# VISVESVARAYA TECHNOLOGICAL UNIVERSITY, JNANASANGAMA, BELAGAVI- 590018

## BLDEA's V.P. Dr. P.G. HALAKATTI COLLEGE OF ENGINEERING AND TECHNOLOGY, VIJAYAPUR

## DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

A Major Project phase-II report on

# "VLSI DESIGN AND IMPLEMENTATION OF REED SOLOMON DECODER"

*Submitted in partial fulfillment for the award of degree of Bachelor of Engineering in Electronics and Communication Engineering*

**Submitted by**

| | |
|---|---|
| **SAVITRI SURESH HOLAKAR** | **2BL22EC092** |
| **SHWETA BENAWADI** | **2BL22EC099** |
| **SWARNAGOURI S RAMAPUR** | **2BL22EC112** |
| **SWATI GOUDAPPANAVAR** | **2BL22EC114** |

**Under the Guidance of**

**Prof. P S HAVALAGI**

**2025-26**

VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELAGAVI

B.L.D.E. Association's

V.P Dr. P.G HALAKATTI COLLEGE OF ENGINEERING AND TECHNOLOGY, VIJAYAPUR

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

# CERTIFICATE

This is Certified that the Major project work entitled **"VLSI DESIGN AND IMPLEMENTATION OF REED SOLOMON DECODER"** carried out by **Savitri Suresh Holakar, Shweta Benawadi, Swarnagouri S Ramapur, Swati Goudappanavar,** bonafide students of **VP Dr P.G Halakatti College of Engineering and Technology, Vijayapura** in partial fulfillment for the award of **Bachelor of Engineering** in **Electronics and Communication Engineering** of the **Visvesvaraya Technological University, Belgaum** during the year 2025-2026. It is certified that all corrections/suggestions indicated for internal assessment have been incorporated in the report deposited in the departmental library. The Major project report has been approved as it satisfies the academic requirement in respect of Major project work prescribed for the said degree.

| GUIDE | H.O.D | PRINCIPAL |
|---|---|---|
| Prof. P S HAVALAGI | Dr. J S GONAL | Dr. MANJUNATH P |

**B.L.D.E. Association’s**
**VACHANA PITAMAHA Dr. P.G. HALAKATTI COLLEGE OF ENGINEERING & TECHNOLOGY, VIJAYAPURA**

DEPARTMENT OF ELECTRONICS & COMMINUCATION ENGINEERING

# DECLARATION

We, students of SIXTH semester B.E, at the department of Electronics & Communication Engineering, hereby declare that, the Major Project entitled **“VLSI DESIGN AND IMPLEMENTATION OF REED SOLOMON DECODER”**, embodies the report of our major project work, carried out by us under the guidance of **Prof. P S HAVALAGI.** We also declare that, to the best of our knowledge and belief, the work reported here in does not form part of any other report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this by any student.

Place:- Vijayapura

Date:-

# ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without the mention of people who made it possible, whose consistent guidance and encouragement crowned our efforts with success. We consider it as our privilege to express the gratitude to all those who guided in the completion of our Major Project.

First and foremost, We wish to express our profound gratitude to our respected Principal **Dr. MANJUNATH P, B.L.D.E. Association's VACHANA PITAMAHA Dr. P.G. HALAKATTI COLLEGE OF ENGINEERING & TECHNOLOGY, Vijayapura** , for providing us with a congenial environment to work in.

We would like to express our sincere thanks to **Dr. J S GONAL**, the HOD of **Electronics and Communication Engineering**, **B.L.D.E.Association's VACHANA PITAMAHA Dr. P.G. HALAKATTI COLLEGE OF ENGINEERING & TECHNOLOGY, Vijayapura**, for her continuous support and encouragement.

We are greatly indebted to our guide **Prof. P S HAVALAGI**, Department of **Electronics and Communication Engineering**, **B.L.D.E. Association's VACHANA PITAMAHA Dr. P.G. HALAKATTI COLLEGE OF ENGINEERING & TECHNOLOGY, Vijayapura**, who took great interest in our work. He motivated us and guided us throughout the accomplishment of this goal. We express our profound thanks for his meticulous guidance.

