SPI-SAR-ADC Controller: Digital Design, Verification and Implementation

by Saumya Raj Singh

Design and Architecture

System-Level Overview & Modular Breakdown

The architecture is partitioned into a top-level spi_adc wrapper and a dedicated ADC controller core to separate host communication from signal acquisition. The SPI module operates as a slave using a standard 4-wire bus (CS, SCK, MOSI, MISO) to manage the internal register map, which handles control, status, and data storage. The ADC controller utilizes a main Finite State Machine (FSM) to drive the Successive Approximation Register (SAR) datapath, coordinating the external DAC and comparator to resolve the 12-bit digital output. To optimize energy consumption, the clock generation block includes a prescaler and clock gating logic, ensuring the high-speed ADC clock is active only during conversion phases.
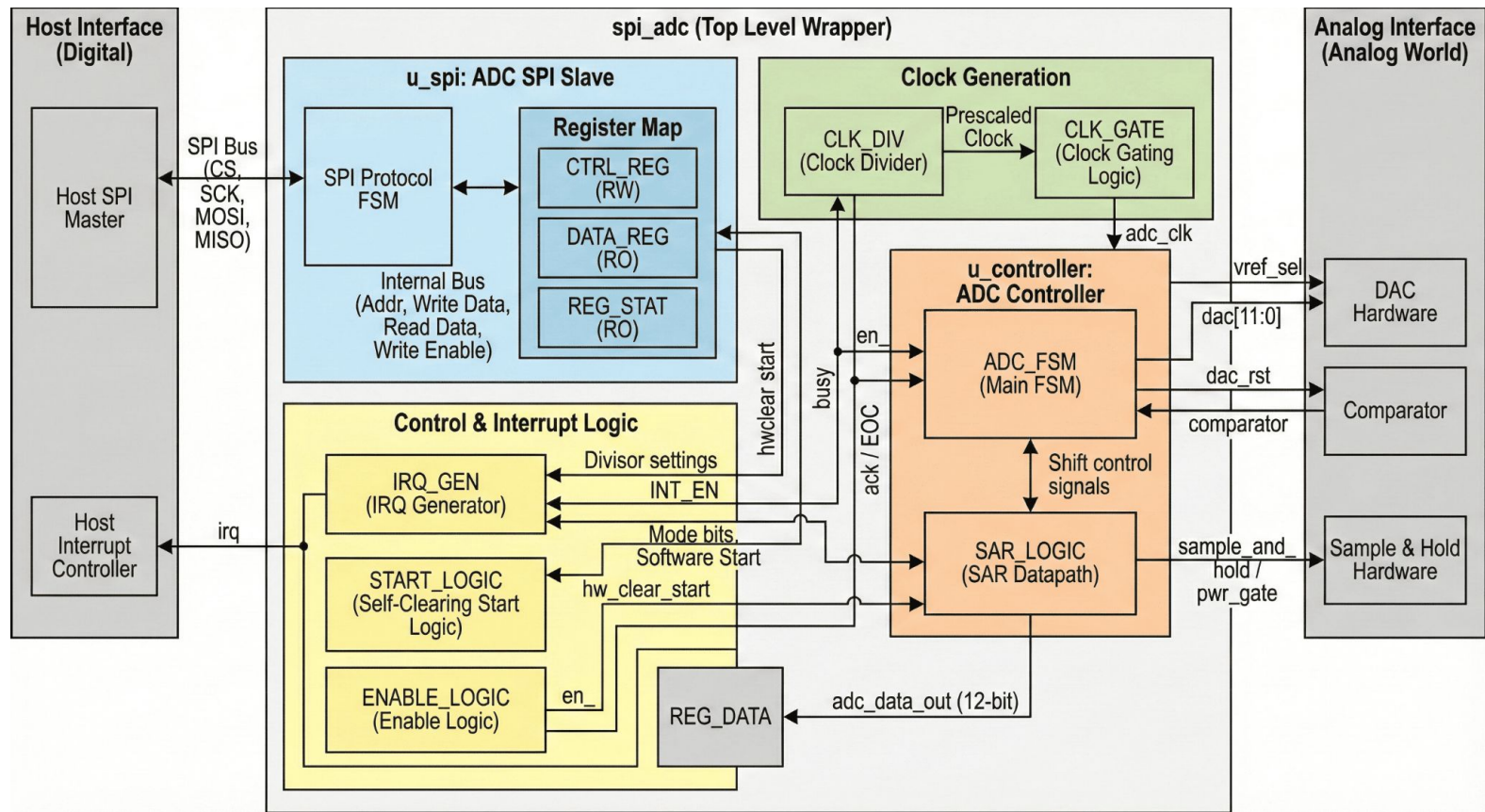
SYSTEM BLOCK DIAGRAM
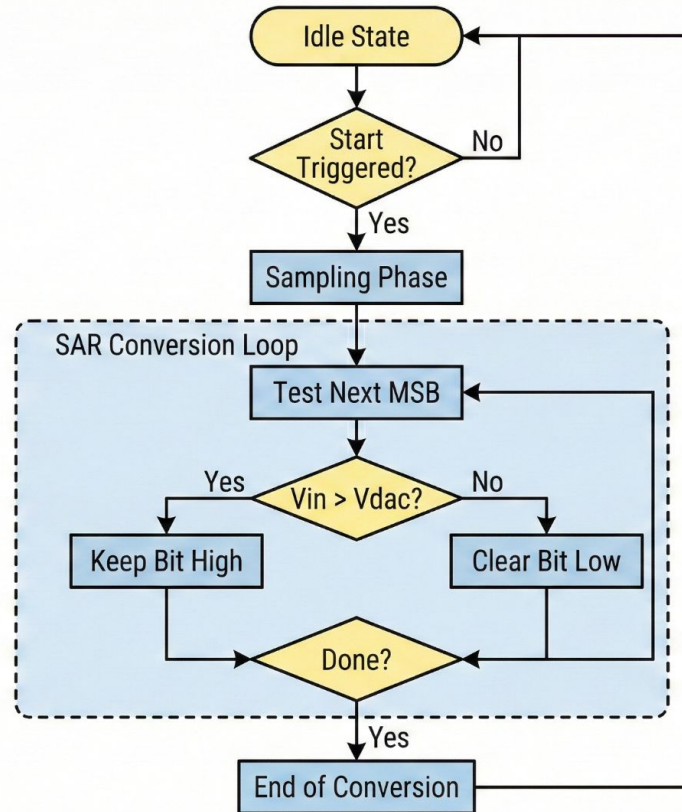
ADC CONTROLLER STATE DIAGRAM : SAR CONVERSION ALGORITHM

