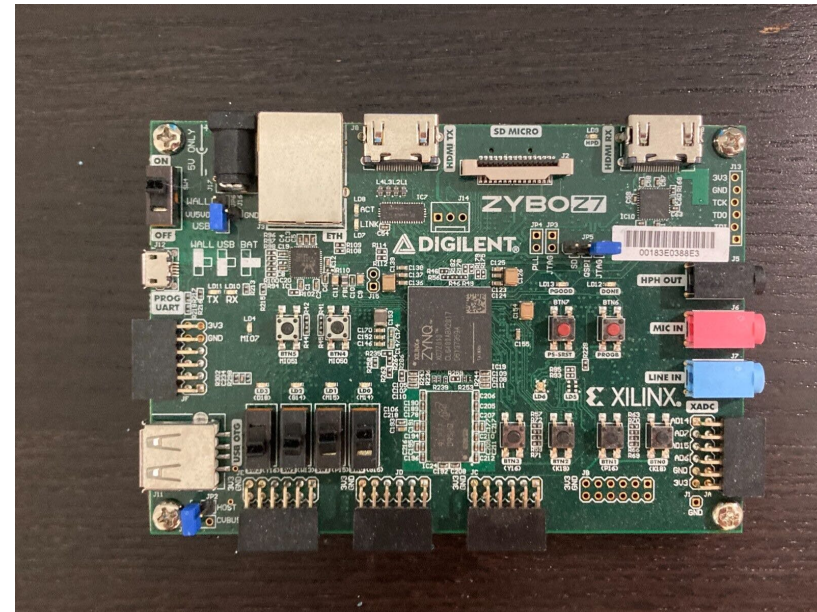
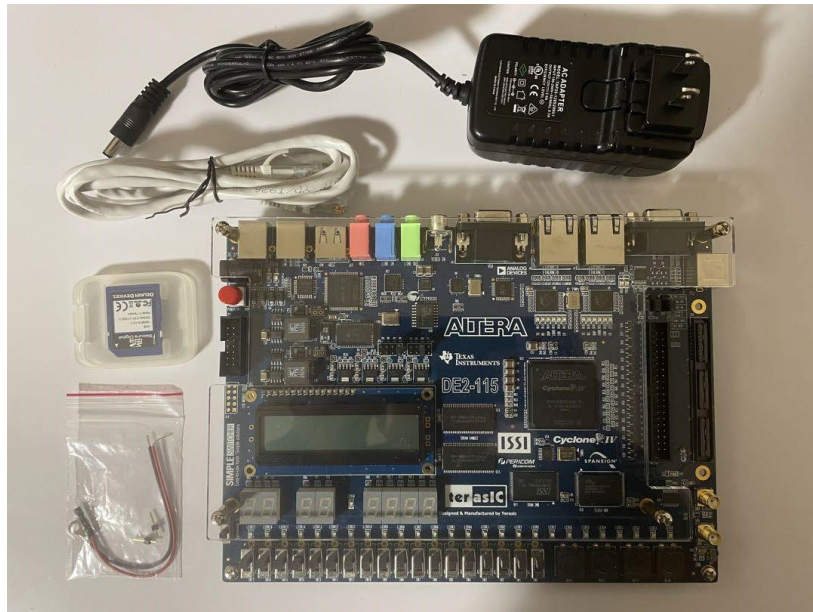


Introduction to Zynq SoC

Erwin Setiawan

FPGA vs. SoC FPGA

- ❖ SoC FPGA biasanya memiliki hard processor.
- ❖ Hard processor yaitu processor yang sudah terfabrikasi sebagai silicon di dalam chip FPGA.

