RTL DESIGN – SERDES

6 MONTHS INTERN AT SYNOPSYS HYDERABAD

DONE BY: P R MADHU VARSHAN (108121067)

Ramakrishna.U

SIGNATURE OF MANAGER

RAMAKRISHNA UPPALAPATI

SIGNATURE OF GUIDE

DR. B. CHANDRABABU NAIK

INTRODUCTION:

Overview:

- The project focuses on developing a pattern generator and pattern matcher to test the functionality and performance of circuits. The pattern generator operates in four distinct modes, each serving different testing purposes.

Importance:

- Pattern Generators: Essential for creating test patterns that simulate real-world scenarios, helping in the verification and validation of circuit designs.
- Pattern Matchers: Crucial for comparing the generated patterns with expected outcomes to ensure the circuit operates correctly.

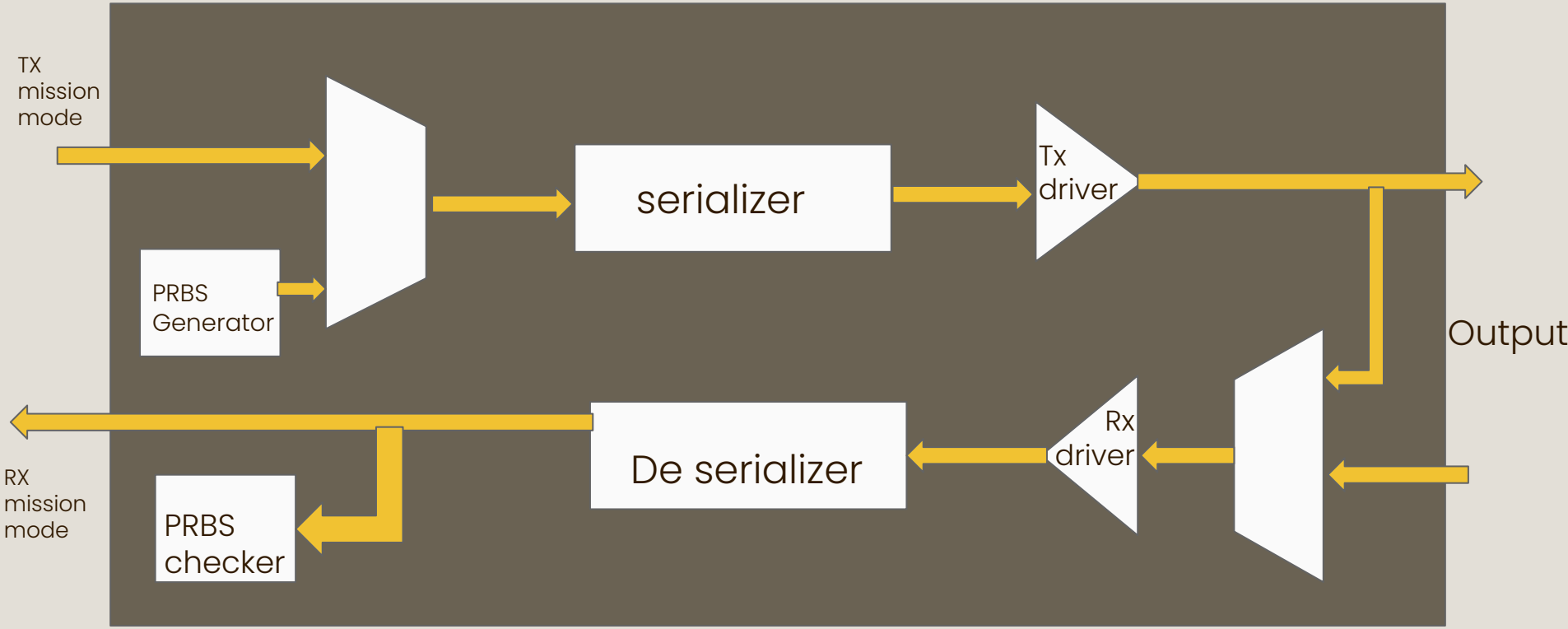
Tools and Technologies:

- Spyglass: Used to check correctness of RTL Code in terms of Syntaxes and CDC.
- Verdi: Used to check the functionality .
- Linux: Operating system environment for development.

OBJECTIVE:

- To develop a pattern generator with multiple modes of operation.
- To implement a pattern matcher for circuit testing error count.
- To use Spyglass and Verdi for debugging, functionality, and performance checks.
- To develop a synchronizer for capturing signals in different clock domains.
- To optimize power and clock gating in the design.

Literature Review:



Proposed Methodology

Pattern Generator:

- Uses 4 modes of operation.
- Mode 1: Different types of linear feedback shift registers (LFSRs) with polynomials (7, 9, 11, 13, 15, 16, 23, 31).
- Mode 2: PAT0 for user-defined transmission with DC balanced word.
- Mode 3: DC balanced word conversion to 20-bit code.
- Mode 4: Extension mode (000, PAT0, FFF, ~PAT0).

Pattern Matcher:

- Contains a pattern generator circuit inside to validate the signals generated by Pattern Generator.