SPI Interface (Type 0)

SPI Register Summary

Address	Register Name	R/W	Description
0b00	CTRL_REG	R/W	Main control register.
0b01	STATUS_REG	R	Combined status flags and device info.
0b10	DATA_REG	R	Holds the 12-bit conversion result.
0b11	INFO_REG	R	Stores the offset from the calibration.

0b00 CTRL_REG (ADC Control Register)

Bit(s)	Name	Function
0	ADC_EN	1 = ADC Enable, 0 = Shutdown
1	START	1 = Start Conversion (Self-clearing)
2	AUTO	1 = Auto Mode, 0 = Manual Mode
3	VREF_SEL	1 = External V_ref, 0 = Internal V_ref
4	INT_EN	1 = Interrupt Enable, 0 = Interrupt Disable
6	CLK_SEL	0 = 8 kHz, 1 = 16 kHz
[11:7]	-	Reserved (Read as 0)

0b01 STATUS_REG (ADC Status Register)

Bit(s)	Name	Function
0	EOC	End of Conversion. 1 when complete. Cleared on read.
1	BUSY	1 when a conversion or calibration is in progress.
[11:2]	-	Reserved (Read as 0)

0b10 DATA_REG (ADC Data Register)

Bit(s)	Name	Function
[11:0]	ADC_DATA	The 12-bit conversion result.

0b11 INFO_REG (Chip Info Register)

Bit(s)	Name	Function
[11:0]	VERSION_INFO	Info and version no.

Frame Structure

Bits	Field Name	Size	Description
[15:14]	CMD	2 bits	Specifies the operation (Read, Write, etc.).
[13:12]	ADDR	2 bits	Selects which of the four internal registers to access.
[11:0]	PAYLOAD	12 bits	Contains the data for a write or is ignored for a read.

Field Definitions

CMD (Command)
- **2'b00: READ** - Reads data from the register at ADDR.
- **2'b01: WRITE** - Writes the PAYLOAD to the register at ADDR.
- **2'b10: SET MASK** - Sets bits in CTRL_REG using the PAYLOAD as a mask.
- **2'b11: CLEAR MASK** - Clears bits in CTRL_REG using the PAYLOAD as a mask.

ADDR (Address)
- **2'b00:** CTRL_REG (Control Register)
- **2'b01:** STATUS_INFO_REG (Status Register)
- **2'b10:** DATA_REG (Data Register)
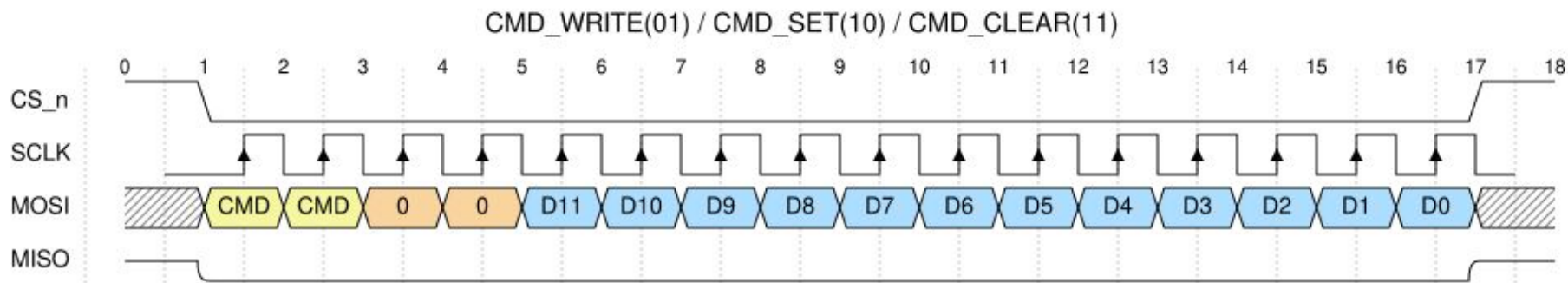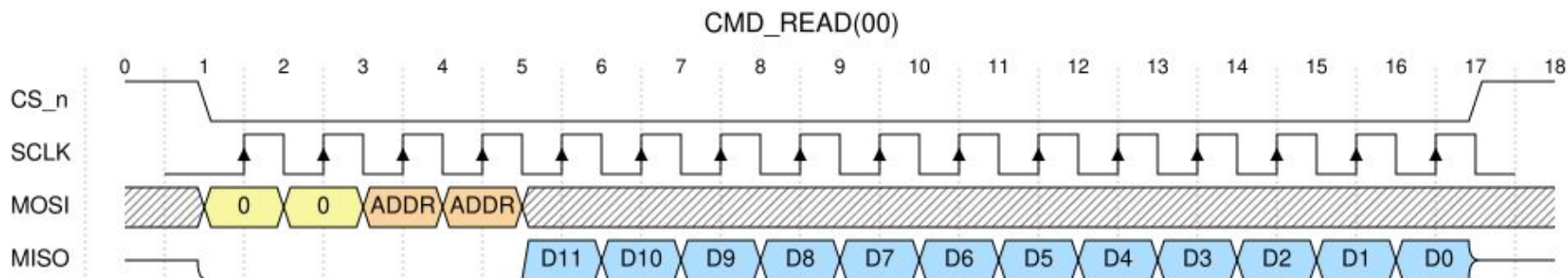- **2'b11:** CAL_OFFSET_REG (Calibration Register)

