Bridge of Life Education

SoC Design

Course Plan

Jiin Lai

Agenda

- Background　(course-plan)
- Objectives
- Lab Platform
- Course Schedule
- Course Policy & Grading
- Lab Submission Guideline　- TA

>----

- Lab Plan (lab-plan)
- Lecture Plan (lecture-plan)
- From Verilog to HLS (verilog-hls)

賴瑾　Jiin Lai

Adjunct Professor at NTU, NTHU, NYCU
Co-Founder, ex-CTO, VIA Technologies Inc.

- ***MSEE, University of Texas, Austin***
- ***BSEE, NTU***

- **Over 30 years of experience in the PC industry**
- **12 years in the storage area.**
- **Co-founded VIA technologies in 1992, remained CTO responsibility including product and architecture development, with an eye toward future computing architecture needs:**
  - **Developed Intel/ AMD compatible chipsets and x86-compatible processors.**
  - **Developed the SSD controller**
  - **Distributed computational storage systems.**
- **Invented 200+ patents including over 50 US patents.**
- **Started from 2020 teaching “Application Acceleration with High-Level Synthesis” at NTU, NTHU, and NYCU.**
- **Founded the Bridge of Life Education Foundation to promote technology education in school.**

Background

Background 1:
Trend of the World

- Affected by the Sino-US trade war and geopolitics,
  - the world is competing to establish its semiconductor industry supply chain and compete for talents.
  - Chip design centers have been established
- Taiwan government quickly passed the "Regulations on Industry-University Cooperation and Talent Cultivation and Innovation Program”.
- Institute of Semiconductors was established in NTU, NTHU, NYCU, and NCKU.
- While increasing the "quantity" in cultivating semiconductor talents, HOW TO IMPROVE TALENT’S QUALITY?

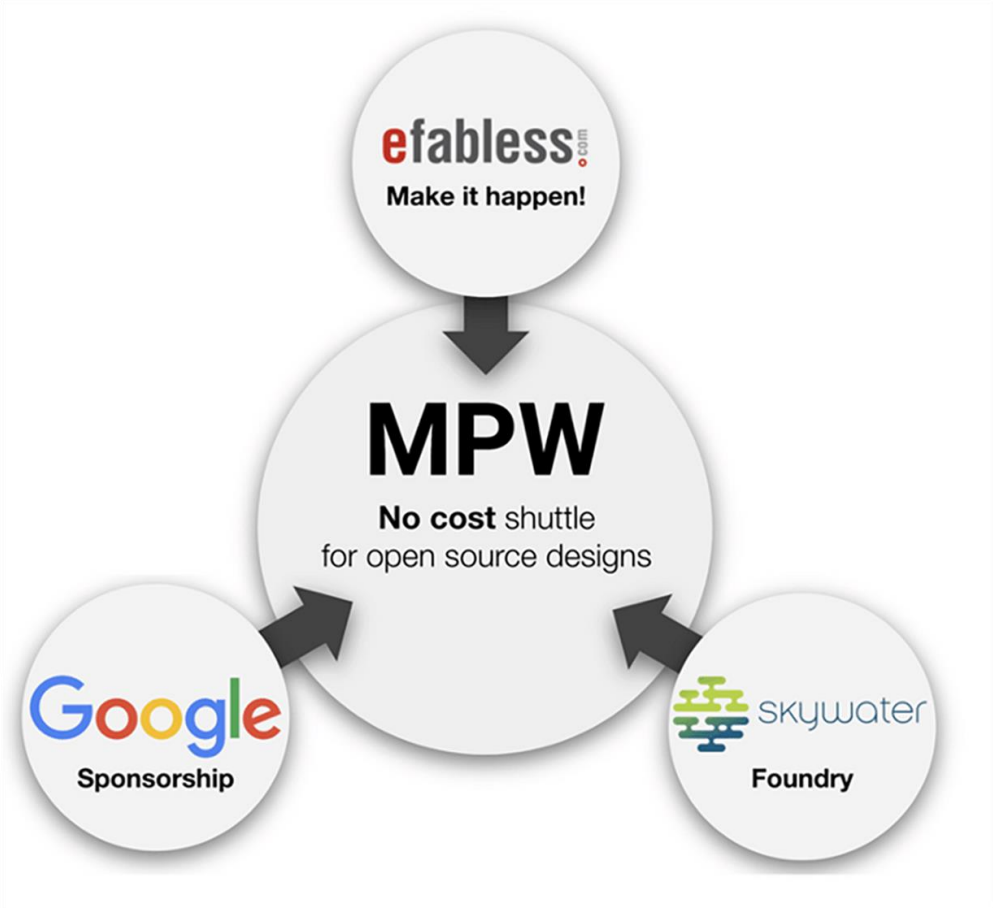
Background 2:
Difficult to cultivate IC design talent?

- Why is it more difficult to cultivate IC design talents than software talents?
  - Developing a chip requires many EDA tools
  - Difficult to obtain IP
  - After the chip is designed, tape-out is required before verification can be done

- **Google Open Source Silicon Program** comes to address the issues

- Chip validation, especially system-level validation, still not addressed.

What is **Google Open Source Silicon Program**



Google Sponsors a

Fully Open-Source Design

Capability for Chips

- Open-source PDK
- Open-source EDA Tools
- Free Fabrication

Six shuttles, 40 slots per shuttle, free to designers of fully open-source IC and IP designs.

