

Double Data Rate 5 Physical layer (DDR5 PHY)

Supervised by: Dr. Hesham Omran

Sponsored by : Si-Vision

Presentation date: Jul 18, 2022





DDR5 PHY

OUTLINE

1. Introduction to Memories.
2. Standard Specifications.
3. DDR5 PHY Blocks.
4. System Integration.
5. FPGA flow.

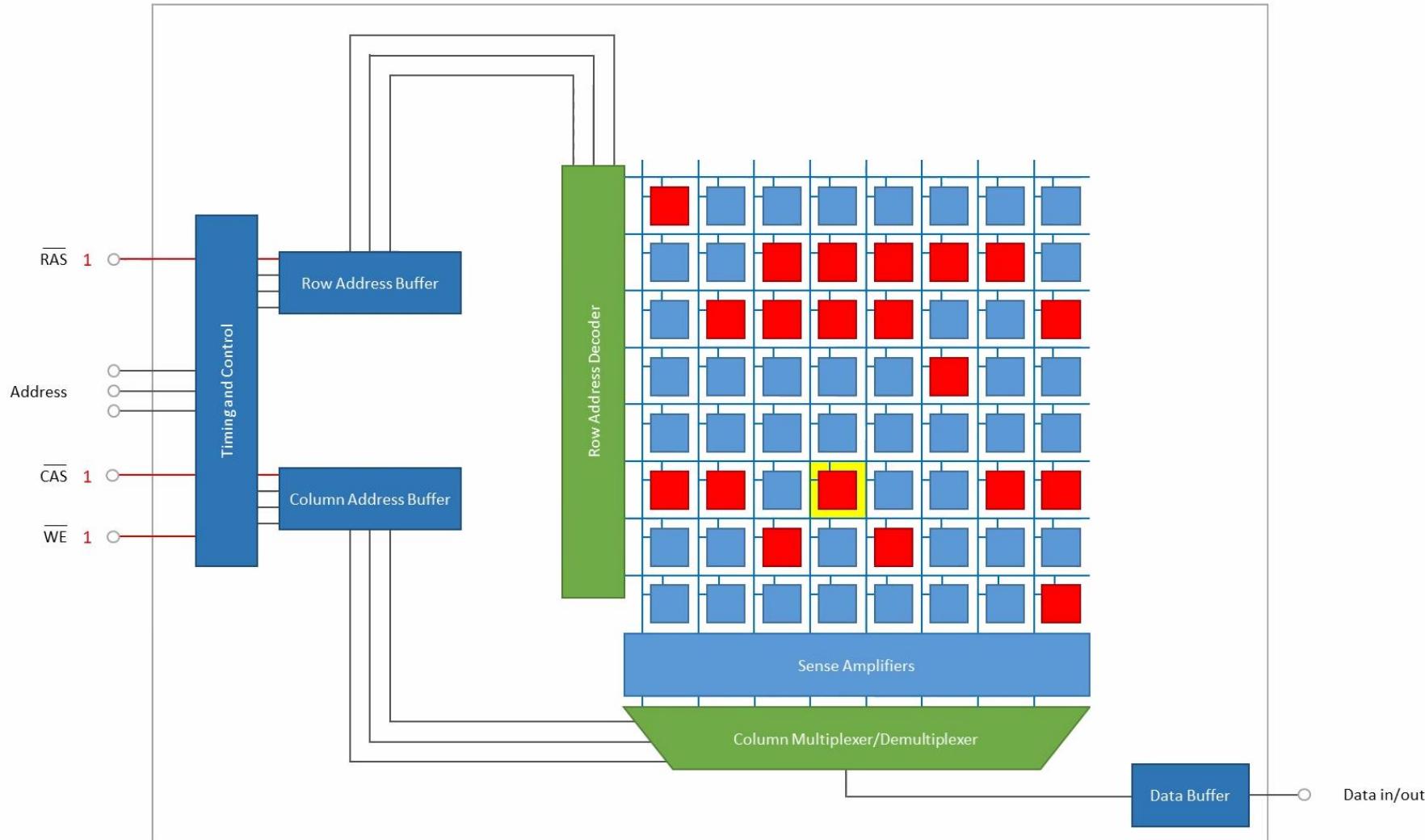


Memories

- Memory is a device or system that is used to store data in computer or related computer hardware and digital electronic devices.
- The data is stored as a stream of bits and each bit is either a zero or a one.
- The smallest storing element in the memory can store only 1 bit, it is called “cell”.
- We are interested in Random Access Memory (RAM) .

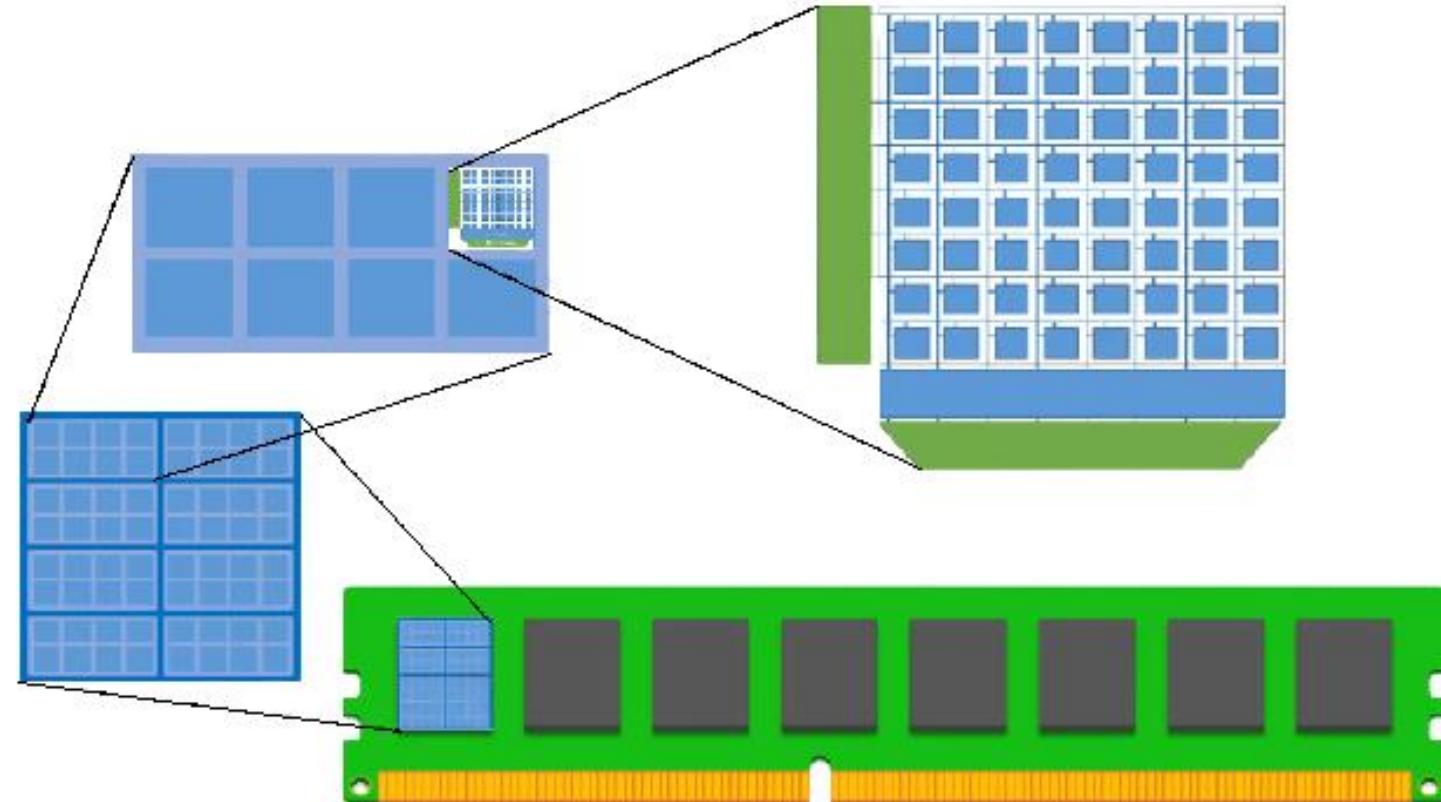


Structure of DRAM





DIMM structure

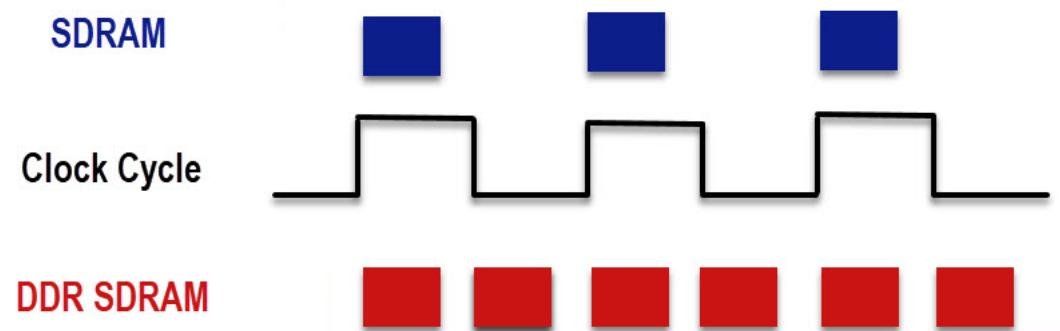


DIMM structure showing chips, banks and arrays⁽³⁾



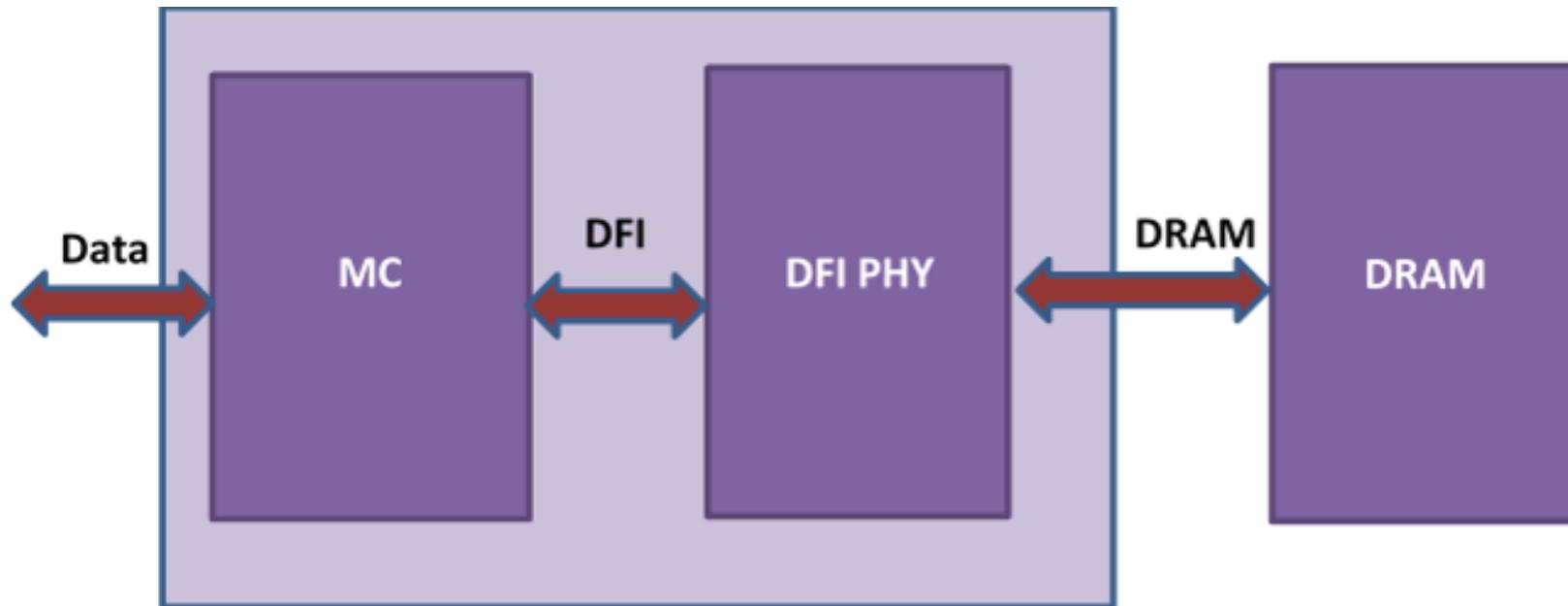
SDRAM Types

- SDRAM was first known as SDR SDRAM since data is transferred only once in each clock cycle.
- DDR SDRAM (Double Data Rate) is the advanced version of SDRAM that transfers data on both the rising and falling edges twice as fast as compared to SDR SDRAM chips.
- After the introduction of DDR new generations were introduced (DDR2 : DDR5), these generations have different internal structure to achieve higher data rates.





DDR subsystem



PHY connecting the MC to DRAM⁽⁴⁾



DDR5 PHY

OUTLINE

1. Introduction to Memories.
2. Standard Specifications.
 1. Standards.
 2. Frequency ratio
 3. Subsystem point of view.
 4. Write operation Flow.
3. DDR5 PHY Blocks.
4. System Integration.
5. FPGA flow.

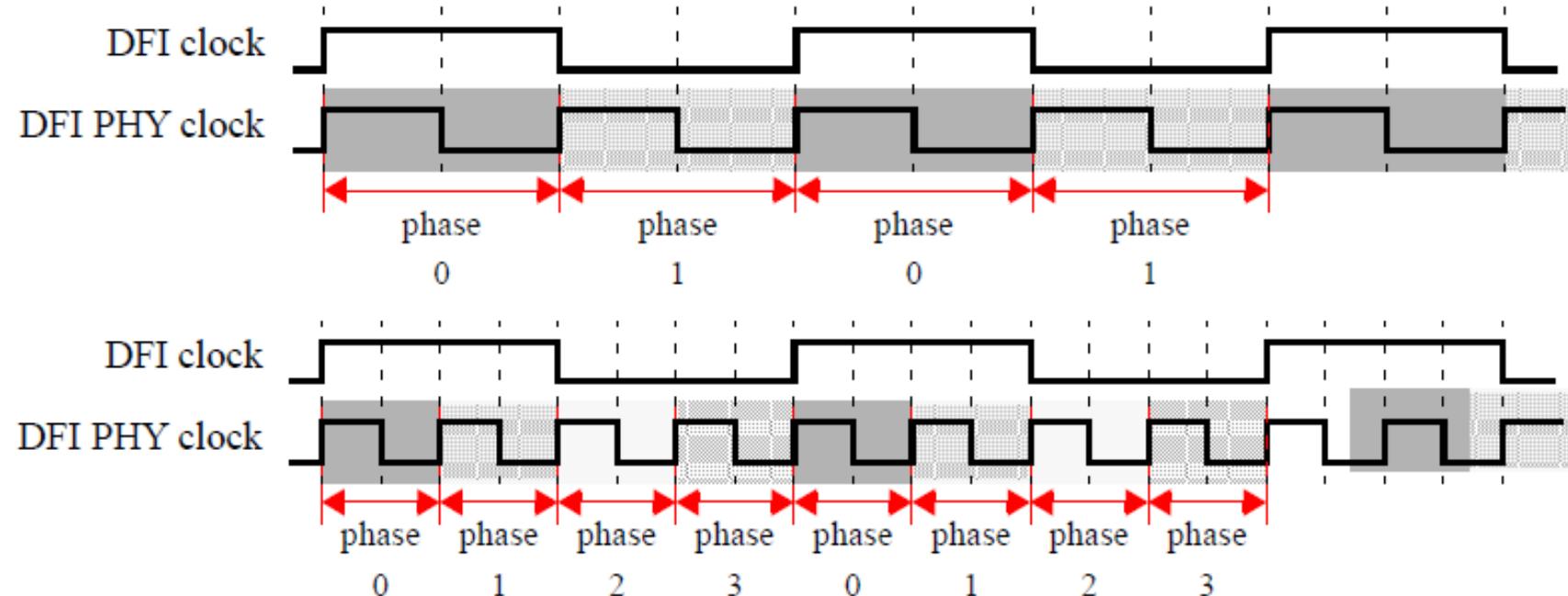


Standards

- The DDR PHY Interface (DFI) is an interface protocol that defines the signals, timing parameters and programmable parameters.
- The DFI specify the signals between MC and PHY, these signals is organized to interface groups, within each interface group are signals and parameters. Some signals are applicable only to certain DRAM types. All of the DFI signals must use the corresponding parameters.
- Joint Electron Device Engineering Council (JEDEC) standard defines the “DDR5 SDRAM” specifications



Frequency ratio



Frequency ratio 1:2 and 1:4 from DFI v5.0 - Frequency Ratio Clock Definition⁽¹⁾



CRC Flow

- **CRC can be generated by MC or PHY “default MC”**
- **PHY defines the value of PHY crc_mode and according to this value CRC is handled**
 - PHY crc_mode = 0 → CRC generation is handled in the MC
 - PHY crc_mode = 1 → CRC generation is handled in the PHY
- In BC8 mode, read CRC and write CRC bits are calculated with the inputs to the CRC engine for the chopped data bursts replaced by all '1's, as shown in figure 16.