Usage Examples

To **WRITE** 12'h5A5 to the CTRL_REG:

CMD:	01
ADDR:	00
PAYLOAD:	10110100101 (5A5)
Full 16-bit frame:	0100 0101 1010 0101 = 16'h45A5
Full 16-bit miso frame:	0000 0000 0000 0000 = 16'h0000

To **READ** from the STATUS_INFO_REG:

CMD:	00
ADDR:	01
PAYLOAD:	000000000000 (Don't care)
Full 16-bit frame:	0001 0000 0000 0000 = 16'h1000
Full 16-bit miso frame:	0000 ____ ____ ____ = 16'h0___

SPI COMMANDS AND REGISTERS

CMD_READ(00)

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
CS_n																			
SCLK																			
MOSI		0	0	ADDR	ADDR														
MISO						D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0		

CMD_WRITE(01) / CMD_SET(10) / CMD_CLEAR(11)

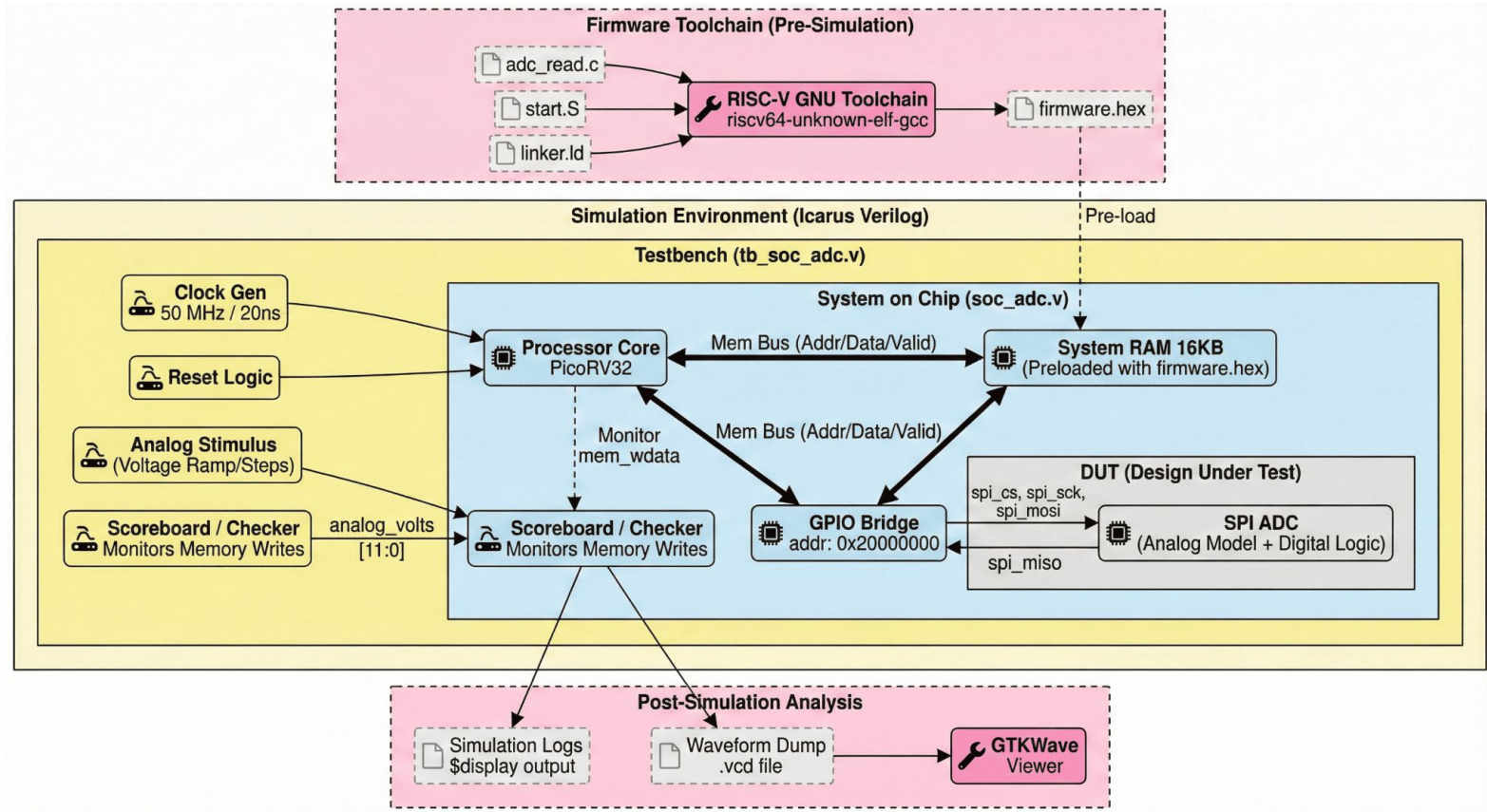|	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
CS_n																			
SCLK																			
MOSI		CMD	CMD	0	0	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0		
MISO																			

SPI TIMING DIAGRAM

Functional Verification

SoC-Level Validation via Hardware-Firmware Co-Simulation

Verification is conducted within a complete SoC environment where a PicoRV32 processor core executes C-based firmware (adc_read.c) to drive the Design Under Test (DUT). The testbench architecture integrates a sophisticated scoreboard and checker that monitor memory writes via the system bus to verify that digital outputs match the analog input stimuli. Extensive testing covers multiple scenarios, including register Read/Write checks, interrupt generation, and auto-sampling modes, ensuring the logic correctly handles both single-shot and continuous conversion requests under varied timing conditions.