SkyWater open-source 130nm PDK

https://developers.google.com/silicon

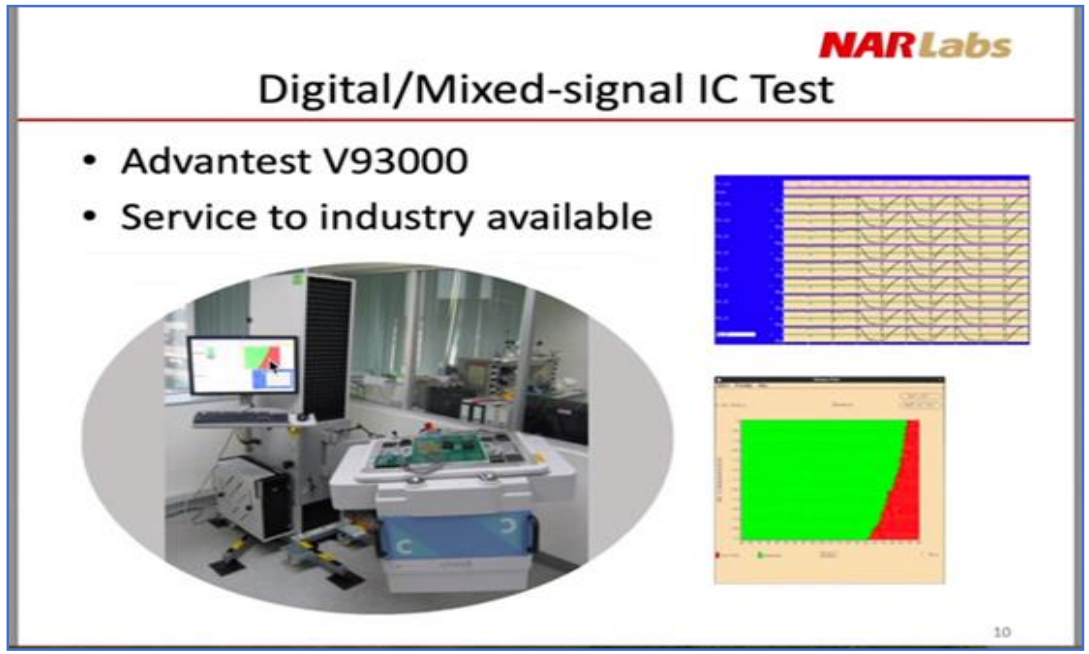
System-level Validation not addressed

- Importance
  - Silicon validation is often the bottleneck for time to market
  - Incomplete system validation causes tremendous damage to the company
  - Unfortunately, school does not teach it.

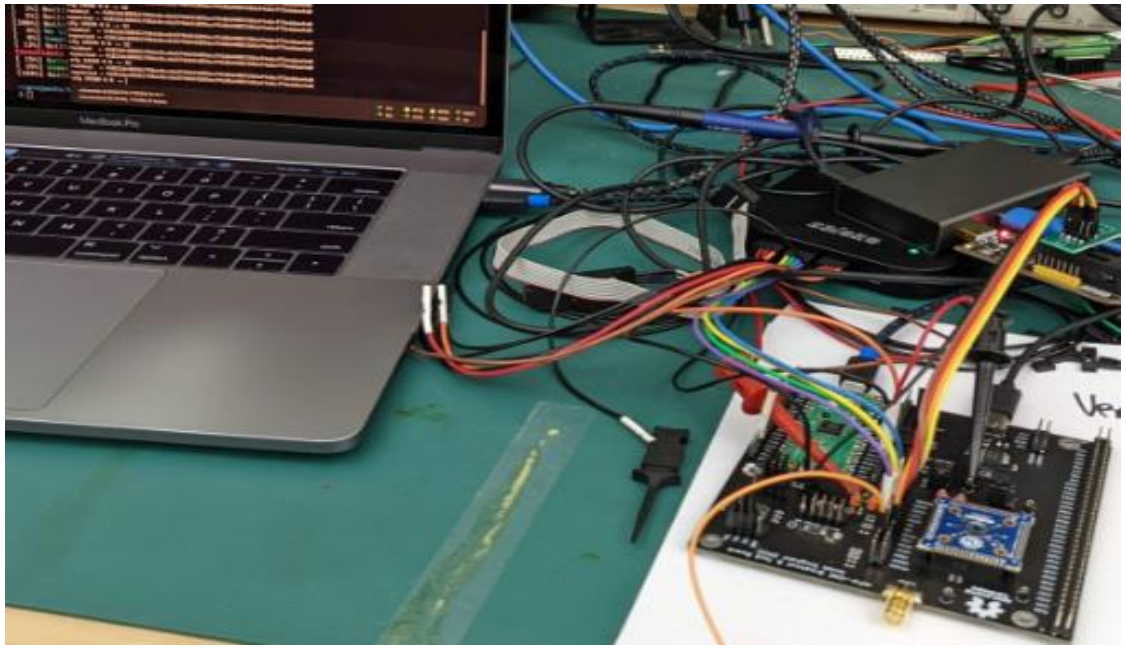
- Difficulty
  - Designer does not design-in necessary observability and controllability
  - IC designer lacks of system operation knowledge
  - It is difficult to build validation platform
  - System debugging is often by software means

Silicon Validation – IC Testing v.s. System Validation

IC Testing
- Testing with Tester, e.g. V93000
- Open-short / electrical test
- Manufacture test
- Function test



System Validation
- System test
- Customized Board design
- Test fixture, control / probe points



Both are difficult to deploy

Objectives

Objectives

Educate a full-stack IC designer with

- IC design
- FPGA design
- Embedded Programming

By Hands-on Labs on SOC-level Platform

One Year Program - SOC Design to train a full-stack Designer (Offered in NTU, NTHU, NYCU)

- Logic Design
- FPGA Design
- Embedded Programming

SOC Design (1st semester)

Objective:

- Learn Verilog and HLS Design Implementation on FPGA and ASIC
- Implement an User Project and Integrate it into Caravel SOC
- Study the SOC design and emulate it in CaravelFPGA

Design & Lecture

1. Introduction to HLS and tools
2. Verilog & Logic Design
3. Caravel SOC
4. Processor
5. Memory
6. Peripheral
7. Embedded Programming
8. SOC – Interconnect
9. Static Timing Analysis
10. Synthesis & Optimization
11. Verification & Simulation

Design Flow

1.Tools – Tcl, Perl, Makefile
2.FPGA flow – Xilinx Vivado
3.Simulator
4.Synthesis
5.Timing analysis
6.Verification Methodology

Laboratory

1.Vivado Tool Installation
2.HLS - FIR Filter (AXI master, AXI Stream)
3.Caravel SOC Simulation
4.Caravel SOC FPGA
5.SOC Design Labs
    1.Interrupt
    2.User RAM
    3.UART
    4.SDRAM
6.Workload Optimized SOC (WLOS) Baseline
7.Final Project

Advanced SOC Design (2nd Semester)

Objective:

1.Learn Advanced topics in IC Design, SOC chip-level design

2.Develop an Application Accelerator

3.Complete IC design flow, and be ready for tape out.