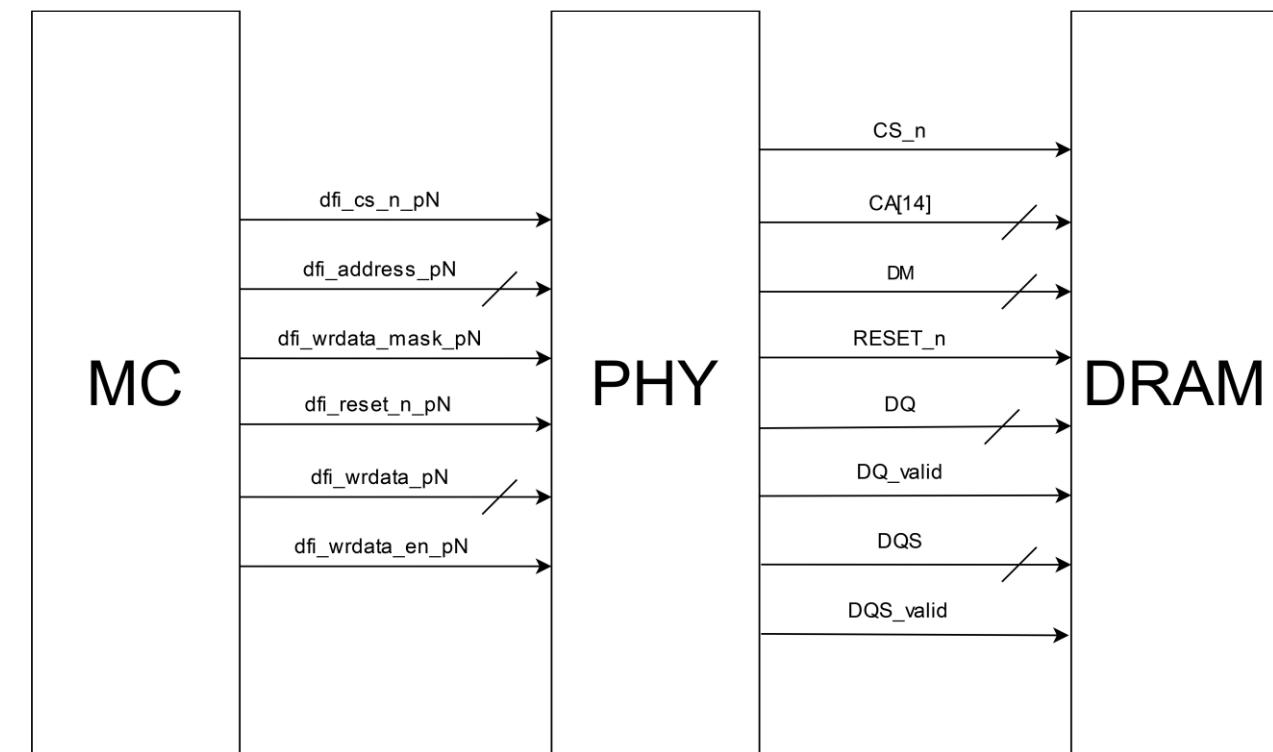
	Transfer																	
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
DQ0	d0	d4	d8	d12	d16	d20	d24	d28	1	1	1	1	1	1	1	CRC0	CRC4	
DQ1	d1	d5	d9	d13	d17	d21	d25	d29	1	1	1	1	1	1	1	CRC1	CRC5	
DQ2	d2	d6	d10	d14	d18	d22	d26	d30	1	1	1	1	1	1	1	CRC2	CRC6	
DQ3	d3	d7	d11	d15	d19	d23	d27	d31	1	1	1	1	1	1	1	CRC3	CRC7	



Sub system point of view

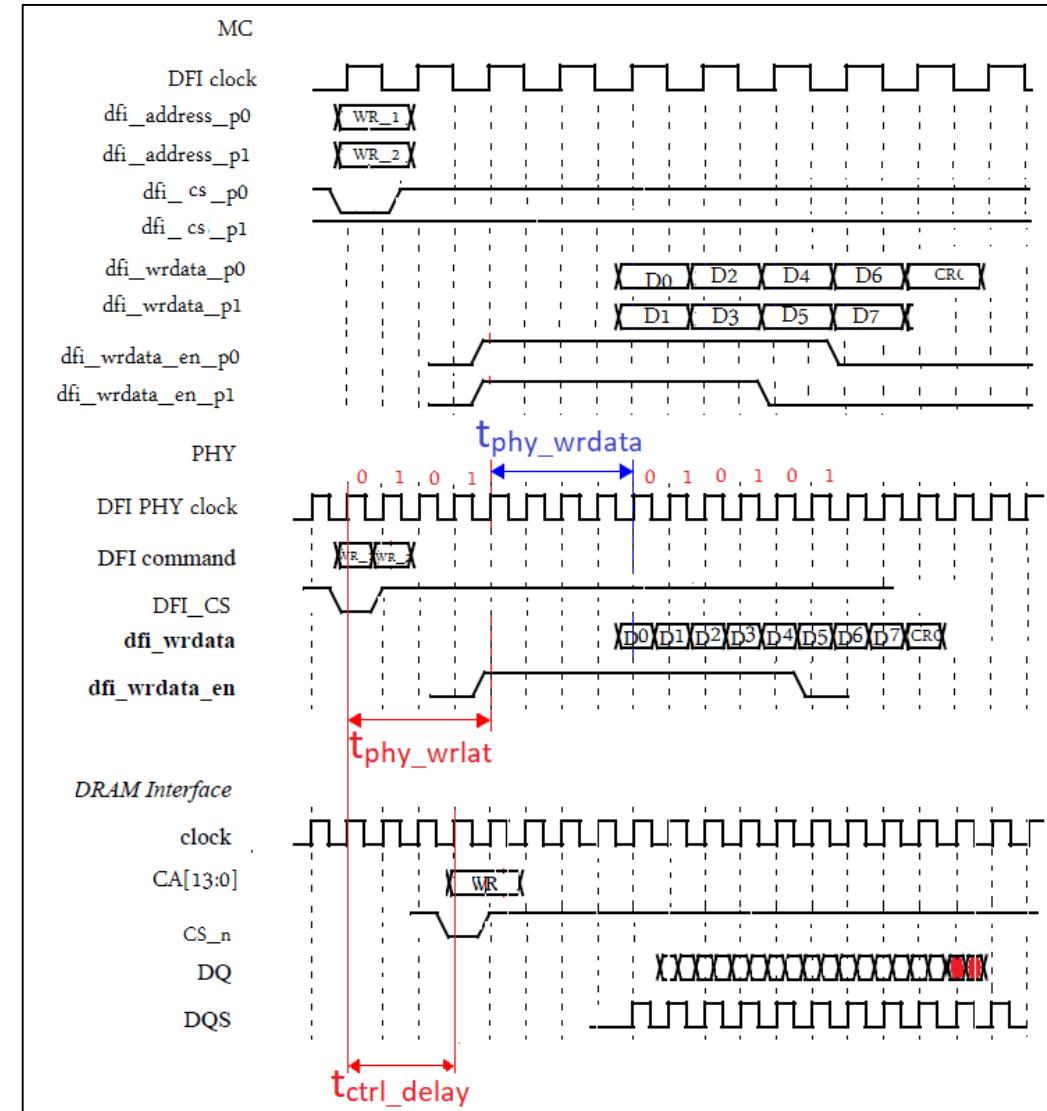
- **Signal mapping:**

- Dfi_address -> CA [13:0].
- Dfi_cs_n -> CS_n.
- Dfi_wrdata -> DQ.
- Dfi_wrdata_mask -> DM.
- Dfi_reset_n -> RESET_n.





Whole Write operation flow from MC to DRAM





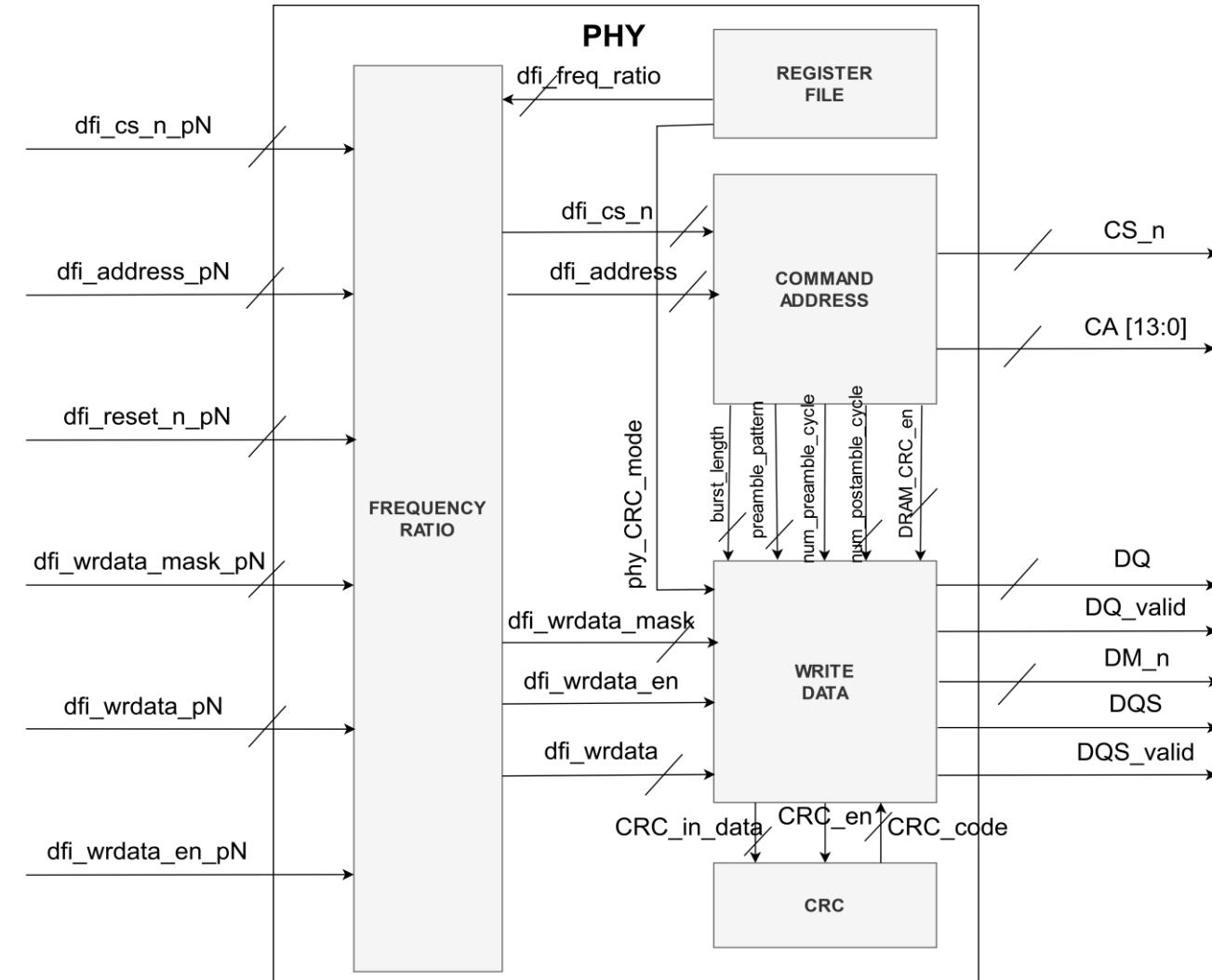
DDR5 PHY

OUTLINE

1. Introduction to Memories.
2. Standard Specifications.
3. DDR5 PHY Blocks.
 1. Frequency Ratio.
 2. Command Address.
 3. Write Manager.
 4. CRC
4. System Integration
5. FPGA flow