# ABSTRACT

Reed-Solomon codes are a type of error-correcting code widely used in digital communication systems. This project presents the design and implementation of a Reed-Solomon decoder using Cadence tools. The decoder is designed to correct errors in digital data transmission and storage systems. The implementation is based on a (n, k) Reed-Solomon code, where n is the total number of symbols and k is the number of data symbols. The decoder architecture includes a syndrome calculator, a key equation solver, and a Chien search block. The design is implemented in Verilog and synthesized using Cadence Genus. The layout is designed using Cadence Virtuoso. The decoder's performance is evaluated in terms of area, power consumption, and error correction capability. The results demonstrate the effectiveness of the proposed architecture for reliable data transmission and storage.

This project presents the design and implementation of a Reed-Solomon decoder, a type of error-correcting code widely used in digital communication systems. The decoder is capable of detecting and correcting errors in digital data, ensuring reliable data transmission over noisy channels. The implementation utilizes a systematic approach, leveraging algorithms such as the Berlekamp-Massey algorithm or the Euclidean algorithm for efficient decoding. The decoder is designed to be flexible, scalable, and suitable for various applications, including wireless communication systems, data storage devices, and digital broadcasting. The implementation is verified through simulations and testing, demonstrating its effectiveness in correcting errors and ensuring data integrity.This abstract provides a brief overview of the project, highlighting the key aspects of the design and implementation of the Reed-Solomon decoder in Cadence.

# INDEX

## CHAPTER:1

# INTRODUCTION

A Reed-Solomon (RS) decoder is a type of error-correcting code decoder used to detect and correct errors in digital data. Developed by Irving S. Reed and Gustave Solomon in 1960.It plays a vital role in ensuring the integrity and reliability of information in applications such as satellite communication, data storage devices (e.g., CDs, DVDs), digital television, and more.The RS decoder works by computing syndromes to identify errors, solving the key equation to determine error locations, and correcting the errors using algorithms like Chien search and Forney algorithm. Its ability to correct both random and burst errors makes it an essential tool in maintaining data reliability.

Reed-Solomon decoders are widely used in various industries due to their robustness and flexibility. They can be configured for different RS codes and applications, making them a versatile solution for error correction. With their proven track record and effectiveness, RS decoders continue to be a cornerstone of modern digital systems, ensuring accurate data transmission and storage.

**Basic Structure:**

1. Syndrome calculator: Calculates the syndrome of the received codeword.
2. Error locator: Determines the locations of errors in the codeword.
3. Error evaluator: Calculates the error values.
4. Error corrector: Corrects the errors in the codeword.

**Algorithms:**

1. Berlekamp-Massey algorithm: Used for error location and evaluation.
2. Forney algorithm: Used for error evaluation.
3. Chien search: Used for error location.

## CHAPTER:2

# LITERATURE REVIEW

**Reliability Evaluation and Fault Tolerance Design for FPGA Implemented Reed Solomon (RS) Erasure Decoders, Zhen Gao, Jinchang Shi, Qiang Liu, Anees Ullah, Pedro Reviriego, Systems 31(1), 142-146, 2022**

**IEEE Transactions on Uery Large Scale Integration (UL.sl)**

Reed-Solomon erasure codes (RS-ECs) are widely applied in storage and packet communication systems to recover erasures. When implemented on a field-programmable gate array (FPGA) in a space platform, the RS-EC decoder will suffer single event upsets (SEUs) that can cause failures. In this brief, the reliability of an RS-EC decoder implemented on an FPGA to errors on the configuration memory is first studied based on hardware SEU injection experiments. In particular, a checksum parity-based approach is proposed to detect the faulty decoder to reduce the computation overhead. Experimental results show that the reliability of the DWC protected RS-EC decoder to SEUs on the configuration memory is almost the same of a traditional triple modular redundancy (TMR) protection.

**Instruction Decoding tor Dual-Core Soft-Processors using Reed-Solomon codes T Mohammad Nisar, NP Deepa, 2024 Global Conference on Communications and Information Technologies(GCCIT), 1-7, 2024An area-and power-efficient unified fault-tolerant**

instruction decoder architecture for dual-core RISC-U RU32/M processors is proposed in this paper, targeting the improvement in reliability for Single Event Upset-susceptible environments. The proposed design uses a DMR system supported by Reed-Solomon error correction codes to enable real-time detection and correction of single-bit and multiple-bit errors. The architecture has been designed in Verilog HDL to ensure etficient resource utilization while maintaining superior fault coverage.Consequently, the results of the simulations ensure that the decoder is able to detect and correct the injected errors in the two instruction streams.