Firmware Toolchain (Pre-Simulation)

adc_read.c
start.S
linker.ld

RISC-V GNU Toolchain
riscv64-unknown-elf-gcc

firmware.hex

Pre-load

Simulation Environment (Icarus Verilog)

Testbench (tb_soc_adc.v)

Clock Gen
50 MHz / 20ns

Reset Logic

Analog Stimulus
(Voltage Ramp/Steps)

Scoreboard / Checker
Monitors Memory Writes

analog_volts
[11:0]

System on Chip (soc_adc.v)

Processor Core
PicoRV32

Mem Bus (Addr/Data/Valid)

System RAM 16KB
(Preloaded with firmware.hex)

Monitor
mem_wdata

Mem Bus (Addr/Data/Valid)

Scoreboard / Checker
Monitors Memory Writes

GPIO Bridge
addr: 0x20000000

DUT (Design Under Test)

spi_cs, spi_sck,
spi_mosi

SPI ADC
(Analog Model + Digital Logic)

spi_miso

Post-Simulation Analysis

Simulation Logs
$display output

Waveform Dump
.vcd file

GTKWave
Viewer

TESTBENCH ARCHITECTURE

================ STARTING EXHAUSTIVE TESTS ================

TEST 1: Defaults
  [PASS]   Reset CTRL: 0x000
  [PASS]   Info Reg: 0x00a

TEST 2: Register R/W
  [PASS]   Readback: 0x5a5

TEST 3: VREF Pin
  [PASS] VREF High

TEST 4: Single 8ksps (Input 0xA52)
  [PASS] 8ksps Timing OK (120010 ns)
  [PASS]   Result: 0xa52

TEST 5: Data Corners
  [PASS]   Min Input: 0x000
  [PASS]   Max Input: 0xfff

TEST 6: Self-Clearing Start
  [PASS] Bit Cleared

TEST 7: Busy Flag
  [PASS] Busy High

TEST 8: Interrupt Masking
  [PASS] IRQ Low
  [PASS] EOC Set

TEST 9: 16ksps Clock
  [PASS] 16ksps Timing OK (59520 ns)
  [PASS]   16ksps Data: 0x888

TEST 10: Revert 8ksps
  [PASS] 8ksps Timing OK (120000 ns)
  [PASS]   8ksps Data: 0x111

