EE5313 Microprocessor Systems

THE UNIVERSITY OF TEXAS
AT ARLINGTON

SDRAM Controller Design

By,

Samyam Santosh 1001994596

Pranav Alle 1002058154

Under the Guidance of,

Instructor: Dr. Shahriar Shahabuddin

INDEX

AIM:

The aim of the project is to design an SDRAM controller which allows SDRAM memory to be interfaced with microprocessor (80386Dx) having synchronous memory support. This report shows the complete design and theory of operations.

COMPONENTS REQUIRED AND SPECIFICATIONS:

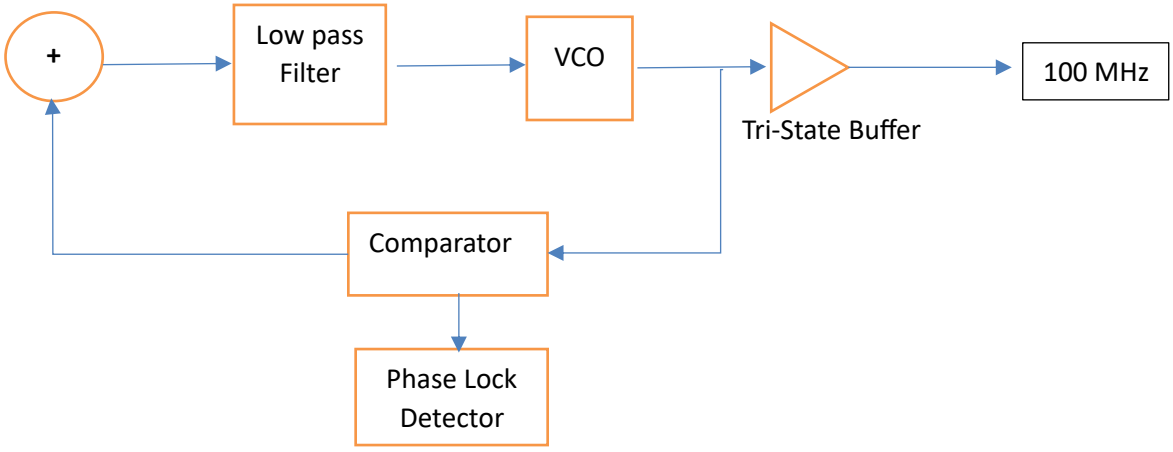
- SDRAM Controller
- SDRAM
  MT48LC16M4A2 – 4 Meg x 4 x 4 banks
  4096 rows x 1024 columns x 4 banks
  Operating Frequency = 100 MHz
  Clock cycle time Tclcl: 10 ns
  Speed Grade: -75 with access time: 5.4 ns
  Setup time: 1.5 ns and hold time: 0.8 ns.
  Burst length: 8
  CAS Latency = 2
  Refresh time: 64 ms for 4096 rows
- 80386DX processor
  32-bit address line and 32-bit data line.

INTERFACING:

- Clk (66 MHz) from the microprocessor is given to the controller.
- Clk for the SDRAM is 100 MHz this is attained by passing Clk2 to phase locked loop.
- 8086Dx microprocessor consists of 32-bit address lines in which A0 and A1 act as bank enable signals for BE0 to BE3.
- 32-bit data bus D0-D31 of 8086Dx microprocessor is meant for data transfer and receive from SDRAM device.
- Ready signal is given to the microprocessor from the controller to indicate the termination of current CPU bus cycle information.
- HOLD is also given to processor from controller.
- HLDA is hold acknowledgement is given to controller form 8086Dx processor.
- ADS, RD, WT signals generate read and write signals for controllers.
- CAS, RAS, WE are the signals given to the SDRAM for memory read and write operations.
- CKE is a clock enabled signal used to enable the operations in the SDRAM.
- VDD is supplied to all devices.
- VSS is the ground.

CLOCK GENERATION:

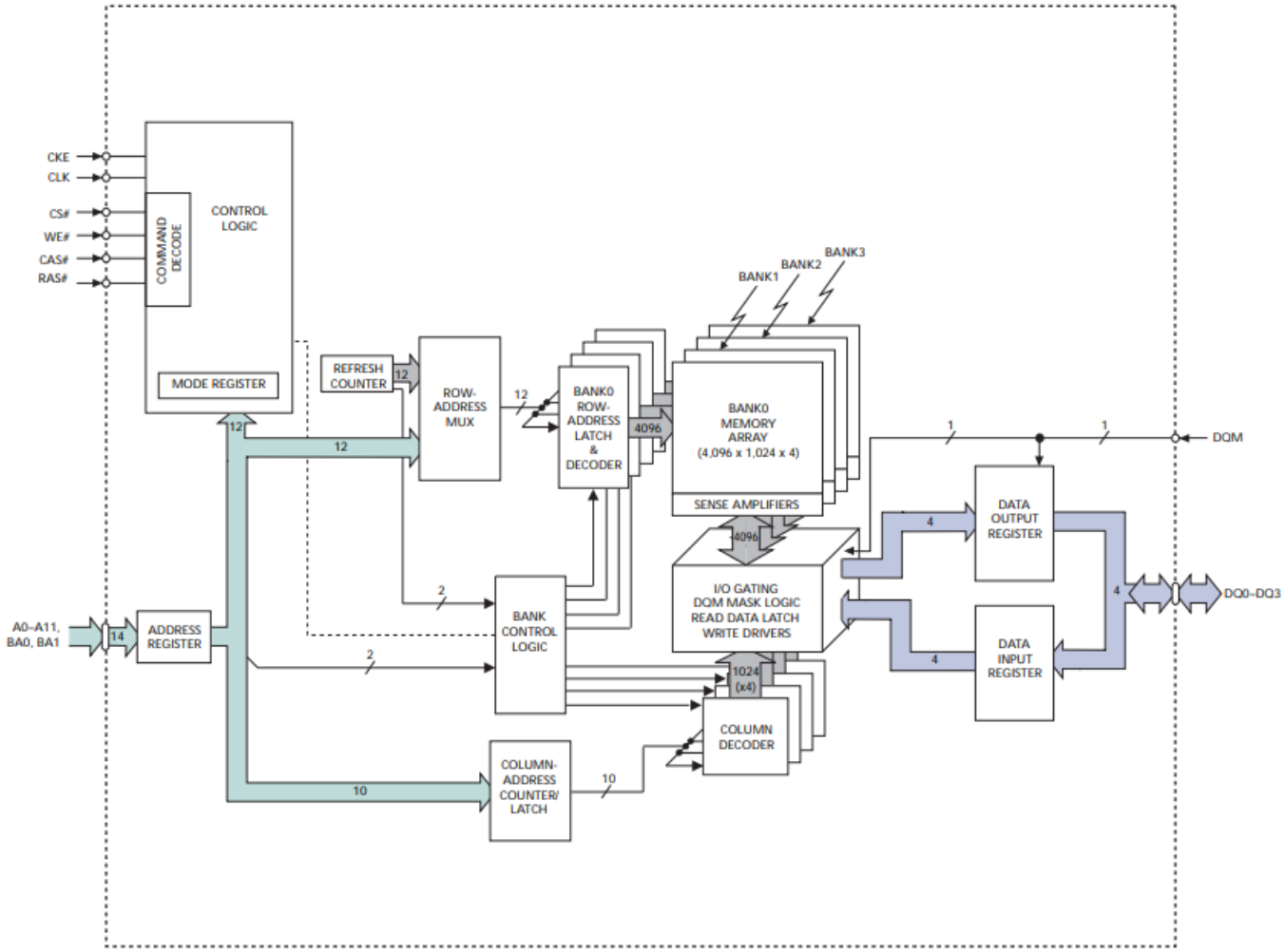
- The CLK2 of 8086Dx processor is given as an input clock with 66MHz clock frequency.



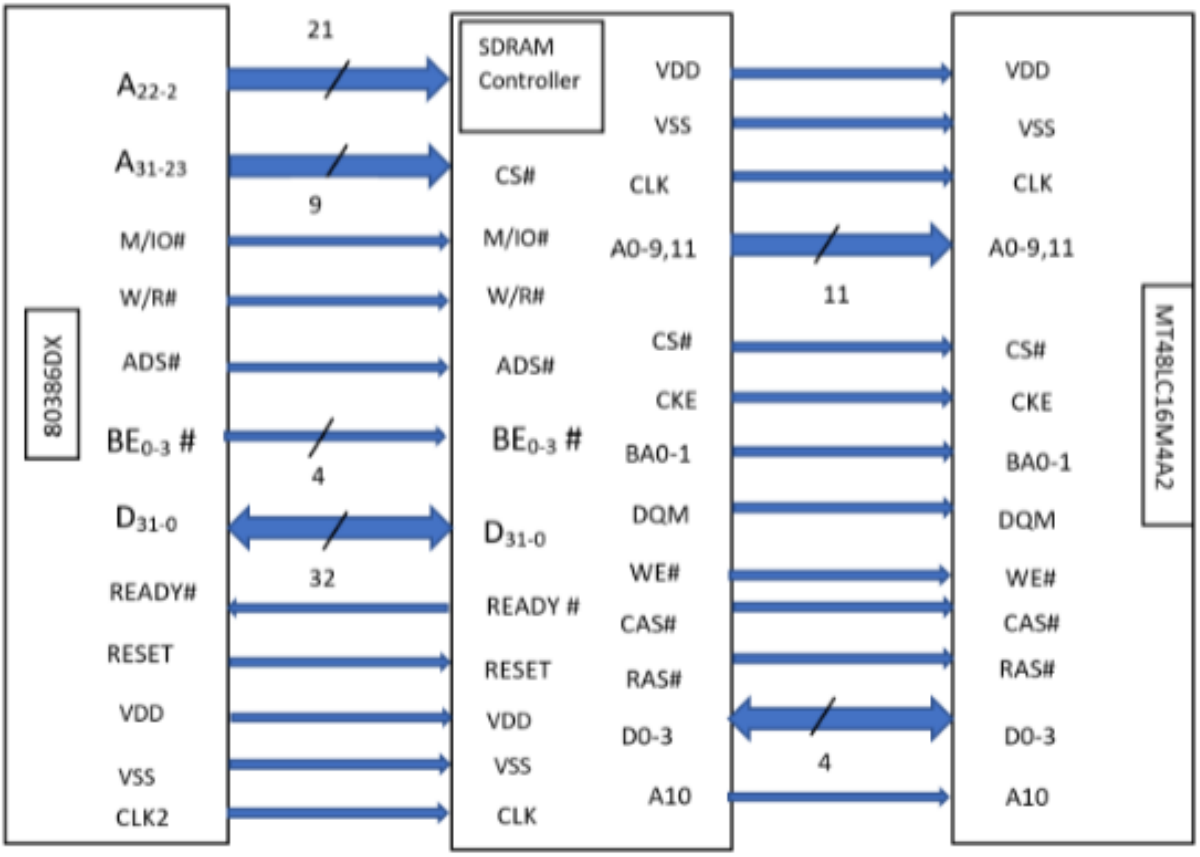
- To convert Clk frequency from 66 MHz to 100 MHz we are using Phase Locked Loop (PLL).
- PLL consists of lowpass filter, Voltage controlled Oscillator (VOC), Phase Lock Detector, Tri-State Buffer, and a divide by 2 phase comparators.
- This is used to generate output signal whose phase is related to the phase of an input signal.
- The 66 MHz from processor is sent to PLL unit to obtain the desired frequency of 100 MHz.
- When 100 MHz frequency is attained, it is given as an input to the controller and SDRAM.