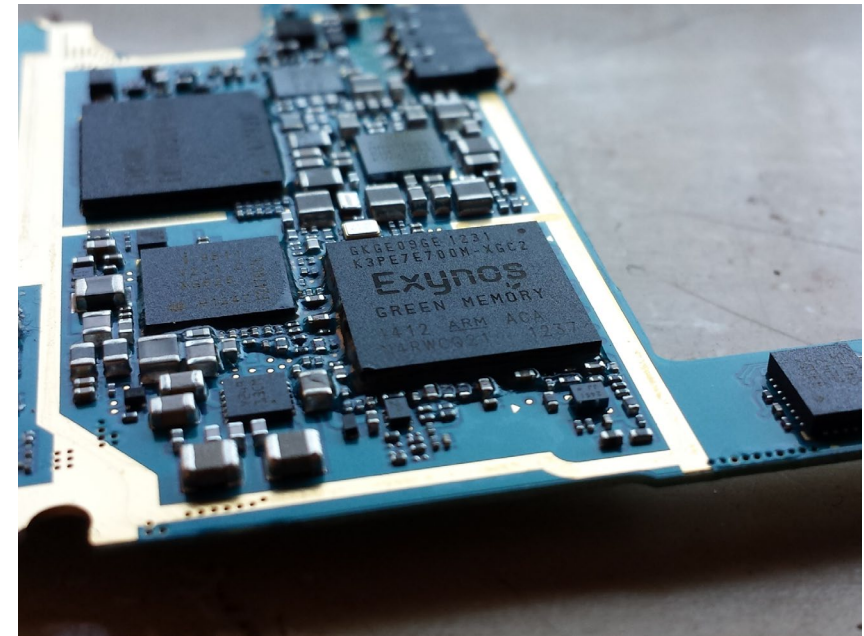
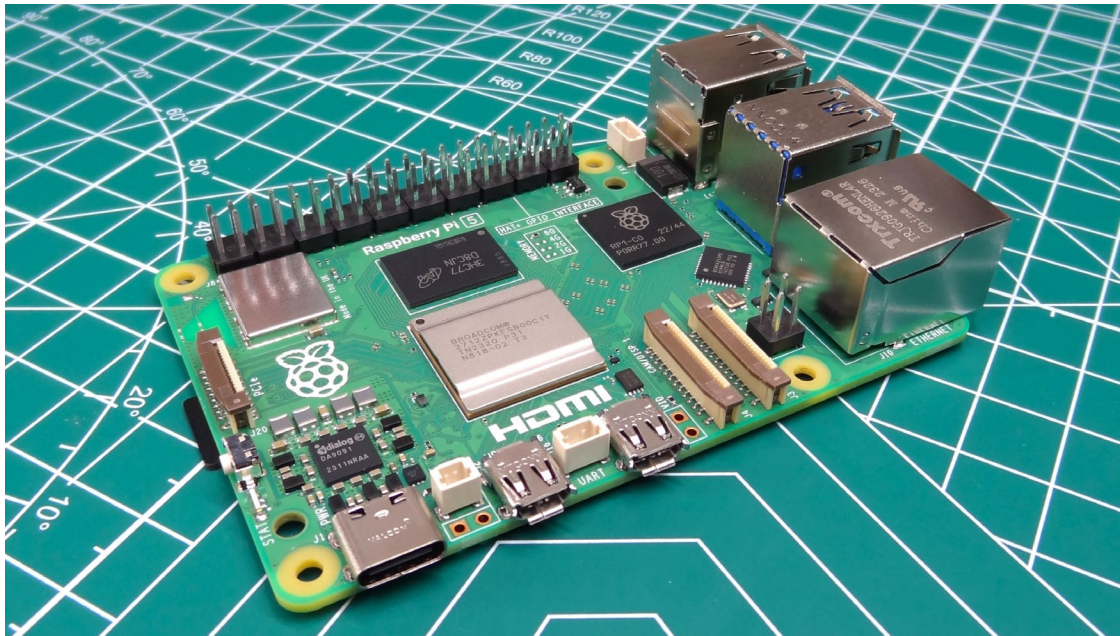