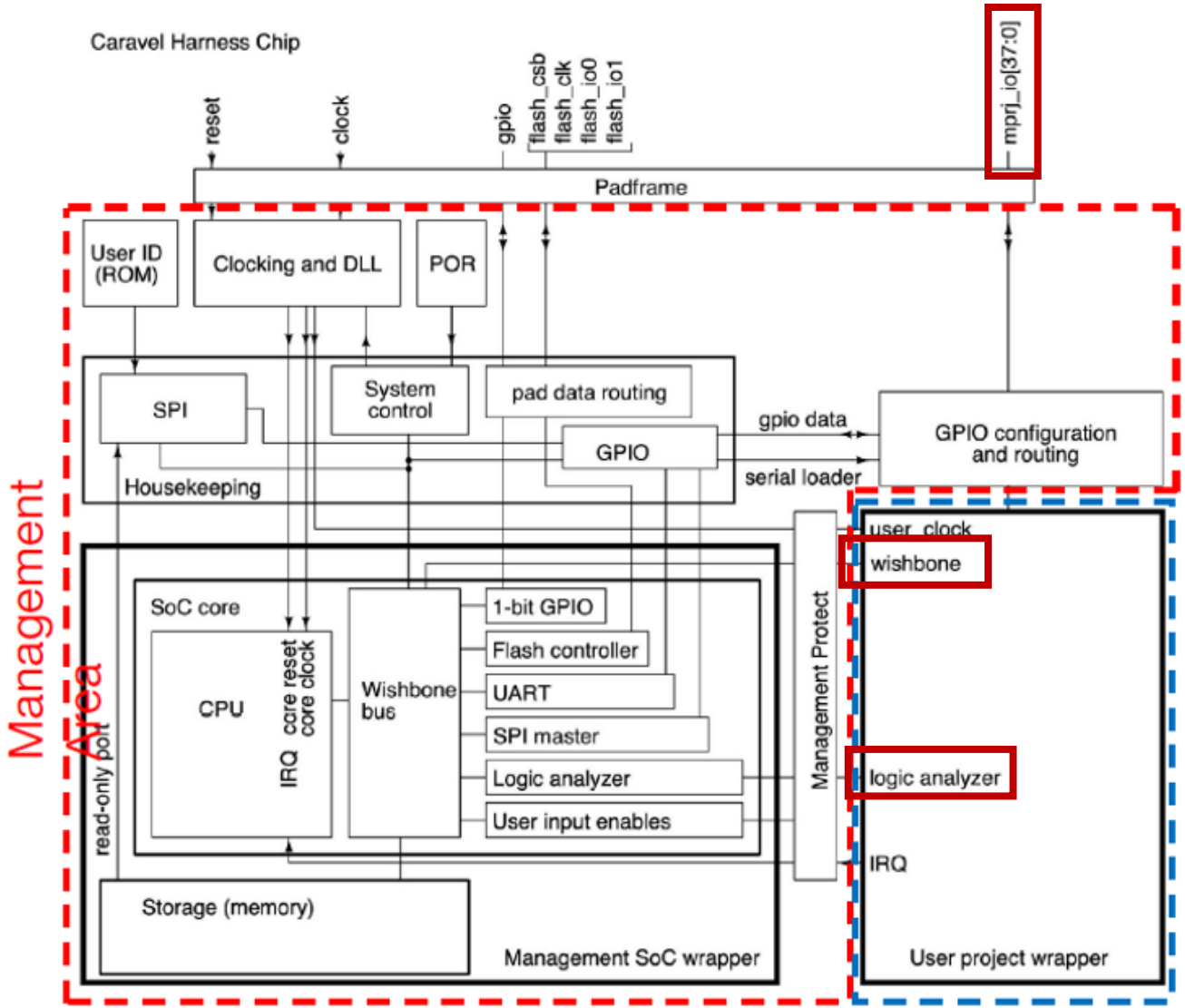
Design
1.Selected topics on high-performance Design
2.Advanced HLS Topics
3.Advanced Static Timing Analysis
4.SOC Chip level Design Issues
5.Advanced Embedding Programming
6.Advanced Design issues

Design Flow
1.Catapult (ASIC HLS)
2.Floorplan
3.Placement
4.Clock tree
5.Routing
6.DFT and Testing
7.LVD, DR · ERC
8.Post-layout Timing Analysis

Laboratory
1.Algorithmic HLS Lab
2.FSIC Module Design & Verification
3.User Project Development
4.FSIC Integration and Simulation
5.FSIC Physical Implementation and Tapeout (in June)