FUNCTIONAL BLOCK DIAGRAM:

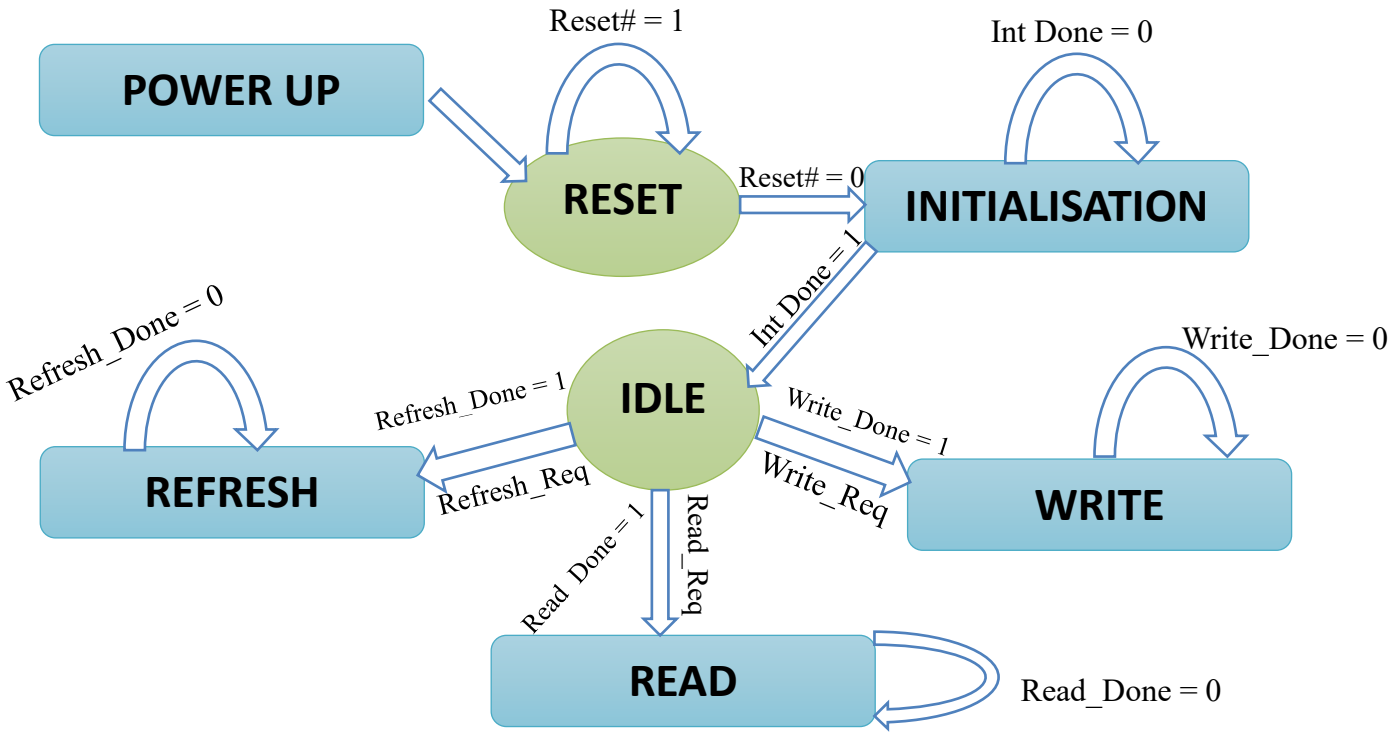
Figure 1: 16 Meg x 4 SDRAM



IMPLEMENTATION:



TOP LEVEL VIEW OF FINITE STATE MACHINE:



The SDRAM Controller FSM consists of main state diagram. It has 4 sub-state machines INITIALIZATION, READ, WRITE and REFRESH.

TOP LEVEL STATE TRANSITION TABLE:

Current State	Conditions	Next State
Power Up	--	Reset
Reset	Reset# = 1	Reset
Reset	Reset# = 0	Initialization
Initialization	Int_Done = 0	Initialization
Initialization	Int_Done = 1	Idle
Idle	Read_Request	Read
Read	Read_Done = 0	Read
Read	Read_Done = 1	Idle
Idle	Write_Request	Write
Write	Write_Done = 0	Write
Write	Write_Done = 1	Idle
Idle	Refresh_Request	Refresh
Refresh	Refresh_Done = 0	Refresh
Refresh	Refresh_Done = 1	Idle

MEMORY ADDRESS MAPPING:

DCD_SDRAM	BA [1:0]	ROW	COL
A31-23	A22-21	A20-9	A8-2
9bits	2bits	12bits	7bits

INITIALIZATION:

When powering up and initializing SDRAMs, it's important to follow the specified procedures, as other procedures may result in unpredictable behavior. After applying power and ensuring a stable clock signal, there should be a delay of at least 100 microseconds before issuing any command other than a NOP. During this 100 microsecond period, NOP commands should be continuously applied. Once this delay has passed and at least one NOP command has been applied, a PRECHARGE command should be issued.

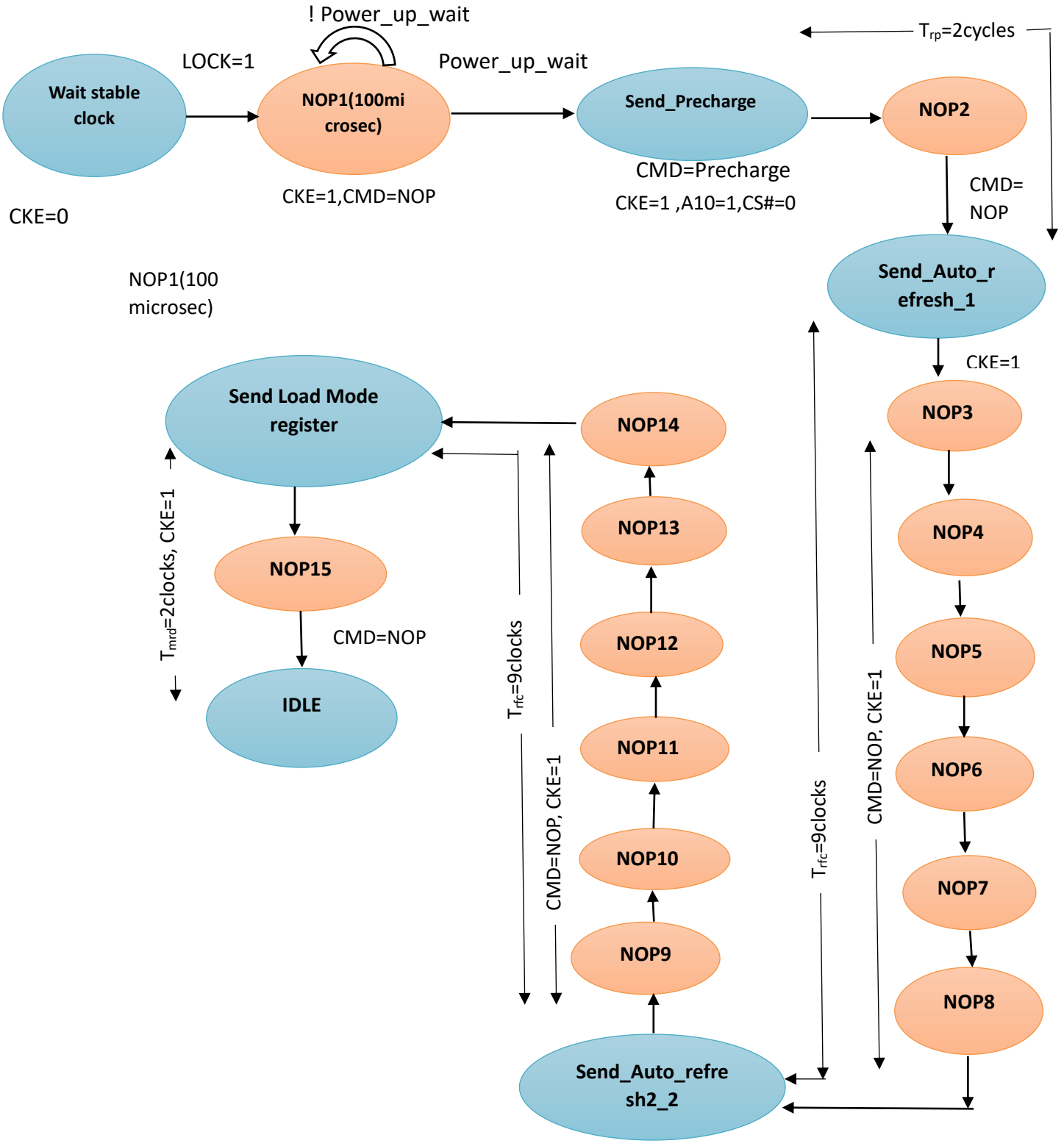
After issuing the PRECHARGE command, it's necessary to wait for at least tRP time. During this time, a NOP2 command should be given. This ensures that all banks complete their precharge operations. After the precharge operation is complete, the SDRAM should move to the send_auto_refresh command. During this time, it should wait for at least tRFC time, which is equivalent to 7 clock cycles. NOP3 through NOP8 commands should be given during this waiting period.

After this waiting period, at least two Auto refresh cycles must be performed. During this time, NOP3 through NOP14 commands should be given.

After completing the auto refresh cycles, the SDRAM is ready for mode register programming. Since the mode register starts up in an unknown state, it should be programmed with the desired bit values before any operational command is applied. This can be done using the LOAD MODE REGISTER command to program the mode register.

After programming the mode register, a waiting period of at least tMRD time is necessary. In this case, as tMRD is 2 clocks, a NOP15 command should be applied during this waiting period. Once this waiting period is over, the SDRAM is ready for any valid command.