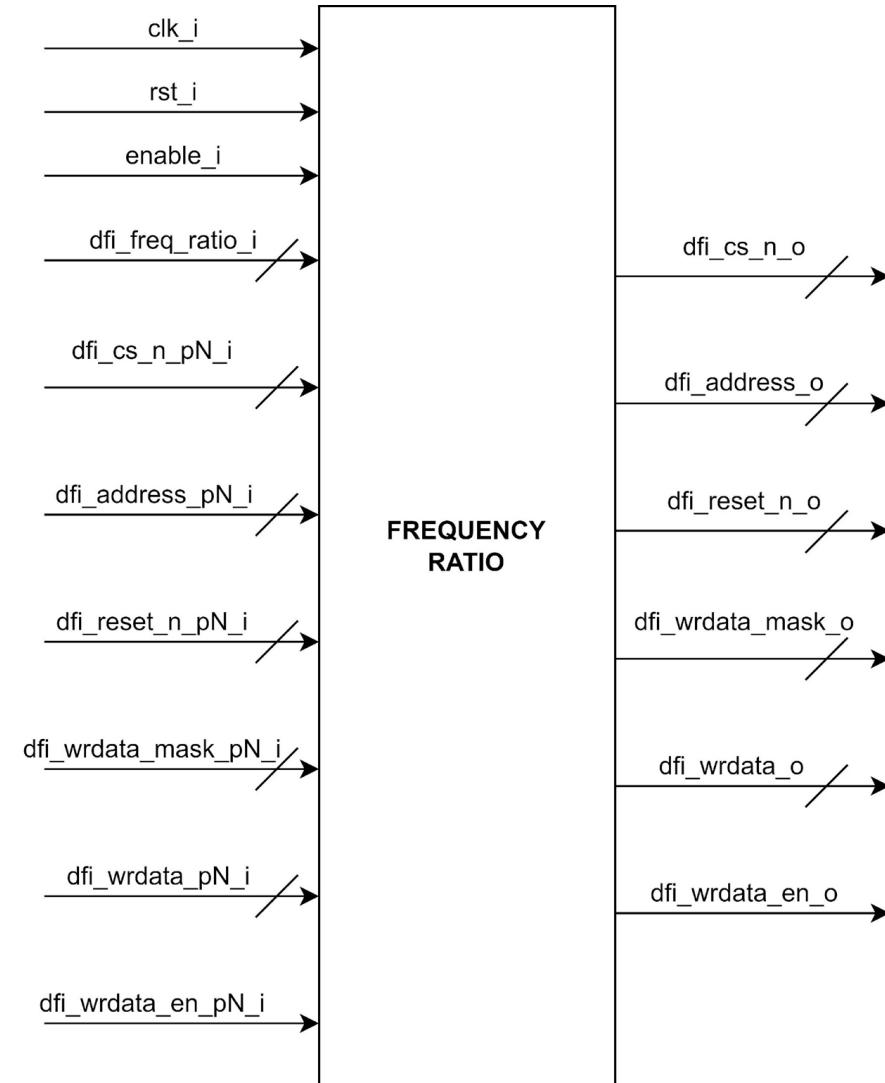


Block Diagram



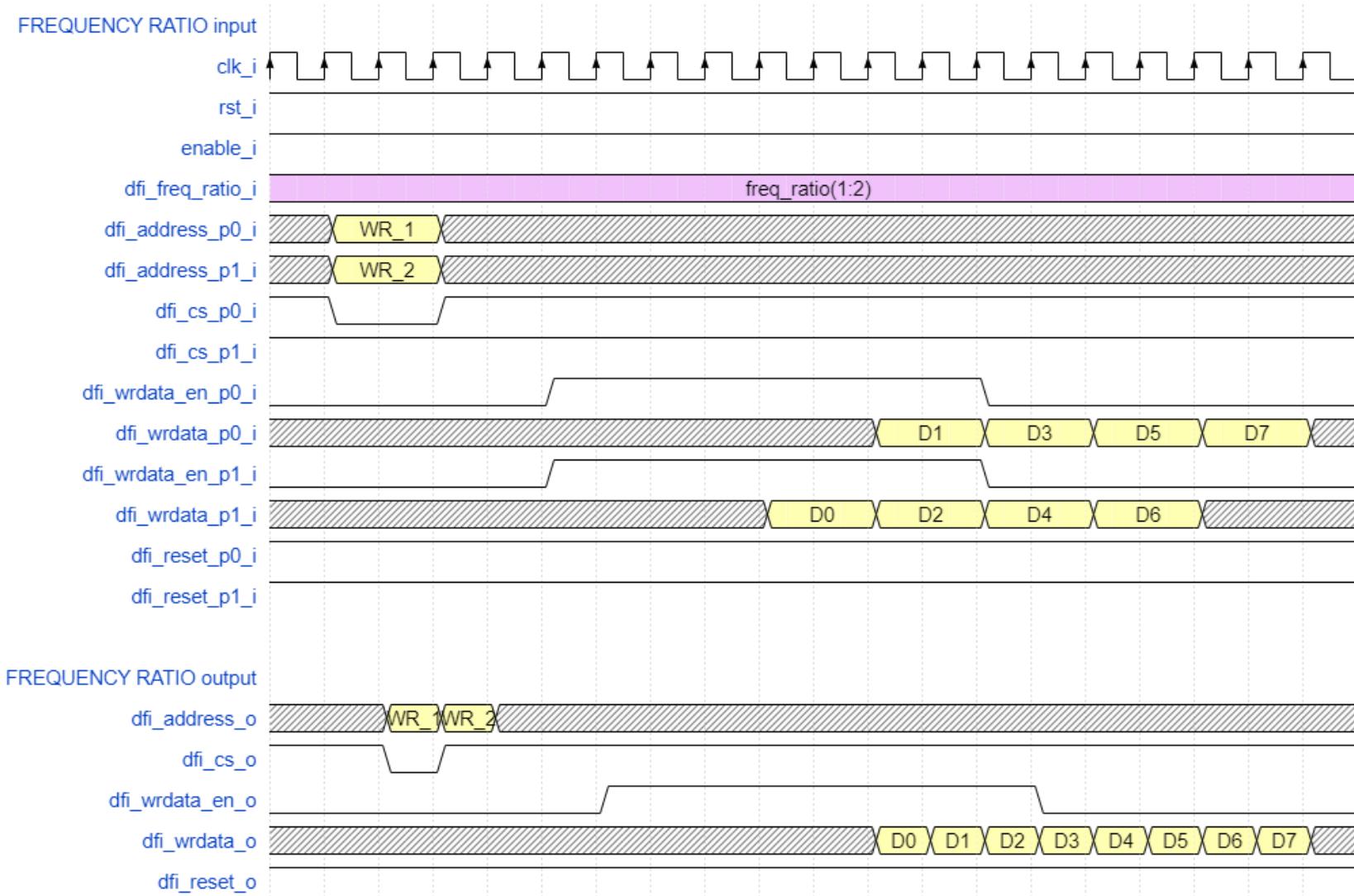


Frequency Ratio Block Diagram



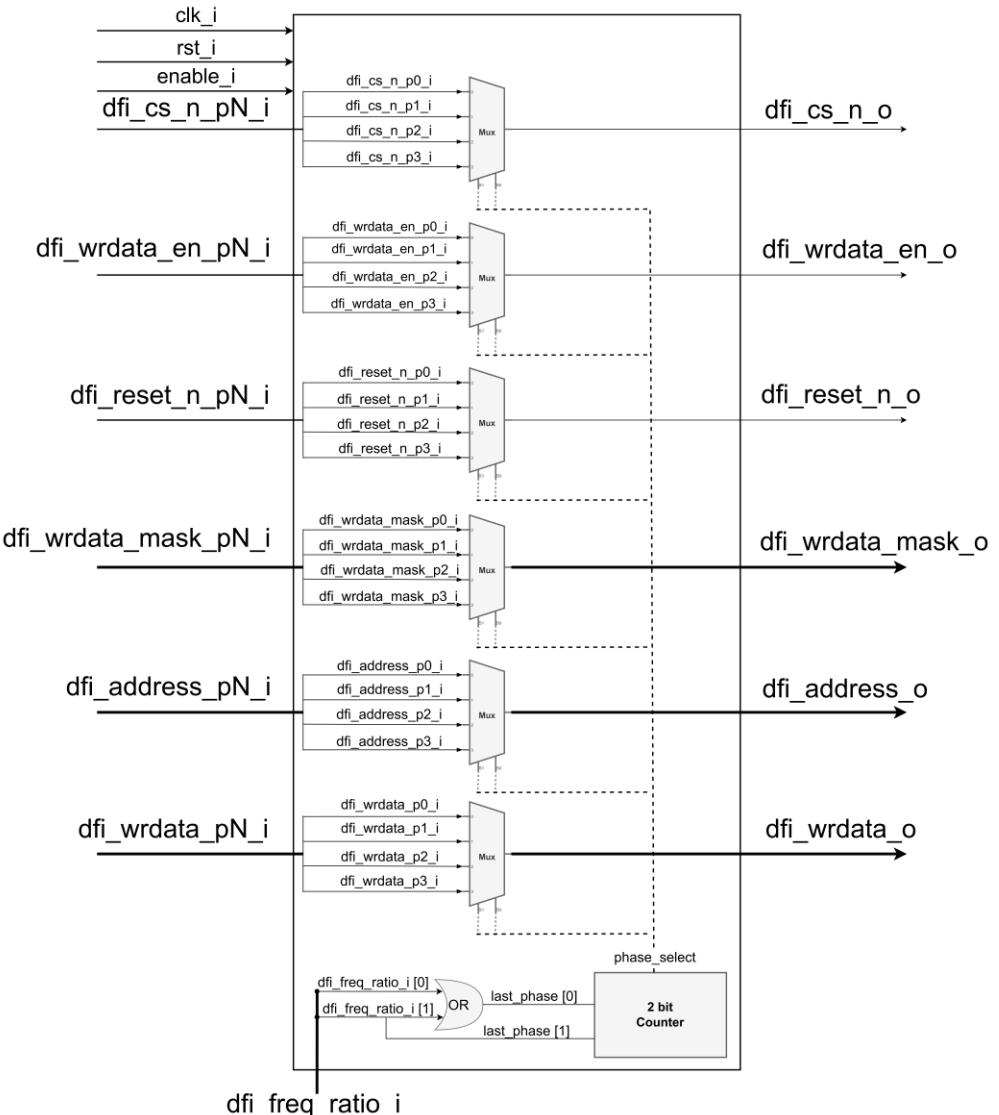


Timing Diagram



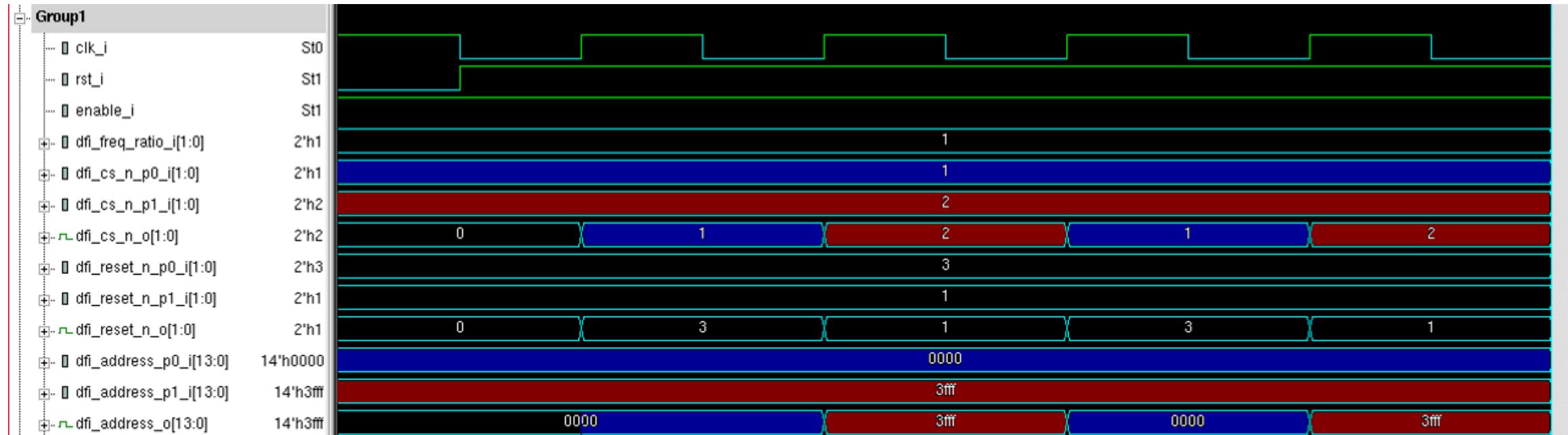


Block Architecture



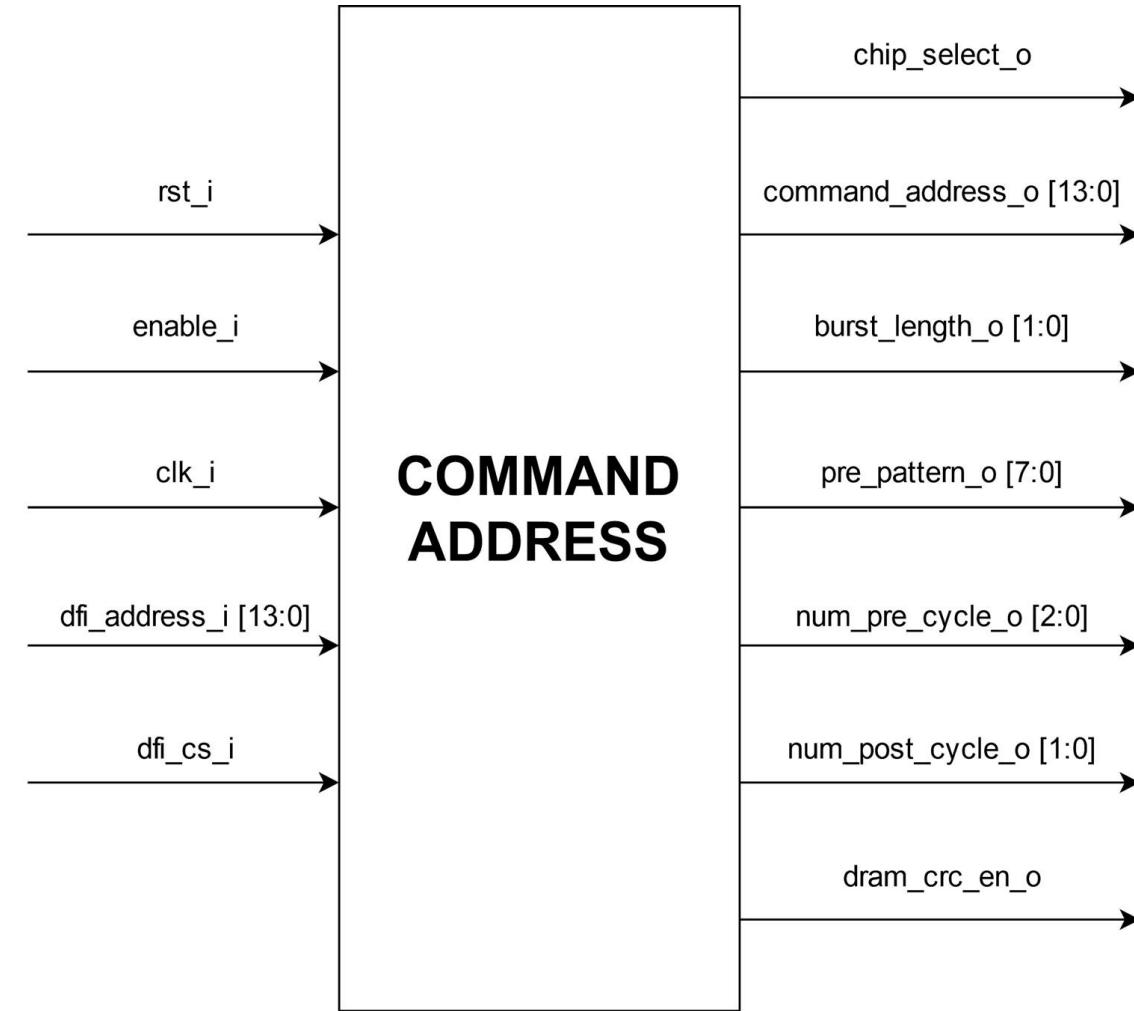


Simulation Results





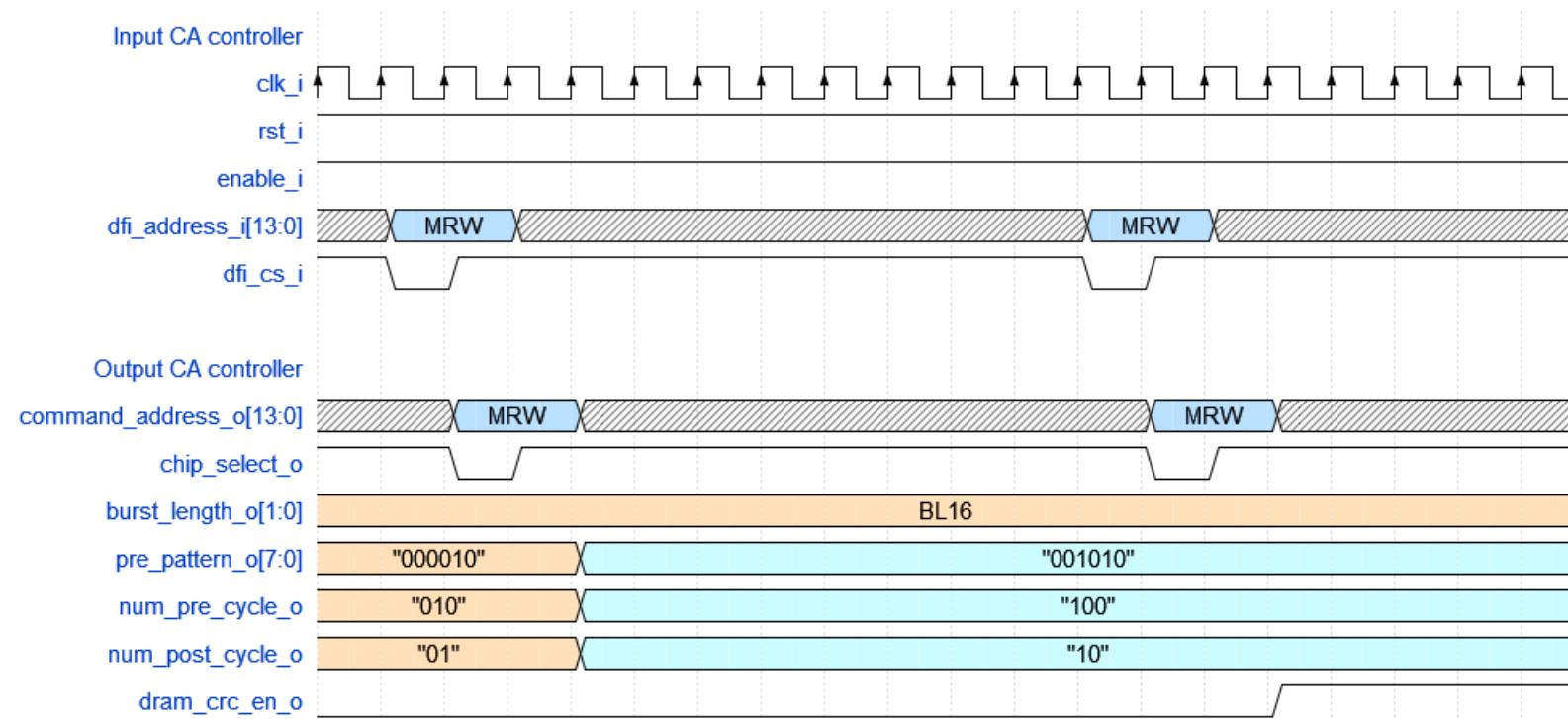
Command Address Block Diagram





Timing Diagram

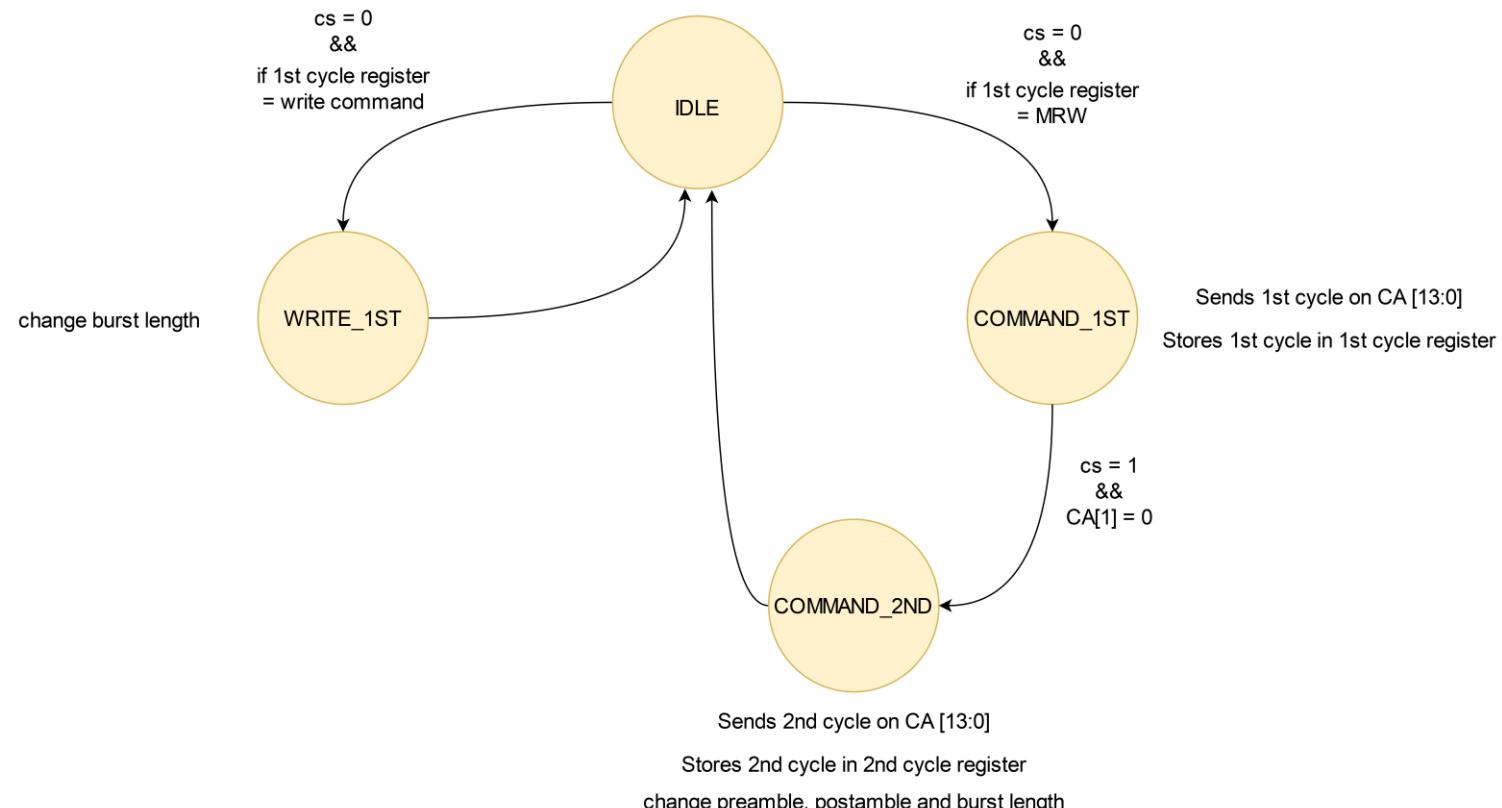
This case is mode register write command to change pre-amble and post-amble followed by another mode register write command to change the dram CRC enable.





Block Implementation

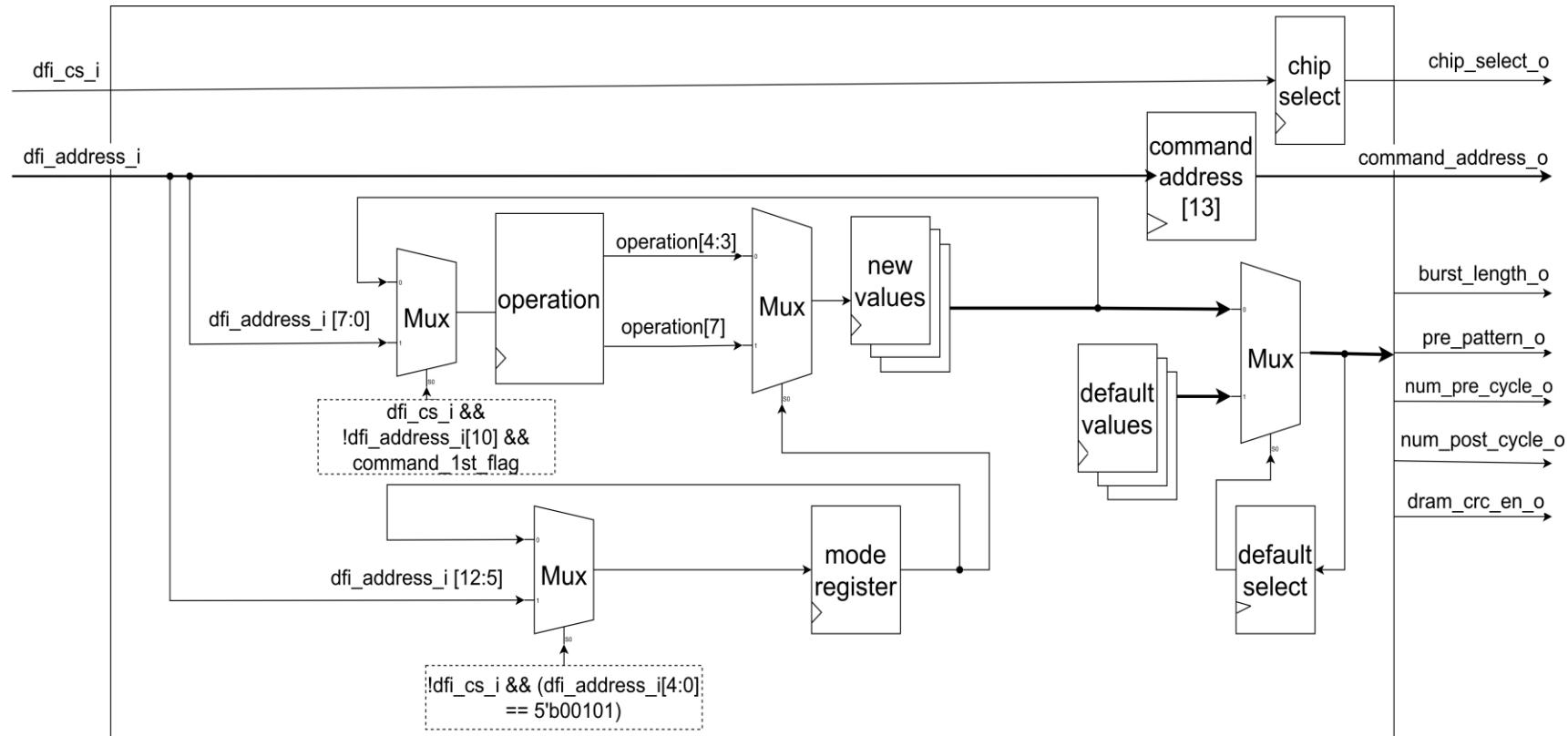
