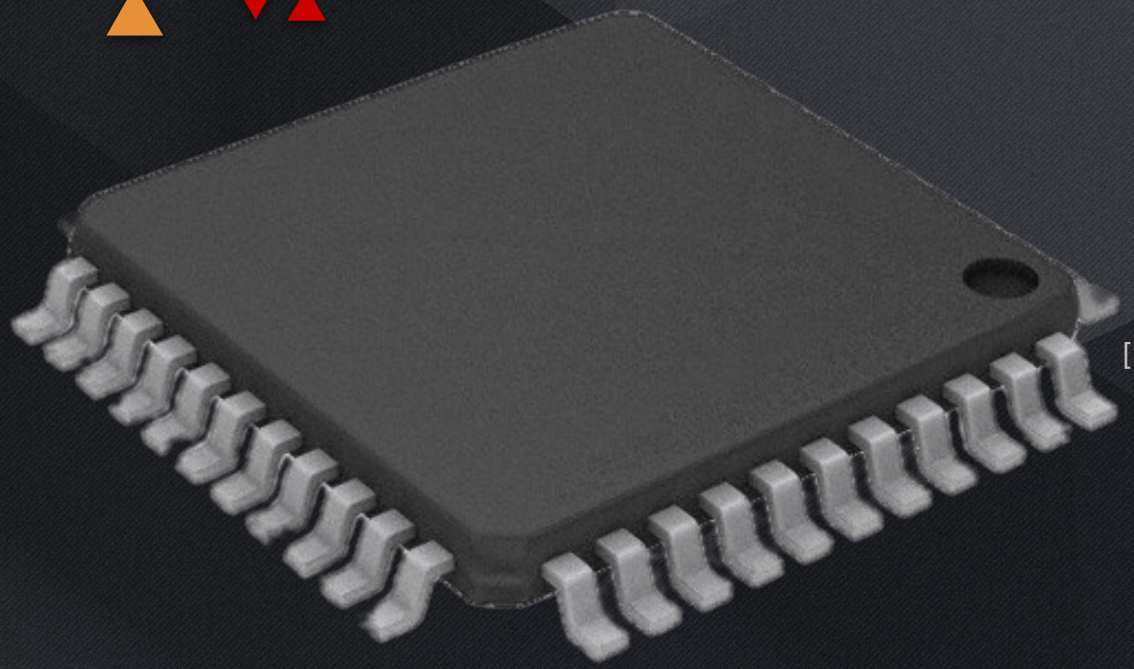

[1]

Digital ASIC Design
Faculty Panel Presentation

Gregory Ling
Will Galles
Jake Hafele
Cade Breeding

Class: CPR E 491
Group: sddec23-06
Client/Advisor: Dr. Duwe
Website: http://sddec23-06.sd.ece.iastate.edu/