TEST 11: Auto Mode (NO IRQ)
  [PASS]   Auto 1: 0x200
  [PASS]   Auto 2: 0x999

TEST 12: Global Disable
  [PASS] Inactive

================ ALL TESTS PASSED ================

BLOCK-LEVEL VERIFICATION RESULTS

```
------------------------------------------
1. Building Firmware (adc_read.c)...
------------------------------------------
------------------------------------------------------
🔧 Compiling for RISC-V (RV32I)
   Script Root: /home/kyouma-hououin/Documents/VLSI/designs/spi_sar_adc/riscv_toolchain
   Source:      riscv_toolchain/source/adc_read.c
   Helpers:     start.S, linker.ld (in Script Root)
   Output:      /home/kyouma-hououin/Documents/VLSI/designs/spi_sar_adc/riscv_toolchain/firmware/firmware.hex
------------------------------------------------------
/usr/lib/gcc/riscv64-unknown-elf/13.2.0/../../../riscv64-unknown-elf/bin/ld: warning: /home/kyouma-hououin/Documents/VLSI/designs/spi_sar_adc/riscv_toolchain/fi
rmware/firmware.elf has a LOAD segment with RWX permissions
✅ Success! Hex file generated at:
   /home/kyouma-hououin/Documents/VLSI/designs/spi_sar_adc/riscv_toolchain/firmware/firmware.hex
------------------------------------------
2. Compiling Verilog Hardware...
------------------------------------------
------------------------------------------
3. Running Simulation...
------------------------------------------
VCD info: dumpfile tb_soc_adc.vcd opened for output.

 --- ADC SOC SIMULATION (COMPREHENSIVE) START ---

[RAM] Firmware wrote Result:      2566
[Time         831350000] CPU wrote      2566 (0x00000a06) to 0x00000400
✅ SINGLE SHOT (16kHz) PASSED! (Got        2566) (Expected        2567)

-> Changing Input to 1543 for Auto Mode (8kHz)...
[Time        2094210000] CPU wrote      1542 (0x00000606) to 0x00000404
✅ AUTO SAMPLE 1 (8kHz) PASSED! (Got        1542) (Expected        1543)

-> Changing Input to 2654...
[Time        2948470000] CPU wrote      2653 (0x00000a5d) to 0x00000408
✅ AUTO SAMPLE 2 (8kHz) PASSED! (Got        2653) (Expected        2654)

-> Changing Input to 3129...
[Time        3800210000] CPU wrote      3128 (0x00000c38) to 0x0000040c
✅ AUTO SAMPLE 3 (8kHz) PASSED! (Got        3128) (Expected        3129)

-> Starting Abort Test (Disabling ADC)...
[Time        4642110000] CPU wrote     43783 (0x0000ab07) to 0x00000410
✅ ABORT TEST PASSED! Busy flag cleared.

==========================================
🎉 ALL TESTS PASSED (Single, Auto, Clock, Abort)
==========================================
```
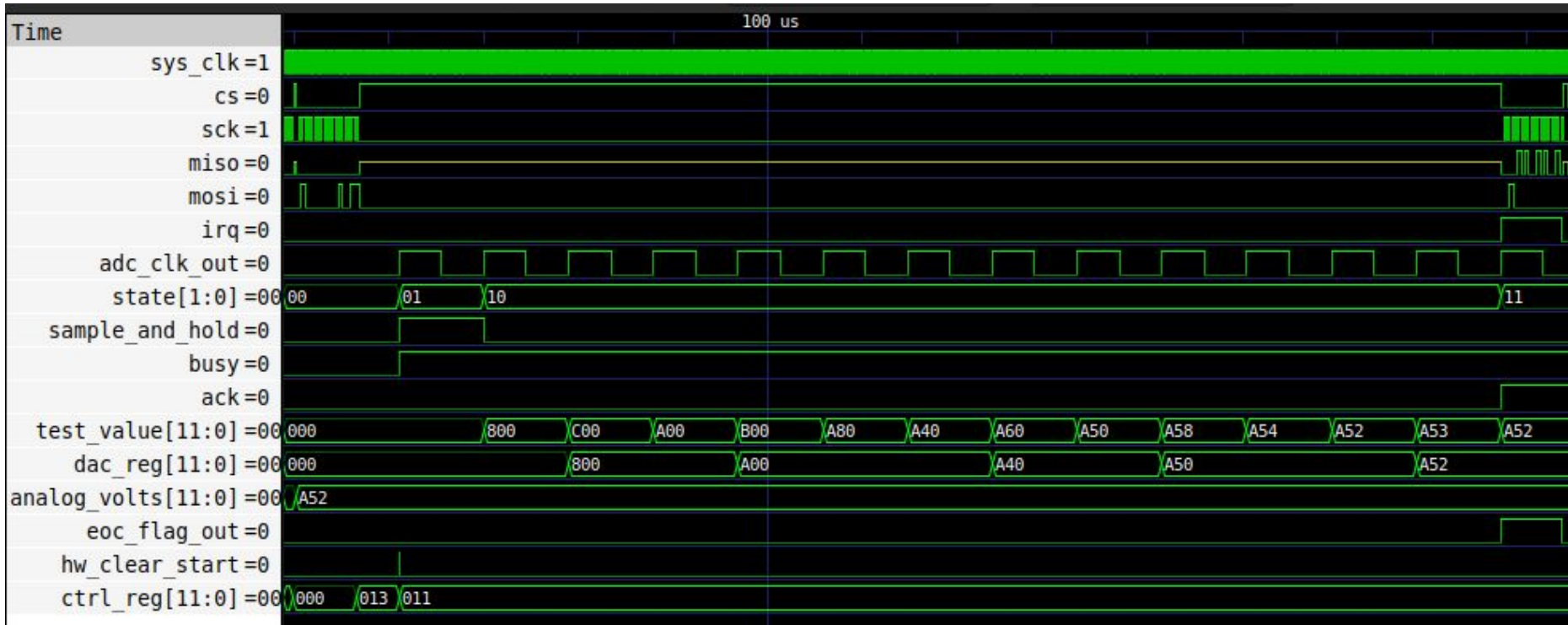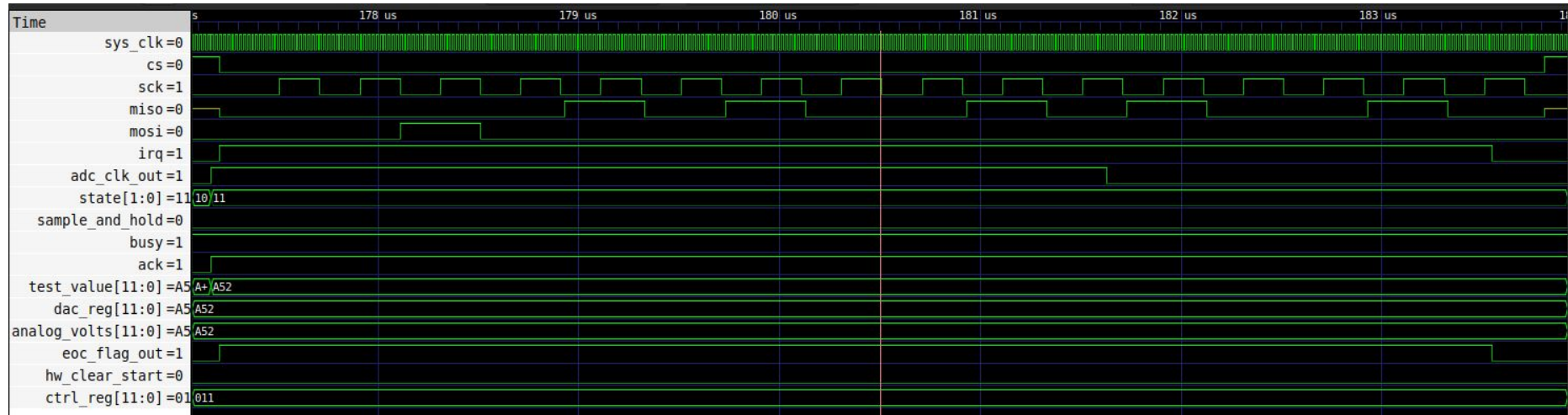
SOC-LEVEL VERIFICATION RESULTS

