CPEG 422/622
EMBEDDED SYSTEMS DESIGN

MW 3:35 – 4:50, SHARP 116

Chengmo Yang

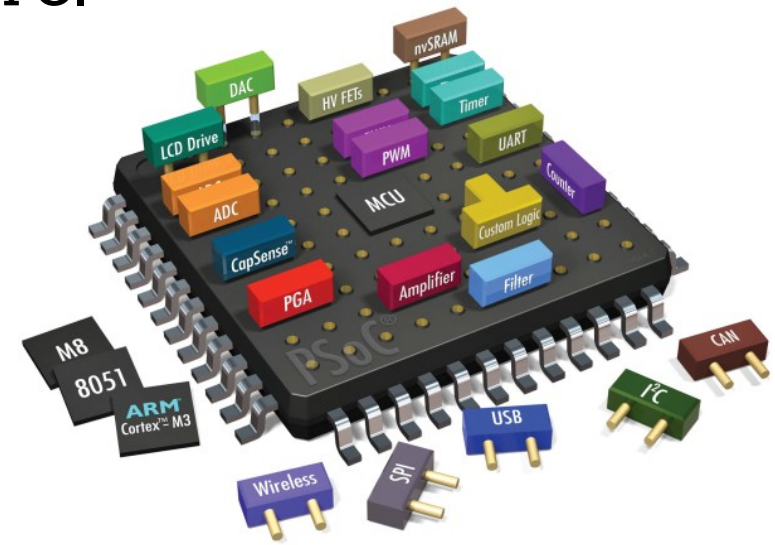
chengmo@udel.edu

Evans 201C

LECTURE 1
COURSE INTRODUCTION

WHAT IS AN EMBEDDED SYSTEM?

- A type of computer system.
- Some of the Most Common Traditional Definitions :
  - An embedded system is designed to perform a dedicated function.
  - Embedded systems are more limited in hardware and/or software functionality then the PC.
  - An embedded system usually has real-time computing constraints.
  - …