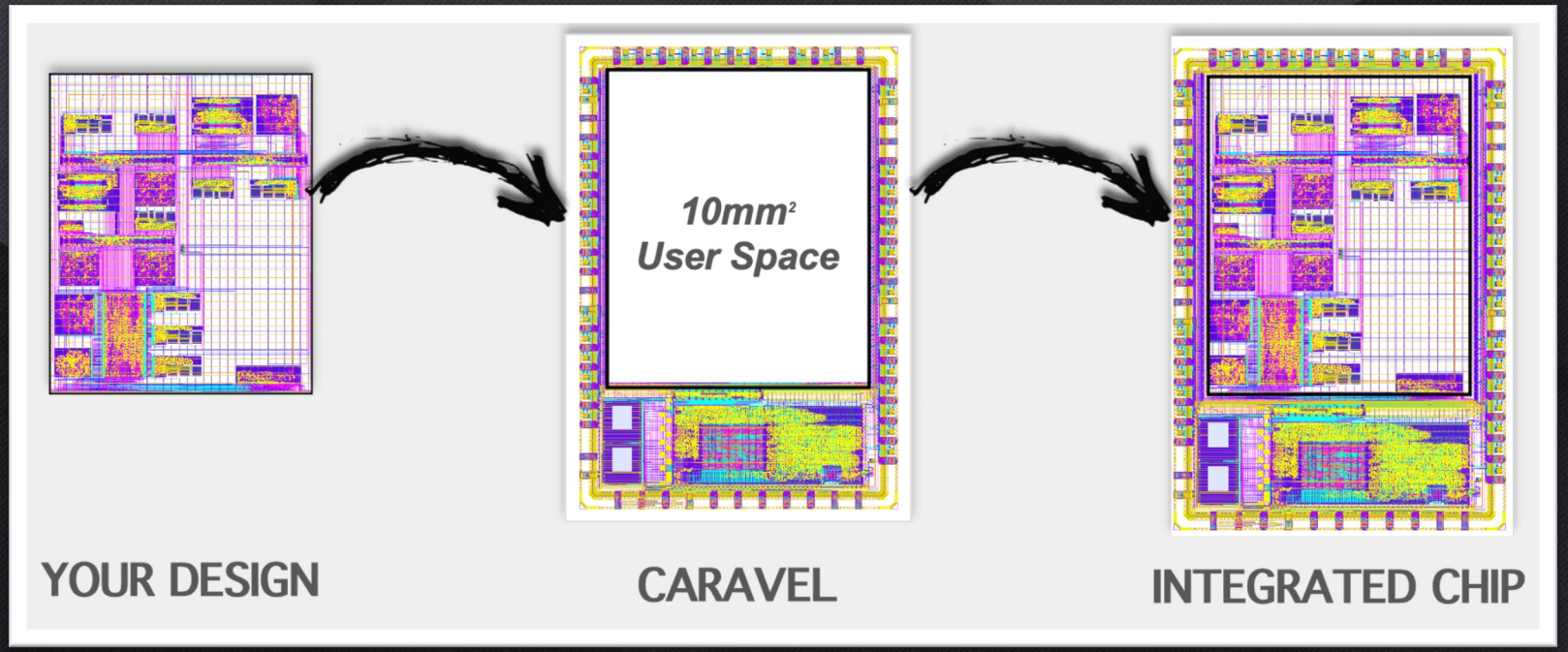
Introduction

Gregory Ling

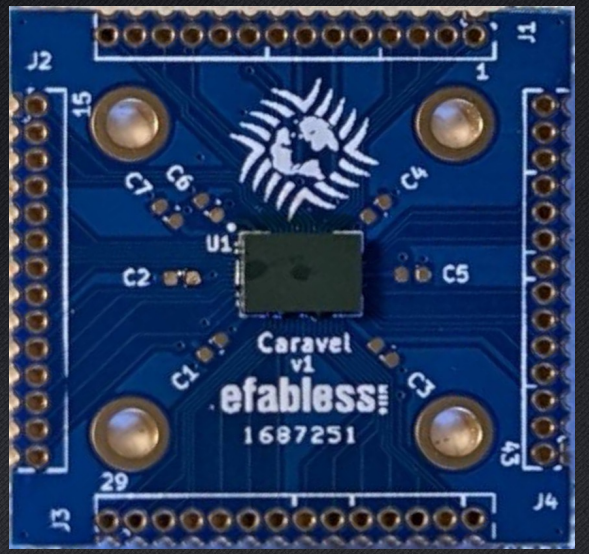
Overview
Users
Requirements
Constraints
Market Survey

Overview

- ASIC Design
- OpenMPW (eFabless)
- Open Source



[2]



[3]

Engineering Standards

- IEEE 1364-2005 – IEEE Standard Verilog Hardware Description Language
- ISO/IEC 9899:2018 – C programming language (C17)
- TIA/EIA 232-F – RS232 (UART) Protocol
- Serial Peripheral Interface (SPI) Protocol
- Wishbone Bus
- Inter-Integrated Circuit (I2C) Protocol