STATE MACHINE INITIALIZATION:

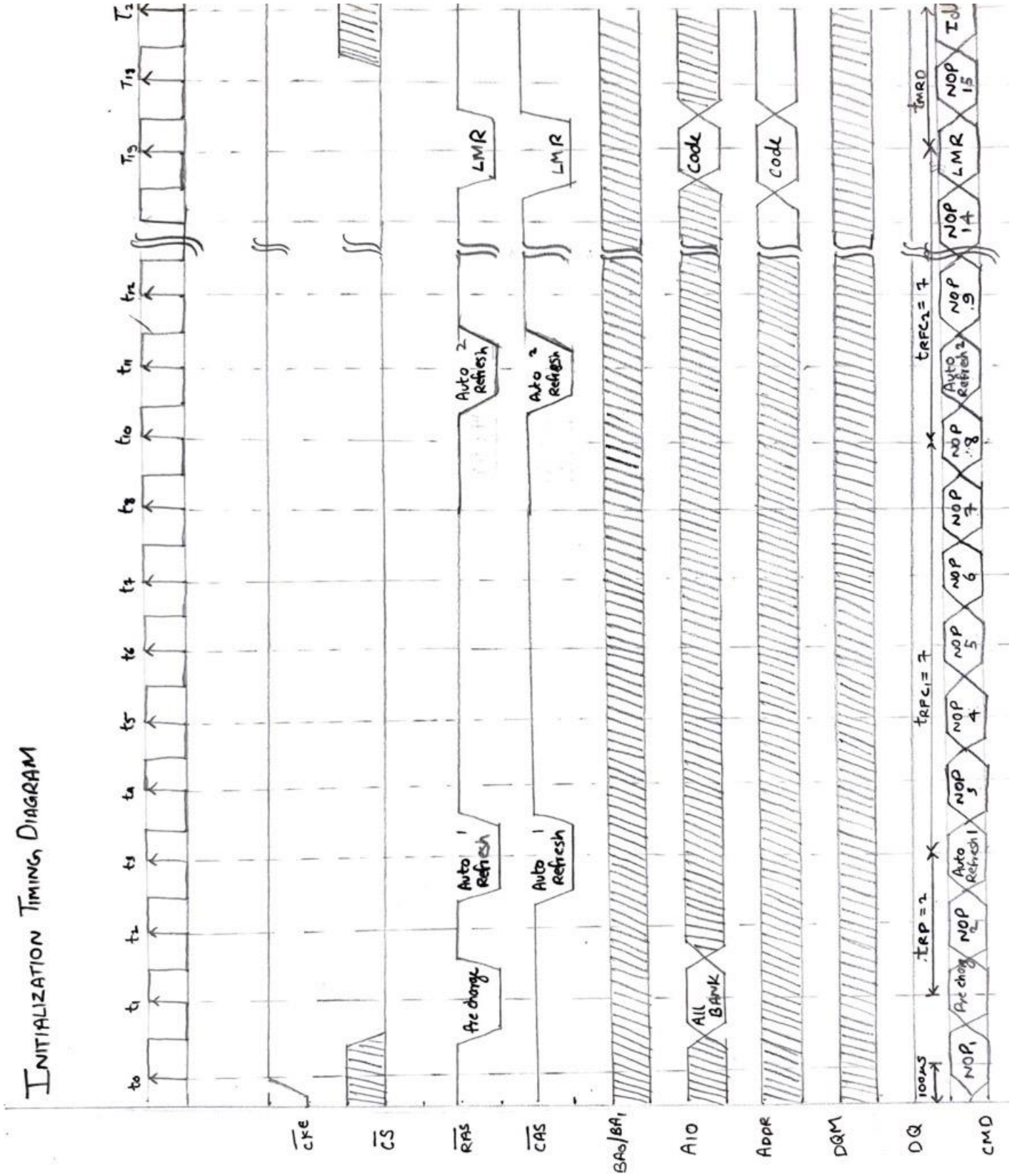


STATE TRANSITION TABLE FOR INITIALIZATION:

Current State	Condition	Next State
NA	Reset#=0	Reset
Reset	Next Clock, CKE=0	Wait Stable Clock
Wait Stable Clock	Lock=1	NOP1(100microsec)
NOP1	Power_up_wait! =0	NOP1
NOP1	Power_up_wait=0, CKE=1	Send_Precharge
Send_Precharge	--	NOP2
NOP2	--	Send_Auto refresh 1
Send-Auto refresh 1	--	NOP3
NOP3	--	NOP4
NOP4	--	NOP5
NOP5	--	NOP6
NOP6	--	NOP7
NOP8	--	Send_Auto refresh2
Send-Auto refresh2	--	NOP9
NOP9	--	NOP10
NOP10	--	NOP11
NOP11	--	NOP12
NOP12	--	NOP13
NOP13	--	NOP14
NOP14	--	Send Load Mode register
Send Load Mode register	--	NOP15
NOP15	--	IDLE

Clock Calculations	Value(ns)	No. of clocks
NOP1 (100 microsecond)	100 microsecond	83,000
Precharge Period (Trp)	20ns	2 Clocks
Auto Refresh Period (Trfc)	66ns	7 Clocks
Load Mode Register Period (Tmrd)	20ns	2 Clocks

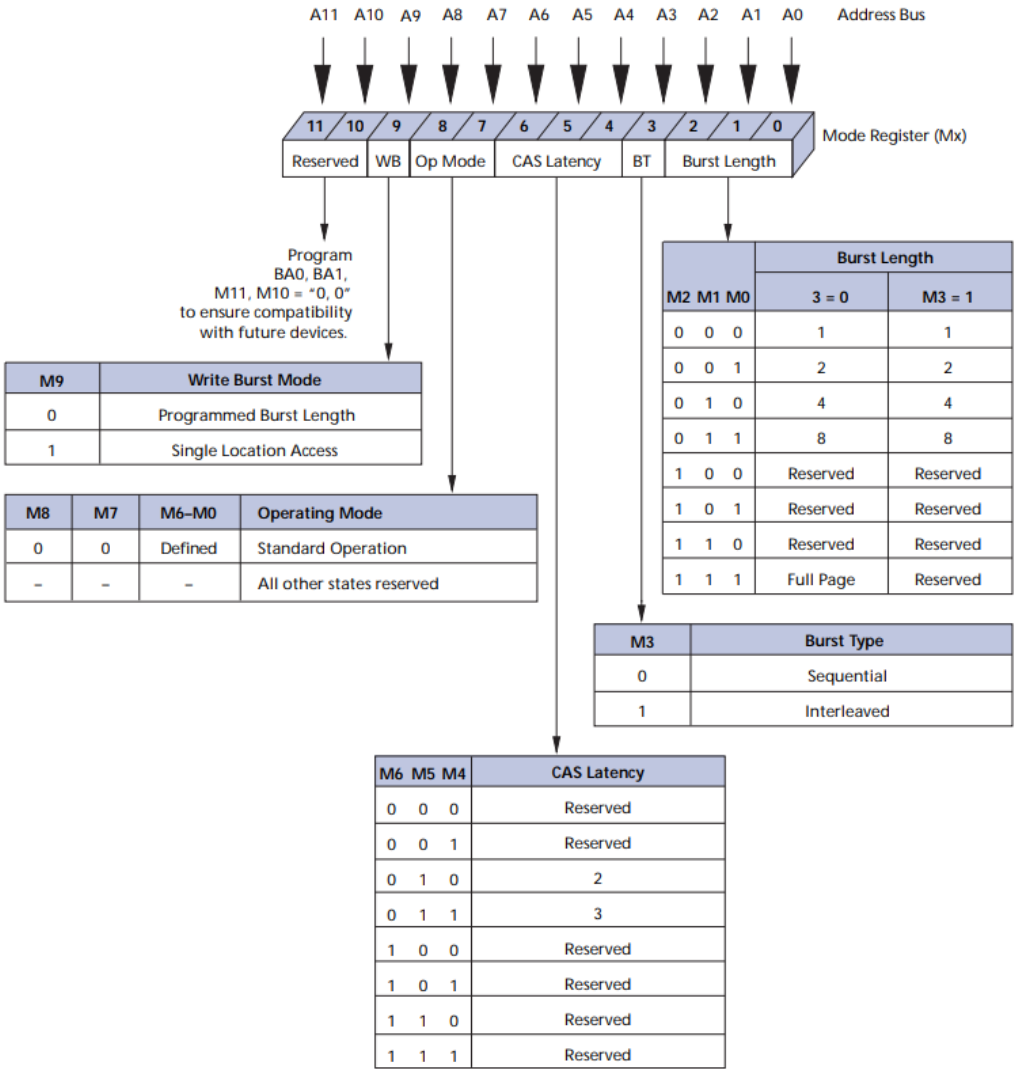
TIMING DIAGRAM:



LOAD MODE REGISTER:

According to this diagram from the data sheet, Load Mode Register is set according to our specifications.

Figure 6: Mode Register Definition



According to our specifications, LMR for initialization is:

Function	Function Value	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
Burst length (BL)	8										0	1	1
Burst Type	Sequential									0			
CAS Latency	2						0	1	0				
Op mode	Standard operation				0	0							
Write Burst Mode	Programmed Burst Mode			0									
Reserved	--	0	0										