ADC SINGLE SHOT CONVERSION WAVEFORM: START PHASE

ADC SINGLE SHOT CONVERSION WAVEFORM: CONVERSION PHASE

ADC SINGLE SHOT CONVERSION WAVEFORM: END PHASE

Physical Implementation

Automated RTL-to-GDSII Flow & Physical Signoff

The physical design is implemented using the LibreLane "Classic" flow, which automates the progression from RTL linting and synthesis (Yosys) to floorplanning, placement, and detailed routing (OpenROAD). The layout undergoes Multi-Corner Static Timing Analysis (STA) to verify setup and hold compliance across different process variations, alongside physical verification (DRC/LVS) using Magic and Netgen to ensure manufacturability. A final comparative Power, Performance, and Area (PPA) analysis confirmed the optimal configuration, achieving a total power consumption of 108.9 μW at 10Mhz clk by utilizing high-density cells and active clock gating.
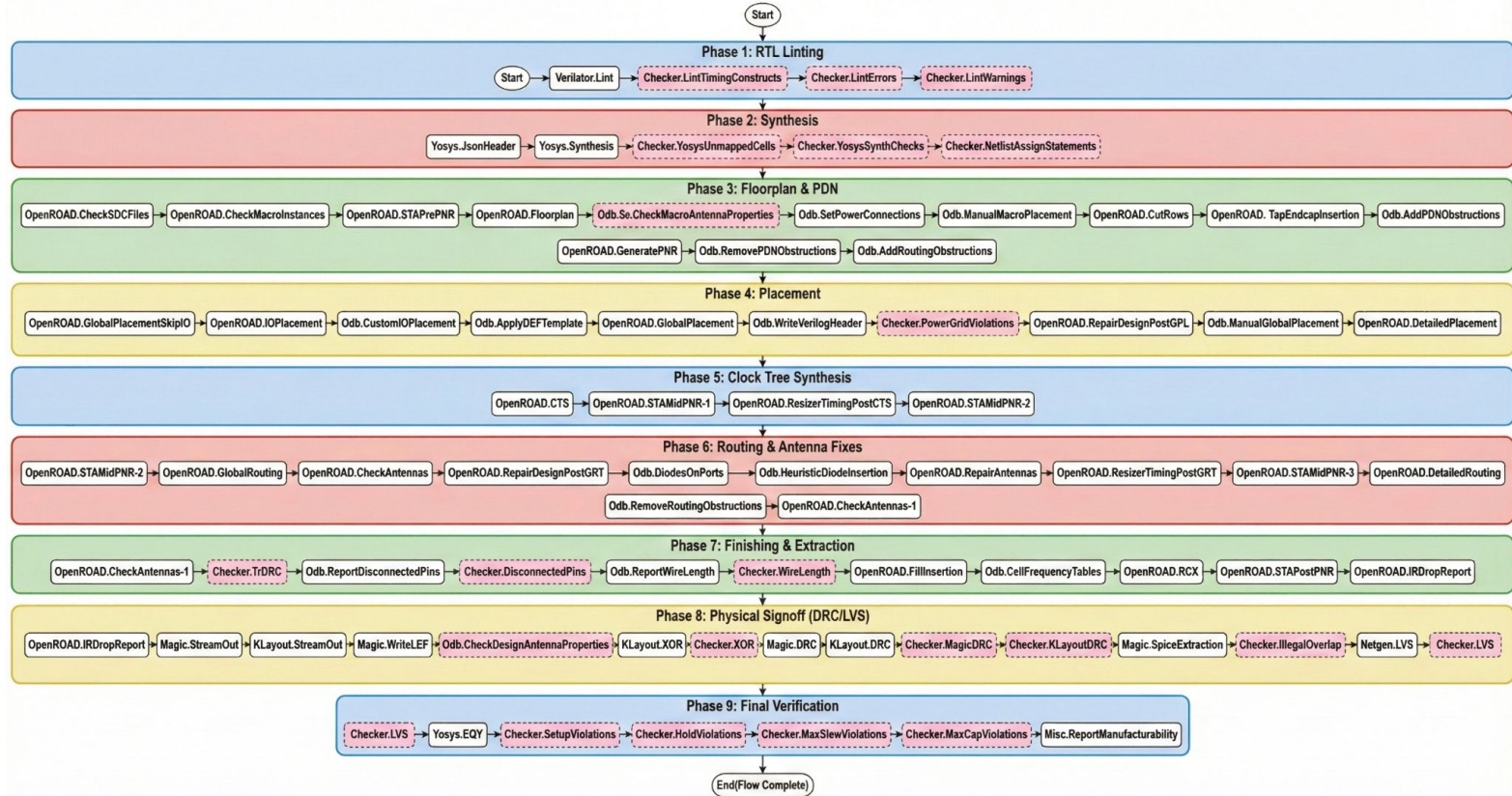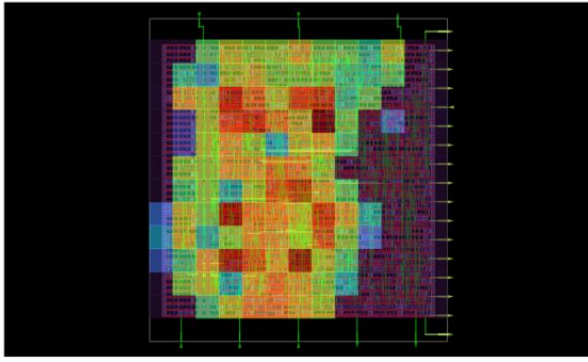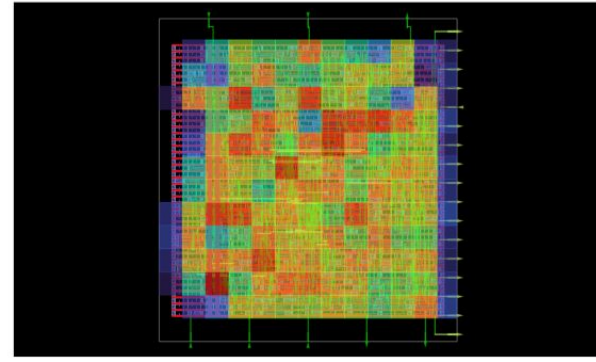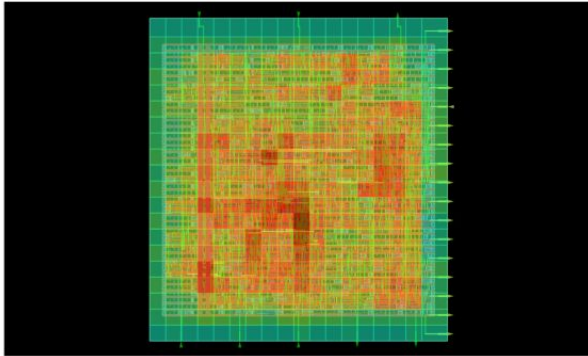
LIBRELANE "CLASSIC" SYNTHESIS FLOW
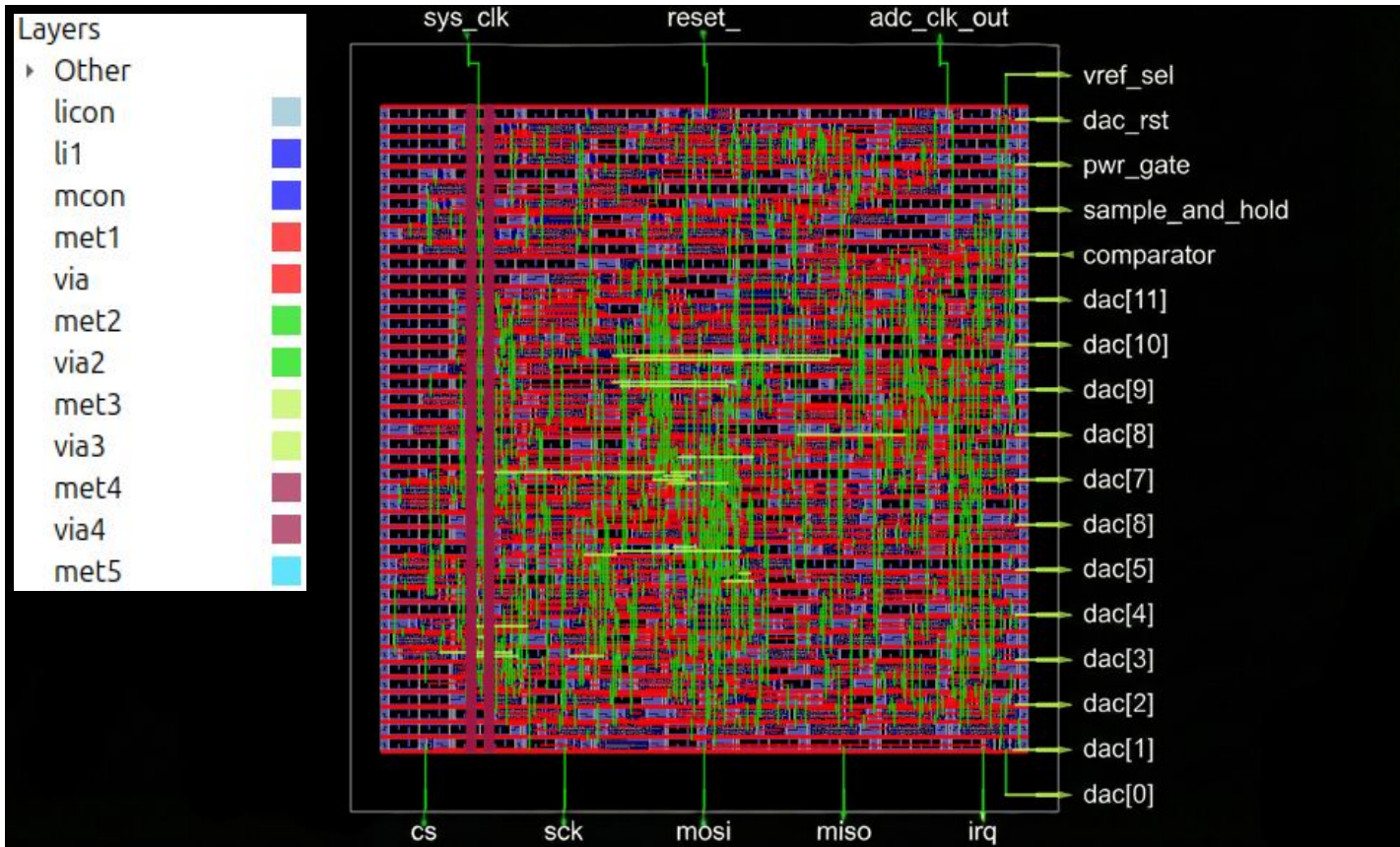
OPENROAD DATABASE FORMAT (OBD) VIEW
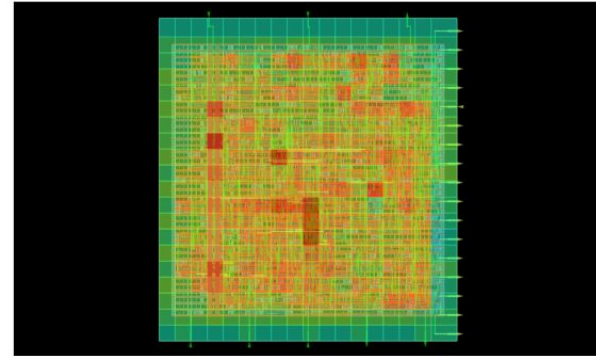
(a) Power Density Map