Users

- Our Team
- Research Groups
- Future Teams
- Open Source Community

Requirements

Explore

Modular design to test different limits of the fabrication process

Learn

Challenge ourselves to work in new areas

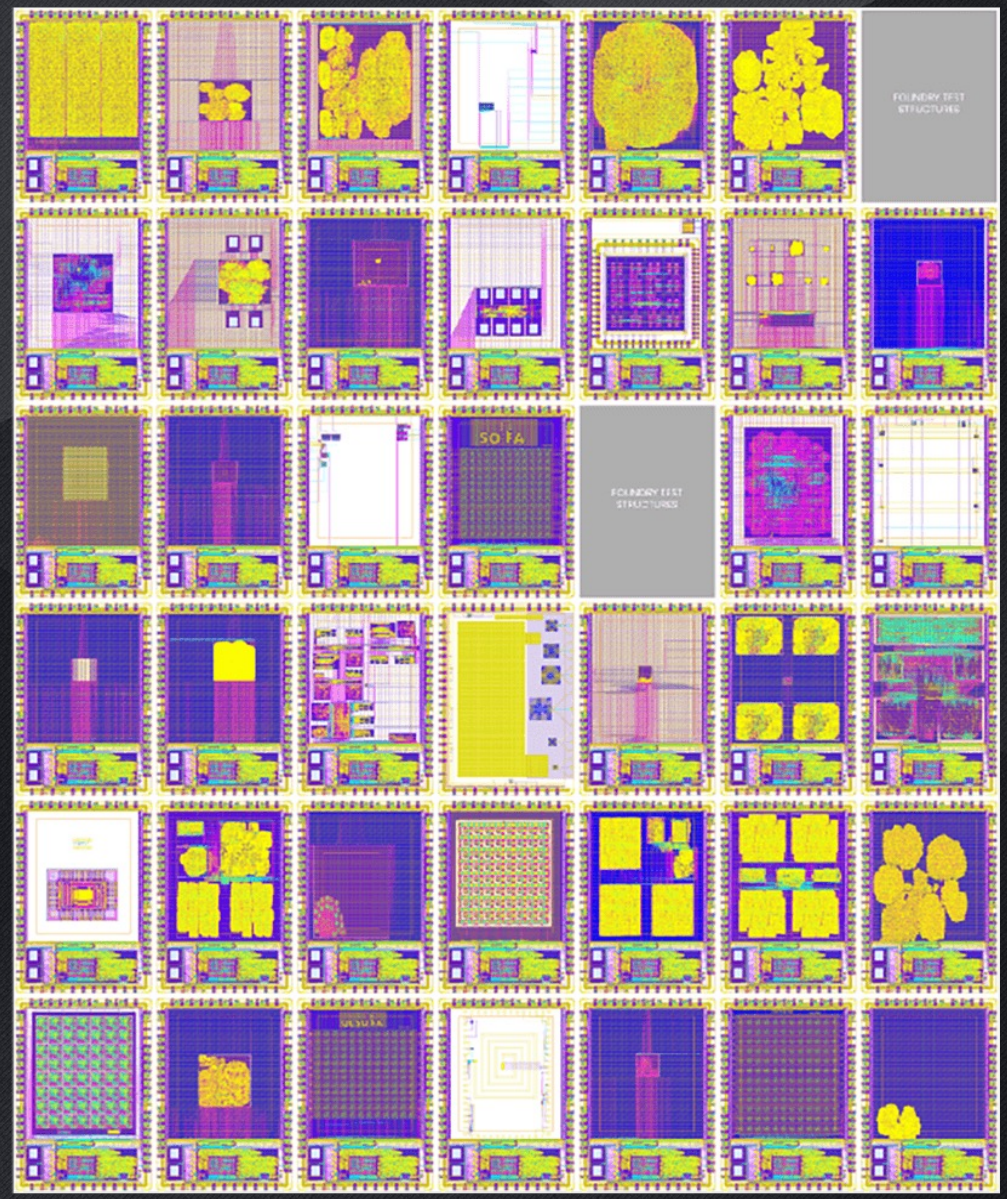
Bring-Up Plan

Create a plan for a future group to test our design

Constraints

eFabless

- 10 mm^2 user area
- SkyWater 130 nm fabrication process
- Verilog
- Specific folder structure
- MPW Precheck
- Open-source repository



[4]

Market Survey

Production Run

- ~$1M for full ASIC

Multi-Project Wafer

- OpenMPW – eFabless, open-source, funded by Google
- ChipIgnite – eFabless, paid ($9,750/project, 10mm^2, 4-5 months)
- Muse Semiconductor – TSMC, paid ($1,250/mm^2, 42 days)

