Python Training
Day 5: Final Project

Elkhan Julian Brillianshah
Anindhita Nayazirly Sukarno
Emmanuella Rumanti

Course Structure

- Section 1: Introduction
- Day 1: Introduction

- Section 2: Core Python
- Day 2: Basic Constructs
- Day 3: Modules and Libraries

- Section 3: Practical Applications
- Day 4: Files, Logs, and Processes
- **Day 5: Final Project**

Day 5 Objective

- The objectives of today's training is to utilize everything you've learned so far to validate a few implementation of an IP.

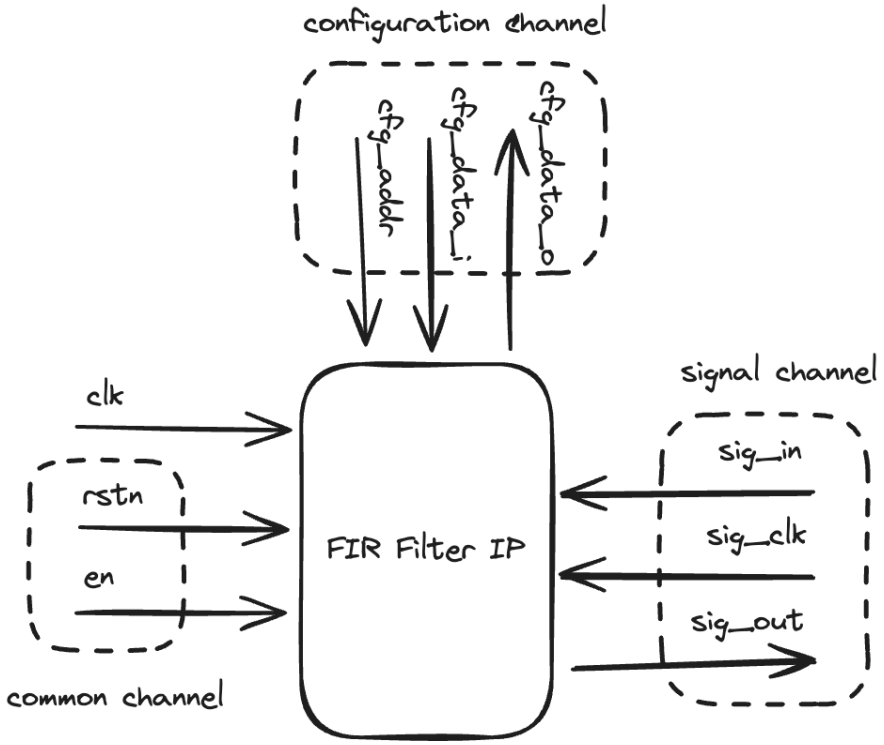
- You will be role playing as a post-silicon validation engineer in a startup company today. ☺

Scenario

- You are a PSV engineer in a company called Dawnstar. Your company recently taped out a die that contains a few instances of a newly developed hard IP. Your job now is to validate the compliance of those implementations to its specifications.