(b) Placement Density Map

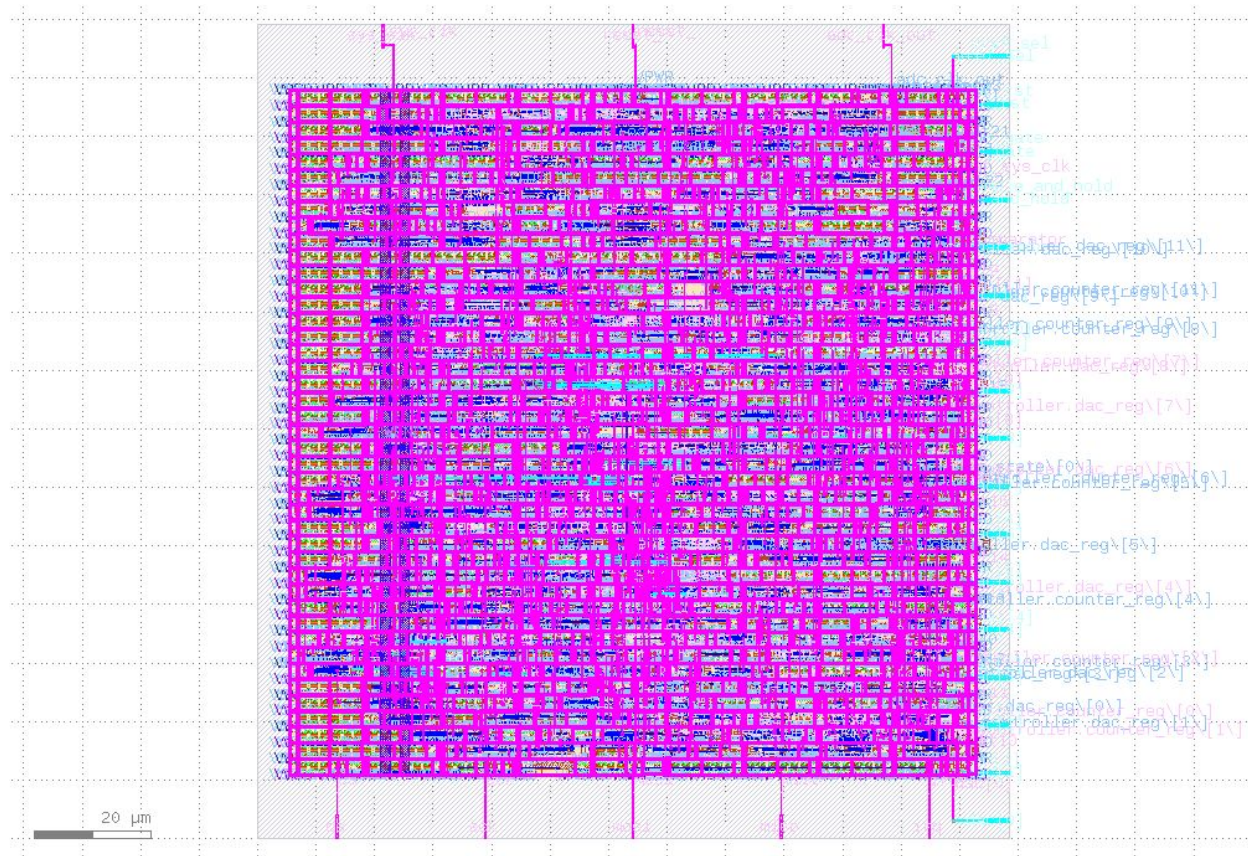(c) Estimated Congestion (RUDY)

(d) Routing Congestion

HEAT MAP ANALYSIS

KLAYOUT GRAPHIC DATA SYSTEM (GDS) VIEW

Constraint: `CLOCK_PERIOD` = 100.0 ns (10 MHz).

Power Component	Value	Percentage
Total Power	108.9 μW	100.0%
Internal Power	79.8 μW	73.3%
Switching Power	29.1 μW	26.7%
Leakage Power	8.28 nW	$< 0.01\%$

- **DRC:** 0 Violations (Magic).

- **LVS:** 0 Mismatches (Netgen).

- **Antenna:** 0 Violations (OpenROAD/Arc).

STATIC POWER ANALYSIS SUMMARY

Constraint: `CLOCK_PERIOD` = 100.0 ns (10 MHz).

Corner	Hold WNS (ns)	Hold TNS (ns)	Setup WNS (ns)	Setup TNS (ns)
Overall	0.1081	0.0000	58.9116	0.0000
nom_tt_025C_1v80	0.3147	0.0000	59.2983	0.0000
nom_ss_100C_1v60	0.8657	0.0000	58.9178	0.0000
nom_ff_n40C_1v95	0.1105	0.0000	59.4411	0.0000
min_tt_025C_1v80	0.3114	0.0000	59.3048	0.0000
min_ss_100C_1v60	0.8592	0.0000	58.9283	0.0000
min_ff_n40C_1v95	**0.1081**	**0.0000**	**59.4456**	**0.0000**
max_tt_025C_1v80	0.3181	0.0000	59.2946	0.0000
max_ss_100C_1v60	**0.8715**	**0.0000**	**58.9116**	**0.0000**
max_ff_n40C_1v95	0.1129	0.0000	59.4383	0.0000

MULTI_CORNER STATIC TIMING ANALYSIS SUMMARY

1. **Config A (Baseline):** 45% Utilization, Clock Gating Enabled.

2. **Config B (High Density):** 60% Utilization, Clock Gating Enabled.

3. **Config C (Low Power Baseline):** 45% Utilization, Clock Gating Disabled.

Metric	Config A (Base)	Config B (High Dens.)	Config C (No Gating)
Instance Count	747	709	747
Cell Area (μm^2)	**7124.33**	7129.34	7148.11
Total Power (W)	**1.089e-4**	1.147e-4	1.093e-4
Switching Power (W)	2.908e-5	3.095e-5	3.071e-5
Setup WNS (ns)	59.30	58.91	59.30
Clock Skew (ns)	0.63	0.58	0.63

COMPARATIVE POWER PERFORMANCE AREA (PPA) ANALYSIS

Project Repository:

https://github.com/SaumyaRaj188/spi_sar_adc