- We introduced two approaches to implement this block:
 - First approach is to implement this block using finite state machine (FSM), the FSM is consist of 4 states.





Block Implementation

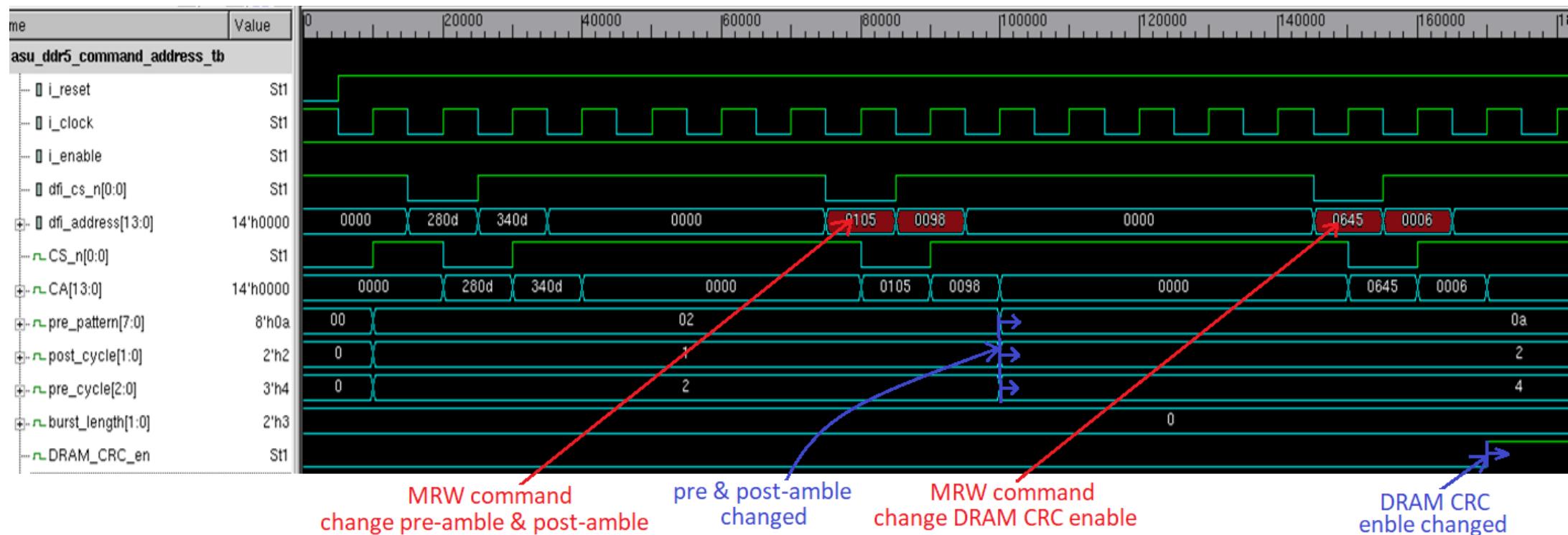
- Second approach is implementing the block using a combinational and sequential elements, e.g. multiplexers and registers.





Simulation Results

This case is write command followed by mode register write command to change pre-amble and post-amble after that mode register write command is inserted to change the dram CRC enable.





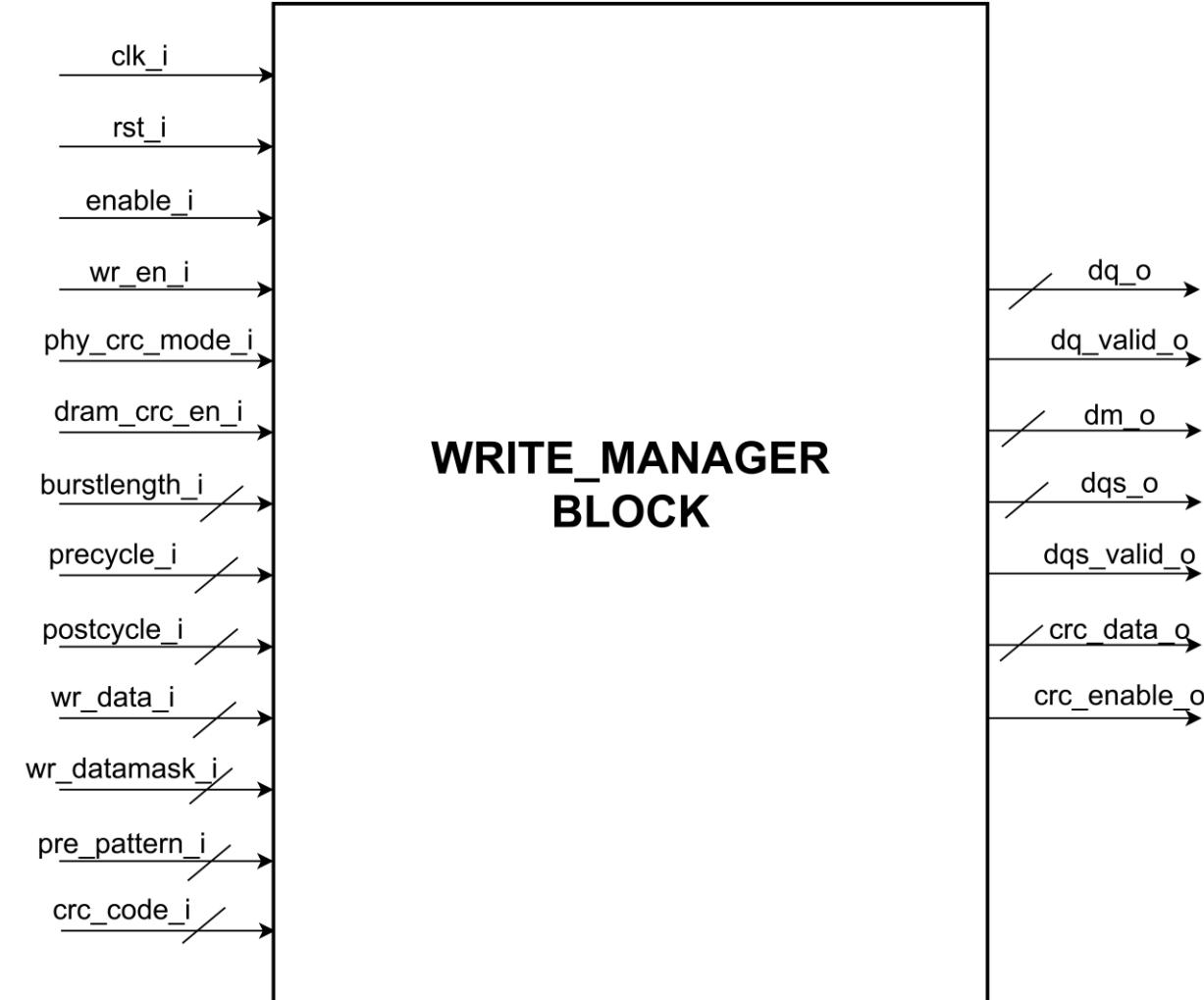
Design Optimization

The block is synthesized on Design Compiler (DC) using 130 nm technology.