Design

Will Galles

Functional Modules
Top Level Design
Potential Risks
Resource Cost Estimate

Module Overview

Wishbone Test

Gregory Ling

Clock Gating

Cade Breeding

Backdoor SPI

Jake Hafele

Custom Cell Test

Gregory Ling

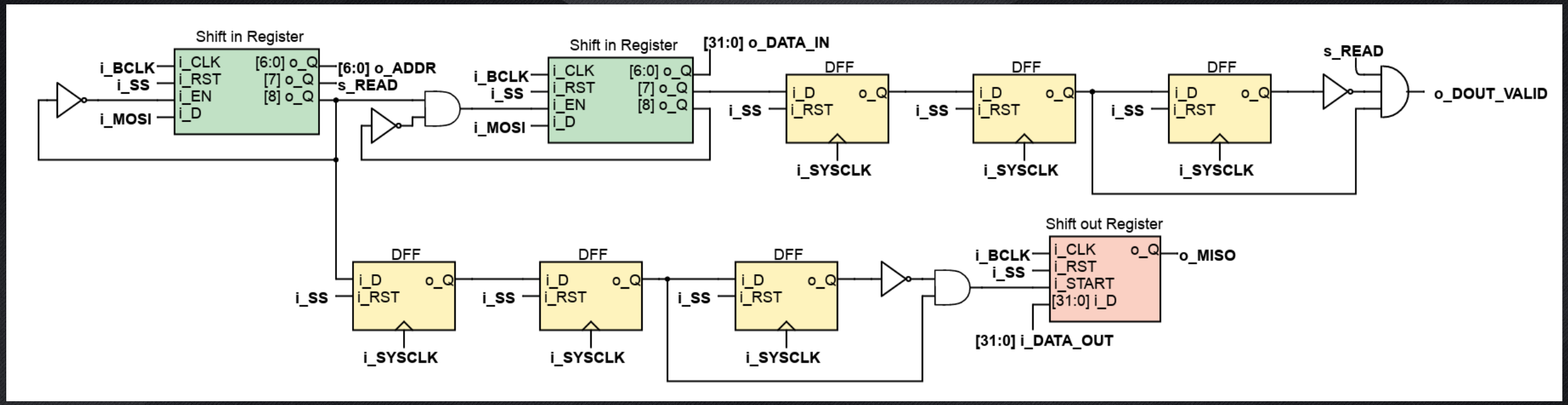
Standard Cell Test

Cade Breeding

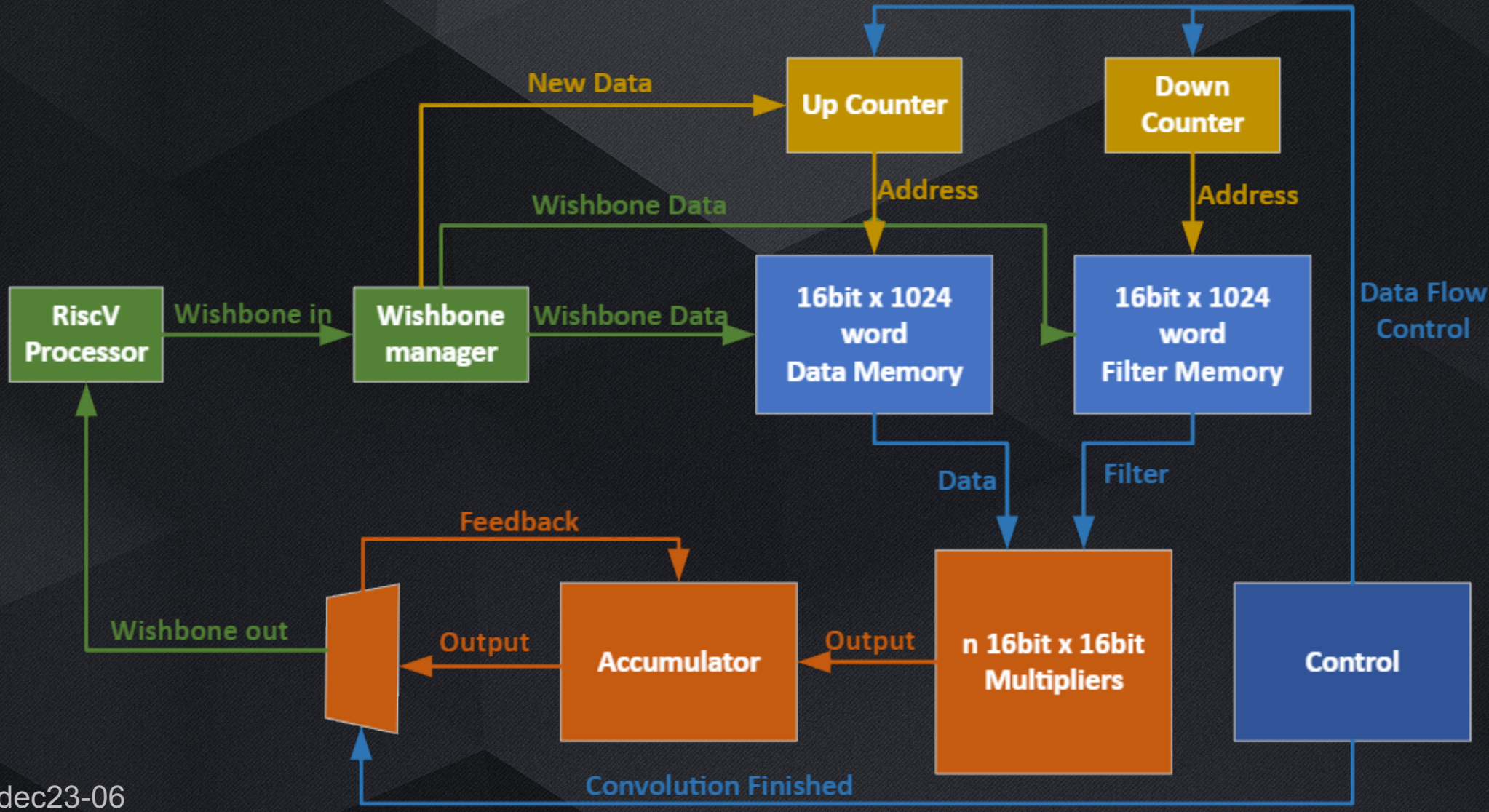
Voice Road-Noise Isolation

Will Galles

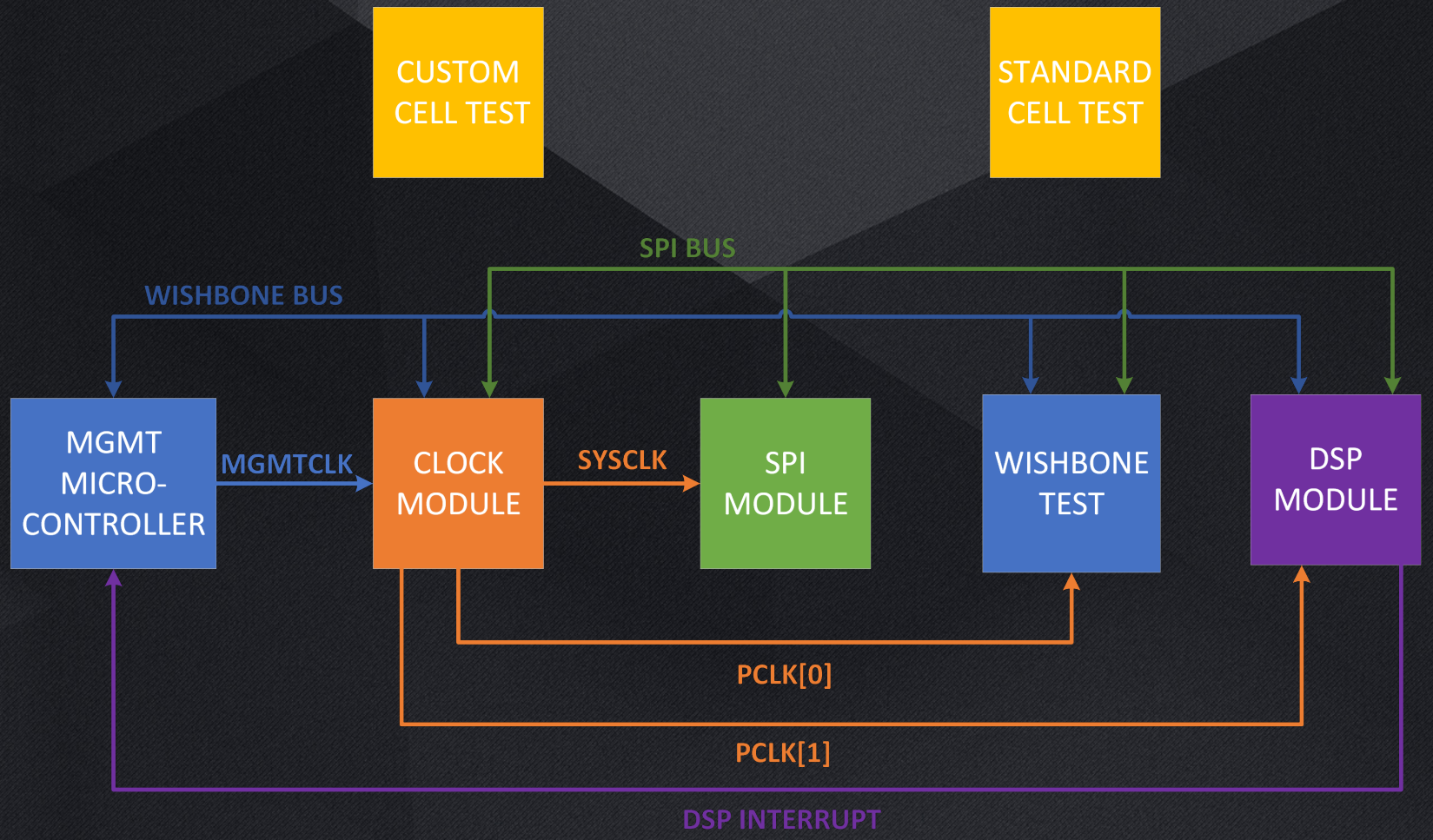
Backdoor SPI Module



Voice Road Noise Isolation Module (DSP)



Top Level Design



Potential Risks

Risk	Probability
No OpenMPW submission is available	~40%
DSP module cannot both fit in user area and run at real time	~40%
Wishbone bus is unable to interact with user modules after fabrication	~5%
Fabrication error causes an individual module to fail	~15%

Resource Cost Estimate

- Only real cost of the project is our time
- eFabless Open MPW Shuttle program funded by Google
- All design tools are open source
- GitLab and Microsoft Teams provided by the ECpE department