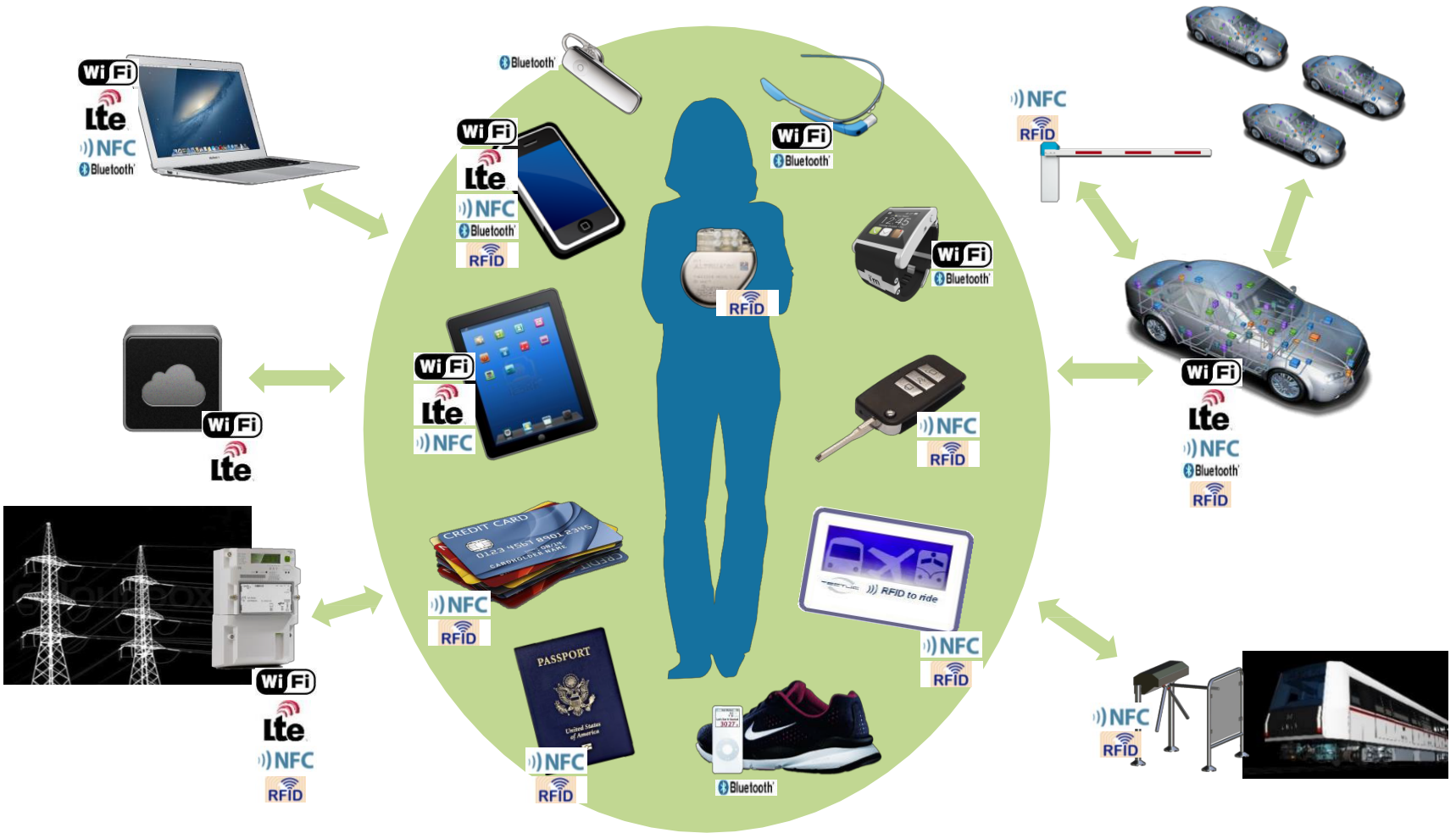


EMBEDDED SYSTEM EXAMPLES

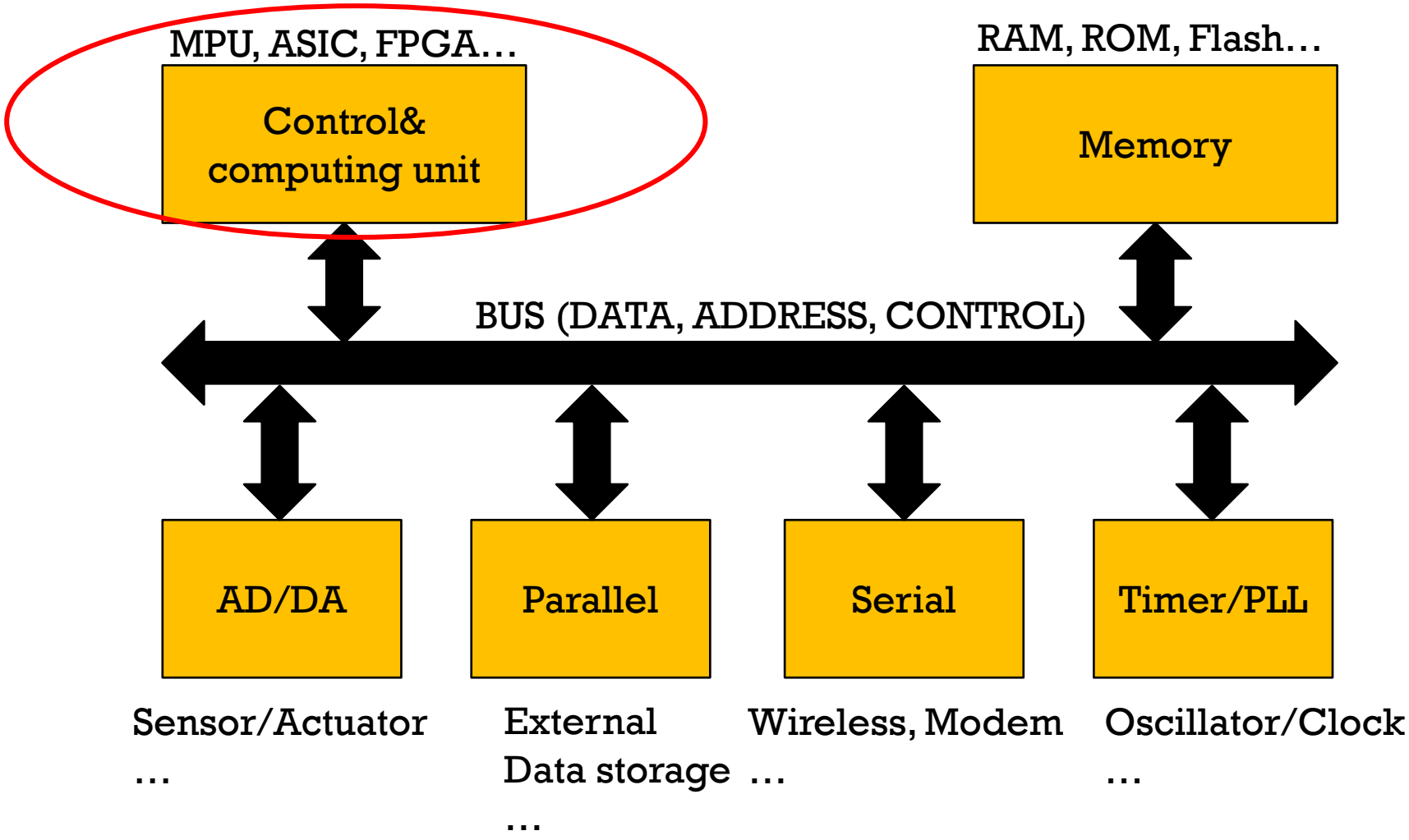
- Automotive
  - i.e. : Ignition Systems, Engine Control, Antilock Braking System, …
- Consumer Electronics
  - i.e. : TVs, STBs, appliances, toys, automobiles, cell phones …
- Industrial Control
  - i.e. : robotics, control systems…