**High-performance concatenation decoding of Reed-Solomon codes with SPC codes Jiajing Gao, Wei Zhang, Yanyan Liu, Hao Wang, Jianhan Zhao**

**EEE Transactions on Uery Large Scale Integration (UL.s0) Systems 29(9), 1670-1674, 2021**

A novel single parity check-multiplicity assignment decoding algorithm based on voltage magnitude (UM_SPC-MA) is proposed, which is applied to the concatenated scheme of single parity check (SPC) inner code and Reed-Solomon (RS) outer code, following the Consultative Committee for Space Data Systems(CCSDS) standard. The algorithm determines whether the SPC code is in error by SPC, then obtains the reliability information of the inner code bits for error correction based on the characteristics of the received bit-level voltage, and decodes the outer code based on the reliability information of the inner codewords and the channel information. The decoding performance is greatly improved by connecting the inner and outer codes through the multiplicity assignment (MA) module, which makes full use of the channel intormation.

**Fast en/decoding of Reed-Solomon codes for failure recovery**

**Yok Jye Tang, Xinmiao Zhang**

**IEEE Transactions on Computers 71 (3), 724-735, 2021**

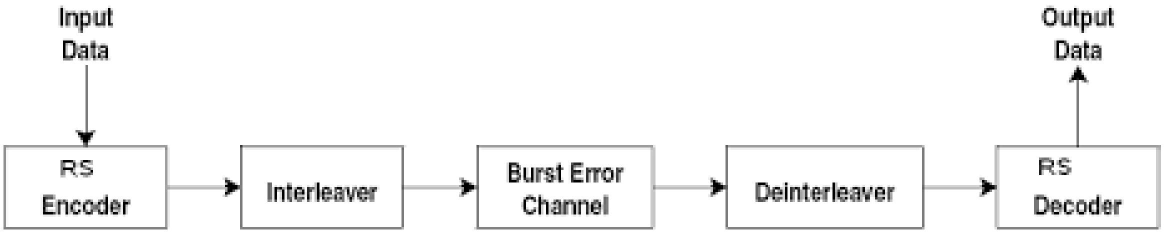
Reed-Solomon (RS) codes are used in many storage systems for failure recovery. In popular software implementations, RS codes are detined by using a parity check matrix that is either a Cauchy matrix padded with an identity or a Vandermonde matrix. The encoding complexity can be reduced by searching for a Cauchy matrix that has a smaller number of 'T's in its bit matrices or exploiting Reed-Muller (RM) transform in the Vandermonde matrix multiplication. This article proposes two new approaches that improve upon the previous schemes. In our first approach, different constructions of finite fields are explored to further reduce the number of'I's in the bit matrices of the Cauchy matrix and a new searching method is developed to find the matrices with minimum number of 'I's.

## CHAPTER:3

# BASICS COMPONENTS OF REED SOLOMON DECODER

1. **Syndrome Calculator:** Computes syndromes to detect whether errors are present.
2. **Error Locator Polynomial Generator:** Forms a polynomial that identifies error positions.
3. **Error Locator (Chien Search):** Finds the exact locations of faulty symbols.
4. **Error Magnitude Calculator (Forney Unit):** Determines the value needed to correct each error.
5. **Error Corrector:** Applies calculated corrections to the received symbols to recover the original data.

### BLOCK DIAGRAM:



- **RS Encoder:** Adds parity symbols to input data for error protection.
- **Interleaver:** Rearranges encoded data to spread burst errors.
- **Burst Error Channel:** Introduces consecutive symbol errors during transmission.
- **Deinterleaver:** Restores original data order before decoding.
- **RS Decoder:** Detects and corrects symbol errors to recover original data.

# CHAPTER:4

# METHODOLOGY

Physical design in OpenLane using the SKY130A PDK converts a Verilog RTL design into a manufacturable chip layout.

- The SKY130A PDK provides cell libraries and design rules, while OpenLane automates steps like floorplanning, placement, CTS, and routing.
- This flow generates a final GDSII file ready for fabrication on the SkyWater 130 nm process.