Testing

Jake Hafele

Platforms Used
Test Plan
Testing Thus Far

Platforms Used

OpenROAD

- RTL synthesis through GDS Layout

Magic

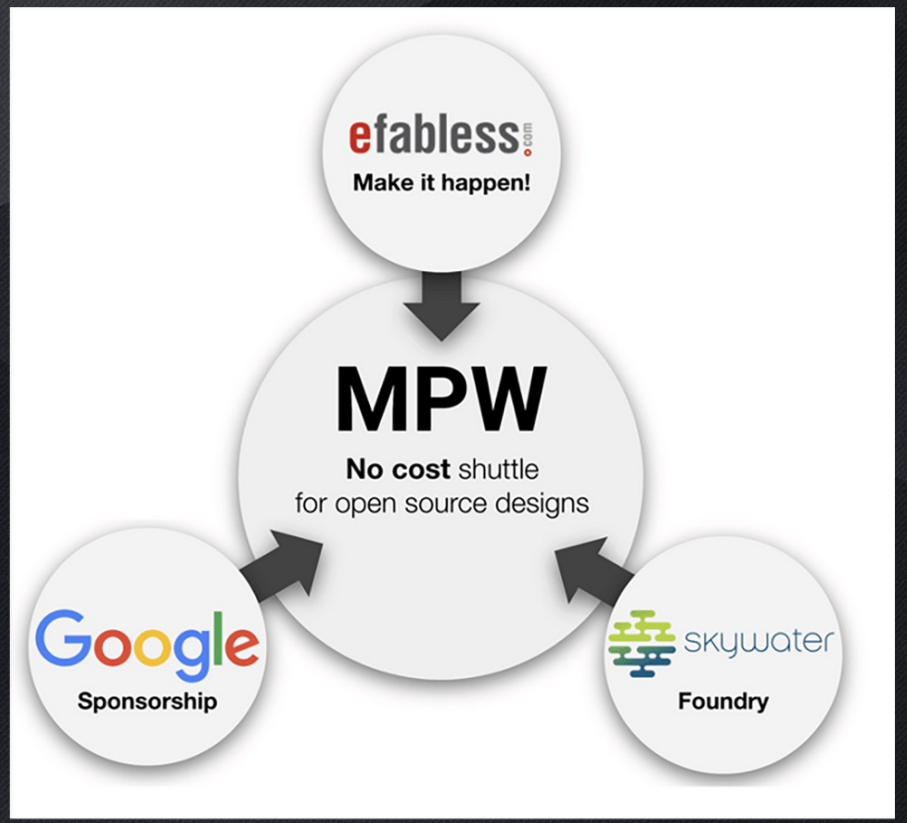
