End

AGENDA DETAILS

Hardware

1. Selecting a Trenz module
    1. Interfaces, periphery
    2. FPGA/SoC
        1. Resources
        2. Power requirements
    3. Series and variants
        1. Customization
        2. Upgrade ability
2. Trenz carrier
    1. Get ready for prototyping
    2. Limitations
3. Custom carrier design
    1. Module HW configuration
        1. IO planning / MIO setup (Software excursion)
        2. Gain compatibility
    2. Power sequencing
    3. Lenght matching
    4. Thermal management
    5. Pitfalls
4. Custom module / Chip-down (NRE)

Software

1. Initial Setup with Trenz-Carrier and Reference Design
    1. Introduction via Public Doc (hands on Trenz Wiki / Docu)
    2. Board Variants
    3. Possible Flows / Scripting
    4. Boot Options Support
2. Initial Verification of custom PS/PL
    1. IO Planning / MIO Setup / Constraints Check (will be done earlier)
3. From Reference Design to Individual Test Design with examples
    1. Adding periphery (PS/EMIO/IP)
        - I2C
        - SPI
        - Proprietary
        - GTs
    2. Testing strategy
    3. Automation for Tests

Debug

After the design phase of an FPGA board has been completed, it is typically time for bringing up and testing of the board. For this step it is good to have powerful debugging tools available.

AMD-XILINX offers the following tools, which we will briefly examine in detail:

1. ILA / System ILA

    The Integrated Logic Analyzer offers the possibility to record signals and interfaces in the FPGA in real time. The tool has many features that are known from discrete logic analyzers.

2. VIO

    With the Virtual Input/Output unit, stimuli can easily be brought into the design and individual signals of the design can be displayed. You can think of this function as virtual DIP-Switches and LEDs.

3. JTAG to AXI

    A master functionality that allows access to AXI interfaces via TCL script and JTAG. Read/write and burst accesses are thus possible without own software in the FPGA.

4. xsct Console

    This console accesses a processor system in the FPGA via JTAG and offers functions such as register dump, read/write memory areas, singlestep and much more.

5. Add Probe

    Offers the possibility to route single signals from a routed design to free pins of the FPGA. The existing design is not changed and the FPGA can be accessed with external measuring equipment.

6. Traffic Generator / Performance Monitor

    In designs with many AXI interfaces, it is often not easy to find out whether the desired performance is still given. Bottlenecks can be identified very elegantly with the Traffic Generator / Performance Monitor.