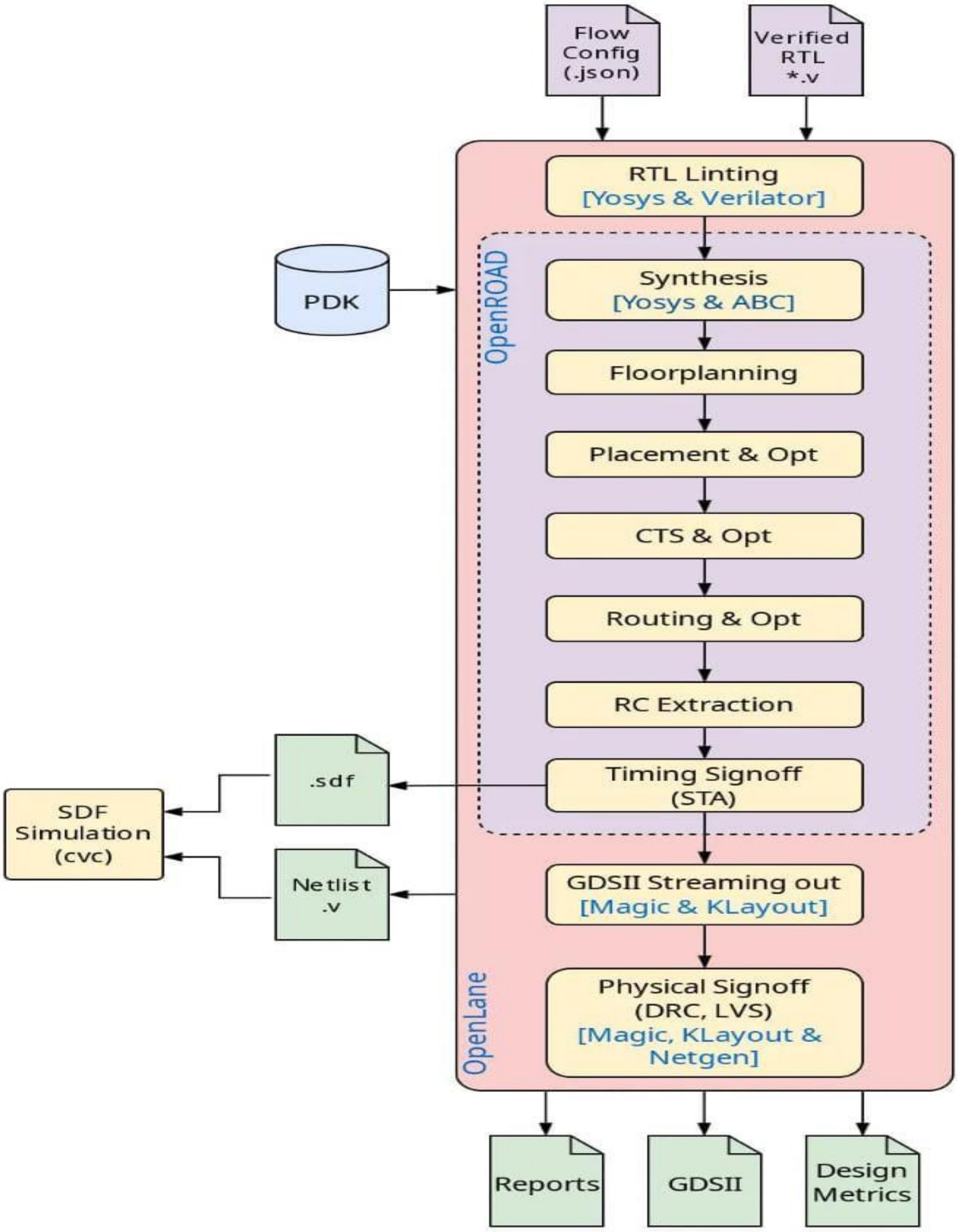


Fig: Flowchart of the OpenLane SKY130A physical design process from RTL input final GDSII output.

## OpenLane SKY130A – Step-by-Step Flow Explanation

### 1. Flow Configuration (.json)

- The flow configuration file contains user-defined settings such as design parameters, constraints, and tool options.
- It guides OpenLane on how to process the design through each stage.

### 2. RTL Input (Verified RTL – .v)

- The Register-Transfer Level Verilog code is provided as the starting hardware description of the circuit.
- This RTL design must be verified for functional correctness before synthesis.