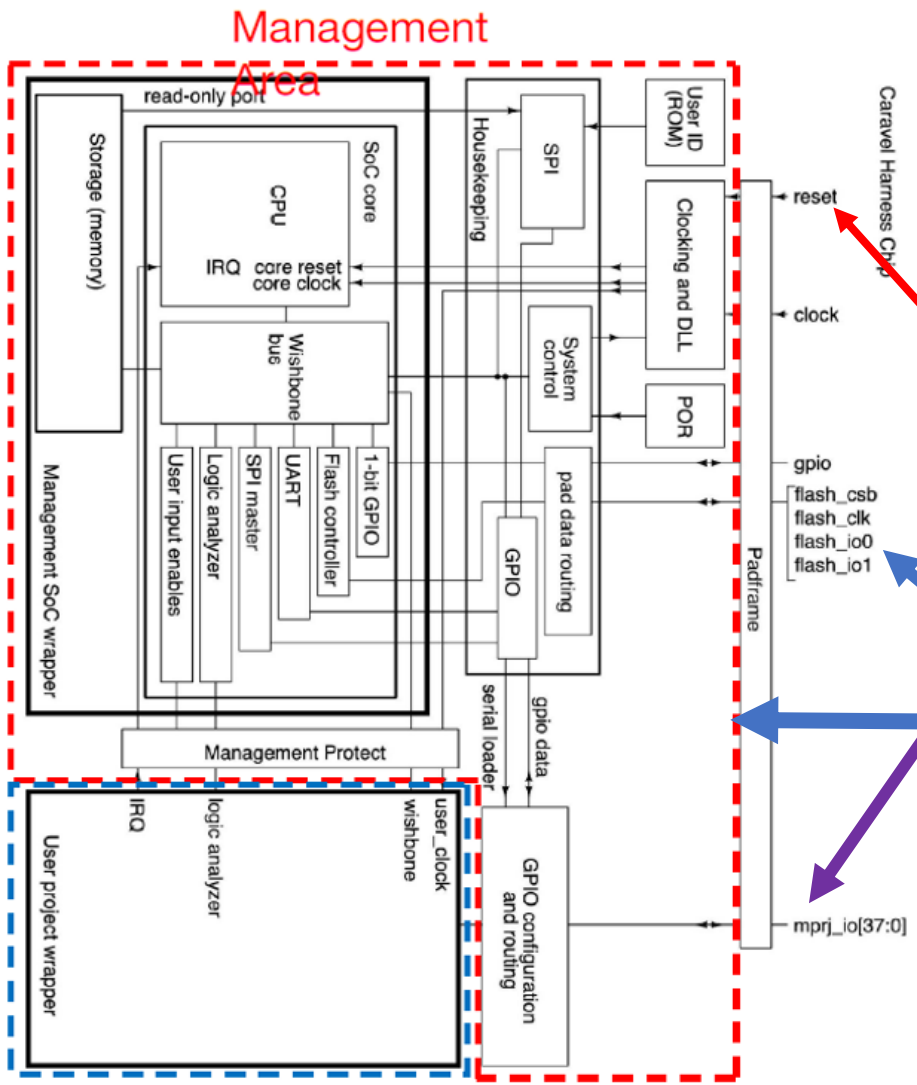
Lab Platform – CaravelFPGA©

Caravel Harness



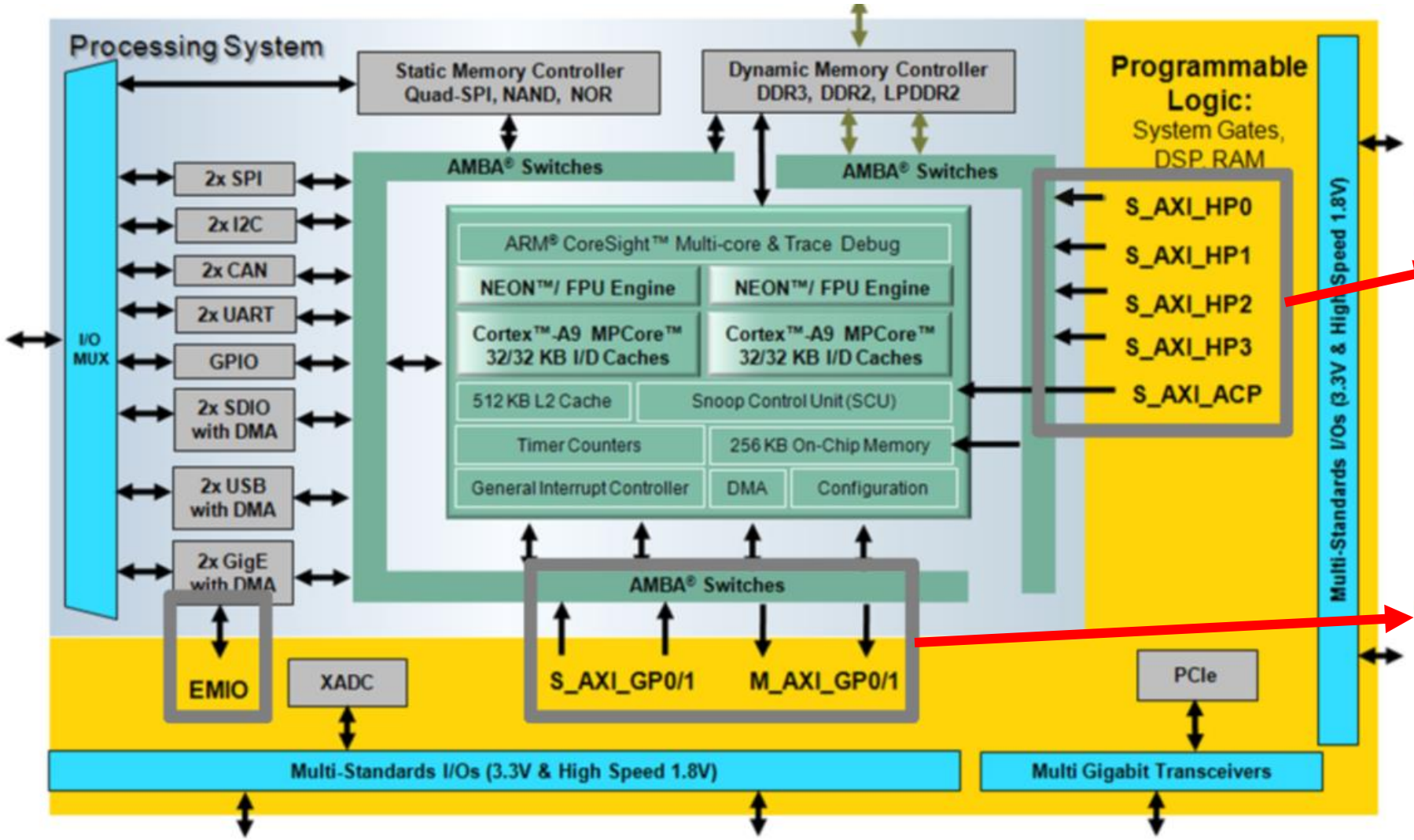
Caravel Simulation Verification System

Verilog Testbench




```
module counter_tb;
Initial beigin
 wait(checkbits == 16'hAB40);
 wait(checkbits = ..
end

Initial begin
  RSB <= 1'b0;
  RSB <= 1'b1;
end

caravel uut(    );

spiflash #( .FILENAME("counter_la.hex"} spiflash ( );

tbuart tburat (     );

endmodule
```