- Medical
  - i.e. : Infusion Pumps, Dialysis Machines, Prosthetic Devices, Cardiac Monitors …
- Networking
  - i.e. : routers, hubs, gateways, …
- Office Automation
  - i.e. : fax machines, photocopiers, printers, monitors …

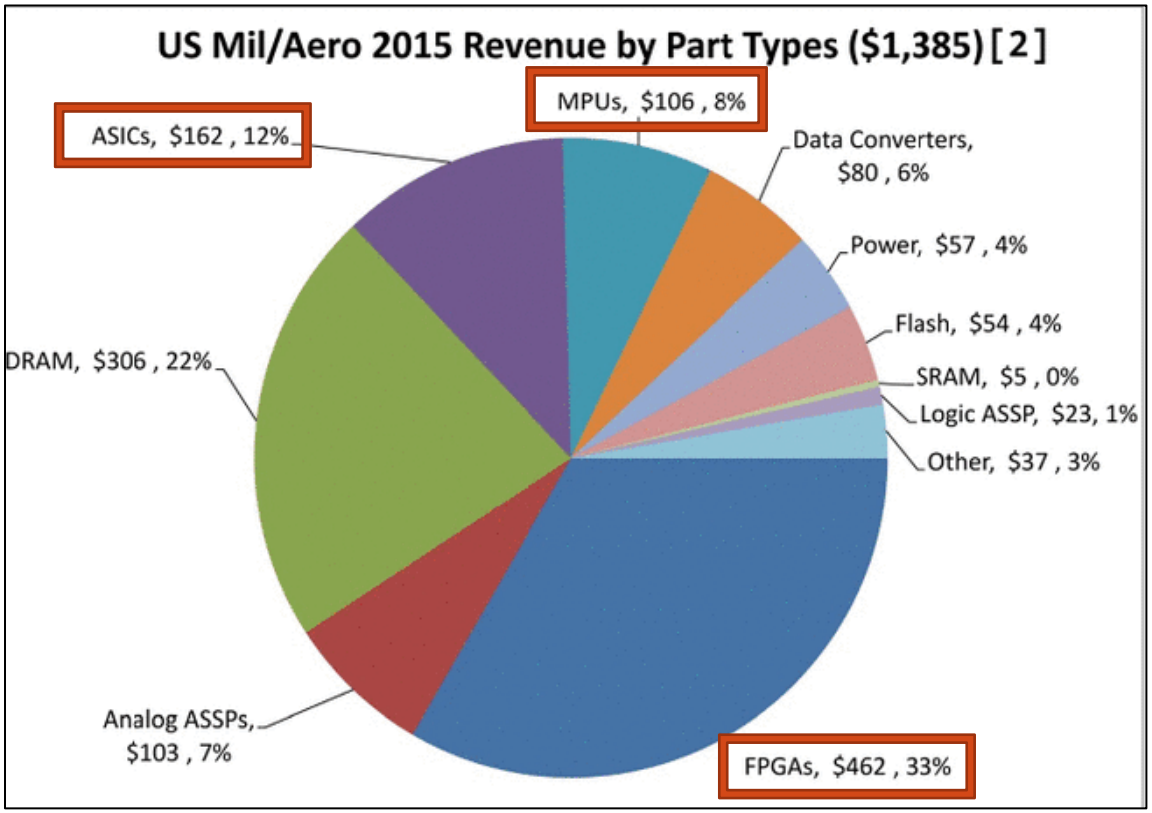
INTERNET OF THINGS (IOT)