Design parameter	First Approach	Second Approach
Number of combinational cells	213	187
Number of sequential cells	51	52
Number of cells	264	239
Total area (um) ²	88700	84500
Total dynamic power (mW)	1.68	1.67
Cell leakage power (uW)	2.5	2.2
Total power (mW)	1.689	1.675
Longest path slack (clk period 30 ns)	+0.02	+2.95

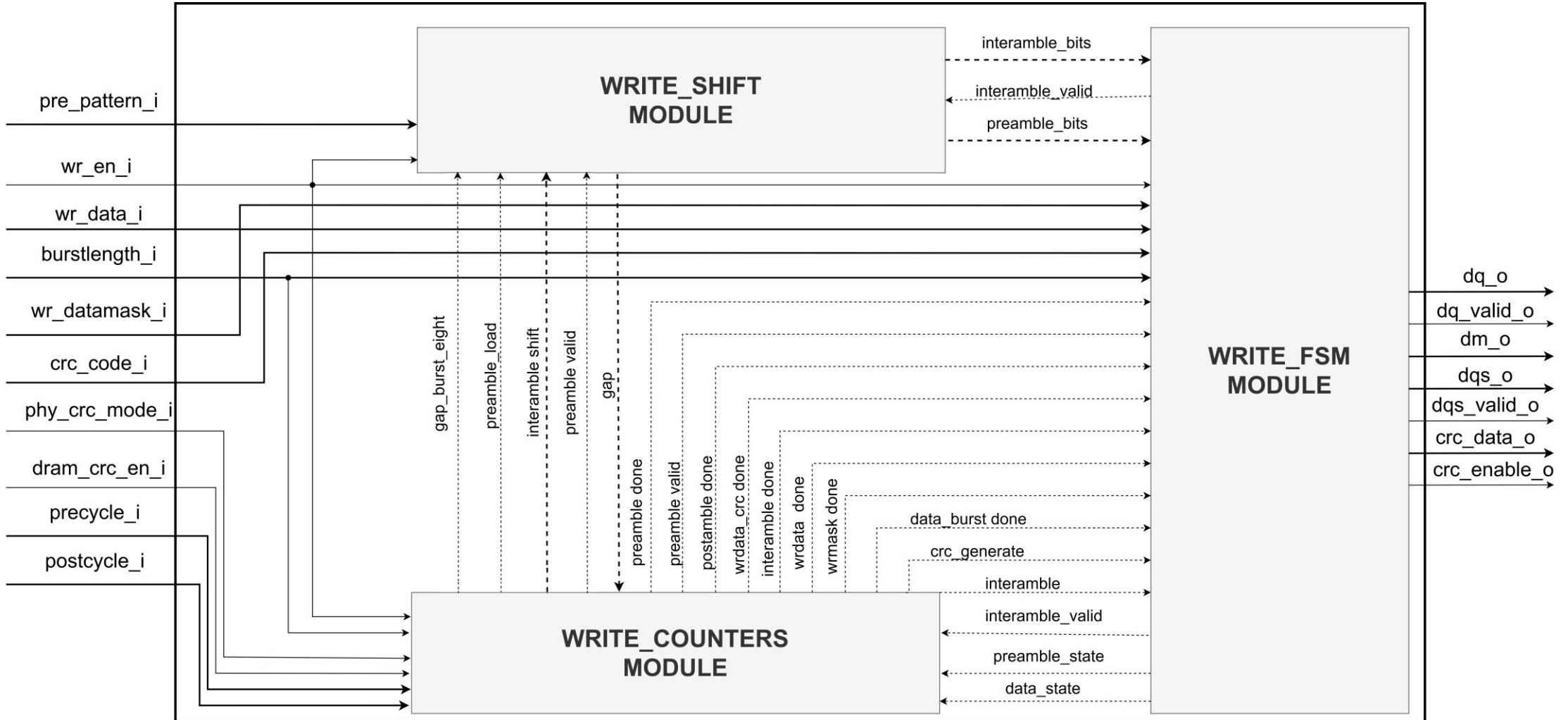


Write Manager Block Diagram



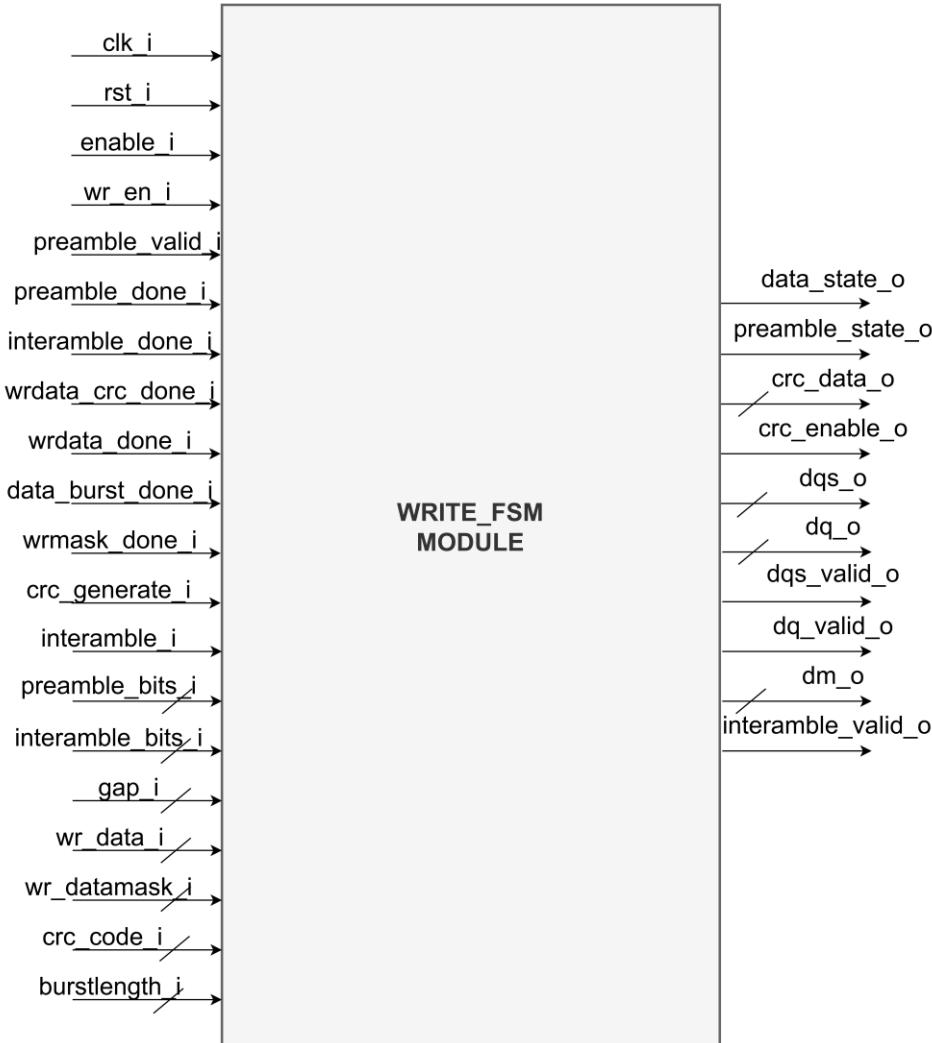


Write Manager Internal Connections





Write FSM Module Block Diagram





Write FSM State Diagram





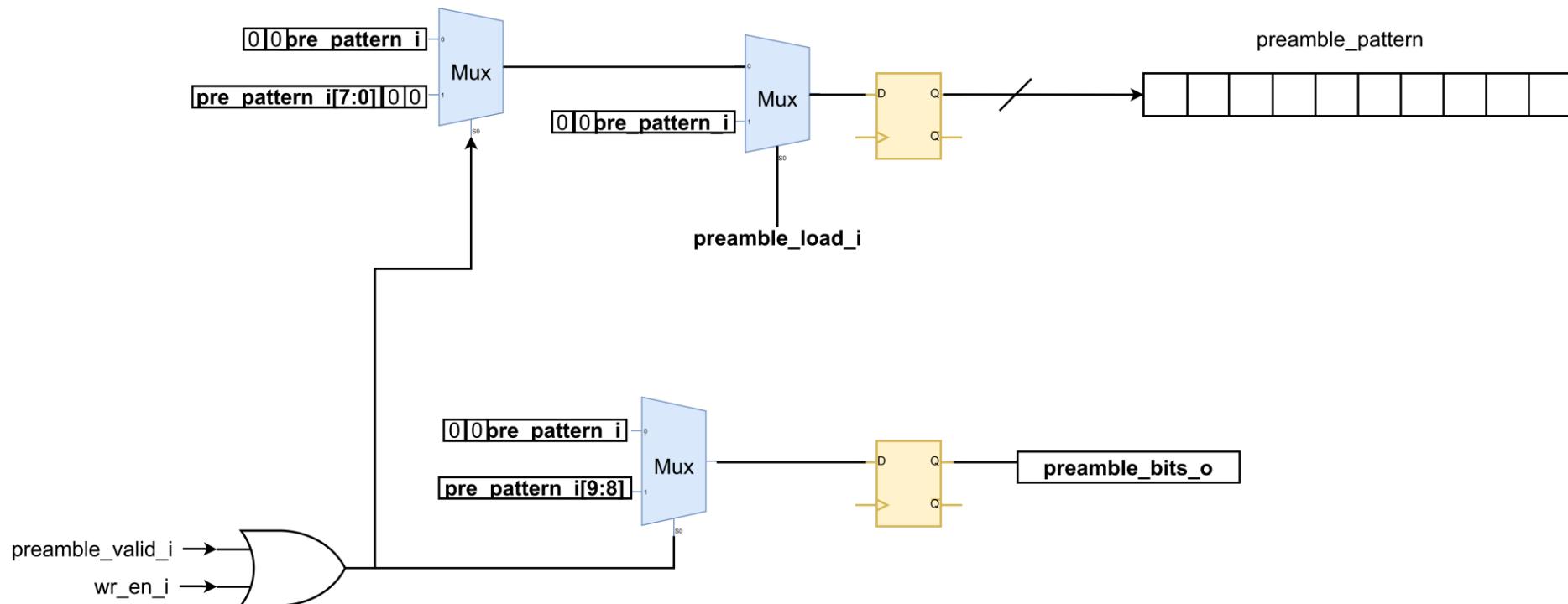
Write Shift Block Diagram





Write Shift block Functionality and Description

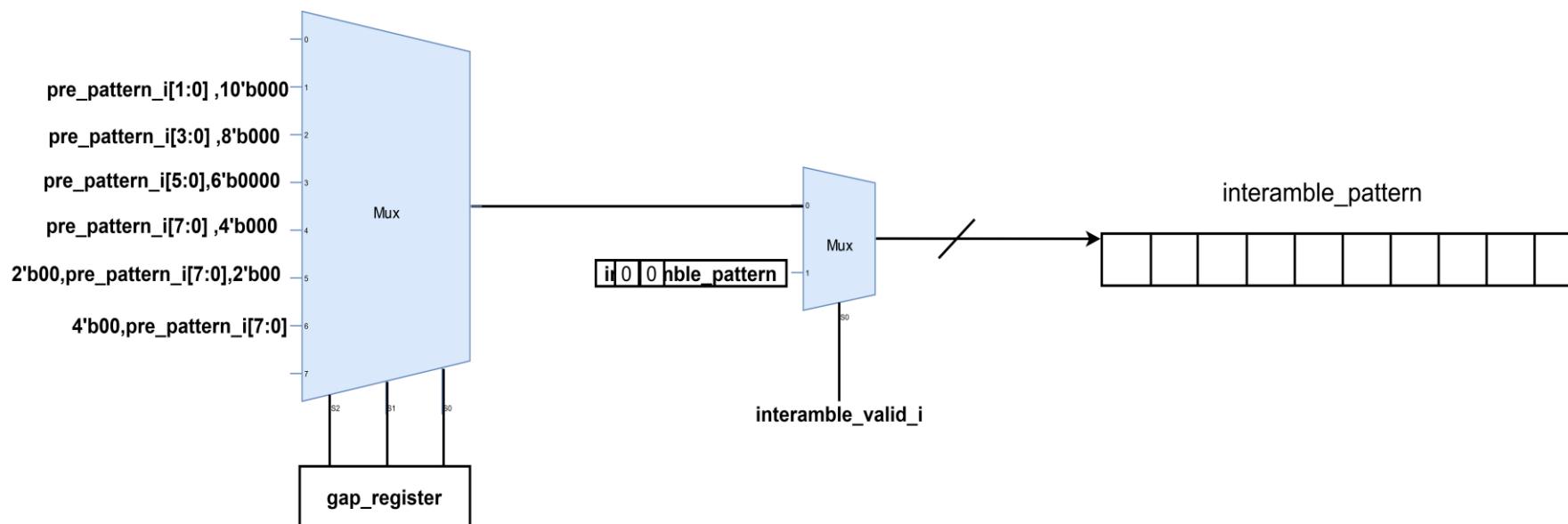
- This block is responsible for 3 main operations:
 1. Shifting preamble pattern.





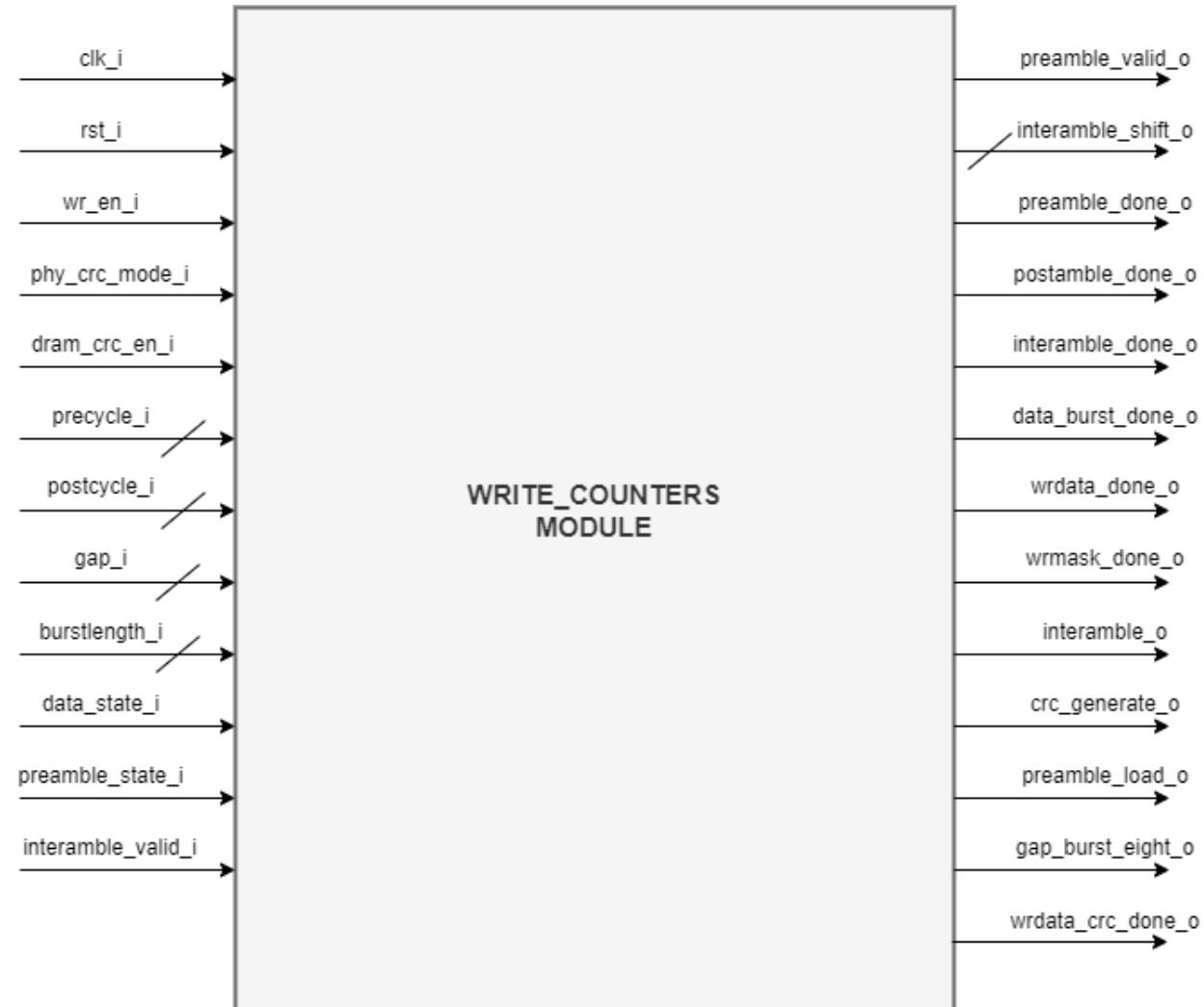
Write Shift block Functionality and Description

2. Calculating gap (no of cycles at which write enable is low):
 - o When wr_en_i becomes low, counter will start to count, the value of counter will be stored in gap_register.
 - o After wr_en_i becomes high, we will reset the counter and check the value stored in register to detect the correct interamble_pattern according to the gap value.
3. Shifting interamble pattern.