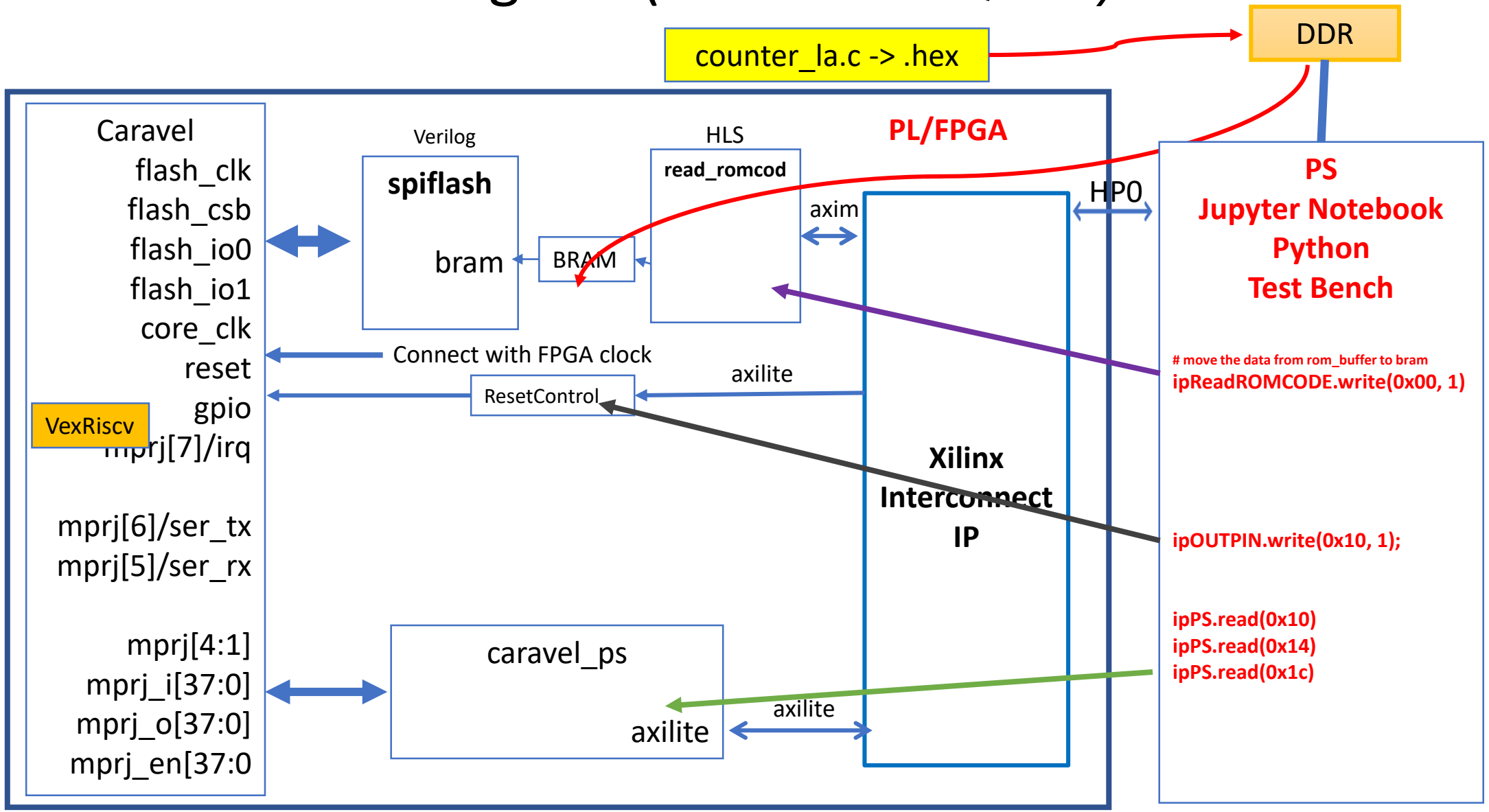
Zynq Block Diagram (PYNQ-Z2 or KV260)



- **AXI High Performance**
  - 4x Slave (64-bit)
    - 1K FIFOs
- **ACP**
  - 1x Slave (64-bit)
  - Cache access

- **AXI General Purpose ports**
  - 2x Master (32-bit)
  - 2x Slave (32-bit)

CaravelFPGA Block Diagram (Based PYNQ-Z2)



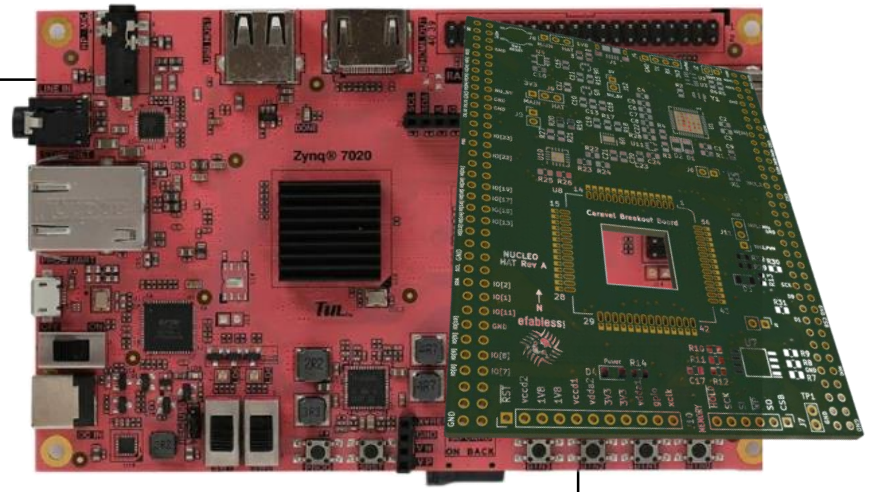
Benefits of CaravelFPGA

- Can validate your design much faster at hardware speed
- Can run real application before tapeout the chip
- Can use FPGA rich peripherals to communicate with outside, e.g. network, HDMI ….

Lab Platform – FSIC FPGA©

(2nd Semester)