LMR: 00 0 00 010 0 011

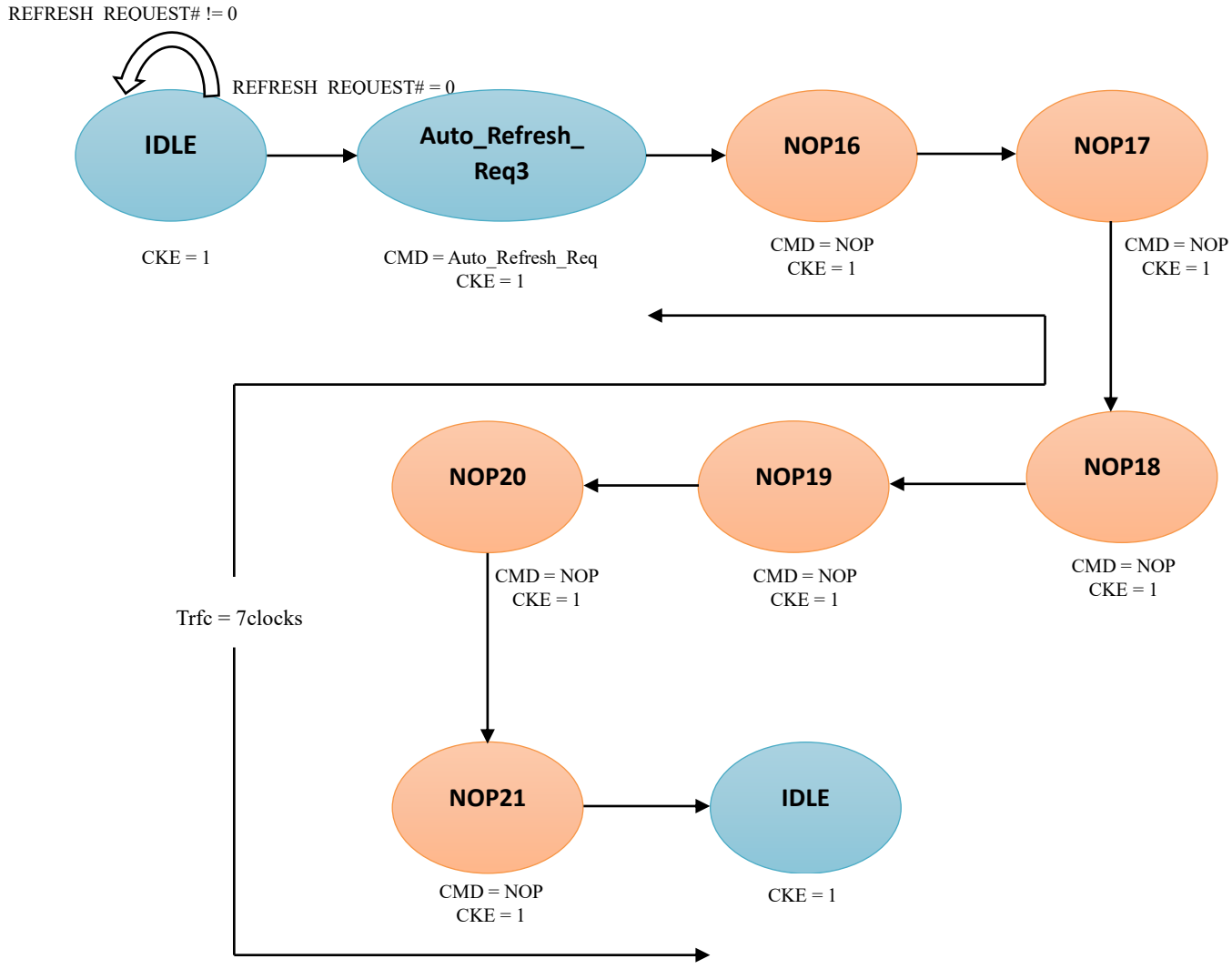
AUTO REFRESH:

During regular operation of the SDRAM, AUTO REFRESH is used to refresh the memory and is similar to the CAS#-BEFORE-RAS# (CBR) refresh in traditional DRAMs. This command is not saved permanently in the memory and must be issued every time a refresh is needed. Prior to issuing an AUTO REFRESH command, all active banks must be PRECHARGED.

It's important to wait for at least the minimum tRP time after the PRECHARGE command before issuing an AUTO REFRESH command, as specified in the Operation section.

During an AUTO REFRESH command, the internal refresh controller generates the addressing, which makes the address bits "Don't Care". For the 64Mb SDRAM, regardless of its width, it's necessary to perform 4,096 AUTO REFRESH cycles every 64ms. To meet this refresh requirement and ensure that each row is refreshed, a distributed AUTO REFRESH command can be provided every 15.625µs. This is done by REFRESH_REQUEST# command. When the signal REFRESH_REQUEST# goes low, the SDRAM enters the Auto_Refresh_Req state. During this state, it goes through NOP16-22 cycles to cover the tRFC time, which is equivalent to 7 clocks. Once this waiting period is over, the SDRAM returns to the idle state.

STATE MACHINE FOR AUTO REFRESH:

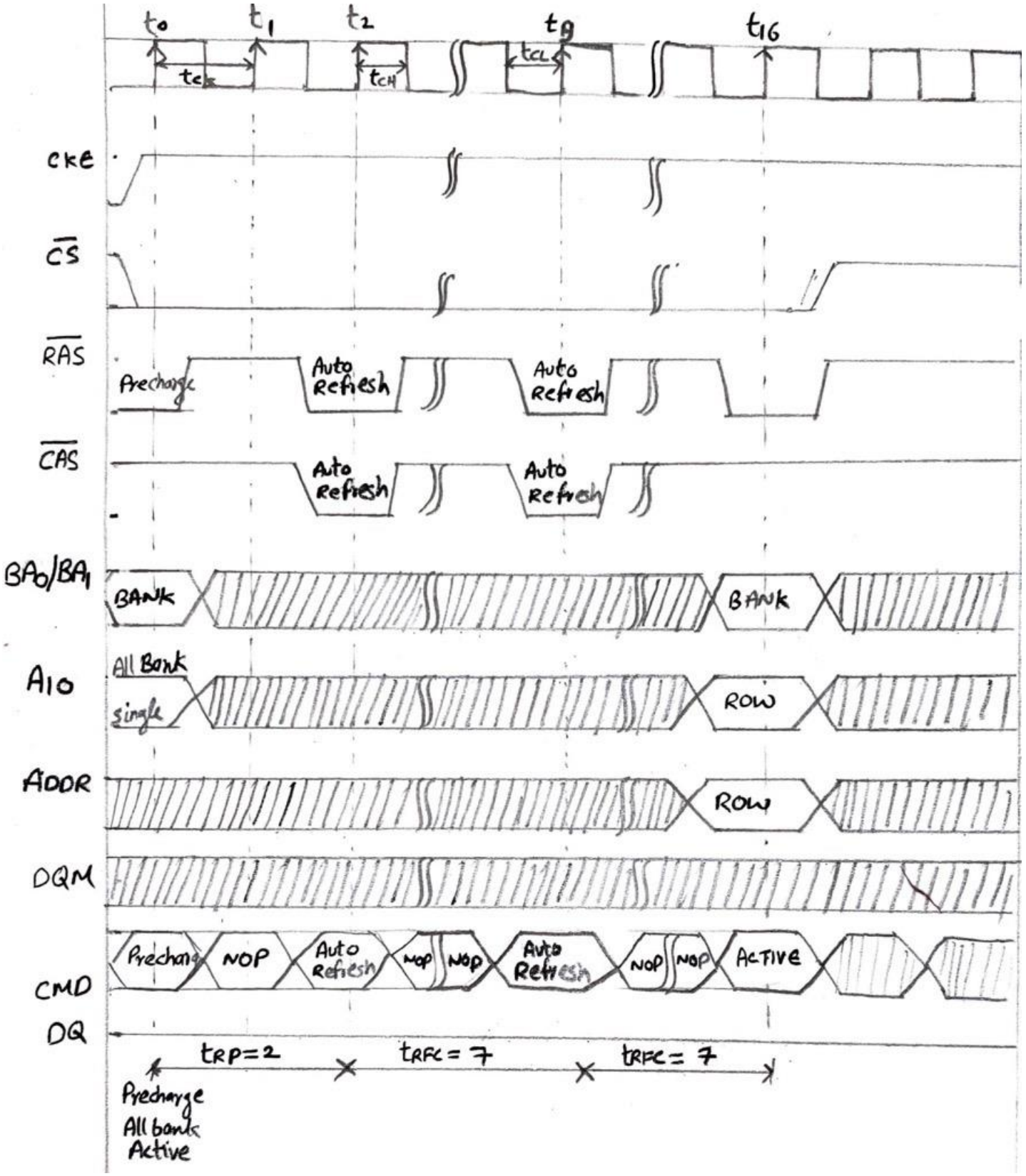


STATE TRANSITION TABLE:

Current State	Condition	Next State
IDLE	Refresh_Request! = 0	IDLE
IDLE	Refresh_Request = 0	Auto_Refresh_Req
Auto_Refresh_Req	--	NOP16
NOP16	--	NOP17
NOP17	--	NOP18
NOP18	--	NOP19
NOP19	--	NOP20
NOP20	--	NOP21
NOP21	--	IDLE

Clock Calculations	Value(ns)	No. of clocks
Precharge Period (Trp)	20ns	2 Clocks
Auto Refresh Period (Trfc)	66ns	7 Clocks

TIMING DIAGRAM:



READ:

Calculations for READ:

Read Cycle Calculations :-

Time Period = 1 sec

Clock Frequency = 100 MHz

Clock Variant = -75

$T_{CLK} = \frac{1}{100 \times 10^6} = 10\,ns$

1) Auto-refresh period for -75 = 66 ns

2) Rows required to refresh = 4096 ie; 2^{12}

3) Refresh time for all rows at once = $4096 \times 66\,ns$
$= 270.336\,\mu s$

4) Refresh period of 4096 rows in -75 is 64 ms

All rows to be refreshed in 1sec = $\frac{1}{64 \times 10^{-3}} = 15.625 = 16$ times

5) Time consumed for refresh in 1sec = $16 \times 270.336\,\mu s$
$= 4.325$ ms

6) Time remaining for read cycle = 1 − Time consumed for refresh
$= 1 - 4.325 \times 10^{-3}\,s$
$= 0.996$ sec
$= 996$ ms.

7) t_{RCD} (Row to Column Delay) for -75 = 20 ns

Number of clocks required = $\frac{t_{RCD}}{T_{CLK}} = \frac{20}{10}\,ns = 2$ clocks

Considering CL = 2 and BL = 8

Number of clocks required for read cycle = $t_{RCD} + CL + BL$
$= 2 + 2 + 8$
$= 12$ clocks