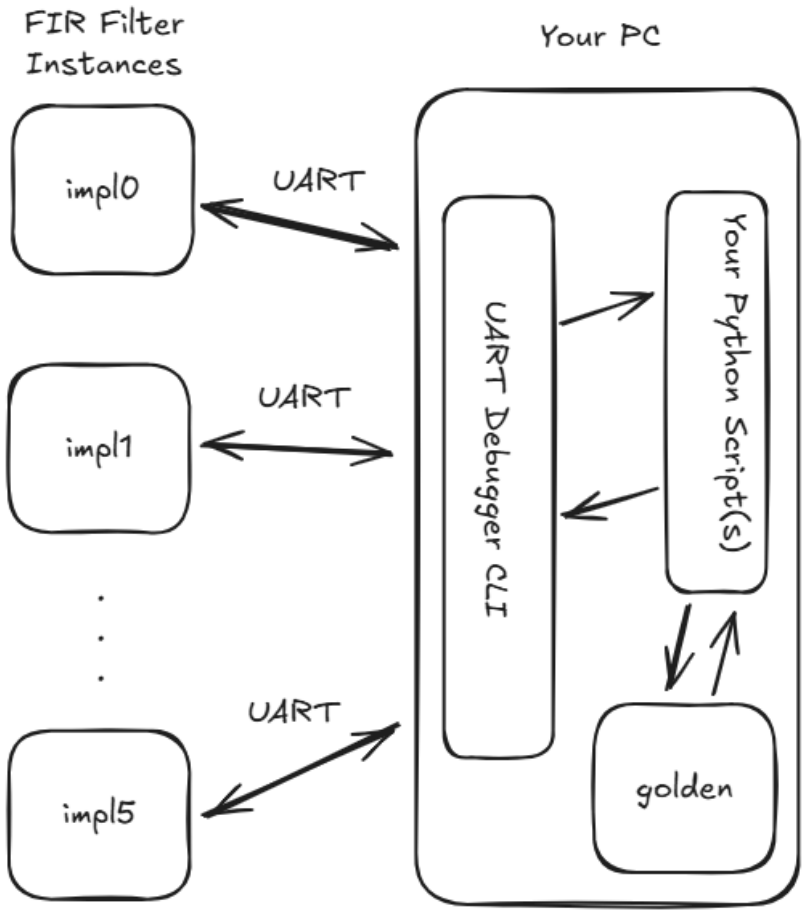
- You validate by executing tests on those instances. The tests are tedious, and you will need to repeat them for every instance that you have. You will now utilize your Python skills to automate the test execution to get speedy and accurate results!

The Tested



- You will be validating a few FIR filter instances. FIR filters are digital signals processors that transforms one digital signals into another based on its filter coefficients.

Validation Setup



- Your test setup is as shown by the figure. You will have 6 filter instances to validate as well as a simulated golden model. The golden model is guaranteed to behave exactly as specified.

- You can interact with the instances by using the UART Debugger CLI, it's an executable program that allows you to

The Tests

- You will need to test for these features on all the instances:

1. Global enable/disable
2. POR register values
3. Input buffer overflow and clearing
4. Filter bypassing
5. Signal processing

Tests: Global enable/disable

- Our IP has a global enable signal which needs to be asserted to use the IP. When the global enable signal is de-asserted, the IP is expected to be inactive and will not respond to any stimuli.

- **Passing criteria:**
- IP channels except for the common channel is inaccessible when enable is de-asserted

Tests: POR register values

- Power-on reset (POR) is mechanism in ICs where a reset signal is asserted when the power is first applied. Registers in the IP we are testing will be reset to their default values when a reset signal is asserted.

- **Passing Criteria**
- All register values after reset matches with the values as specified in por.csv

Tests: Input buffer overflow and clearing

- The FIR filter IP has an input buffer that stores sampled input signal when the filter is halted. It can store up to 255 samples before it loses data and can be cleared by setting the right register field.

- **Passing Criteria**
- Input buffer count is correct, the correct register field is set upon overflow, and that the input buffer count can be cleared.

Tests: Filter bypassing

- A register field in the instances can be set to bypass signal processing.

- **Passing Criteria**
- Output signal matches exactly with the input signal when the bypass is enabled.

Tests: Signal processing

- This is the main feature of the IP. You will need to set the filter's coefficients and drive its input signal. You will be provided with a .cfg file that specifies the coef. values and enables. You will also be provided with a .vec file that contains values of the input signals to drive.