FSIC - IC Validation System

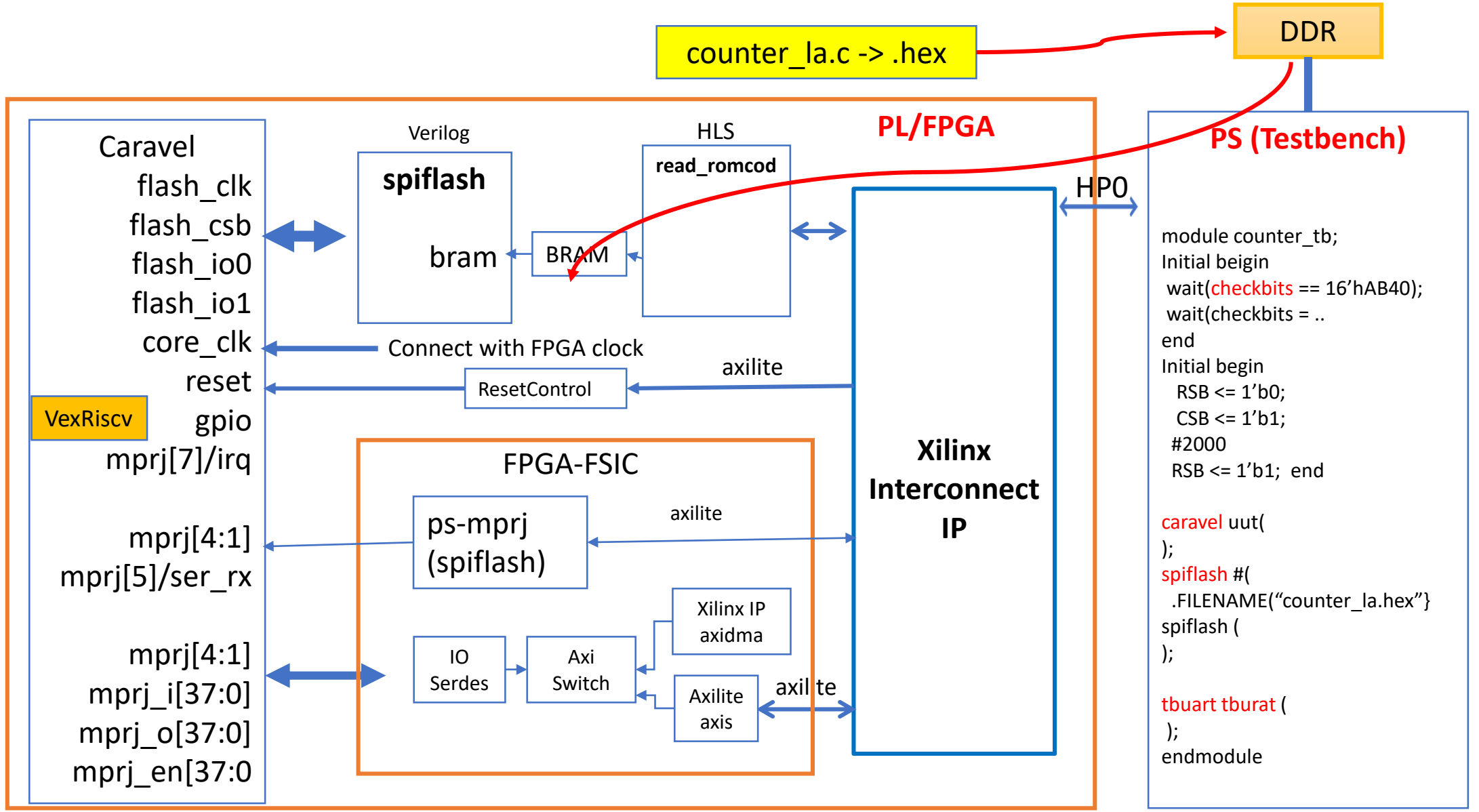


Caravel Chip DUT

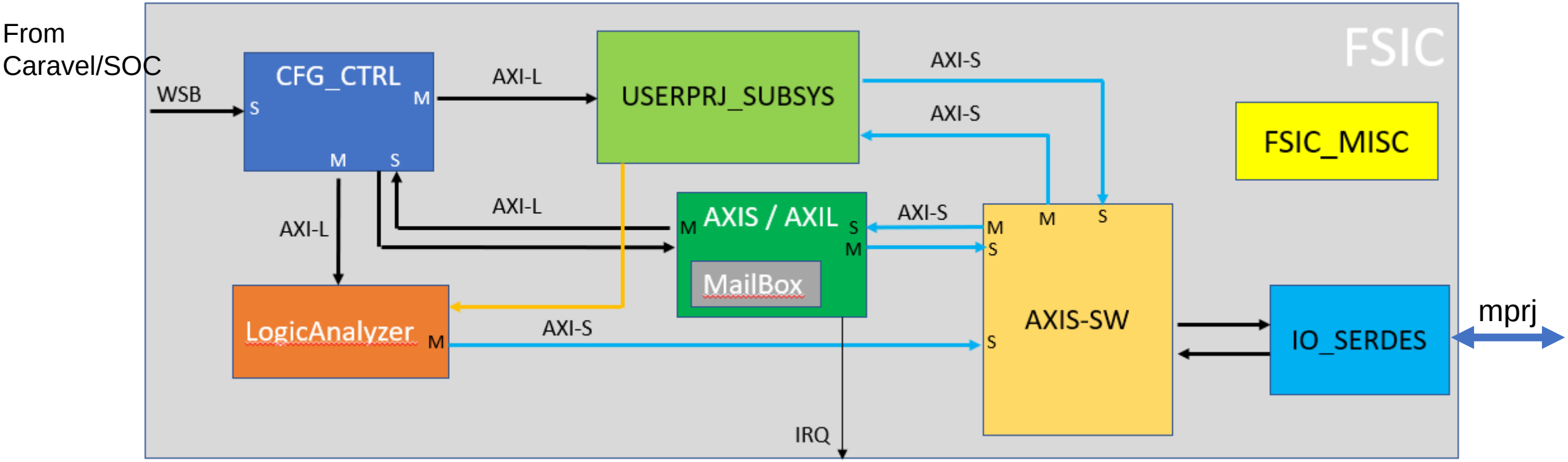
Caravel Harness Chip

reset
clock
gpio
flash_csb
flash_clk
flash_io0
flash_io1
mprj_io[37:0]
Padframe
User ID (ROM)
Clocking and DLL
POR
SPI
System control
pad data routing
GPIO
Housekeeping
gpio data
serial loader
GPIO configuration and routing
user_clock
wishbone
2.92mm x 3.52mm
logic analyzer
IRQ
Protocol Coversion
Logic Analyzer
Management Protect
SoC core
CPU
core reset
core clock
IRQ
Wishbone bus
1-bit GPIO
Flash controller
UART
SPI master
Logic analyzer
User input enables
read-only port
Storage (memory)
Management SoC wrapper
User project wrapper
axis
PYNQ FPGA
PL
Data Mover
PS
WAN
Remote Host
Jupyter Notebook