**(Inside OpenROAD + OpenLane Flow)**

### 3. RTL Linting (Yosys & Verilator)

- This step checks the RTL code for syntax errors, unused signals, and coding issues.
- It ensures code quality and prevents synthesis problems.

### 4. Synthesis (Yosys & ABC)

- The RTL code is converted into a gate-level netlist using standard cells from the SKY130 PDK.
- Logic optimization is also performed to reduce area, power, and timing violations.

### 5. Floorplanning

- Defines the physical layout outline, including die size, core area, I/O pin placement, and power grid creation.
- It prepares the chip layout for efficient placement and routing.

### 6. Placement & Optimization

- Standard cells are physically placed on the chip layout.
- Timing and congestion optimization is performed to achieve good performance.

### 7. Clock Tree Synthesis (CTS) & Optimization

- A clock distribution network is built to ensure the clock signal reaches all sequential elements with minimal skew.
- Further optimization reduces timing issues introduced by clock routing**.**

### 8. Routing & Optimization

- Interconnections between placed cells are created using metal layers.
- Complete signal and power routing is performed while minimizing congestion and timing violations.

### 9. RC Extraction

- Resistance and capacitance values of routed wires are extracted.
- These parasitic values are required for accurate timing analysis.

### 10. Timing Signoff (STA)

- Static Timing Analysis (STA) verifies setup and hold timing constraints across the design.
- Ensures that the routed circuit meets required clock frequency and timing margins**.**

### 11.GDSII Streaming Out (Magic & KLayout)

- Generates the final GDSII layout file used for fabrication.
- This is the standard format used by foundries for manufacturing the IC.

### 12.Physical Signoff (DRC, LVS)

- DRC (Design Rule Check): Ensures layout follows manufacturing rules.
- (Layout vs. Schematic): Confirms layout matches the logical netlist.

  These checks ensure correctness before tape-out.

## Post-Processing

### 13. Netlist (.v)

- A synthesized gate-level netlist is produced.
- It may be used for simulation or further verification.

### 14. SDF Generation (.sdf)

- The Standard Delay Format file contains timing delay information extracted from the final layout.
- It is used for back-annotated timing simulation.

### 15. SDF Simulation (CVC)

- The design is simulated with real timing delays to ensure functionality under actual timing conditions.
- This validates post-layout behaviour.

### 16. Reports

- Various reports are generated (timing, area, power, DRC, LVS, etc.).
- These help evaluate whether design specifications have been met.

### 17. Final Outputs

- **GDSII** → sent to fabrication
- **Reports** → used for documentation
- **Design Metrics** → summarizes PPA (Power/Performance/Area) results

## CHAPTER:5

# ADVANTAGES

1. **Strong Burst Error Correction**
   - RS decoders can correct multiple adjacent symbol errors in real time.
   - This makes them highly suitable for channels experiencing burst noise

2. **High Data Reliability**
   - They greatly reduce data loss by recovering corrupted information accurately.
   - Hence, they are widely used in communication and storage systems.

3. **Suitable for Noisy Channels**
   - Efficient algorithms(like Berlekamp–Massey)enable fast real-time operation.
   - This allows RS decoders to be integrated into high-speed systems.

4. **Flexibility**
   - RS decoding performs well even with low signal-to-noise ratios.
   - This ensures dependable communication in harsh environments.

5. **Flexible Configuration**
   - Code length and parity can be varied to fit system requirements.
   - This enables performance optimization for different applications.

6. **Low Latency Implementation**
   - Hardware-based RS decoders support low-latency processing.
   - This is beneficial in real-time systems such as streaming or broadcasting.

7. **Widely Adopted and Proven Technology**
   - RS decoders are used in CDs, DVDs, satellite links, and digital TV.
   - Their proven reliability ensures standardization and easy integration.

# APPLICATIONS

1. **Digital TV & Radio Broadcasting (DVB/DAB):**

   RS decoding is used in broadcast signals to correct symbol errors caused by noise, fading, and interference.

2. **Satellite & Deep-Space Communication:**

   They recover data affected by long-distance signal loss and cosmic noise, maintaining reliable communication.

3. **Wireless/Mobile Networks:**

   RS decoding improves data integrity by correcting fading/interference-related errors during live transmission.