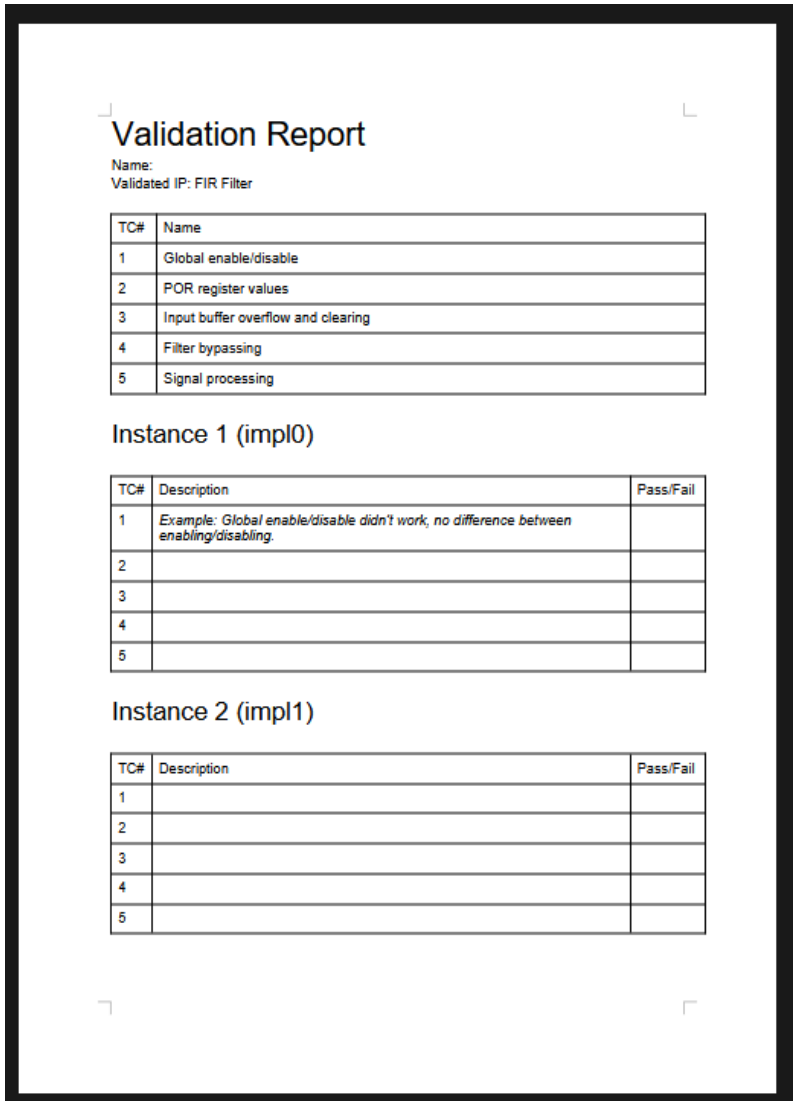
- **Proof of execution requirement**
- A visualization of the input and output signals

- **Passing Criteria**
- Output signals matches exactly with the output signals from the golden model given the same coefficients and input signals.

Validation Report

- After executing all those tests on every instance, you'll need to make a report that contains:

1. A proof of execution (ex: a screenshot of your script execution)
2. A brief analysis on the execution result (ex: register field X does not match with the POR values …)
3. A status of the test execution (pass/fail/wip)



Validation Report

Name:
Validated IP: FIR Filter

TC#	Name
1	Global enable/disable
2	POR register values
3	Input buffer overflow and clearing
4	Filter bypassing
5	Signal processing

Instance 1 (impl0)

TC#	Description	Pass/Fail
1	*Example: Global enable/disable didn't work, no difference between enabling/disabling.*	
2		
3		
4		
5		

Instance 2 (impl1)

TC#	Description	Pass/Fail
1		
2		
3		
4		
5		

Your Resources

- The following resources will be provided to you:

1. A High-Level Architecture Specification (HAS) document of the IP
2. A brief manual on how to use the UART Debugger Interface
3. A golden model to be used for script development and or test execution
4. Six executables of the UART Debugger Interface that corresponds to the 6 instances
5. Miscellaneous text files (por.csv, various filter config file, input signal file)
6. Some starter code

- Also, **feel free to collaborate** with your peers and use any resources you have access to.

Thank You