- Custom cell layout

KLayout

- Layout viewer

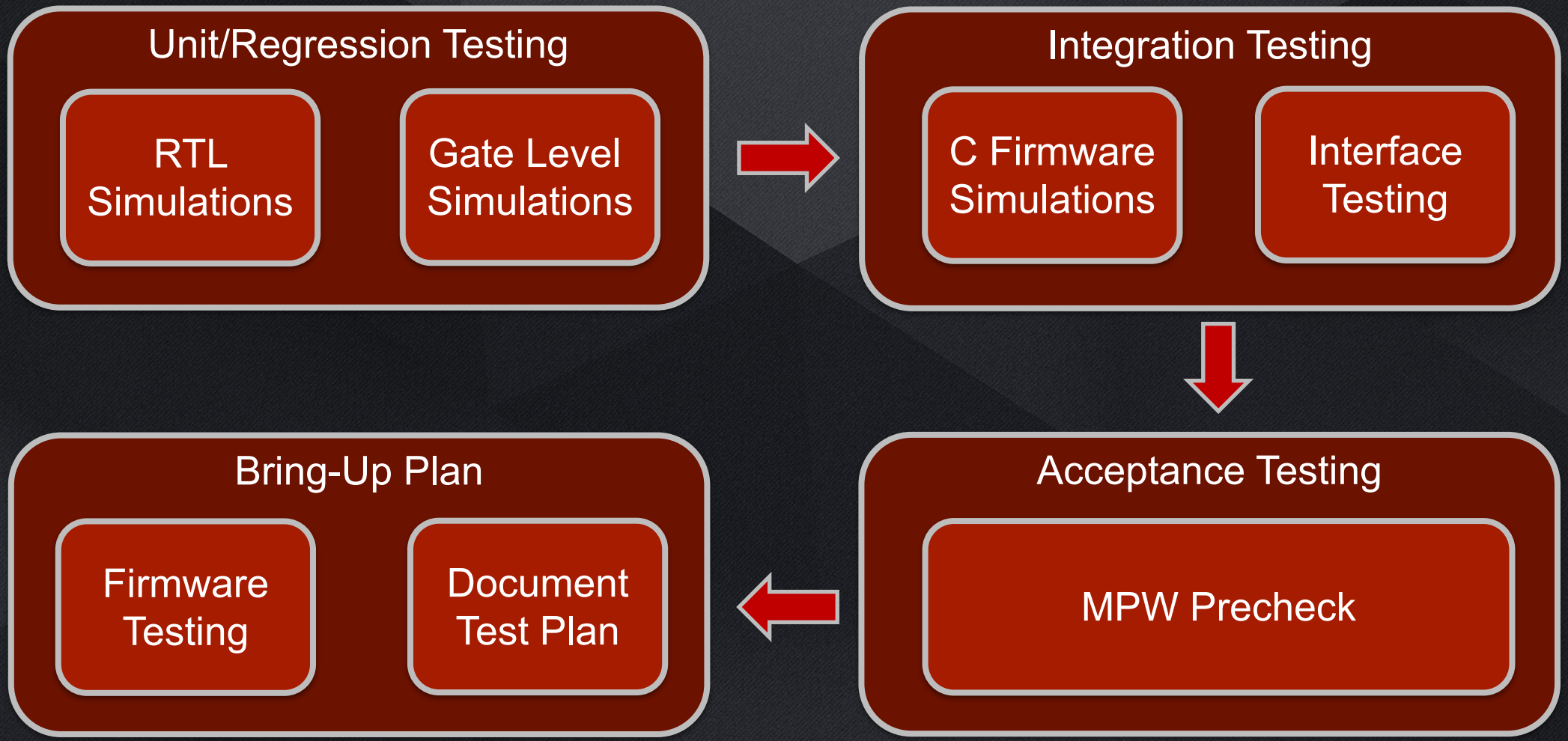
GTKWave

- Waveform viewer



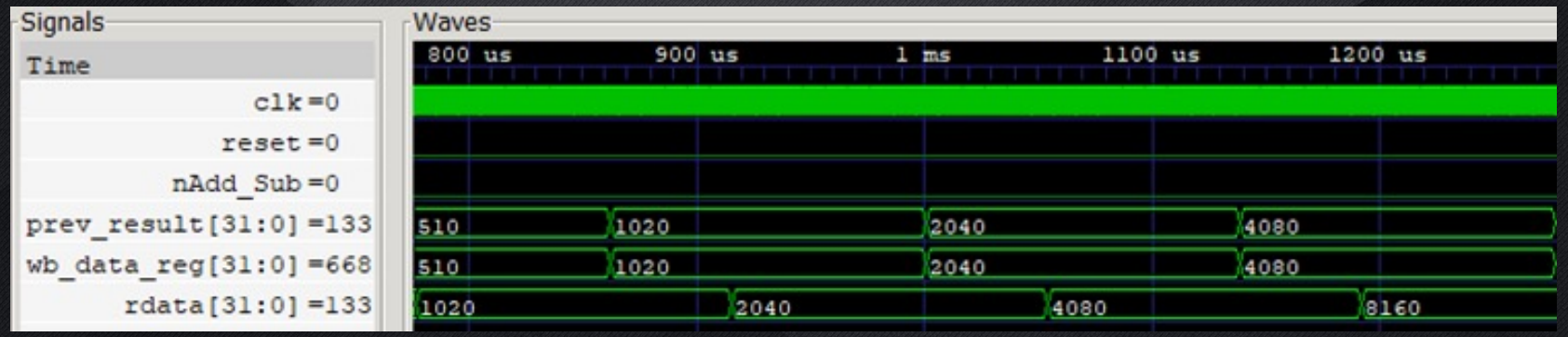
[4]