$T_{RC} = 12 \times T_{CLK} = 12 \times 10 = 120\,ns.$

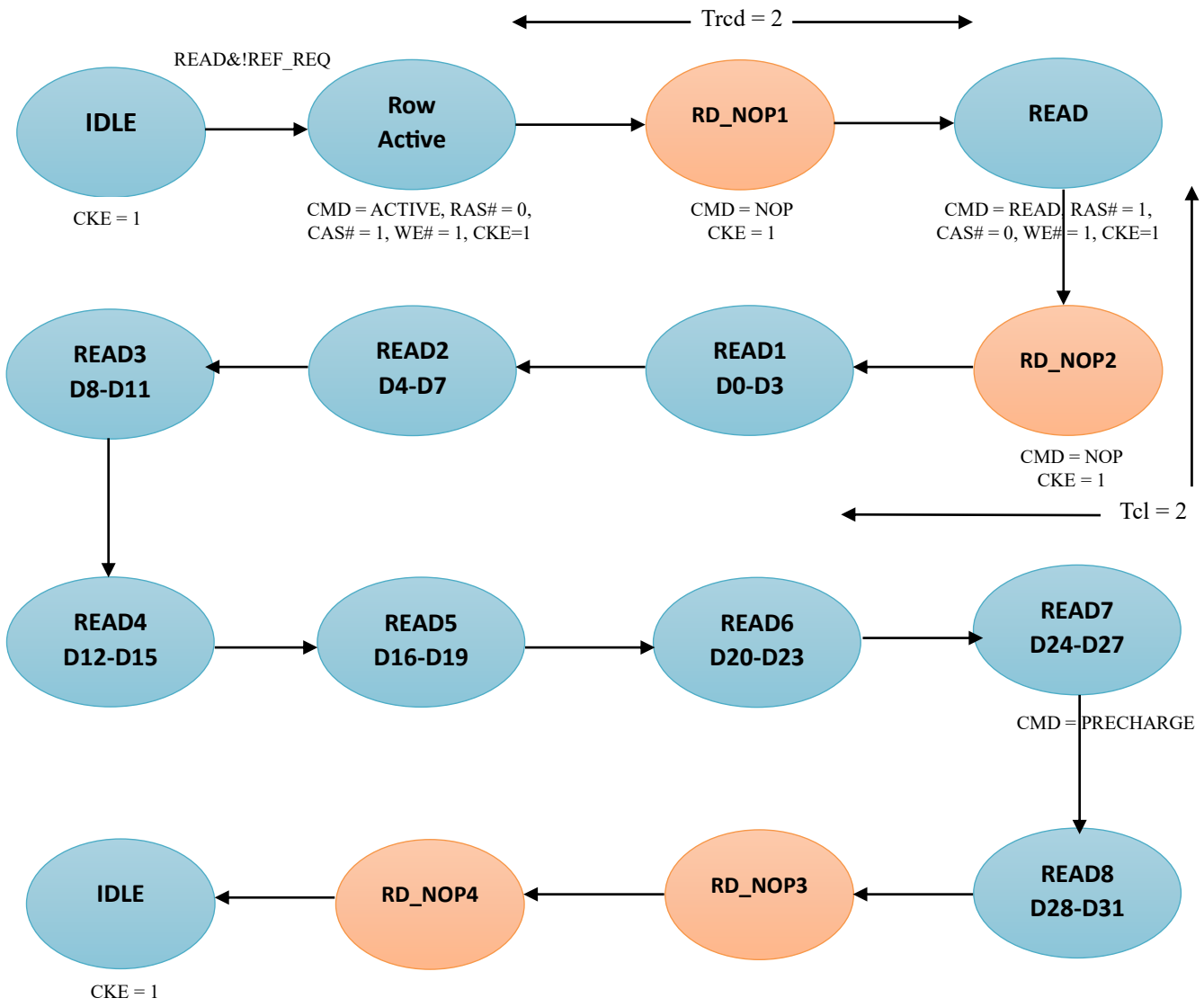
8) Number of read cycles = $\frac{\text{Time remaining for read cycle}}{T_{RC}} = \frac{996\,ms}{120\,ns}$

$= 8.3 \times 10^6 = 8300000$ cycles.

STEPS FOR READ:

The READ command is used to read data from an active row. When the signal READ&!REF_REQ goes low, the SDRAM moves to the Active state where the row and bank are activated by setting RAS#=0, CAS#=1, and WE#=1. This is followed by the RD_NOP1 state, after which it moves to the READ state. In the READ state, the column address is provided along with the value of input A10 which determines whether auto precharge is used or not. In this case, auto precharge is used and A10 is set to high. After the READ state, the SDRAM must wait based on the value of CAS Latency, which in this case is 2. Therefore, it moves to the RD_NOP2 state, and in the next cycle, it moves to READ1. This process continues up to READ8, and in each read, 4 bits of data are obtained, resulting in a total of 32 bits of data. After the READ operation is complete, the READY signal is generated, and a minimum of 2 wait cycles is required because the SDRAM clock is twice as fast as the microprocessor clock.

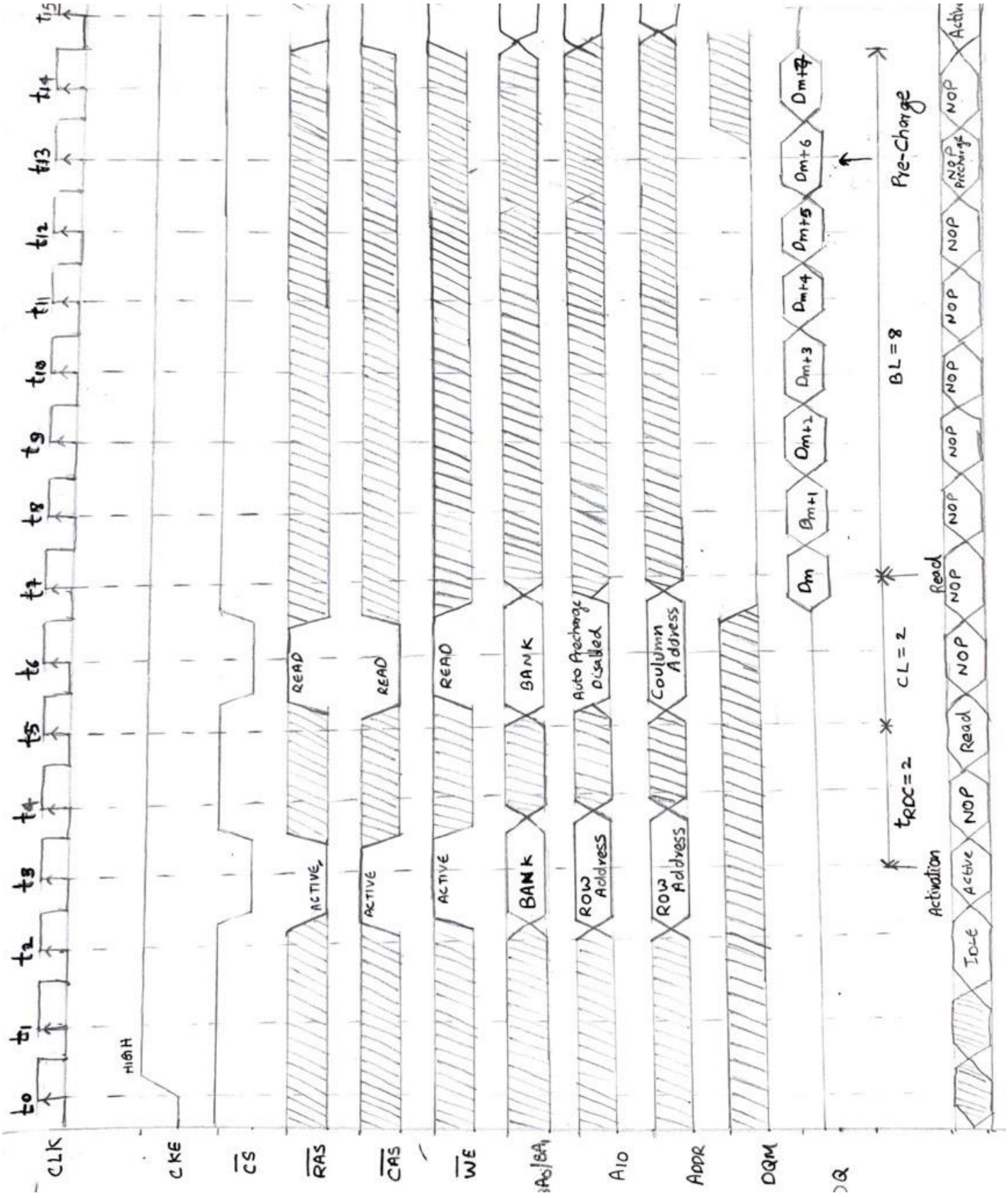
STATE MACHINE FOR READ:



STATE TRANSITION TABLE:

Current State	Condition	Next State
IDLE	RD&! REF_REQ=0	ROW ACTIVE
ROW ACTIVE	--	RD_NOP1
RD_NOP1	--	READ
READ	--	RD_NOP2
RD_NOP2	--	READ1
READ1	--	READ2
READ2	--	READ3
READ3	--	READ4
READ4	--	READ5
READ5	--	READ6
READ6	--	READ7
READ7	--	READ8
READ8	--	RD_NOP3
RD_NOP3	--	RD_NOP4
RD_NOP4	--	IDLE

TIMING DIAGRAM:



WRITE:

Calculations for WRITE:

Trcd Active to Read/Write delay = 20ns = 2 clock cycles

Twr Write Recovery Time = 1clk + 7.5ns = 2 clock cycles

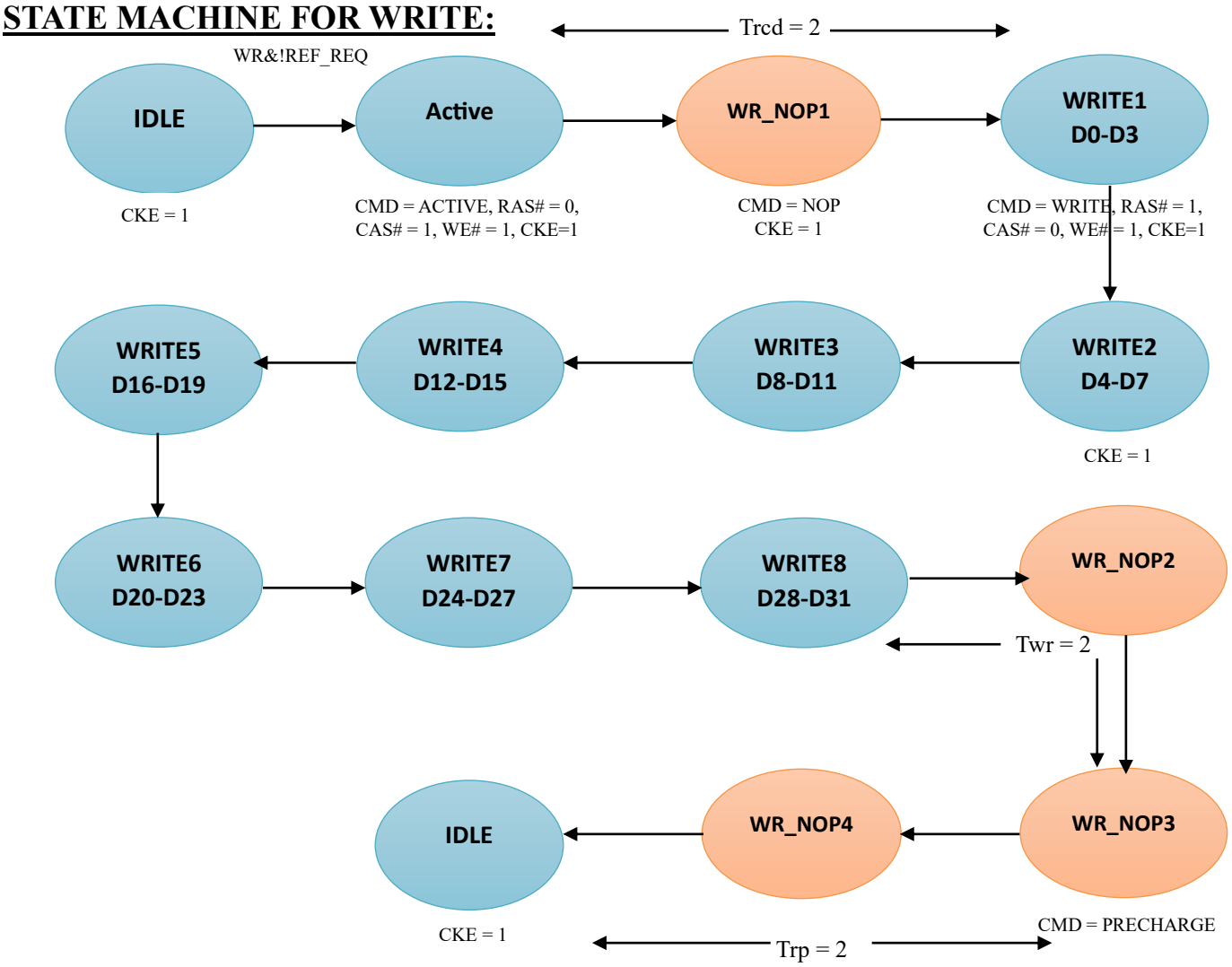
Trp Precharge Command Period = 20ns = 2 clock cycles