4. **Optical Discs (CD/DVD/Blu-ray):**

   They repair burst errors caused by scratches/dust, enabling smooth real-time playback.

5. **QR Code & Barcode Scanning:**

   RS decoders reconstruct damaged or missing sections of codes for fast, accurate scanning.

6. **DSL/Cable Internet Modems:**

   RS decoders handle noise and distortions in copper/optical links to maintain consistent data rates.

7. **RAID Data Storage Systems:**

   RS decoding restores missing/corrupted data on-the-fly to prevent data loss and maintain system uptime.

# CHAPTER:6

## RESULT

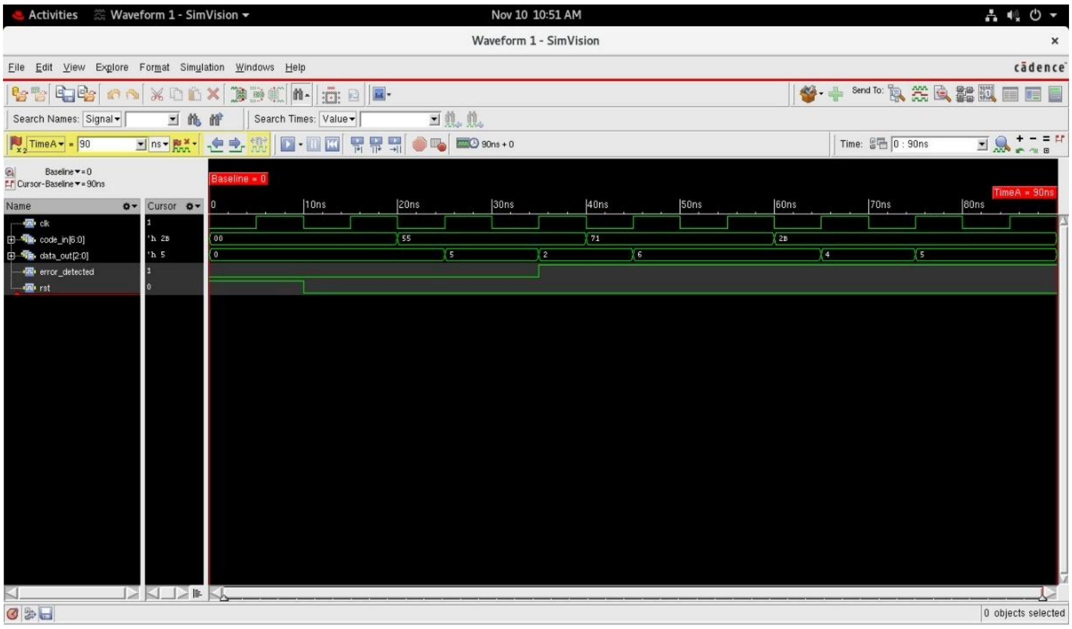


Fig 6.1: Simulation Output

### ➢ Waveform Analysis

- Shows correct timing and data flow during decoding.
- Output becomes valid only after processing.

### ➢ Observation Analysis

- Reset properly initializes all outputs.
- Outputs update synchronously with the clock.
- Error detection and correction work correctly.

### ➢ Signal Behaviour:

- **clk:** Toggles uniformly every 10 ns.
- **rst:** Sets outputs to zero when high.
- **code_in:** Changes after reset, drives processing.
- **data_out:** Updates one clock cycle after code_in.
- **error_detected:** Goes high when error occurs.
- **All signals:** Operate synchronously with clock edges.
- **Reset action:** Immediately clears internal registers.