EMBEDDED SYSTEM ARCHITECTURE



MPU VS ASIC VS FPGA



Microelectronics in Department of Defense 2015

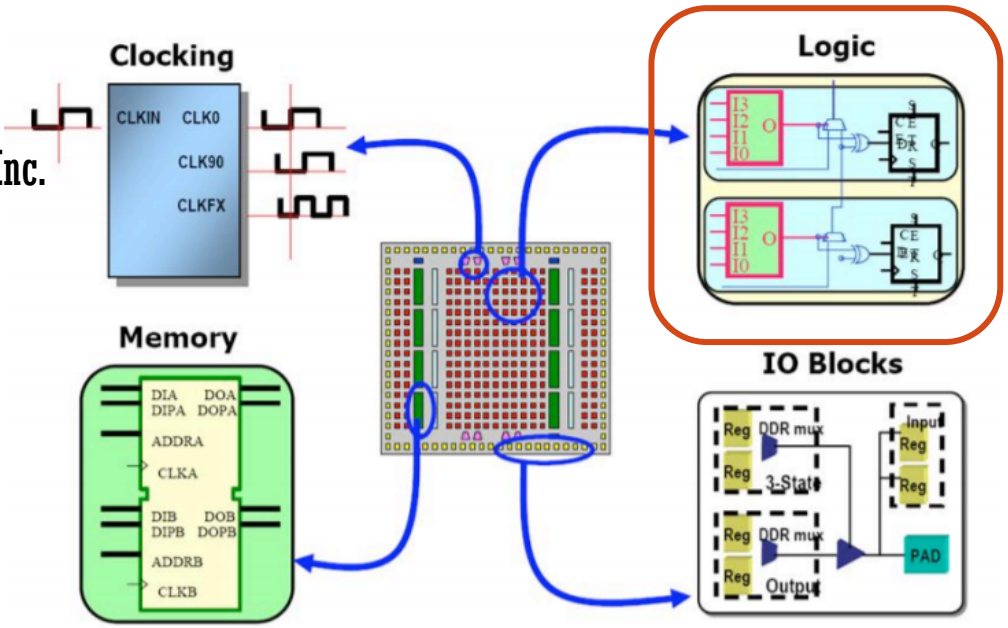
MPU VS ASIC VS FPGA

	MPU	ASIC	FPGA
Reconfigurable	No	No	**Yes**
Non-recurring engineering cost	Medium	Expensive	**Cheap**
Efficiency	Low	**High**	Medium
Performance	**High**	**High**	Low
Programming language	High level (C/C++/C#…)	-	Hardware language (VHDL, Verilog…)
Application range	**Wide**	Narrow	Medium
User's flexibility	Medium	Low	**High**
Time to market	Slow	Slow	**Fast**

WHAT IS AN FPGA?

- A **field-programmable gate array** (**FPGA**) is an IC designed to be configured by a customer after manufacturing.

Introduced in 1985 by Xilinx Inc.



hardware logic reprogrammable!