Tclcl = 10ns

STEPS FOR WRITE:

The WRITE command is used to initiate a write access to an active row in SDRAM. When the WRITE command along with the signal WR&!REF_REQ is received, the SDRAM moves to the Active state by activating the row and bank where RAS#=0, CAS#=1, WE#=1. This state is followed by WR_NOP1 state, and then it moves to the WRITE state where the column address and data to be written are provided. The write operation occurs in WRITE1 state, and this continues up to WRITE8 state, where a total of 32 bits of data (4 bits per write cycle) are written. After completion of WRITE, the SDRAM moves to WR_NOP2 state as per tWR timing requirements, and it takes 2 cycles from WRITE8 to WR_NOP3 state. Then, after a delay of trp=2 cycles, it moves to the IDLE state.

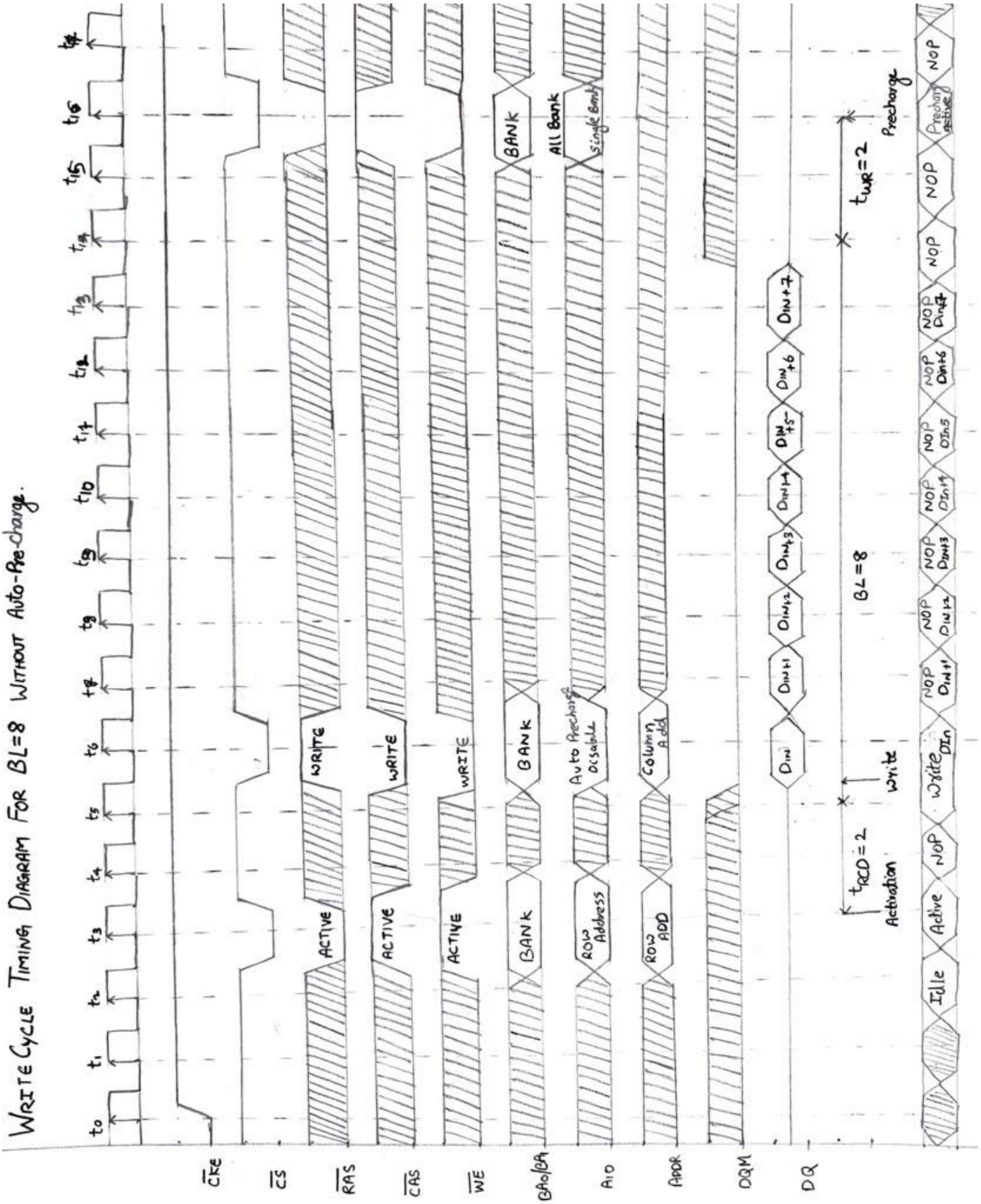
STATE MACHINE FOR WRITE:



STATE TRANSITION TABLE:

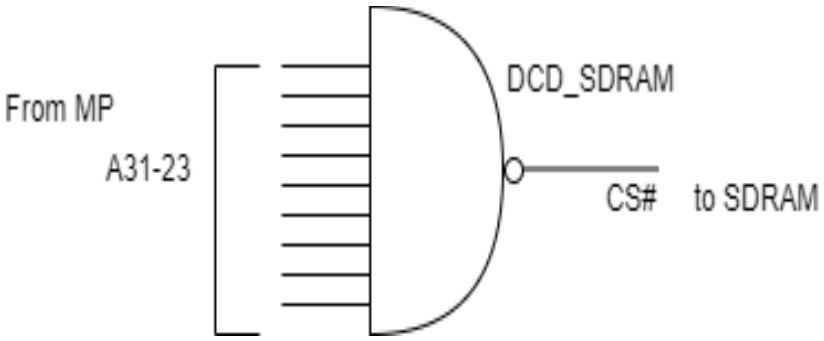
Current State	Condition	Next State
IDLE	WR&!REF_REQ=0	ACTIVE
ACTIVE	--	WR_NOP1
WR_NOP1	--	WRITE1
WRITE1	--	WRITE2
WRITE2	--	WRITE3
WRITE3	--	WRITE4
WRITE4	--	WRITE5
WRITE5	--	WRITE6
WRITE6	--	WRITE7
WRITE7	--	WRITE8
WRITE8	--	WR_NOP2
WR_NOP2	--	WR_NOP3
WR_NOP3	--	WR_NOP4
WR_NOP4	--	IDLE

TIMING DIAGRAM:

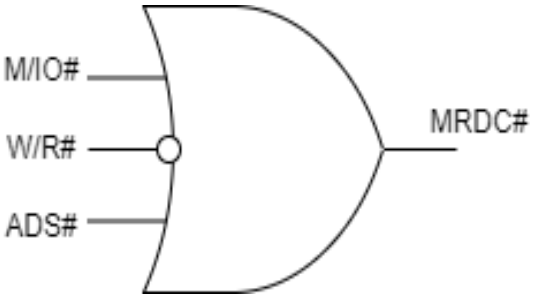


GENERATION OF CHIP SIGNALS:

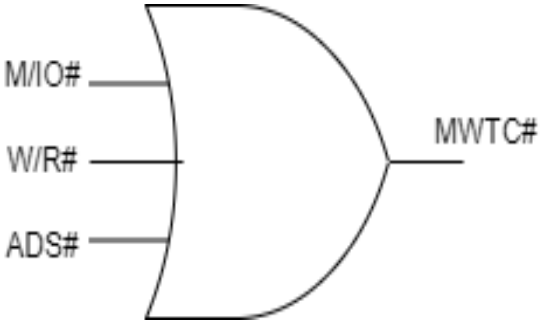
Generation of Chip Select:



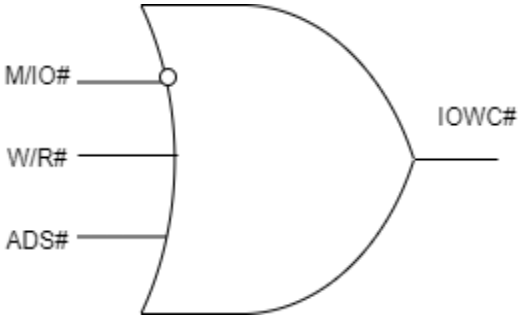
Generation of MRDC#:



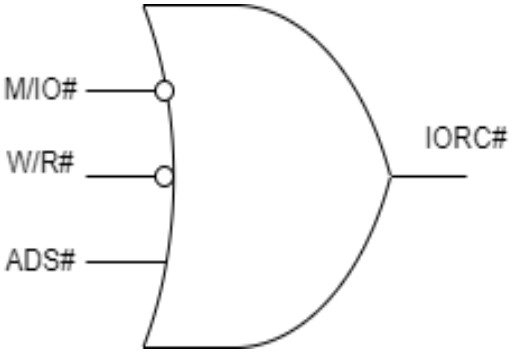
Generation of MWTC#:



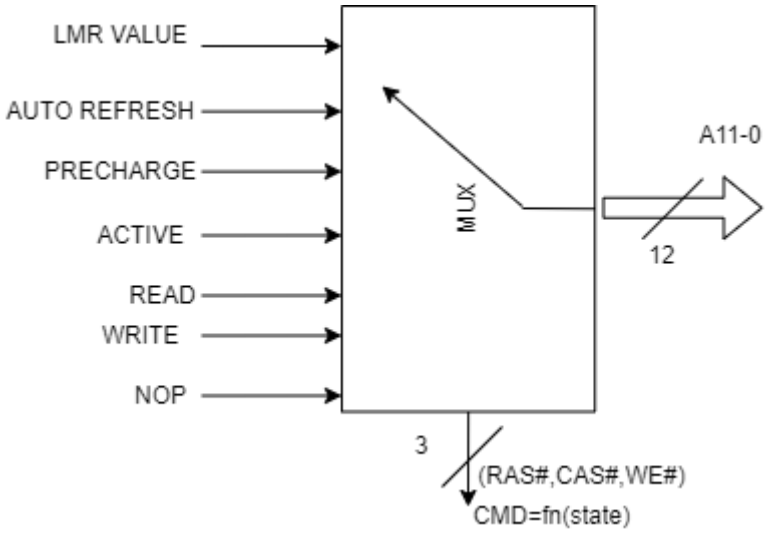
Generation of IOWC#:



Generation of IORC#:

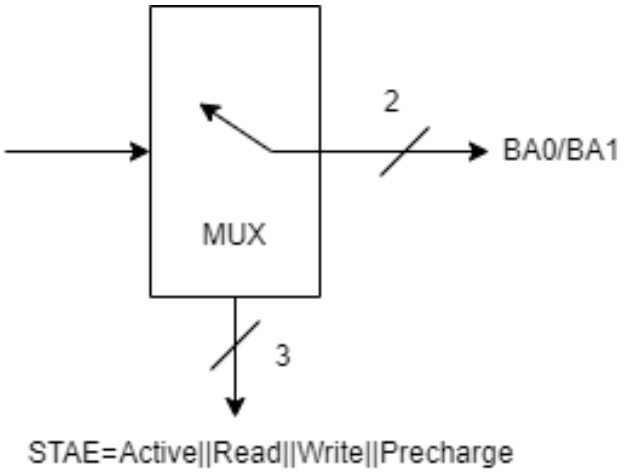


Generation of functional command signal:

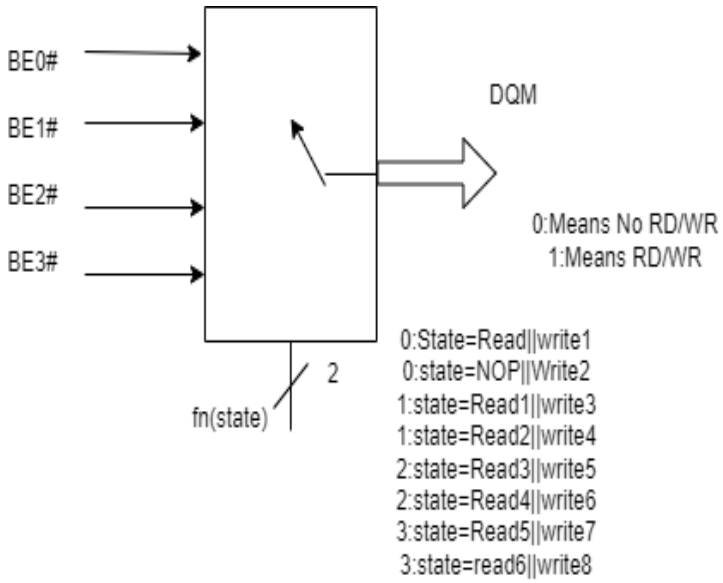


Name (Function)	RAS#	CAS#	WE#
LMR VALUE	L	L	L
AUTO REFRESH	L	L	H
PRECHARGE	L	H	L
ACTIVE	L	H	H
READ	H	L	H
WRITE	H	L	L
NOP	H	H	H

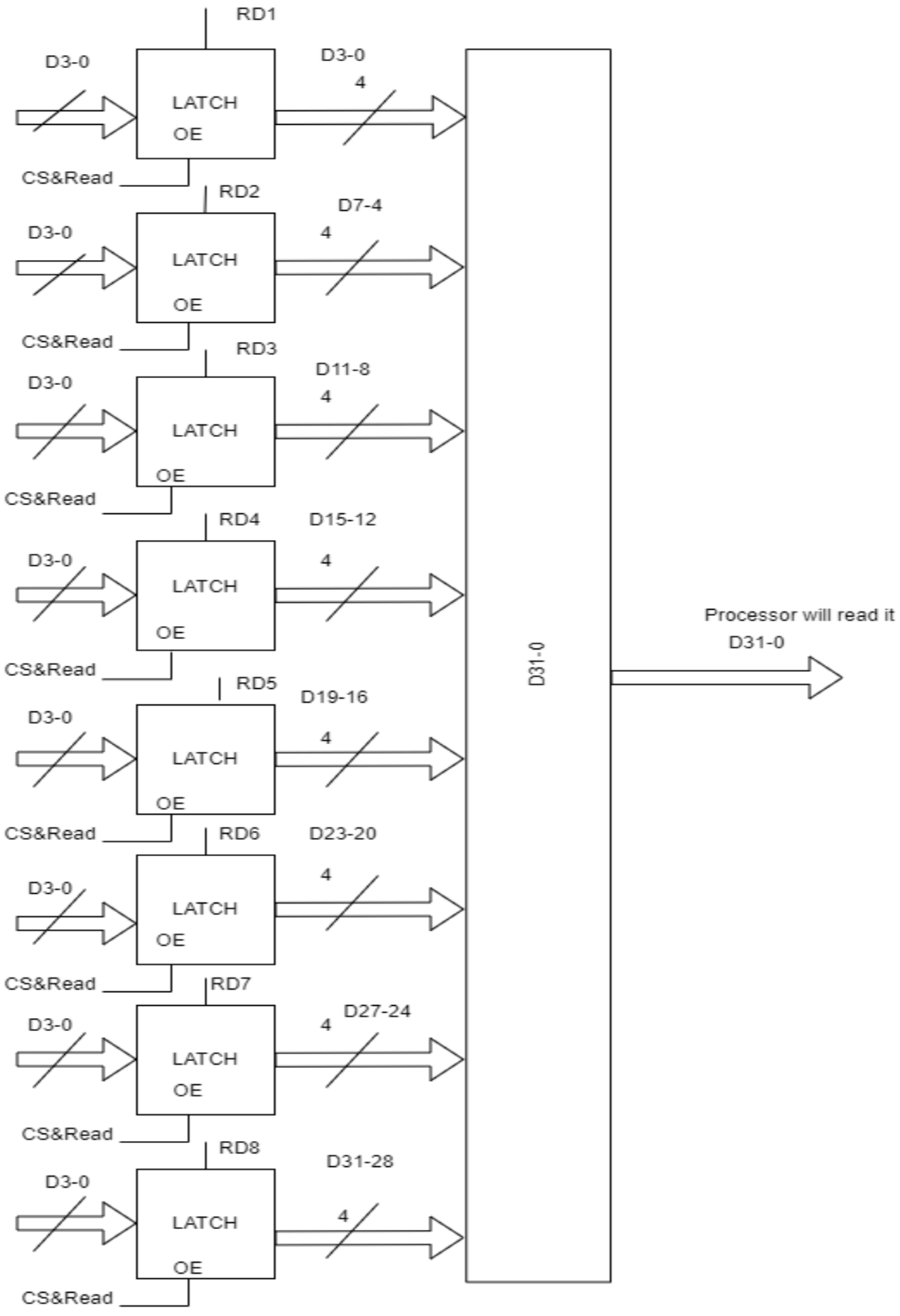
Generation of bank address:



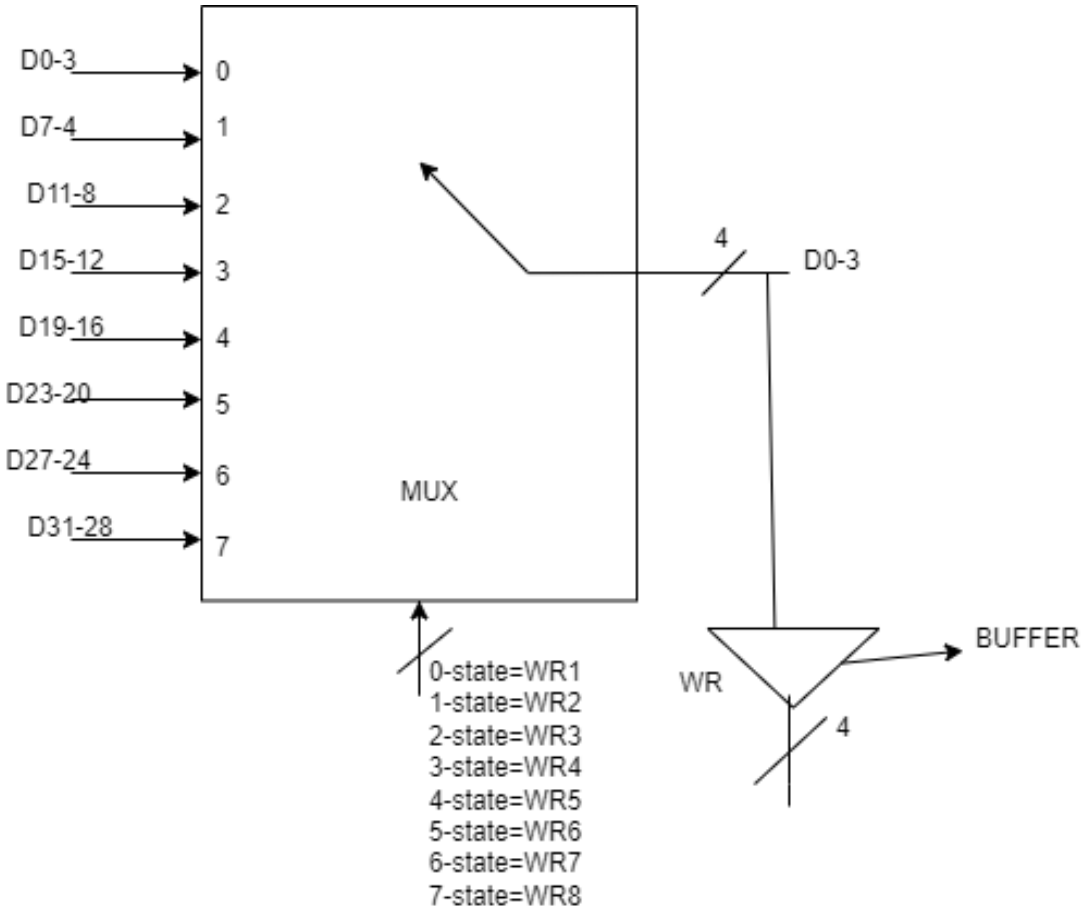
Generation of DQM:



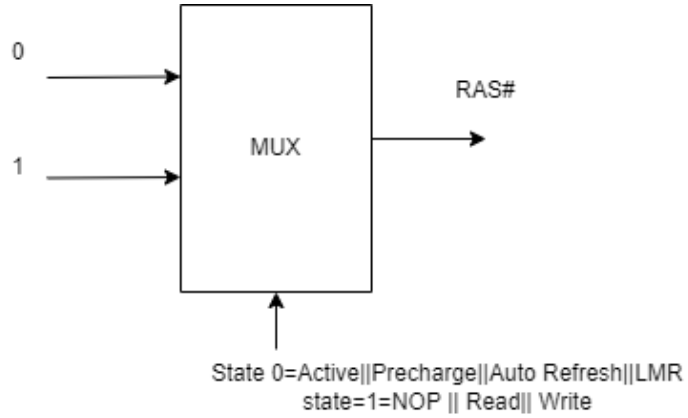
Data Latch for READ:



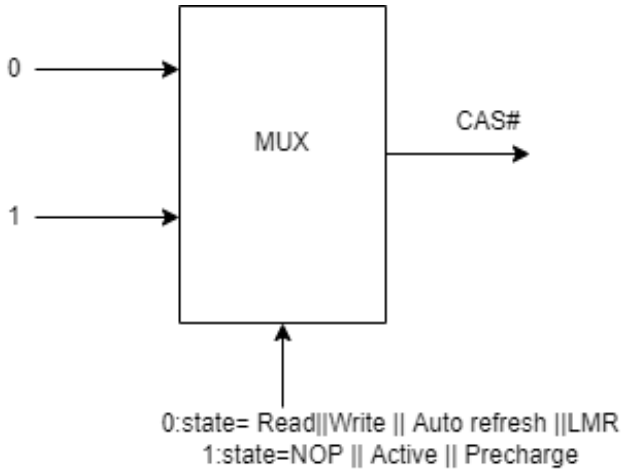
Data Mux for WRITE:



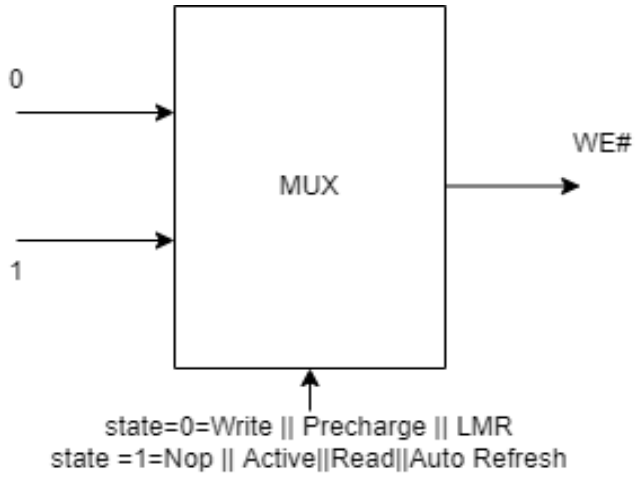
Generation of RAS#:



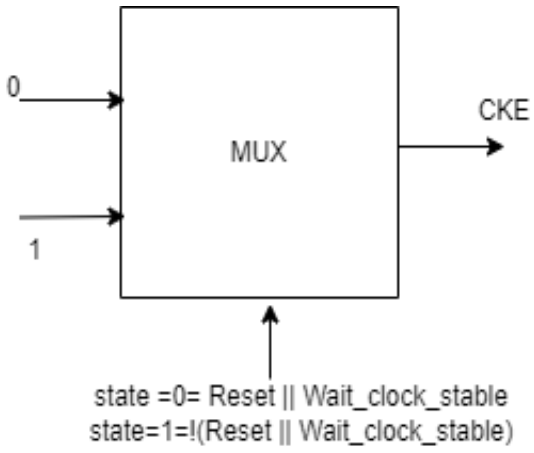
Generation of CAS#:



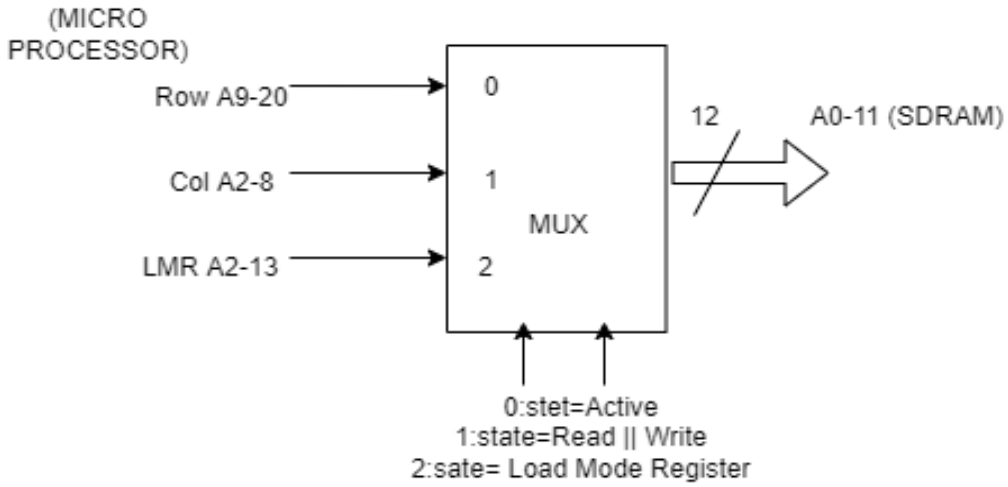
Generation of WE#:



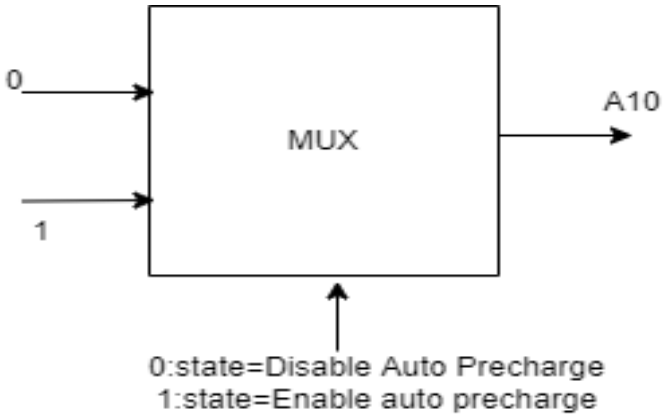
Generation of CKE:



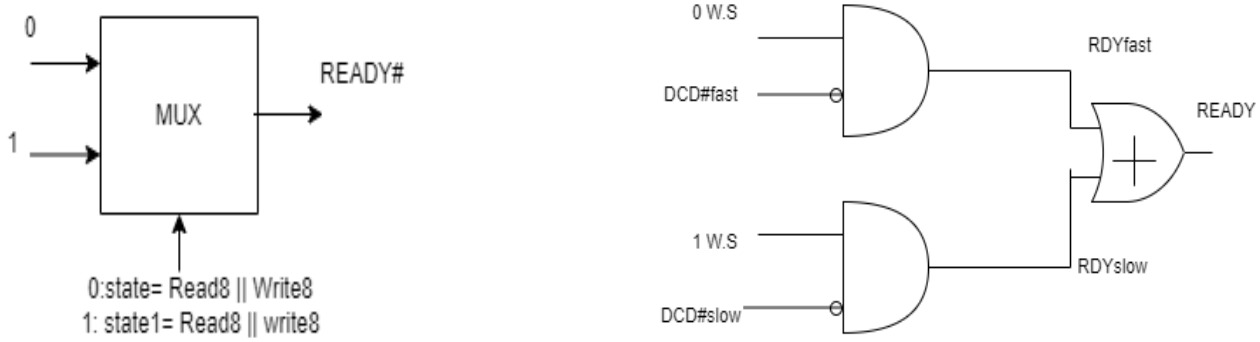
Generation of Row and Column:



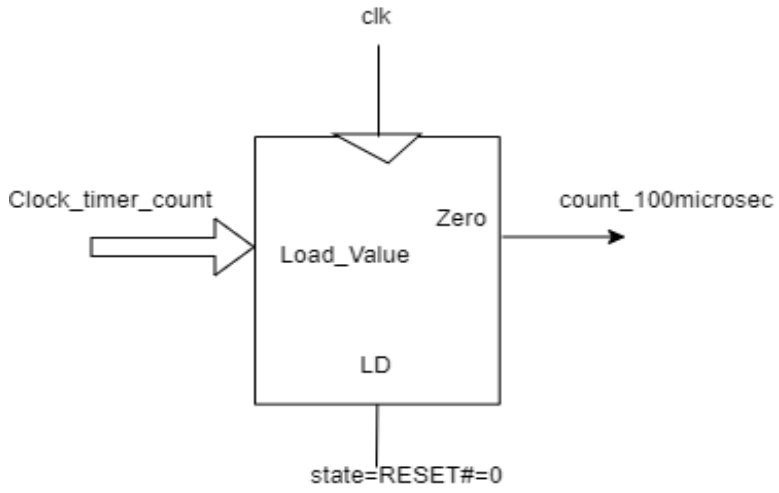
Generation of Auto Precharge (A10):



Generation of READY Signal:

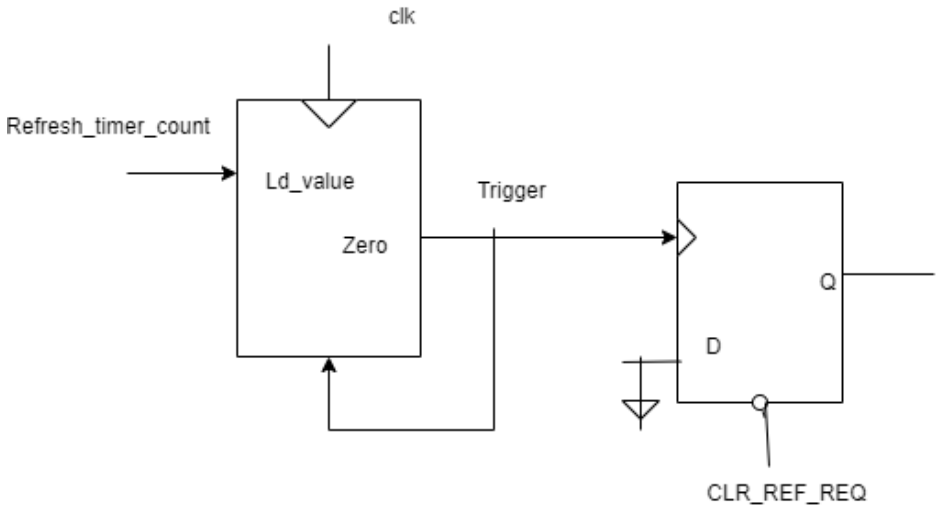


Generation of Clock Timer:



Clock_timer_count = 100microsec*100MHz = 10,000counts

Generation of Refresh Timer:



Refresh_timer_count = 64ms*100MHz = 6,400