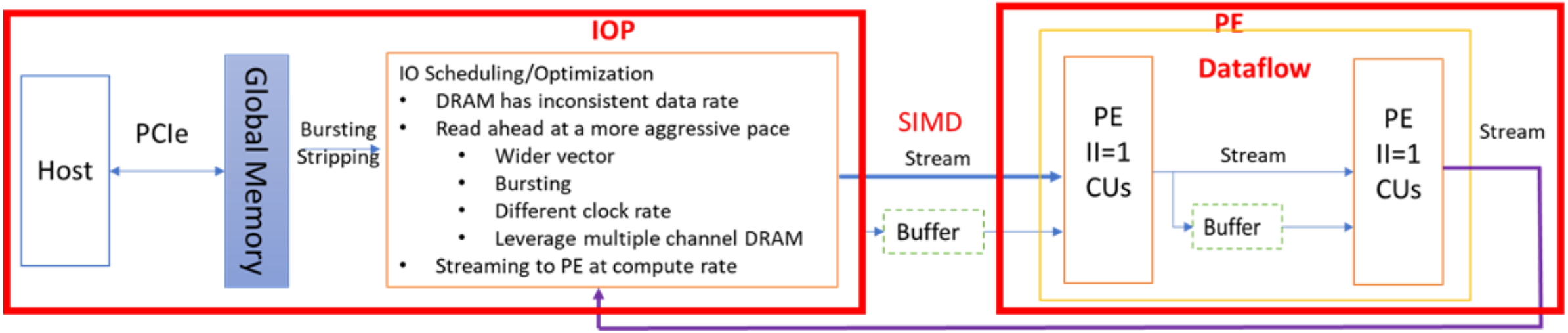
Caravel-FSIC + FPGA-FSIC



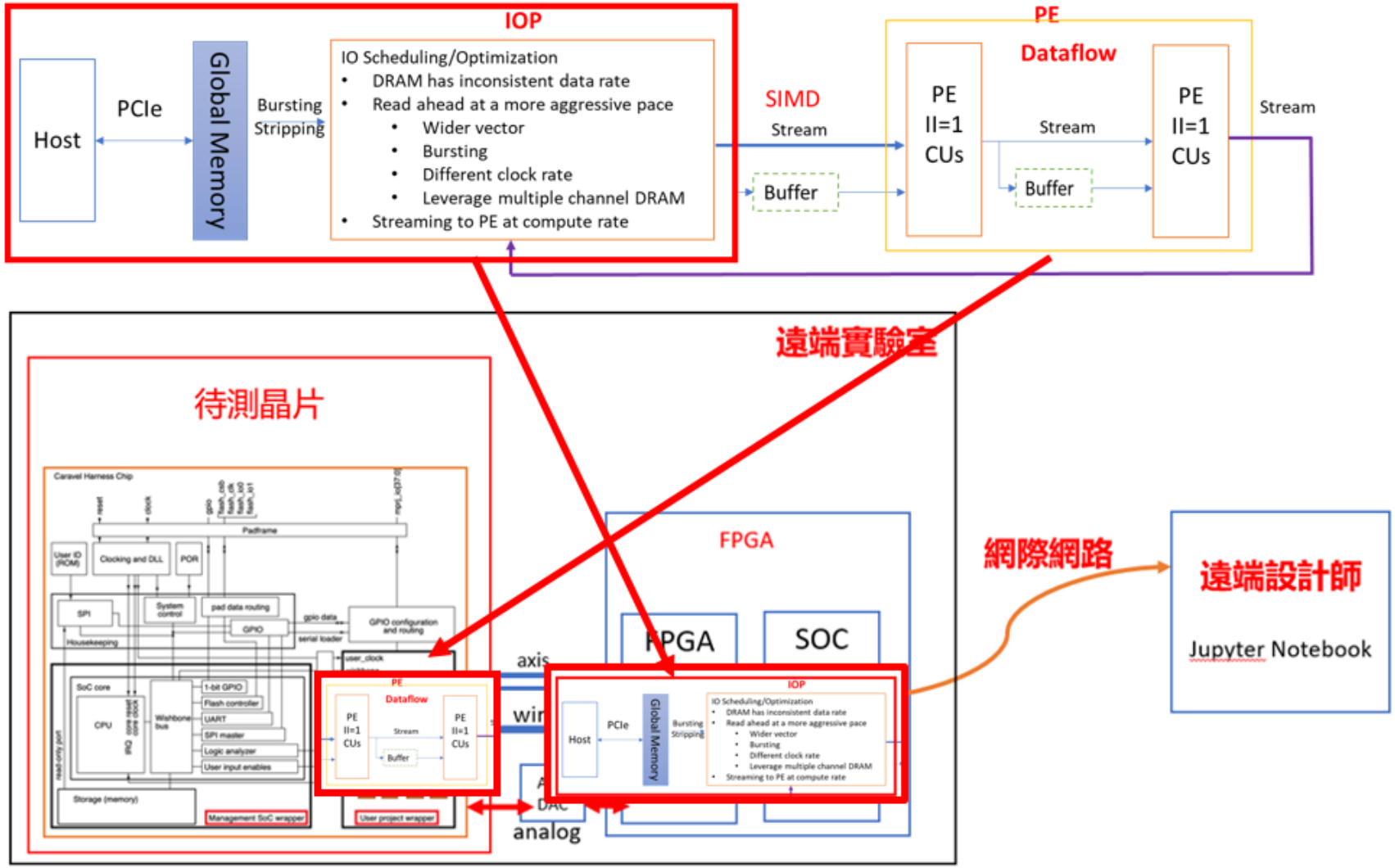
User Project Wrapper



Architecture for Application Accelerator Design



Embed Application Accelerator in FSIC

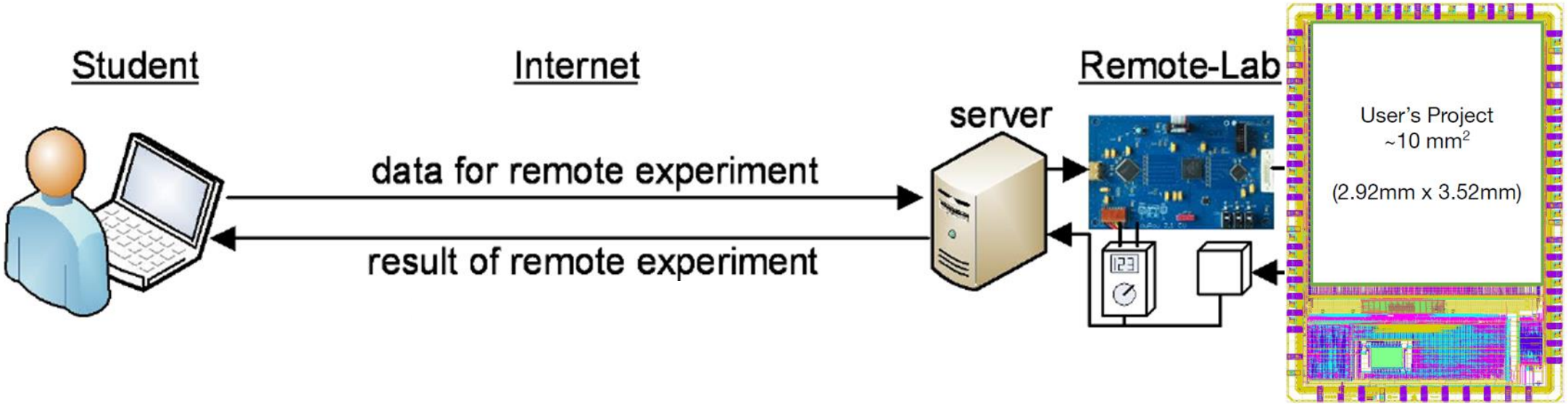


Benefits of FSIC

Build a Full-Stack IC Designer Program to enable

- System/Application level validation
- Scalable deployment – Online, remote debugging
- FSIC design lab training system

Online Lab System



Collaborative Learning & Participate in Q & A

HLS and SOC Design is in its evolving development process. It is hard to teach the subject in academic environment where known and matured knowledge is taught. The only way to learn it is through collaborative learning, through a Q & A process.

Subscribe the following Slack, and Github to communicate idea.

- Slack channel: “soclab”
  - https://boledu.slack.com/archives/C05D406JUSX
- Github/Discussion Forum “HLS-SOC-Discussions”
  - https://github.com/bol-edu/HLS-SOC-Discussions
  - https://github.com/bol-edu/HLS-SOC-Discussions/discussions

For topics related to lecture contents, laboratory work and project implementation problems, please use **Github/Discussion** Forum

Lecture & Lab Schedule

week	Date	In Class: Lecture, Presentation	Lab	Lab Due
1st Semester				
1	**13-Sep**	Course plan / From Gate to HLS	Lab#1 - Tool installation (1w) - individual	27-Sep
2	**20-Sep**	PYNQ-Lab2 Software Hardware Codesign / HLS Introduction / Kernel IO / Structure Design	Lab#2 - FIR (stream/master) (1w) - individual	4-Oct
3	**27-Sep**	Verilog and Logic Design	Lab#3- Verilog FIR & XSIM & GTKWave (2w) - individual	18-Oct
4	**4-Oct**	Caravel SoC System Introduction		
5	**11-Oct**	Computer & Microprocessor Architecture - RISC-V	Lab#4-1 Caravel SOC - Management FW (1w) - team	25-Oct
6	**18-Oct**	SoC Peripherals - UART, SPI, I2C, GPIO, UserProject IO, DMA	Lab#4-2 Caravel User Project - FIR (1w) - team	1-Nov
7	**25-Oct**	SOC Interconnect (Wishbone, AXI, Switch, DMA)	Lab#5 - Caravel FPGA (2w) - team	15-Nov
8	**1-Nov**	SOC Memory - Cache/DDR		
9	**8-Nov**	Caravel SOC Lab Presentation - Lab#4-1, Lab#4-2, Lab#5	Lab#A-E (3w) - team	29-Nov
10	**15-Nov**	Midterm		
11	**22-Nov**	Embedded Programming (ISA, Interrupt, Debugging)		