Internal structure of FPGA

WHAT DO FPGAS ADD TO EMBEDDED SYSTEM DESIGN?

- Performance
  - Hardware Speed vs software speed
- Flexibility
  - Reconfigurable
  - Easy to update
- Cost
  - Pre-qualification for VLSI chip design
- Time–to-Market
  - Short-term development speed

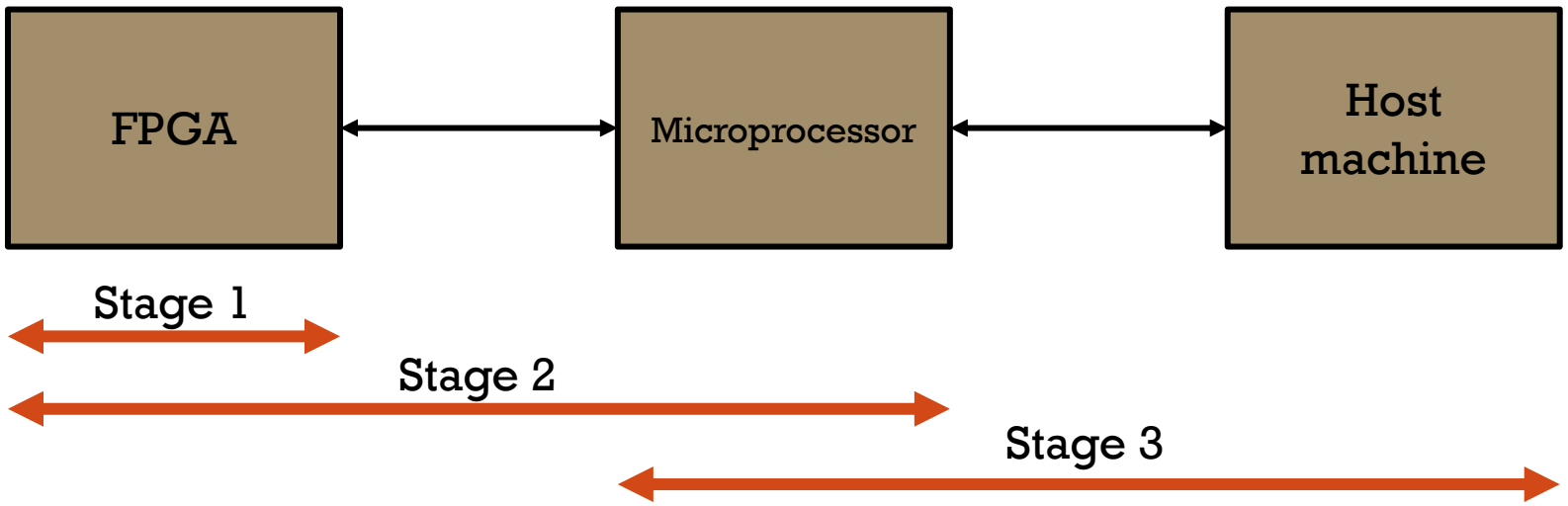
HOW TO PROGRAM AN FPGA?

- Use a **Hardware Description Language (HDL)**
- A specialized computer language used to describe the structure and behavior of electronic circuits, and most commonly, digital logic circuits.
- Example languages:
  - **VHDL,** Verilog, System C
- Difference with C/C++
  - No dataflow or data structure, but hardware structure in textual
  - Concurrent programming
  - Hierarchy and timing are always important in design

WHAT WILL WE LEARN IN CPEG422/622?

- How to design and implement an embedded system
- Vivado
- Zybo board
- VHDL , C, and Python
- Create custom IP cores
- State Machine
- Pipelining
- Design principles
- Security and Reliability