Write Counters Block Diagram





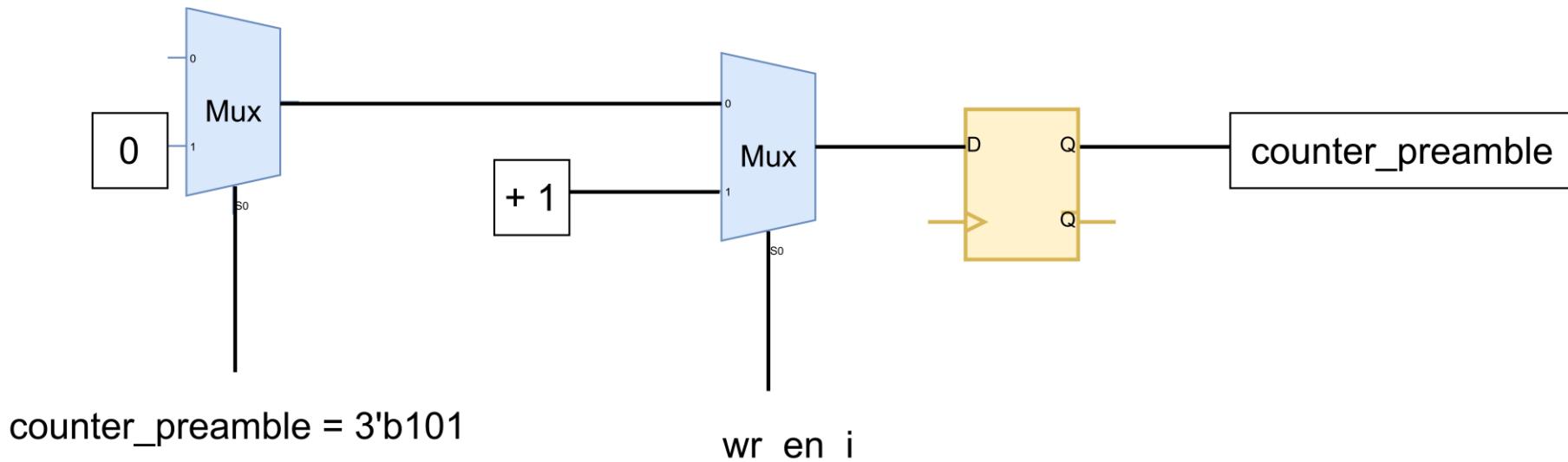
Write Counters Functionality and Implementation

- **Write counters block is responsible for:**
 - generating done signals which will be used in transitions between different states in write FSM block.
 - Determining either if there will be an inter-amble or not.
 - Determining whether there will be a CRC code generation to the DRAM or not.
- **This block consists of 3 counters:**
 - Preamble Counter
 - Write Data Counter
 - Interpost Counter



Counter 1: Preamble Counter

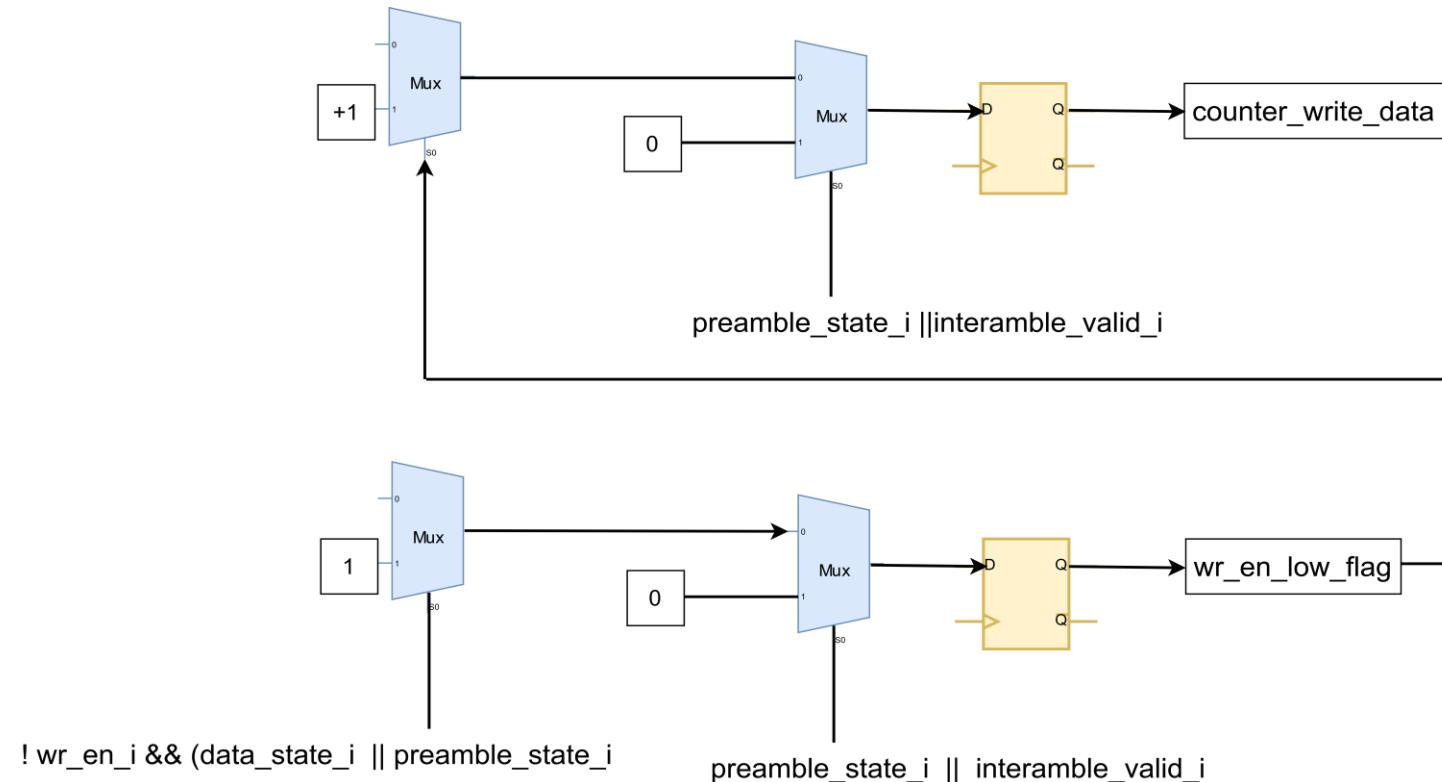
- Pre-amble counter works simultaneously with write enable signal independent on the state.
- Activating write enable signal means there is a data coming after 6 cycles. So, preamble counter start counting





Counter 2: Write Data Counter

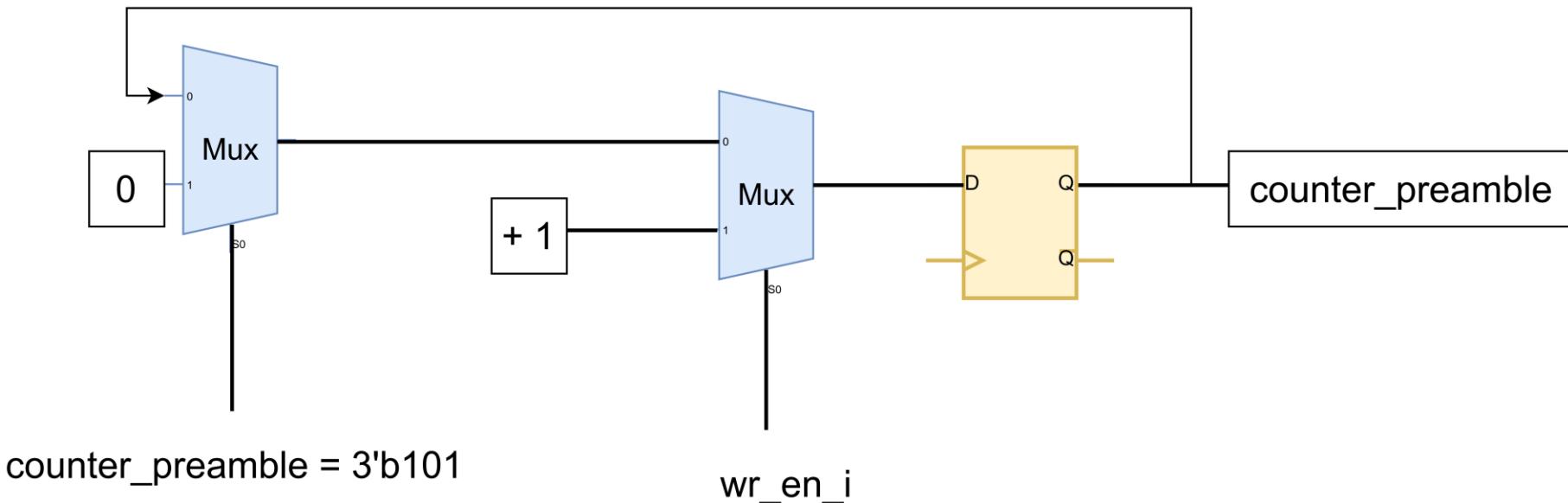
- write data counter only counts in data state especially when write enable is de-activated.
- After 5 cycles from de-activating write enable in data state, the full data burst will finish, and a done signal will be activated high telling the controller to move from data state.





Counter 3: Interpost Counter

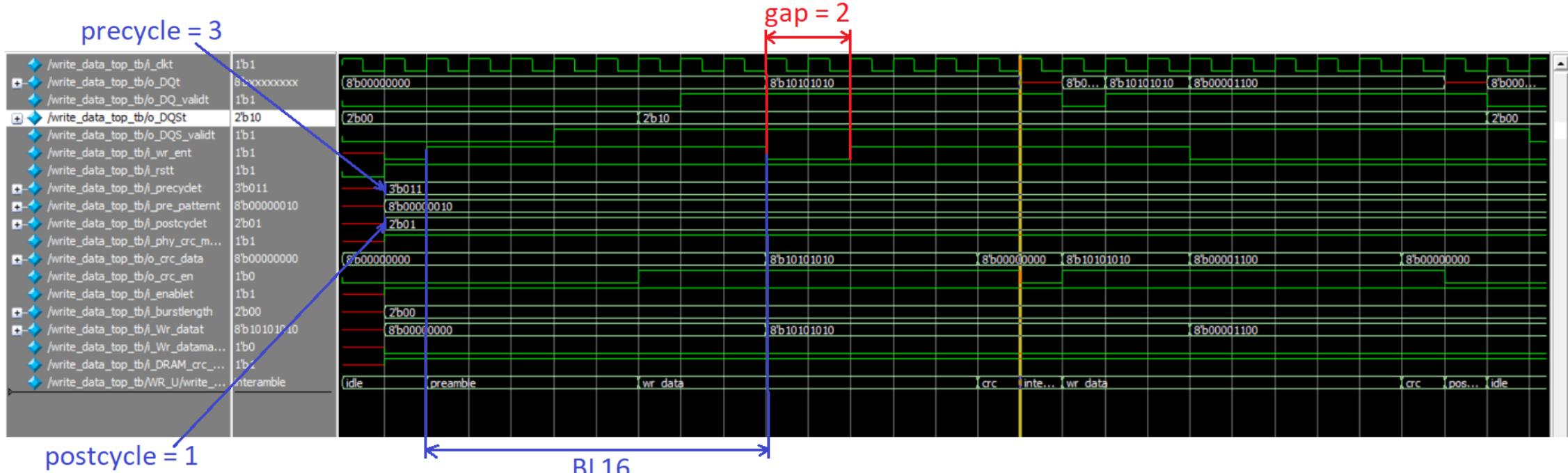
- This counter counts only during inter-amble state and post-amble state.
- Post-amble and inter-amble states cannot occur at same time. This illustrates the reason for using one counter for them both.





Simulation Results

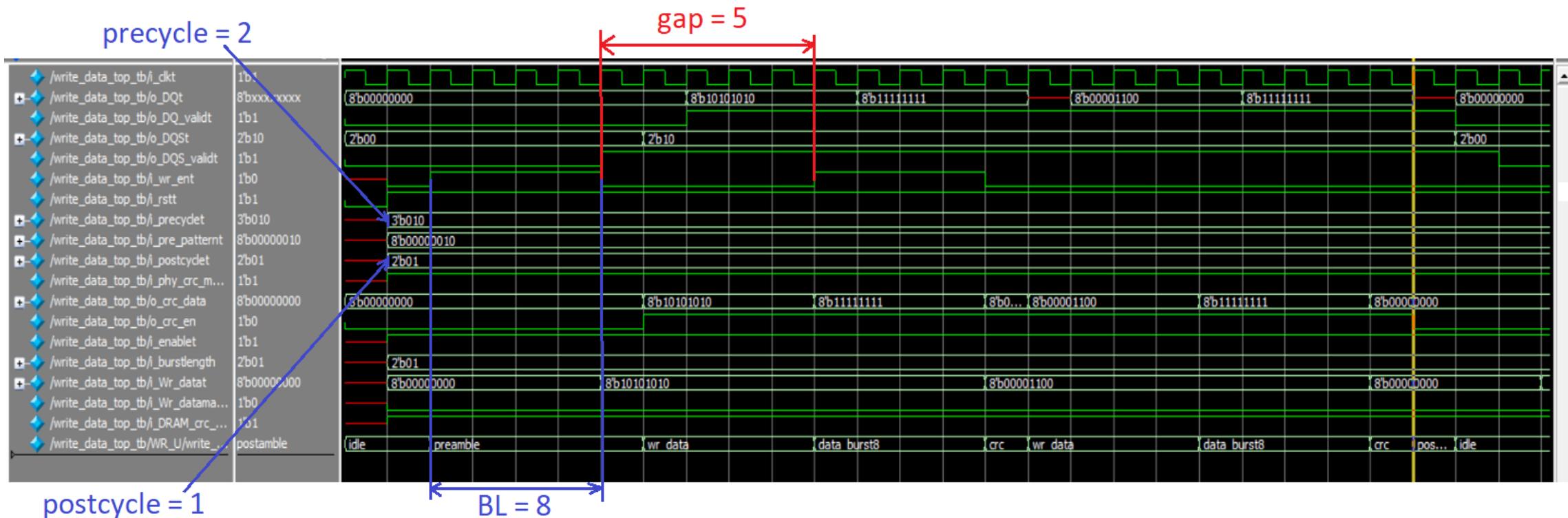
This case is burst length 16 (BL16) with PHY CRC generation.