FPGA vs. SoC FPGA

- Contoh FPGA: Altera DE2
 - Hanya memiliki resource FPGA
 - Tidak ada hard processor
 - Jika membutuhkan processor bias menggunakan soft processor NIOS II.
- Contoh SoC FPGA: Digilent Xilinx Zybo
 - Memiliki resource FPGA
 - Terdapat hard processor ARM cortex A9

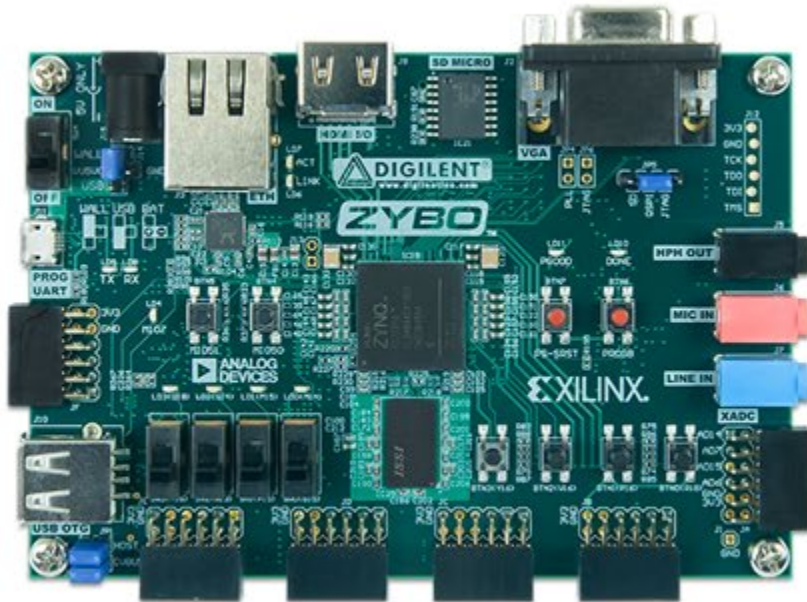
SoC FPGA vs. SoC

- ❖ SoC pada umumnya dipakai di single board computer seperti raspberry, sampai ke gadget seperti smartphone/tablet.
- ❖ Jenis SoC ini biasanya tidak memiliki programmable logic (FPGA).



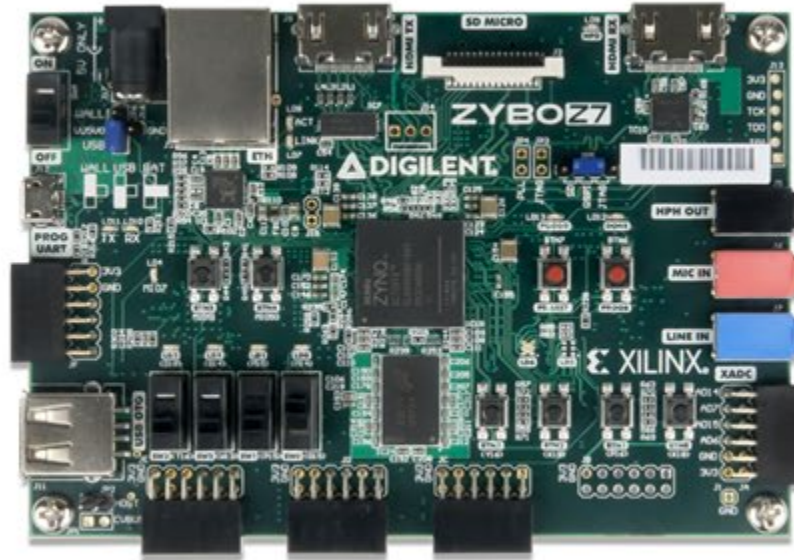
Xilinx Zynq Board (ZyBo)