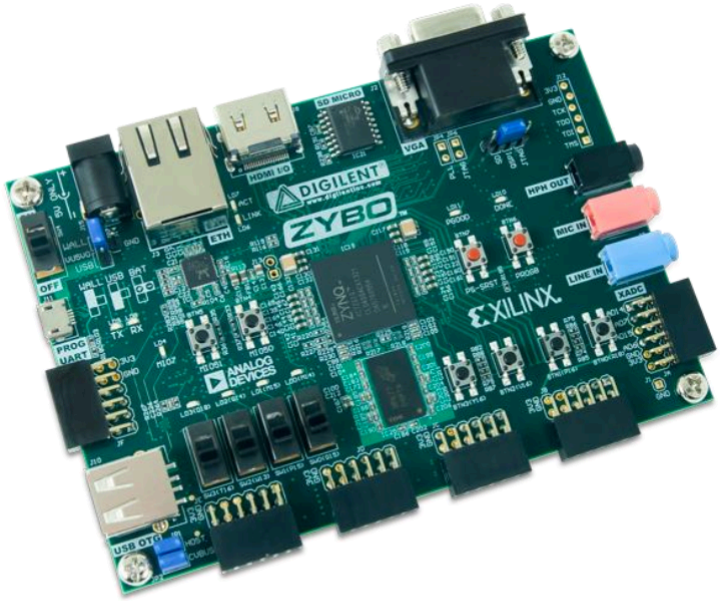
LEARNING STAGES



- Stage 1: learn how to program FPGA with VHDL.
- Stage 2: learn how to co-program FPGA and the microprocessor and let them synchronize
- Stage 3: learn how to program the interface between microprocessor and the host machine

WHAT TOOLS WILL WE USE?

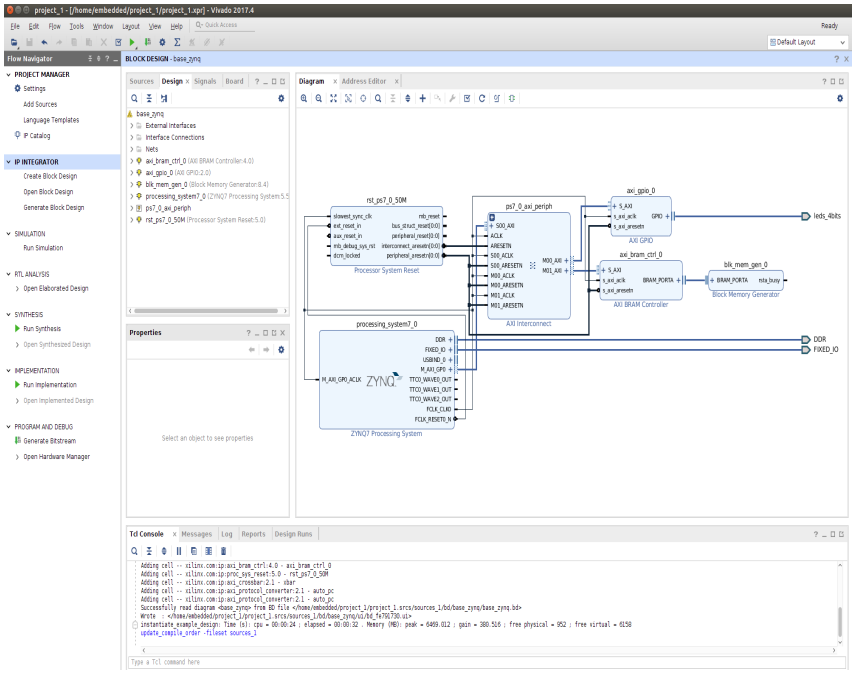
- Zybo development board



The ZYBO Zynq-7000 development board.

https://reference.digilentinc.com/_media/zybo:zybo_rm.pdf

- Vivado IDE



COURSE WORKLOAD

▪ Homeworks	10%	
▪ Quizzes	10%	lowest score dropped
▪ Midterm exam	15%	
▪ Final exam	25%	
▪ Projects	40%	

COURSE RELATED RESOURCES

- Lectures: slides and reading assignments posted on Canvas
- Textbook: "The Zynq Book", can be downloaded from http://www.zynqbook.com/download-book.php
- Review Sessions: before the exams
- Pop quizzes: in class
- Homeworks: posted on Canvas and turned in through Canvas
- Projects: posted on Canvas and turned in through Canvas

HOMEWORK

- Get your Zybo board from Mr. Thomas Lum in Evans 127

NEXT LECTURE

- Introduction to FPGA and VHDL