12	**29-Nov**	Static Timing Analysis	Lab#6 & Final Project - WLOS (4-6 w) - team	10-Jan
13	**6-Dec**	Caravel SOC Lab Presentation - Lab#A-E, Final Project Proposal		
14	**13-Dec**	Synthesis and Optimization		
15	**20-Dec**	Verification & Simulation		
16	**27-Dec**	No Class		
17	**3-Jan**	No Class		
18	**10-Jan**	Final Project Presentation		

Grading

Grading

Item	Content	Submission	Weight
Lab#1	Tool installation (individual)	Screen shot	2
Lab#2	HLS-FIR (individual)	Report	4
Lab#3	XSIM & GTKWave Simulation - Individual	Report	10
Lab#4-1	Management FW (team)	Github & Report	6
Lab#4-2	Caravel User Project - FIR (team)	Github & Report	8
Lab#5	Caravel FPGA (team)	Github & Report	8
Lab#A-E	Choose 2, each 8 points (team)		16
	Extra Lab, each 4 points (team)		12
	A: Interrupt Service	Github & Report	
	B: ExMem - User Project Memory	Github & Report	
	C: UART	Github & Report	
	D: WB-SDRAM	Github & Report	
	E: Software Emulation - Bit Banging	Github & Report	
Lab#6	Baseline WLOS (team)	Github & Report	10
Midterm			10
Final Project	WLOS Optimization (team)	Github & Report & ppt	20
Presentation	Selected presentation (extra credit)	ppt	3
StudyJournal	Github & StudyJournal (individual)	Github & StudyJournal.md	4
Soft Skill	Ask question, offer help, sharing	Send link of evidence to TA	2
		Total	115

Study Journal

- It is By you, and For you
- Benefit
  - Thought-starter and planning
  - Note of accomplishments, keep track progress
  - Material for report
  - Space for reflection, Facilitating a breakthrough
  - Maintaining the writing habit

- Reference: https://hackmd.io/@TonyHo?utm_source=preview-mode&utm_medium=rec

Soft Skills

- Participate Q & A in
  - In-class – Question & Answer
  - Participate in Slack channel: “**soclab**” - https://boledu.slack.com/archives/C05D406JUSX
  - For general topics of HLS and SOC (e.g., lecture contents, labortory work and project implementation problems), use Github/Discussion Forum
    - https://github.com/bol-edu/HLS-SOC-Discussions
    - https://github.com/bol-edu/HLS-SOC-Discussions/discussions
  - Submit question to Public Forum
- Offer help to solve questions
- Share your findings

- Please provide the link of evidence to TA so that it can be put into record