ZYBO

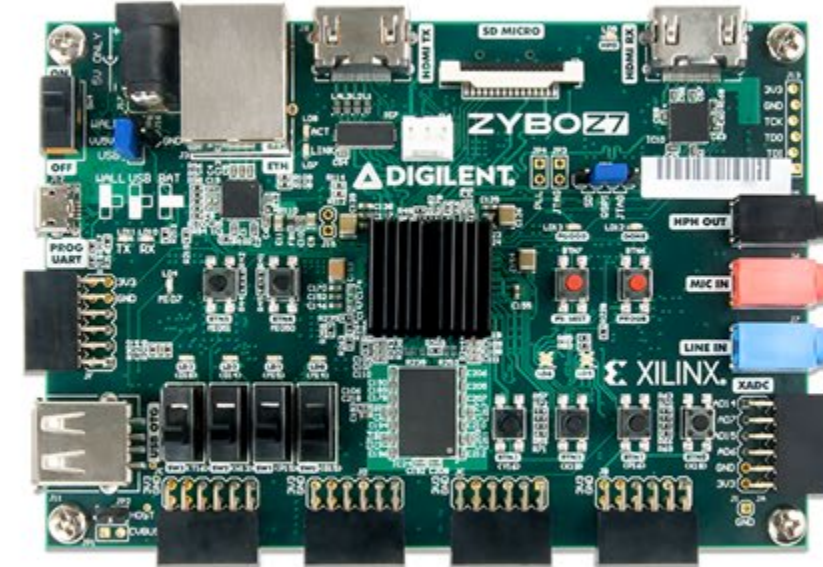


Part:
XC7Z010-1CLG400C

Zybo Z7-10



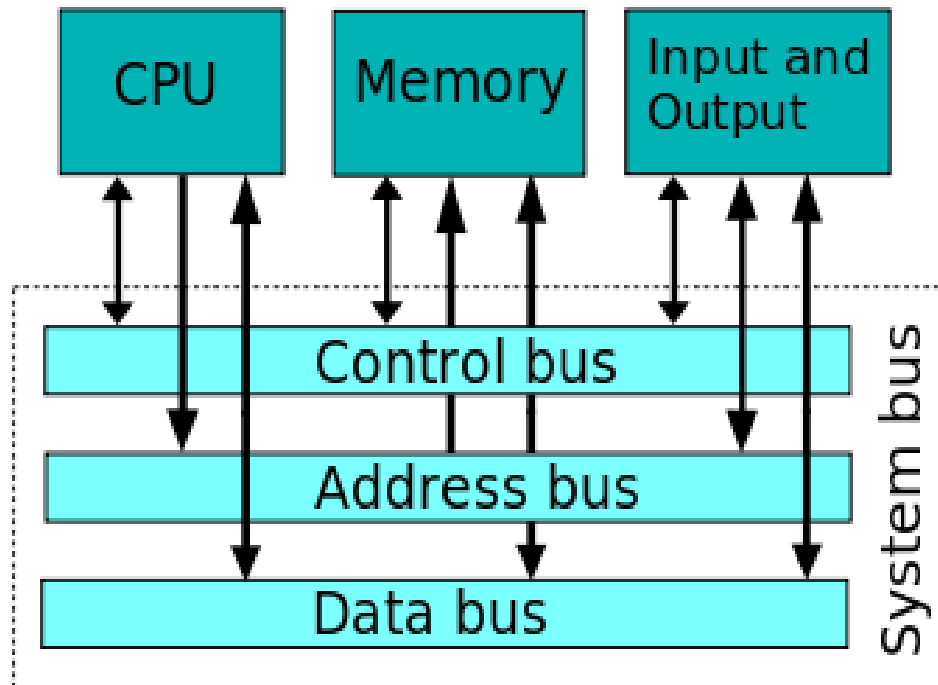
Zybo Z7-20



Part:
XC7Z020-1CLG400C

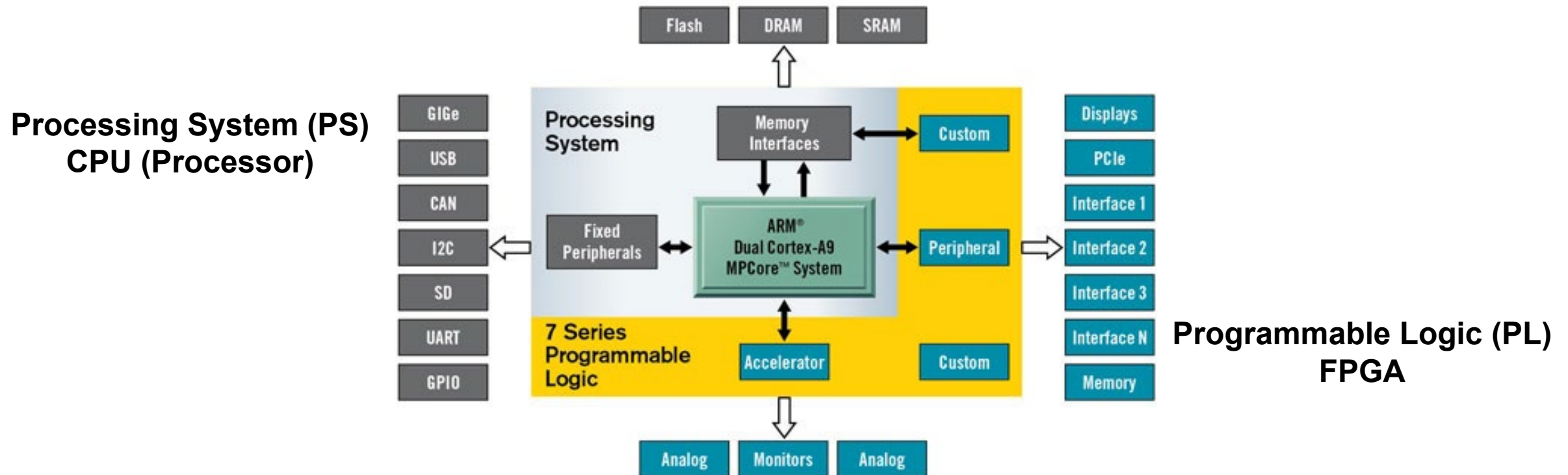
Arsitektur Komputer

- ❖ Arsitektur computer pada umumnya terdiri dari: CPU (processor), memory, I/O, dan System Bus.



Arsitektur Zynq

- Arsitektur zynq sama seperti arsitektur komputer pada umumnya, tetapi ada tambahan FPGA.