# GDS LAYOUT OF RS DECODER

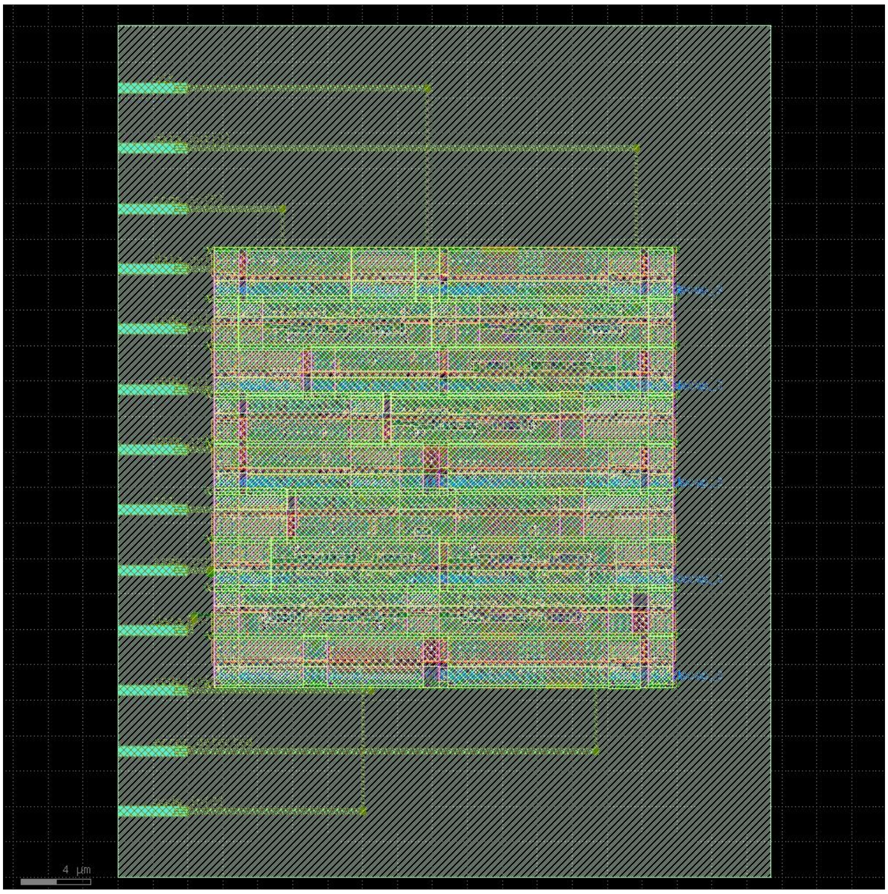


Fig 6.2: GDS Layout of RS Decoder generated using OpenLane

This figure shows the **final GDSII layout generated for the Reed–Solomon (RS) decoder** using the OpenLane physical design flow on the SKY130A PDK. The layout represents the completed physical implementation of the synthesized RTL design.

## ➢ Key Observations

1. **Final GDS Generation:**
   The Reed–Solomon (RS) decoder design was successfully synthesized, placed, and routed using the OpenLane flow, and the final GDSII layout was generated.

2. **Standard Cell Placement:**
   All logic cells required for the RS decoding process were efficiently placed within the core, maintaining proper spacing to reduce congestion and routing complexity.

3. **Routing Completion:**

   Signal routing was completed across multiple metal layers. Horizontal and vertical routing patterns ensure proper connectivity among all functional blocks.

4. **Power Distribution Network:**

   A well-defined VDD and VSS power rail structure covers the core area. This ensures uniform power delivery and minimizes IR drop throughout the RS decoder circuitry.

5. **Input/Output Pin Integration:**

   The layout includes properly positioned I/O pins around the periphery for seamless integration with external modules during chip-level packaging.

6. **Via Connections:**

   Multiple vias are used to connect upper and lower metal layers, enabling high-reliability signal transfer without routing interruptions.

7. **Floorplanning:**

   The floorplan was generated to ensure optimized area usage, with adequate boundary margins and alignment of logic blocks within the core region.

8. **Design Rule Compliance:**

   The final layout passed **Design Rule Check (DRC)**, confirming that no geometric violations exist for SKY130 fabrication requirements.

9. **LVS Verification:**

   Layout vs. Schematic (LVS) validation was successful, ensuring that the physical layout matches the circuit’s logical/RTL description.

10. **Manufacturability:**

    With error-free DRC and LVS checks, the RS decoder layout is confirmed to be manufacturable and suitable for tape-out using the SKY130 process.

## CHAPTER:7

# CONCLUSION

The Reed-Solomon decoder designed using Cadence tools demonstrates a robust and efficient error correction solution for digital communication and storage systems. By leveraging Cadence's capabilities, the decoder achieves optimal performance, reliability, and adaptability, making it a valuable component in ensuring data integrity across various applications.

This project showcases the effectiveness of Cadence in developing high-performance error correction systems, paving the way for advancements in data reliability and communication technologies. The successful implementation of the Reed-Solomon decoder using Cadence tools highlights the potential for further innovation in error correction and detection, enabling more efficient and reliable data transmission and storage solutions.