Test Plan



pp. 32-34

Testing Thus Far – Adder, Standard Cell



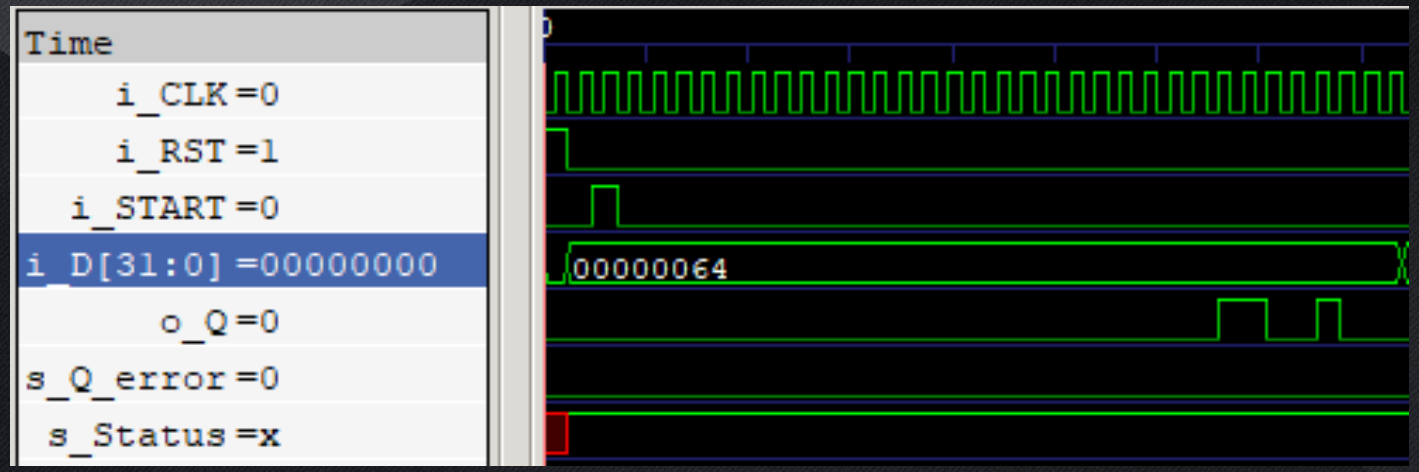
Adder Waveform Results

2x1 Standard Cell MUX Waveform Results

Testing Thus Far – Backdoor SPI, Shift Out Register

```
for(integer i = DATA_WIDTH - 1; i >= 0; i = i - 1)
begin
    if(o_Q != data_in[i]) begin //if enable set
        s_Q_error = 1'b1;
        s_Status = 2'b0;
        repeat(1) @(negedge i_CLK);
        s_Q_error = 1'b0;
    end
    repeat(1) @(negedge i_CLK);
end
```

Shift Out Register Testbench Verification



Shift Out Register Waveform Results

Future Implementation

Cade Breeding

Milestones
Schedule
Plan for the Future

Milestones and Schedule

Task	Deadline	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
Install the open-source tools and simulate sample code	2/19/2023												
Define our project specifications	3/19/2023												
Draw out a top-level diagram of the user area including each individual module	3/26/2023												
Draw out detailed implementation of each module	4/2/2023												
Write initial Verilog implementation of each module	5/14/2023												
Test and iterate using RTL simulations	8/20/2023												
Join modules together and verify final design as they are completed	8/27/2023												
Test and iterate using RTL simulations	9/22/2023												
Verify using gate-level simulation	9/22/2023												
Verify submission using the provided verification tools	10/1/2023												
Submit to MPW Shuttle	10/8/2023												
Create Software to run on embedded microcontroller	10/22/2023												
Create Documentation and bring up plan to test returned project in the future	11/3/2023												

Next Steps

- **RTL Simulation**
- **Integration of Modules**
  - RTL Simulation
  - Gate Level Simulation
- **Verification**
- **Submission**
- **Bring Up Plan**

Test and iterate using RTL simulations	8/20/2023
Create thorough tests that will cover main functionality of module	
Create tests that verify module satisfies top level constraints of the module	
Create tests to cover edge cases outside of typical operating state	
Join modules together and verify final design as they are completed	8/27/2023
Review Verilog modules designed by each team member	
Review Verilog testbenches designed by each team member	
Test and iterate using RTL simulations	9/22/2023
Create timing tests to ensure that all individual modules satisfy the timing requirements of the system	
Create main functionality tests that verify each module gives correct results for main desired task	
Create tests to verify functional interactions between modules	
Verify using gate-level simulation	9/22/2023
Create tests to ensure modules operate with the same functionality as our RTL simulations	
Verify submission using the provided verification tools	10/1/2023
Ensure that final project passes Efabless' precheck tests	
Submit to MPW Shuttle	10/8/2023
Create public repository to satisfy Efabless' open-source requirement	
Create a project on Efabless' website and point at our public repository	
Submit the design to the time applicable OpenMPW shuttle	
Create Software to run on embedded microcontroller	10/22/2023
Create a repository of tested sample code which will verify the integrity of our system	
Create Documentation and bring up plan to test returned project in the future	11/3/2023
Document the bring-up plan in a detailed form for a future user	

Conclusion

Cade Breeding

Current Status
Contributions

Current Status

Wishbone Test
Gregory Ling

Verilog Started

Clock Gating
Cade Breeding

Verilog Started

Backdoor SPI
Jake Hafele

In Testing

Custom Cell Test
Gregory Ling

Exploration

Standard Cell Test
Cade Breeding

Verilog Started

VRNI (DSP)
Will Galles

Verilog Starting

References

[1] https://www.pngall.com/microcontroller-png/
[2] https://efabless.com/open_shuttle_program
[3] https://www.zerotoasiccourse.com/post/mpw1-bringup/
[4] https://efabless.com/open_shuttle_program

Questions?