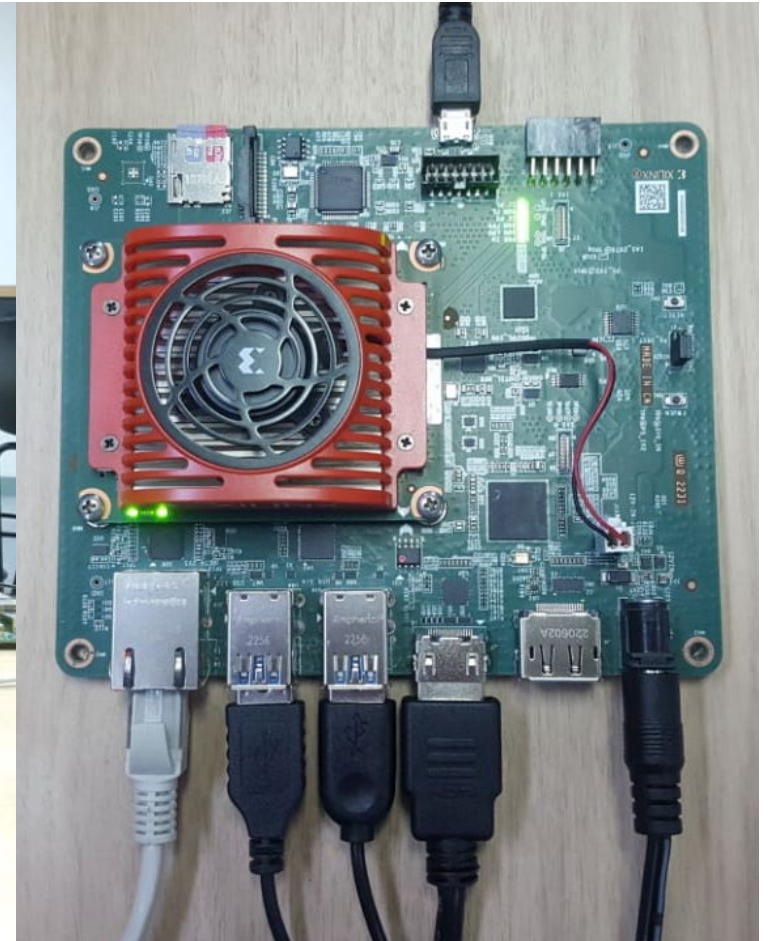
System Bus

- ❖ AHB (Advanced High-performance Bus)
 - ❖ Ethernet, SDIO, USB
- ❖ APB (Advanced Peripheral Bus)
 - ❖ UART, SPI, I2C, GPIO
- ❖ AXI (Advanced Extensible Interface) (PS to PL interface)
 - ❖ AXI memory mapped
 - ❖ AXI-Full
 - ❖ AXI-Lite
 - ❖ AXI stream

PYNQ Framework

- ❖ PYNQ (Python productivity for Zynq) merupakan high level framework dan ekosistem hardware (berbasis Xilinx Zynq SoC FPGA).
- ❖ Analoginya seperti Arduino. Arduino terdiri dari framework, library dan ekosistem hardware-nya.
- ❖ PYNQ framework berjalan pada Operating System Linux untuk board tersebut.
- ❖ PYNQ melakukan abstraksi terhadap kompleksitas linux yang memungkinkan user membuat program aplikasi embedded Linux yang terintegrasi dengan FPGA secara lebih mudah.

