Designing with Versal AI Engine 2: Graph Programming with AI Engine Kernels

ACAP 3

ACAP-AIE2 (v1.0)

Course Specification

Course Description

This course describes the system design flow and interfaces that can be used for data movements in the Versal™ AI Engine. It also demonstrates how to utilize the advanced MAC intrinsics, AI Engine library for faster development and advanced features in adaptive data flow (ADF) graph implementation, such as using streams, cascade stream, buffer location constraints, run-time parameterization and APIs to update and/read run-time parameters.

The emphasis of this course is on:

- Implementing a system-level design flow (PS + PL + AIE) and the supported simulation
- Using an interface for data movement between the PL and AI Engine
- Utilizing advanced MAC intrinsics to implement filters
- Utilizing the AI Engine library for faster development
- Applying advanced features for optimizing a system-level design

Level – ACAP 3

Course Details

- 2 days live instructor led training (in person or online)
- 10 lectures
- 4 labs

Price – $1,600 or 16 Xilinx Training Credits

Course Part Number – ACAP-AIE2

Who Should Attend? – Software and hardware developers, system architects, and anyone who needs to accelerate their software applications using Xilinx devices

Prerequisites

- Comfort with the C/C++ programming language
- Xilinx Rapid Development Embedded Design
- Vitis software for application acceleration development flow
- Designing with Versal AI Engine 1: Architecture and Design Flow

Subsequent Courses

- Designing with Versal AI Engine 3 Kernel Programming and Optimization

Software Tools

- Vitis unified software platform 2020.2

Hardware

- Architecture: Xilinx Versal ACAPs

Check with Morgan Advanced Programmable Systems, Inc. for the specifics of the in-class lab board or other customizations. After completing this comprehensive training, you will have the necessary skills to:

- Describe the system-level flow, which includes PS + PL + AIE (SW-HW-SW) designs
- Describe the supported emulation for a system-level design
- Describe the data movement between the PS, PL, and AI Engines
- Describe the implementation of the AI Engine and programmable logic
- Implement a system-level design for Versal ACAPs with the Vitis tool flow
- Utilize advanced MAC intrinsic syntax and application-specific intrinsics such as DDS and FFT
- Utilize the AI Engine DSP library for faster development
- Apply location constraints on kernels and buffers in the AI Engine array
- Apply runtime parameters to modify application behavior
- Debug a system-level design

Course Outline

Day 1

- **Application Partitioning on Versal ACAPs 1 (Review)**
  Covers what application partitioning is and how an application can be accelerated by using various compute engines in the Versal ACAP. Also describes how different models of computation (sequential, concurrent, and functional) can be mapped to the Versal ACAP. {Lecture}
- **Application Partitioning on Versal ACAPs 2**
  Explains how image and video processing can be targeted for the Versal ACAP by utilizing the different engines (Scalar Engine, Adaptable Engine, and Intelligent Engine). Also describes the AI engine development flow. {Lecture}
- **ACAP Data Communications 1**
  Describes the implementation of AI Engine and programmable logic (PL) kernels and how to implement the functions in the AI Engine that take advantage of low power. {Lecture}
- **ACAP Data Communications 2**
  Describes the programming model for the implementation of stream interfaces for the AI Engine kernels and PL kernels. Lists the stream data types that are supported by AI Engine and PL kernels. {Lecture}
- **System Design Flow**
  Demonstrates the Vitis compiler flow to integrate a compiled AI Engine design graph (libadf.a) with additional kernels implemented in the PL region of the device (including HLS and RTL kernels) and link them for use on a target platform. You can call then these compiled hardware functions from a host program running in the Arm® processor in the Versal device or on an external x86 processor. {Lecture, Lab}
- **Introduction to Advanced Intrinsic Functions**
  Describes how to implement filters using advanced intrinsics functions for various filters, such as non-symmetric FIR, symmetric FIR, or half-band decimators. {Lecture}

Day 2

- **Versal AI Engine DSP Library Overview**
  Provides an overview of the available DSP library, which enables faster development and comes with ready-to-use example designs that help with using the library and tools. {Lecture, Labs}
- **Advanced Graph Input Specifications 1**
  Learn advanced features such as using initialization functions, writing directly using streams from the AI Engine, cascade stream, core location constraints, and buffer location constraints. {Lecture}
- **Advanced Graph Input Specifications 2**
  Describes how to implement runtime parameterization, which can be used as adaptive feedback and to switch functionality dynamically. {Lecture, Lab}
- **Versal AI Engine Application Debug and Trace**
  Shows how to debug the AI Engine application running on the Linux OS and how to debug via hardware emulation that allows simulation of the application. {Lecture}

Register Today

Morgan Advanced Programmable Systems, Inc. (Morgan A.P.S.) delivers public and private courses in locations throughout the central US region; including Iowa, Illinois, Kansas, Minnesota, Missouri, Nebraska, North Dakota, South Dakota, and Wisconsin.

Visit **morgan-aps.com/training**, for full course schedule and training information.

Morgan
Advanced Programmable Systems

- You must have your tuition payment information available when you enroll. We accept credit cards (Visa, MasterCard, or American Express) as well as purchase orders and Xilinx training credits.

Student Cancellation Policy

- Student cancellations received more than 7 days before the first day of class are entitled to a 100% refund. Refunds will be processed within 14 days.
- Student cancellations received less than 7 days before the first day of class are entitled to a 100% credit toward a future class.
- Student cancellations must be sent here.

Morgan A.P.S. Course Cancellation Policy

- We regret from time-to-time classes will need to be rescheduled or cancelled.
- In the event of cancellation, live on-line training may be offered as a substitute.
- Morgan A.P.S. may cancel a class up to 7 days before the scheduled start date of the class; all students will be entitled to a 100% refund.
- Under no circumstances is Morgan A.P.S. responsible or liable for travel, lodging or other incidental costs. Please be aware of this cancellation policy when making your arrangements.
- For additional information or to schedule a private class contact us here.

Online training with real hardware

- During the Covid-19 period, some companies do not allow their staff to participate in live in-person training.
  - Consequently, Morgan Advanced Programmable Systems, Inc. has set up a training VPN where engineer participants can take classes online using the same computers and devCards used during in-person training.
  - Even better, and upon request, you can use these computers after hours on training days to experiment with labs. This is not possible for in-person training.
  - Additionally, just like in-person training, the laptops and devCards, tools, OS, and licensing are setup in advance.
  - In some ways, live online-training is better than in-person…for example, you can grant the instructor permission to look at your Vivado, PetaLinux terminal, or Vitis for extended periods of time if your lab is not going exactly has planned to a missed step.
  - This is often more comfortable than two engineers crowding around a laptop screen.
  - Taking remote training also allows you to learn some tips and tricks for working remote. Whether your devCard is in the lab down the hall, or across the world via VPN, you can control your Xilinx based device quickly and efficiently.