Synchronizer:

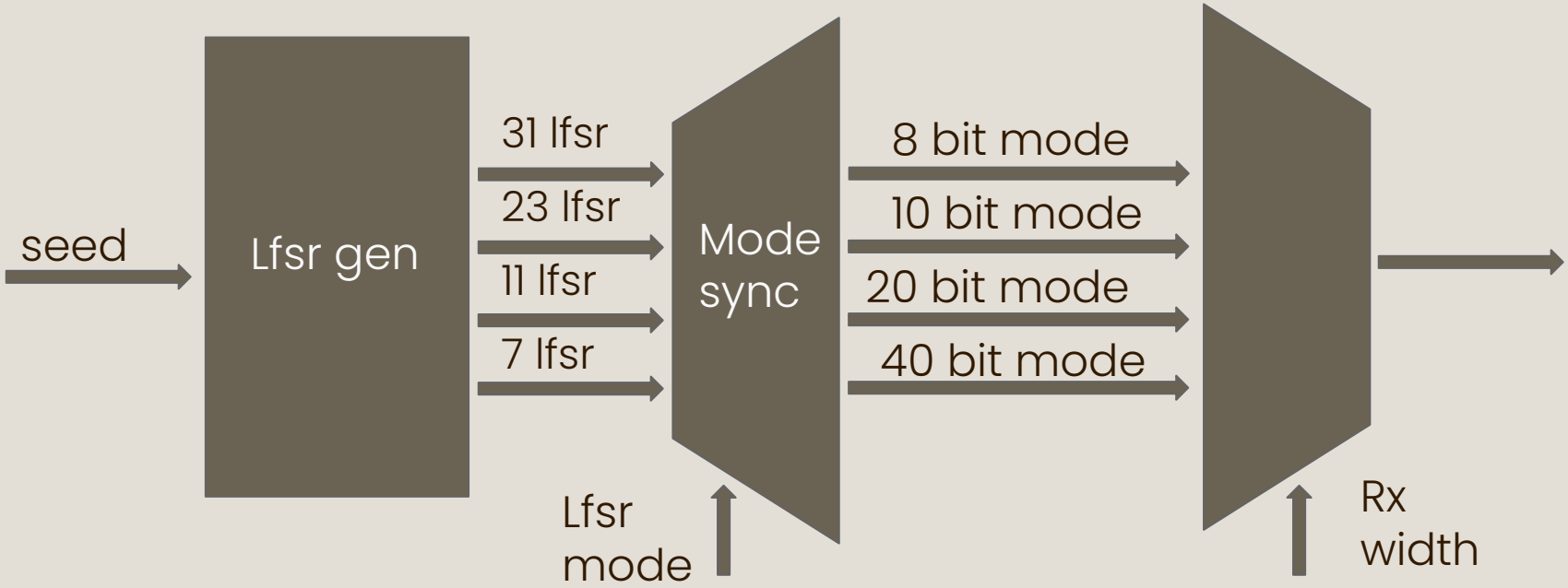
- To fix the clock Domain crossings in cases where Data will propagate from one clock domain to another.

Power Gating Model:

- Optimizes power and clock gating.

Work Done So Far

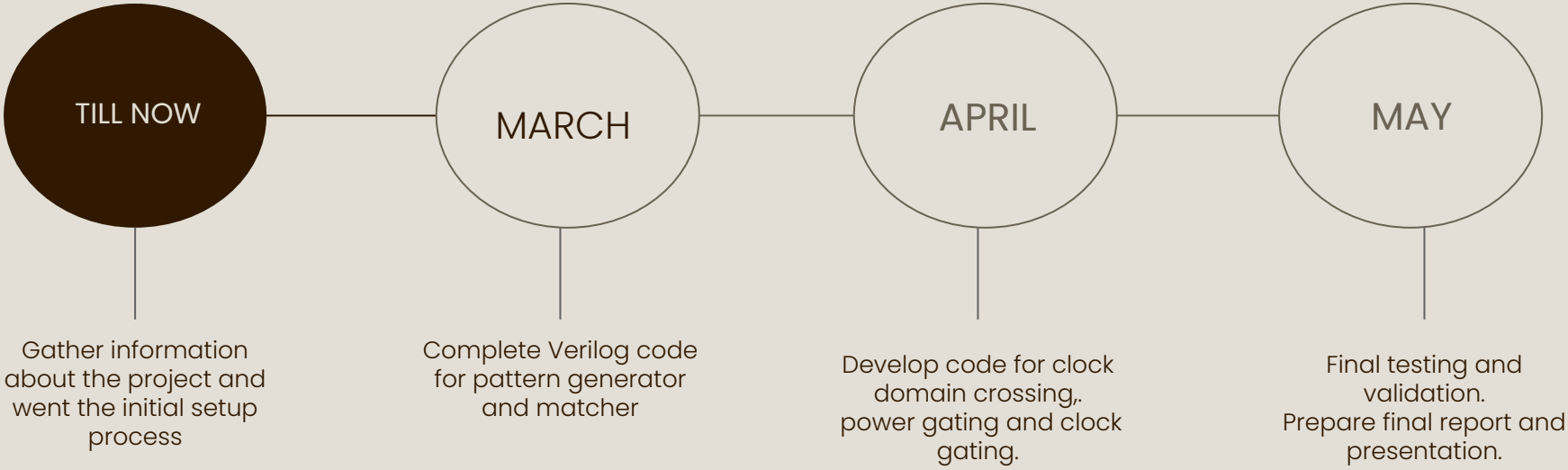
Gained necessary knowledge to start with Verilog code in Linux.
Studied relevant materials from company portals.



Future Works

- Implement Verilog code for pattern generator and matcher.
- Test and debug the design using Spyglass and Verdi.
- Optimize power and clock gating

Timeline



References

- Synopsys recorded videos
- Datasheets
- P. Moorthy and S. S. Bharathy, "An efficient test pattern generator for high fault coverage in built-in-self-test applications," 2013
- B. Moryani and D. K. Mishra, "Low power test pattern generator with modified clock for BIST," 2017

THANK YOU