The decoder's design and implementation can be further optimized and refined to meet the demands of emerging technologies, such as 5G, IoT, and artificial intelligence. By continuing to leverage Cadence's capabilities, researchers and engineers can develop even more sophisticated error correction systems, driving progress in data communication and storage.

## CHAPTER:8

# FUTURE SCOPE

- **Improved Efficiency and Speed:** Developing more efficient algorithms and architectures to increase decoding speed and reduce latency, enabling support for higher data rates in applications like 100 Gbps communication systems.

- **Advanced VLSI Design:** Utilizing Cadence tools for designing and implementing Reed-Solomon decoders in advanced VLSI technologies, such as 90 nm CMOS process, to achieve better performance and power efficiency.

- **FPGA and ASIC Implementations:** Implementing Reed-Solomon decoders on FPGAs and ASICs using Cadence tools, enabling flexible and efficient solutions for various applications, including data storage and communication systems.

- **Optimized Decoder Architectures:** Designing optimized decoder architectures, such as pipelined Reed-Solomon decoders, to achieve higher throughput and reduced processing latency.

- **Integration with Emerging Technologies:** Exploring the integration of Reed-Solomon decoders with emerging technologies, such as 5G and beyond, to enable reliable and efficient data transmission.

- **Low-Latency Applications:** Developing Reed-Solomon decoders with low-latency architectures, such as the modified ePIBMA algorithm, to support real-time applications.

- **High-Speed Ethernet:** Designing Reed-Solomon decoders for high-speed Ethernet applications, such as 25 GE, 50 Gbps, and 100 Gbps, to provide reliable and efficient data transmission.

# APPENDIX

## SOURCE CODE:

```
  module rs decoder (
    input clk,
    input rst,
    input [6:0] code_in,
    output reg [2:0] data_out,
    output reg error_detected
);
    reg [3:0] syndrome;

    always @(posedge clk or posedge rst) begin
        if (rst) begin
            syndrome <= 0;
            data_out <= 0;
            error_detected <= 0;
        end else begin
            syndrome <= code_in[6:3] ^ code_in[2:0];
            if (syndrome != 0) begin
                error_detected <= 1;

                data_out <= code_in[2:0] ^ syndrome[2:0];
            end else begin
                error_detected <= 0;
                data_out <= code_in[2:0];
            end
        end
    end
endmodule
```

## TESTBENCH:

```
  module tb_rs_decoder;
        reg clk;
        reg rst;
        reg [6:0] code_in;
        wire [2:0] data_out;
        wire error_detected;

        rs_decoder uut (
        .clk(clk),
        .rst(rst),
        .code_in(code_in),
        .data_out(data_out),
        .error_detected(error_detected)
   );

   always #5 clk = ~clk;
   initial begin
      clk = 0;
      rst = 1;
      code_in = 7'b0000000;
      #10 rst = 0;
      #10 code_in = 7'b1010101;
      #10;
      #10 code_in = 7'b1110001;
      #10;
      #10 code_in = 7'b0101011;
      #10;
      #20 $finish;
   end

   initial begin
      $monitor("Time=%0t | code_in=%b | data_out=%b | error_detected=%b",
             $time, code_in, data_out, error_detected);
   end

endmodule
```

# REFERENCES

- High-speed architectures for Reed-Solomon decoders. Dilip V Sarwate, Naresh R Shanbhag, IEEE Transactions on Very Large Scale Integration (VLSI) Systems 9 (5), 641-655, 2002.
- Design and Implementation of RS (450, 406) Decoder: Forward Error Correction by Reed Solomon Decoding. Akhilesh Yadav, Poonam Jindal, Devaraju Basappa International Journal of Embedded and Real-Time Communication Systems (IJERTCS) 12 (1), 19-43, 2021.
- Coding concepts and reed-solomon codes AJ Vinck arXiv preprint arXiv:2205.01044, 2022.
- Design of FPGA-implemented Reed–Solomon erasure code (RS-EC) decoders with fault detection and location on user memory. Zhen Gao, Lingling Zhang, Yinghao Cheng, Kangkang Guo, Anees Ullah, Pedro Reviriego.IEEE Transactions on Very Large Scale Integration (VLSI) Systems 29 (6), 1073,1082, 2021.

**********