Simulation Results

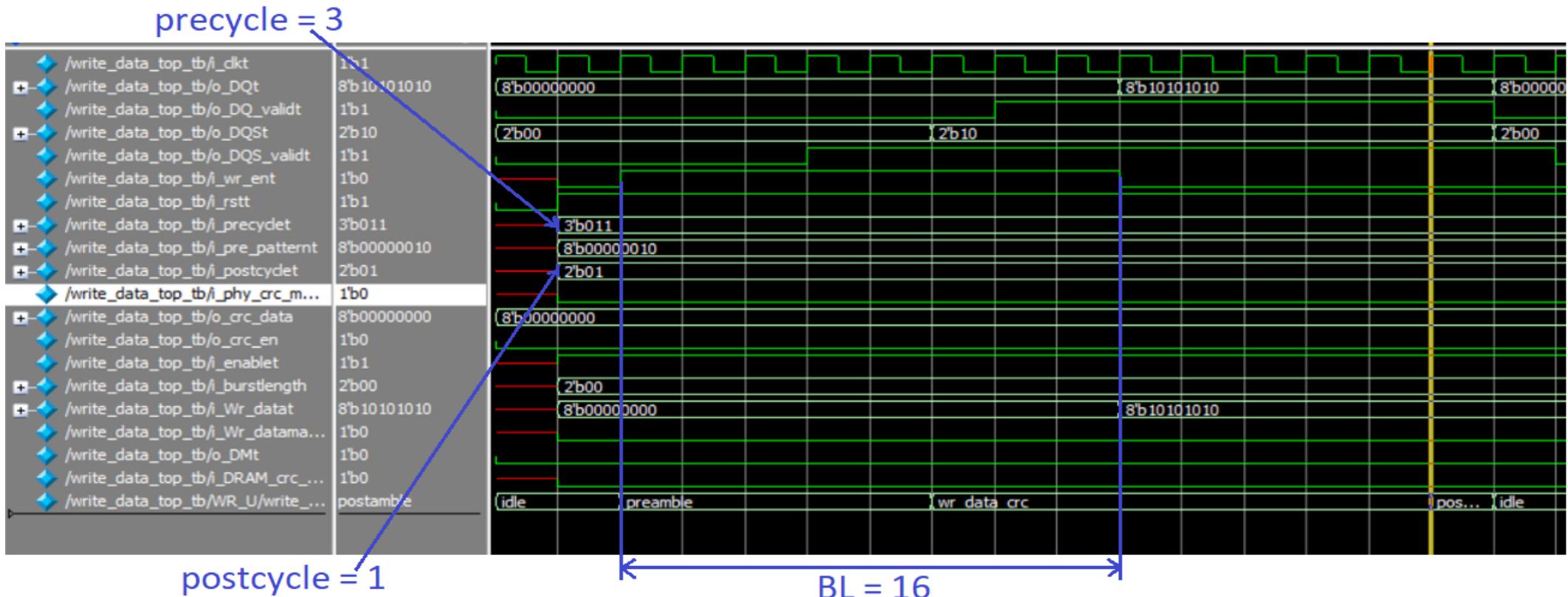
This case is burst length 8 (BL8) with PHY CRC generation.





Simulation Results

This case is burst length 16 (BL16) with Mask operation.



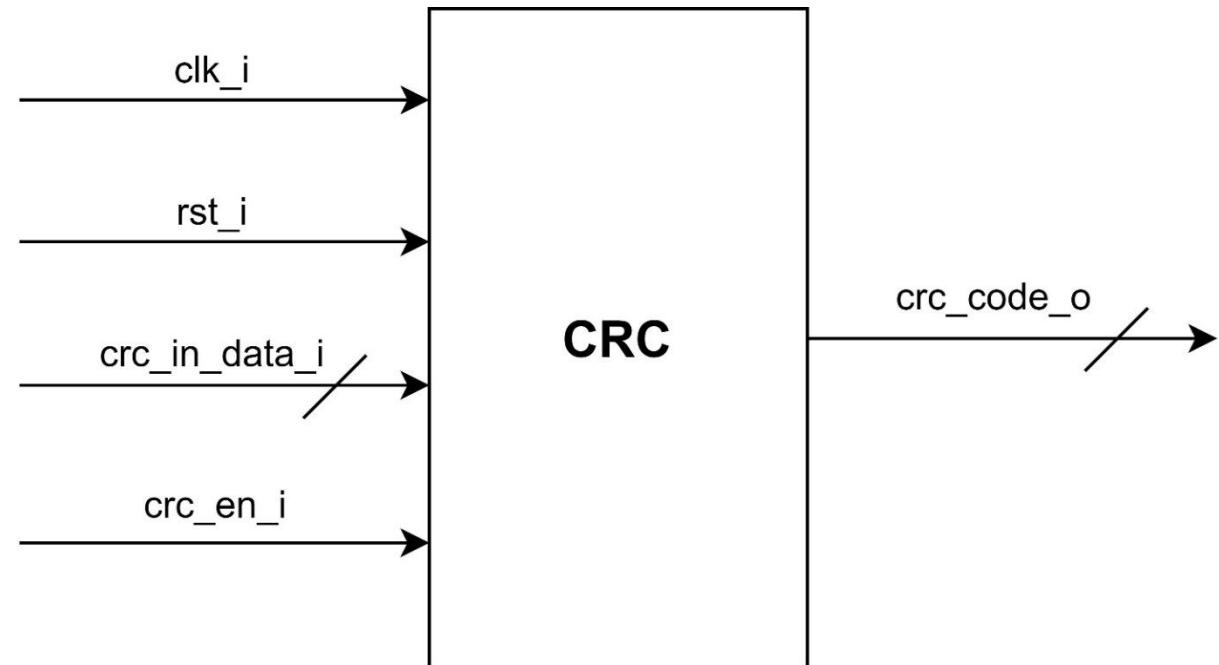


Optimization

	First Approach	Second Approach
Number of cells	629	474
Number of combinational cells	515	408
Number of sequential cells	111	63
Combinational area(μm^2)	6036.47	3427.72
Total cell area(μm^2)	8291.02	5099
Total area(μm^2)	197,011.97	170,580.64
Switching Power(mW)	0.48	0.25
leakage Power(μW)	8.20	3.95
Total Dynamic Power(mW)	1.18	0.55
slack	0.00 (MET)	0.8 (MET)

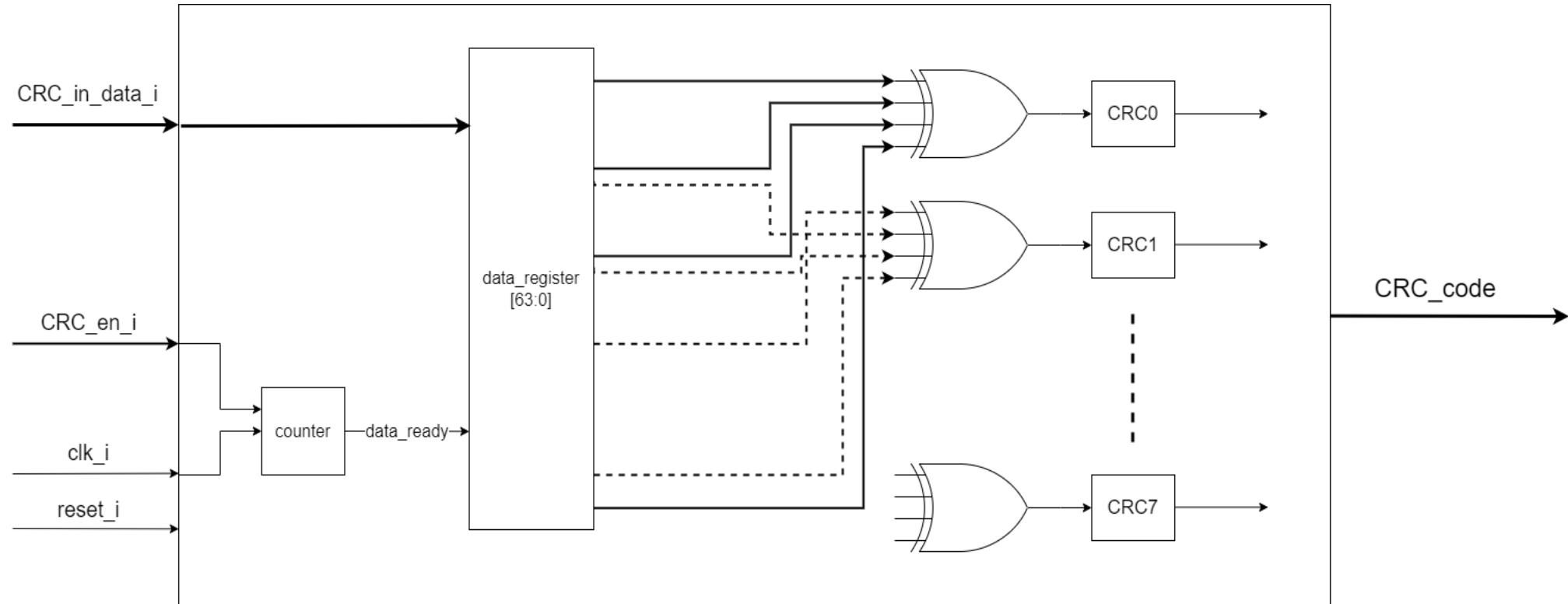


CRC Block Diagram



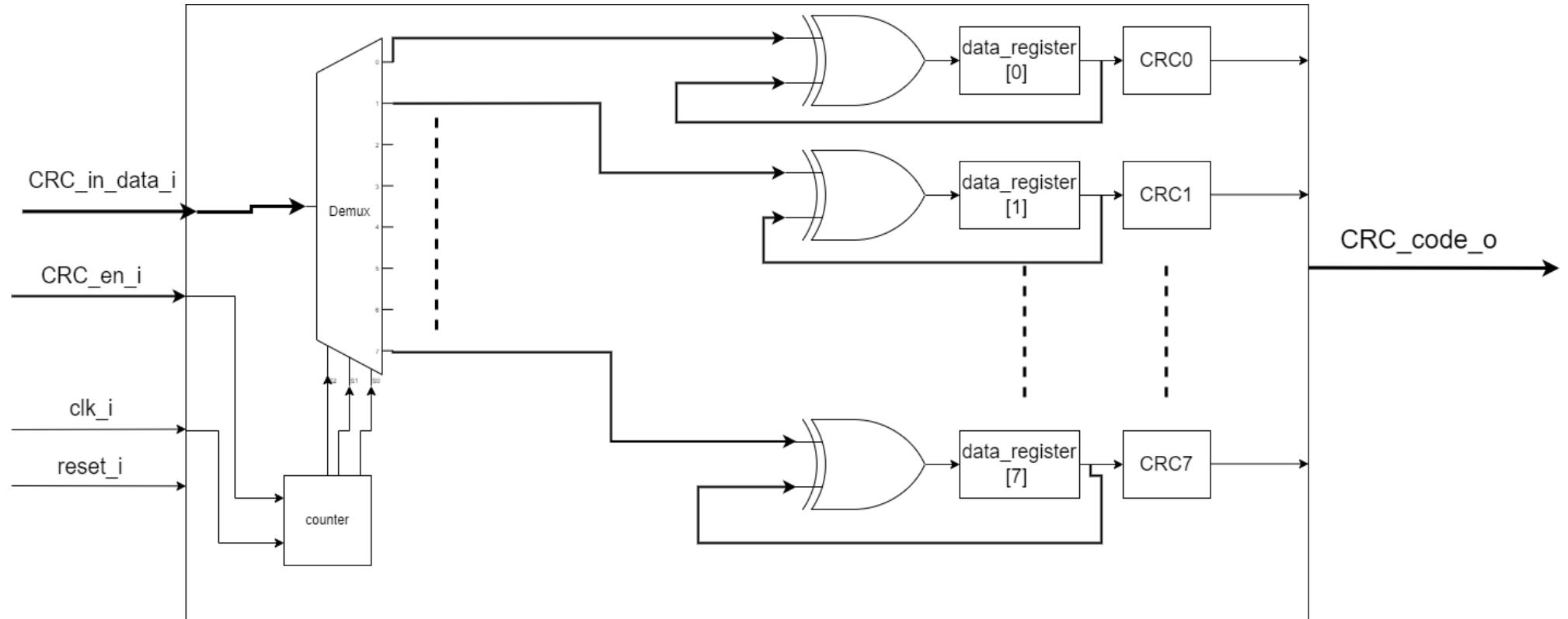


First approach architecture: XOR CRC



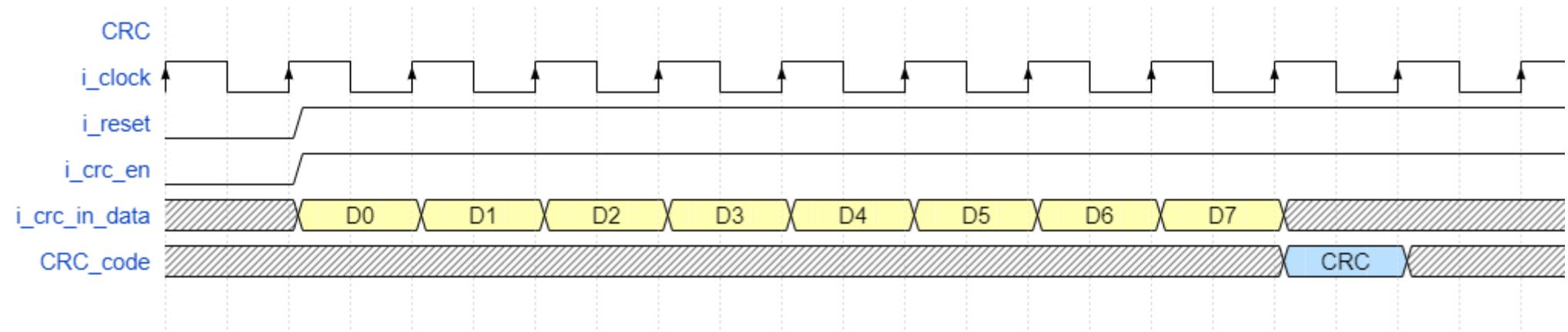


Second approach architecture : Feedback CRC





Timing diagram





Online Calculator

pDRAM_SIZE = 8 (x8), burst length = 16 (BL16)

- The input is: 128'b 1010_1011_1001_1000_1100_1101_0111_0110_1110_1111_0101_0100_1010_1011_0011_0010_1100_1101_0001_0000_1110_1111_1001_1000_0111_0110_0101_0100_0011_0010_0001_0000
- We divided the CRC 16 bits generation into two 8 bits generation

Online CRC Calculation

Be careful: there are several ways to realize a CRC. They differ (at least) in the way which bit is shifted in first and also in the initialization of the flipflops.

Enter your CRC polynomial as bit sequence ("100110001") here:

100000111

This gives the following CRC polynomial (press RETURN to update):

$$P(x) = x^8 + x^2 + x^1 + x^0$$

Enter your message as sequence of hex bytes here. Don't care about whitespaces since they will be ignored.

ABCDEFABCDEF7632

Press RETURN or the Calculate button below to see the CRC checksum here:

\$ 11 (hexadecimal)
% 00010001 (binary, see [calculation details here](#))
! 17 (decimal)

Online CRC Calculation

Be careful: there are several ways to realize a CRC. They differ (at least) in the way which bit is shifted in first and also in the initialization of the flipflops.

Enter your CRC polynomial as bit sequence ("100110001") here:

100000111

This gives the following CRC polynomial (press RETURN to update):

$$P(x) = x^8 + x^2 + x^1 + x^0$$

Enter your message as sequence of hex bytes here. Don't care about whitespaces since they will be ignored.