Linux OS pada Board Kria KV260

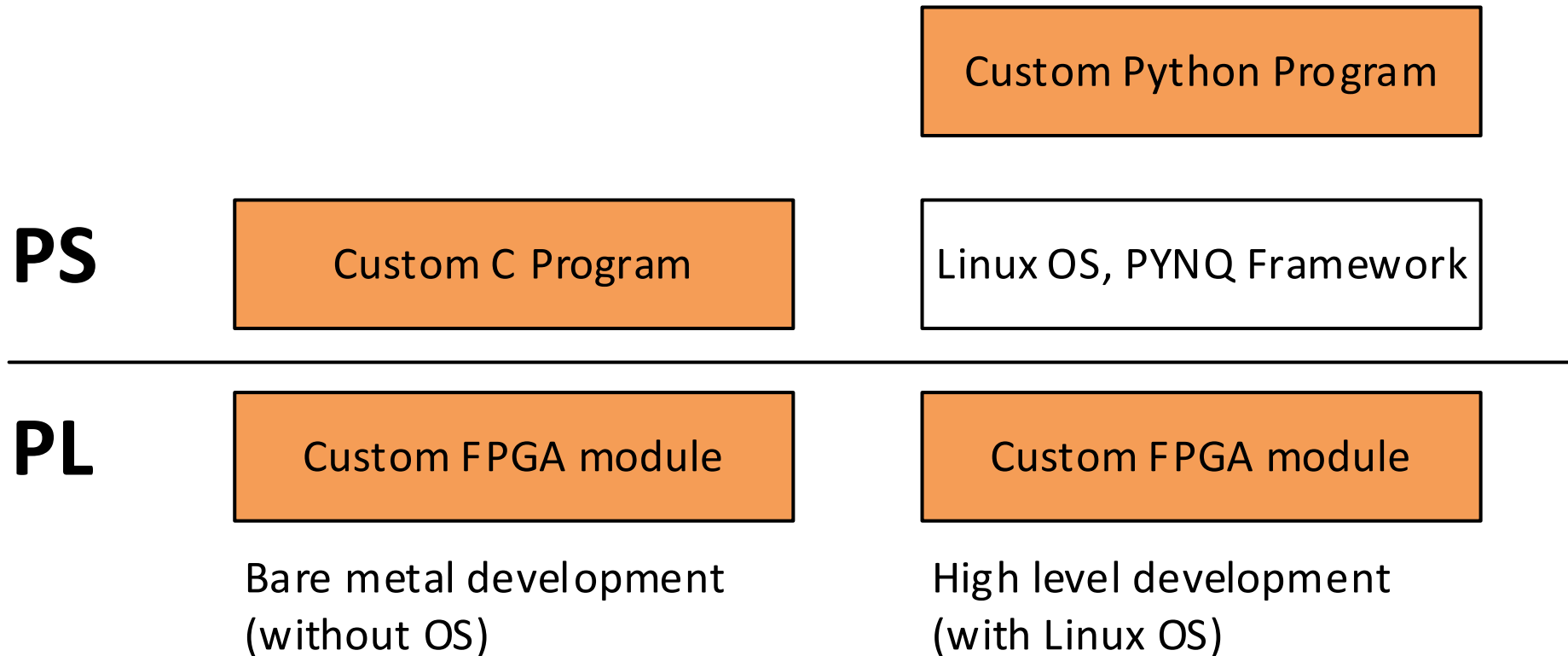


Development Flow

- ❖ Development pada Zynq terdiri dari dua tahap, yaitu development hardware/FPGA/programmable logic (PL) dan software/firmware/Processing system (PS).
- ❖ Development hardware:
 - ❖ Membuat modul RTL dengan verilog atau VHDL yang nantinya akan diprogram ke FPGA/PL.
- ❖ Development software:
 - ❖ Membuat program (C, Python) yang nantinya akan diprogram ke PS.

Development Flow

❖ Ada 2 tipe development pada Zynq.



Software Tools for ZyBo (Bare Metal)

- Vivado **versi 2019.1**:
- Vivado versi 2019.2 - terbaru:
 - **Tidak support** karena tidak pakai SDK, tapi menggunakan Vitis.

Instalasi Vivado 1

- Offline installer:
 - Ukuran 21GB (versi 2019.1)
 - [https://www.xilinx.com/member/forms/download/xef-vivado.html?filename=Xilinx Vivado SDK 2019.1 0524 1430.tar.gz](https://www.xilinx.com/member/forms/download/xef-vivado.html?filename=Xilinx_Vivado_SDK_2019.1_0524_1430.tar.gz)

Instalasi Vivado 2

- Untuk kuliah ini cukup pakai **Vivado versi WebPack (free)**. License bisa registrasi ke website Xilinx (free).
- Jangan menginstal Vivado di **folder atau sub-folder-nya yang memiliki space**. Karena biasanya akan error ketika proses synthesis.
- Contoh:
 - C:\Program Files **(Error)**
 - C:\ProgramFiles **(OK)**
 - C:\ProgramFiles\Kuliah VLSI\Xilinx **(Error)**
 - C:\Xilinx **(OK)**
- Pastikan **checklist opsi SDK** saat menginstall Vivado.

Membuat Project Vivado

- Jangan membuat project Vivado di folder seperti: folder **Users di C:, folder Program Files, atau folder system yang sifatnya protected atau read only** karena biasanya akan error ketika proses synthesis.
- Jangan membuat project Vivado di folder atau sub-folder-nya yang memiliki space.
- Contoh:
 - D:\Kuliah VLSI\Vivado **(Error)**
 - D:\Kuliah_VLSI\Vivado **(OK)**
 - D:\Kuliah_VLSI **(OK)**