

# Compact DSP Design for Versal Using Vitis Model Composer

Online Live

Workshop

Applicable Technologies	Requirements	Contact
Xilinx® Versal® ACAP	Basic understanding of digital signal processing Basic experience with the Matlab software	Michael Schwarz P. +49 7664 91313-15 E. info@plc2.de
Fee (net per person)	Inclusive	Duration
<b>OL</b> € 1,900	Training material	3 days
<b>WO</b> € 2,300	Plus beverages during breaks Lunch	3 days

## Workshop

This course provides experience with using the Vitis™ Model Composer tool for model-based designs focussing on UltraScale+™ and Versal® technologies. With Matlab and Simulink, Vitis™ Model Composer offers the DSP developer a fast approach to development. Model Composer provides three toolboxes: HDL, HLS, and AIE library toolbox.

A particular focus of this workshop is on programming and compiling the AI engines in Versal®. The graph methodology and kernel compilation for acceleration tasks are also being developed for this purpose.

The verification of the code generation is supported on the one hand by the Matlab simulation and

on the other hand by the profiling using the Vitis™ analyzer tool.

The methods of design drafting, compilation, optimization, analysis, and verification are accompanied by practical exercises.

Due to accompanying exercises, the course offers in-depth and practice-oriented training. Attendees of the online live course will do the practical exercises in the afternoon on their own.

## Agenda

- |   |  |
|---|--|
| <ul style="list-style-type: none"> <li>01. Creating a model-based design using the library blocks</li> <li>02. Creating custom blocks in Vitis™ Model Composer</li> <li>03. Implementing DSP functions using Vitis™ Model Composer</li> <li>04. Utilizing design implementation tools</li> <li>05. Transforming algorithmic specifications</li> <li>06. Optimization methodologies using Vitis™ Model Composer</li> </ul> | <ul style="list-style-type: none"> <li>07. Creating Versal® AI Engine graphs and kernels</li> <li>08. Connecting AI Engine blocks and non-AI Engine blocks</li> <li>09. Verifying and debugging AI Engine code using the Vitis™ analyzer</li> <li>10. Simulating and debugging using AI Engine library blocks</li> </ul> |
|---|--|