9876543210985410

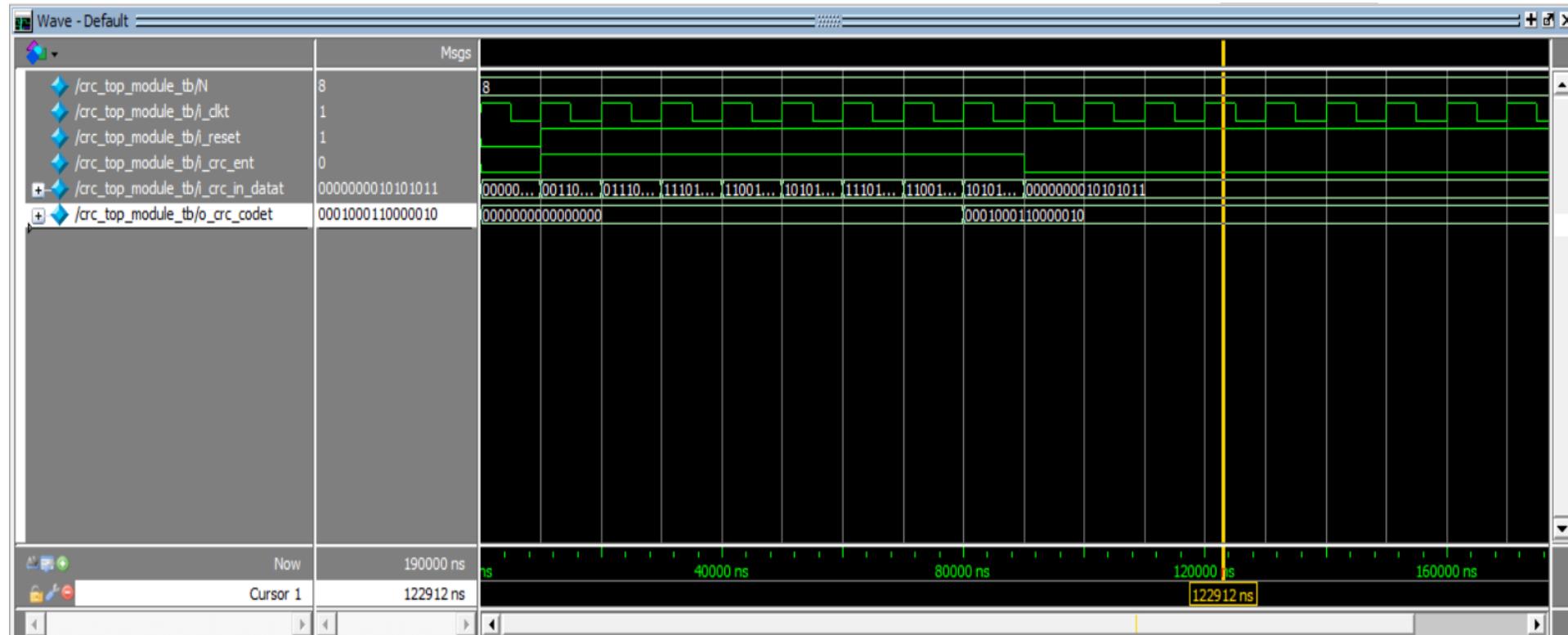
Press RETURN or the Calculate button below to see the CRC checksum here:

\$ 82 (hexadecimal)
% 10000010 (binary, see [calculation details here](#))
! 130 (decimal)



Simulation result

pDRAM_SIZE = 8 (x8), burst length = 16 (BL16)





Optimization

The block is synthesized on Design Compiler (DC) using 130 nm technology.

	First approach	Second approach
Number of cells	1724	1394
Combinational area(μm^2)	22539.68	19474.38
Total cell area(μm^2)	31612.04	21061.75
Total area(μm^2)	556474.56	454643.93
Cell Internal Power(μW)	3.9411	1.8187
Total Dynamic Power(mW)	5.6164	4.9514
data required time	2.63	2.56
slack	-0.26	0.00



DDR5 PHY

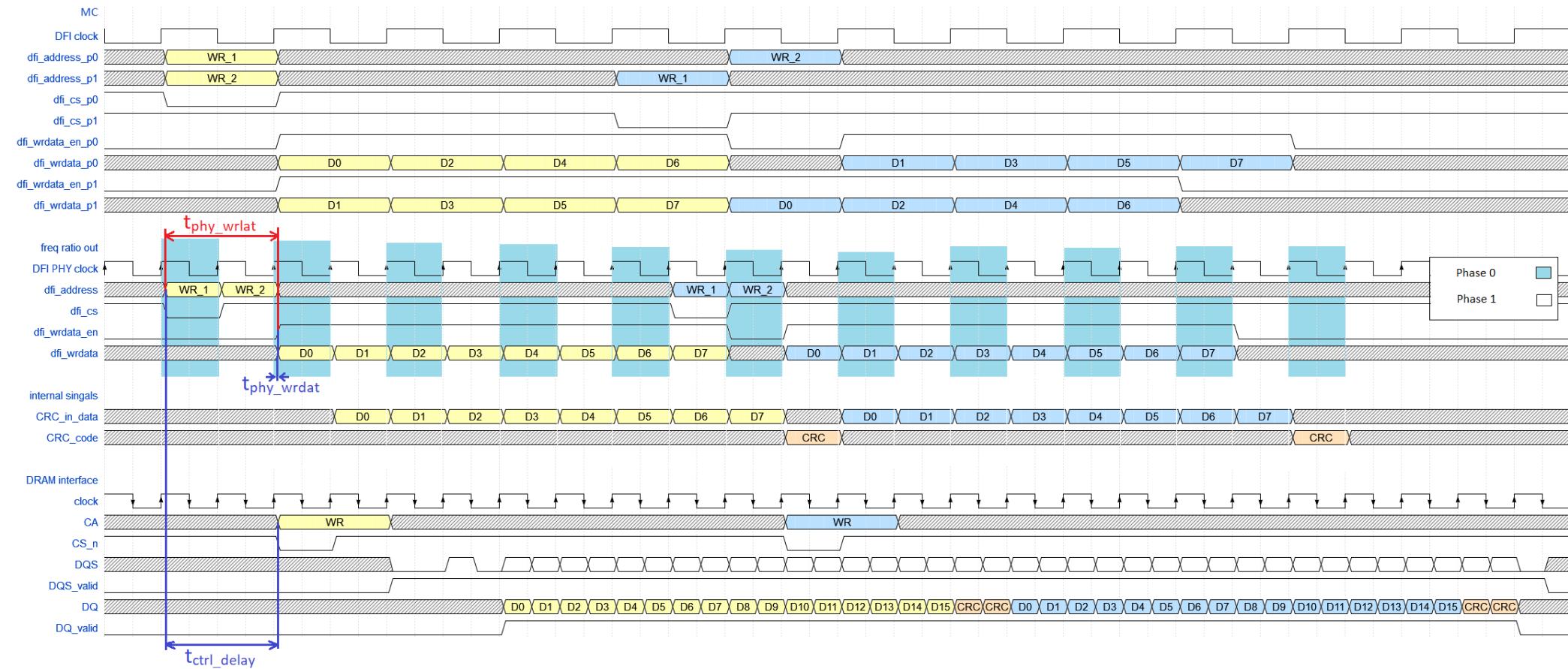
OUTLINE

1. Introduction to Memories.
2. Standard Specifications.
3. DDR5 PHY Blocks.
4. **System Integration.**
 1. Timing Diagram.
 2. Parameters value.
 3. Simulation Result.
5. FPGA flow.



Timing Diagram

- Timing diagram of two back to back write operation with frequency ratio of 1:2 and burst length 16 (BL16) with CRC generated from PHY.





Timing parameters

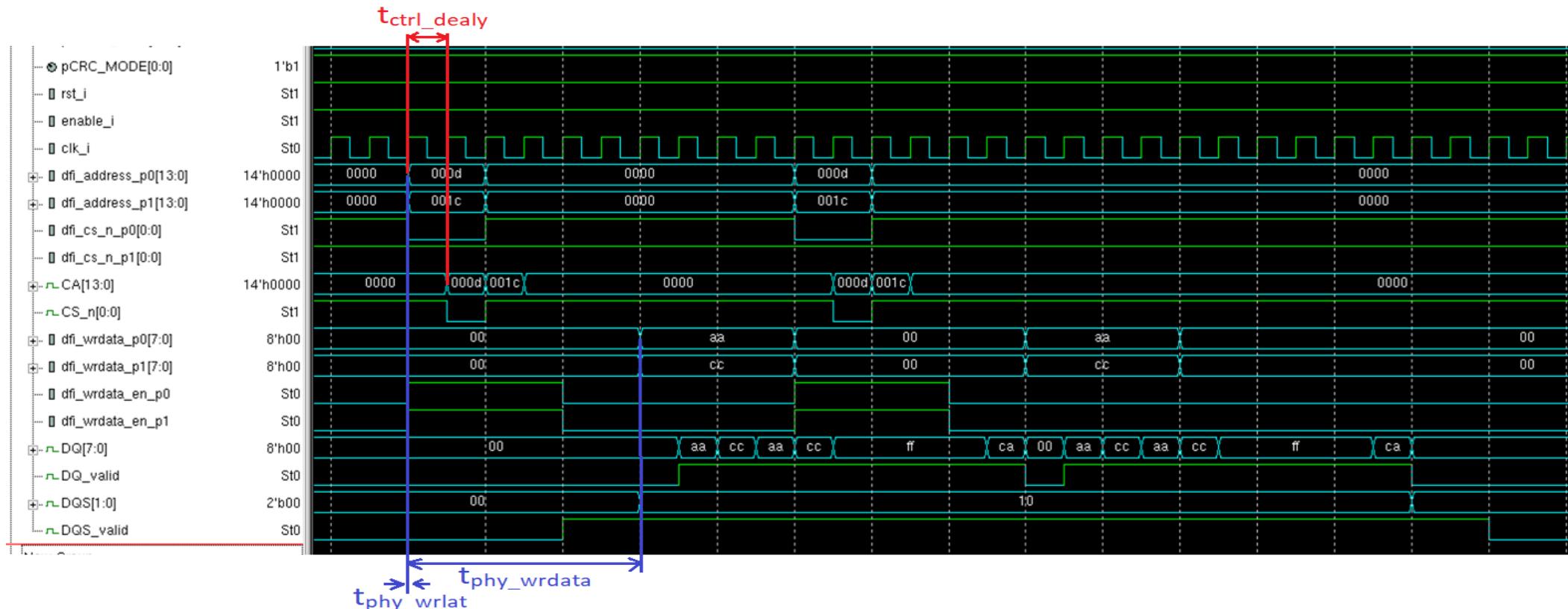
Timing parameter	Value (clk)	Min/Max (clk)	Controlled by	Description
t_{ctrl_delay}	2	0/- ^a	Frequency ratio block and command address block	Each block registering the outputs.
t_{phy_wrlat}	0	0/- ^a	Command address block	The block doesn't need this delay to handle the command
t_{phy_wrdata}	6	0/- ^a	Write data block	We need this delay to calculate the inter-amble correctly

- a. The minimum supportable value is 0; the DFI does not specify a maximum value. The range of values supported is implementation-specific.



Simulation Result

- Frequency ration 1:2 ,Burst length = 8 ,phy CRC support, with preamble: 3 CLK : “000010” , and post-amble : “0000” inter-amble : “10”



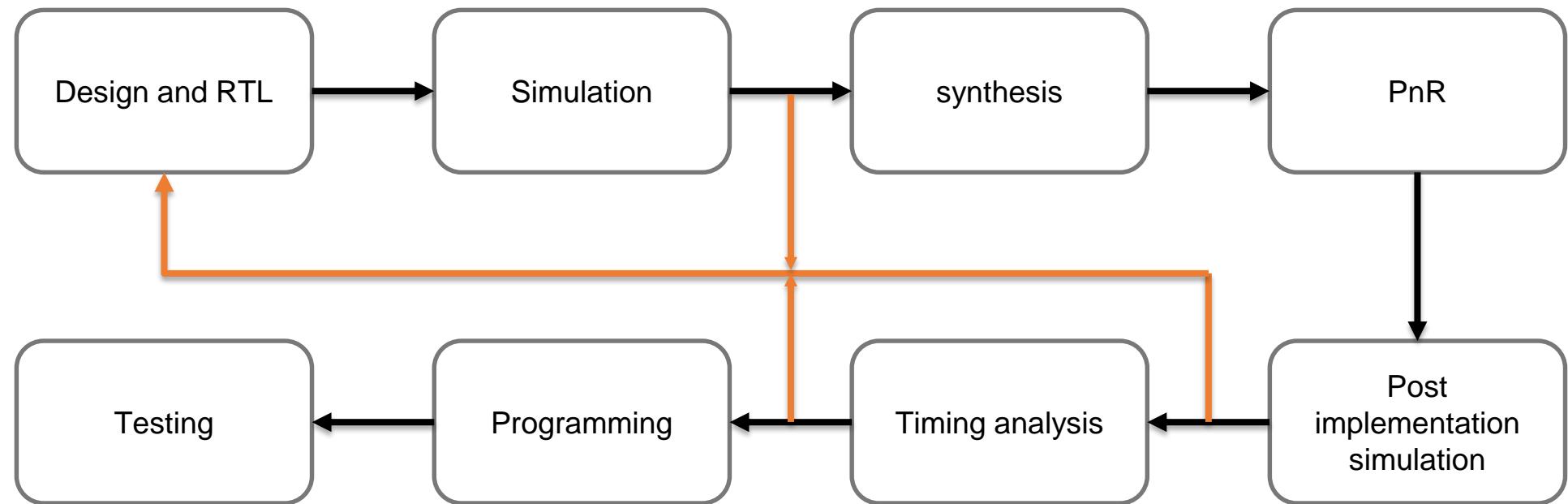


DDR5 PHY

OUTLINE

1. Introduction to Memories.
2. Standard Specifications.
3. DDR5 PHY Blocks.
4. System Integration.
5. FPGA flow.

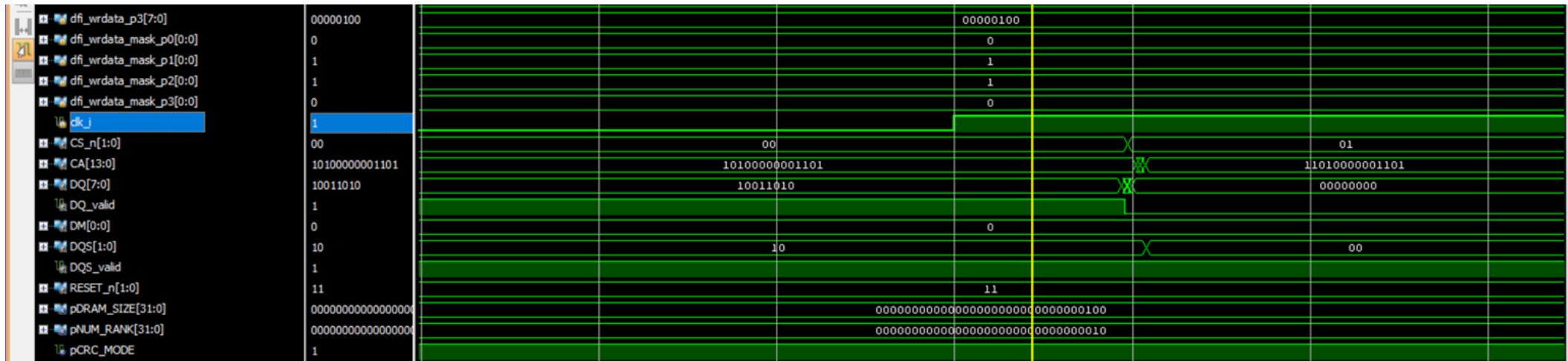
FPGA design flow





Post implementation simulation (zoomed)

- The output changes after rising edge by 10 ps.





Conclusion

- We have studied different versions of DDR and we started reading the standards (DFI and JEDEC).
- We extracted the required specification to implement the write operation and CRC operation and proposed a design for the PHY system.
- The RTL of each block was implemented using System Verilog (SV) and each block was tested using ModelSim and VCS.
- The system was synthesized on DC compiler and the area, power and maximum timing were reported.
- We have gone through FPGA design flow.



Future work

- generating verification environment.
- performing design for testability (DFT) for our system.
- going through PnR flow till generating GDSII files.



References

1. DDR PHY Interface DFI 5.0 Specification.
2. JEDEC STANDARD JESD79-5A.
3. Random Access Memory. Youtube channel “Computer science”. Last accessed 27th June 2022: <https://www.youtube.com/playlist?list=PLTd6ceoshpreExQfQ-akUMU1sEtthFdB>
4. DDR-PHY Interoperability Using DFI. Synopsys. Last accessed 27th June 2022: <https://blogs.synopsys.com/vip-central/2016/09/06/ddr-phy-interoperability-using-dfi/>
5. Classification of Memory. JavaTpoint. Last accessed 23th Nov 2021: <https://www.javatpoint.com/classification-of-memory>
6. DRAM Operation: how does dynamic RAM work. Electronics notes. Last accessed 11th Nov 2021: https://www.electronics-notes.com/articles/electronic_components/semiconductor-ic-memory/dynamic-ram-how-does-dram-work-operation.php
7. Linda Rosencrance. DIMM (dual in-line memory module). TechTarget. Last accessed 18th Nov 2021: <https://www.techtarget.com/searchstorage/definition/DIMM>

THANK YOU...

