MICROPROCESSOR SYSTEM DESIGN

ECE 585

DDR4 Memory Controller (OPEN PAGE POLICY)

Project Report

Instructor: Prof. Mark G. Faust

Team Members-

DINESH KUMAR SIVA(dsiva@pdx.edu)
(PSU ID : 955005529)
PRADEEP REDDY MANTHU (pmanthu@pdx.edu)
(PSU ID: 942289994)
NARENDRA SRINIVAS RAGHAVARAJU (nraghav2@pdx.edu)
(PSU ID: 970226940)
NAVEEN MANIVANNAN (nav9@pdx.edu)
(PSU ID :982204785)

Table of Contents

LIST OF FIGURES

LIST OF TABLES

INTRODUCTION:

DDR4 SDRAM:

Double Data Rate 4 Synchronous Dynamic Random-Access Memory is a synchronous dynamic random-access memory with a high bandwidth interface.

Advantages of DDR4:

- DDR4 RAM has higher overall speed.
- It gives you increased transfer rates and less voltage which means less overall power consumption.
- DDR4 RAM cuts back voltage by 40% and boosts performance and bandwidth by 50% as compared to DDR3.
- One of the main benefits of DDR4 that makes it better than the DDR3 modules is that it has a larger span of clock speeds and timings, uses less power, and features an overall reduction in latency.
- Because the DDR4 is capable of handling more information at once than previous RAM options, it aids long-term system stability. It also serves as a safer choice for overclocking tests because it can handle much more information without getting overworked. And finally, it reduces stress on the computer as a whole.

DDR4 ADDRESS MAPPING:

ROW	HIGH COL	BANK	BANK GRP	LOW COL	BYTE SEL
32 18	17 10	9 8	7 6	5 4 3	2 1 0

TABLE 1: Address Map

DDR4	
Organizaation	1Gib x8
Capacity	8Gib
No of Address pins	30
No of data pins	8
No of Rows	15
No of High Columns	8
No of Banks	4
No of Bank groups	4
No of Low Columns(Burst Index))	3
No of Bytes(Byte Select)	3

TABLE 2: DDR4 – Address Calculation

DIMM	
Organization	8Gib x 64
Capacity	64Gib(8GB)
Total no of chips	8

TABLE 3: DIMM – Memory Map

TIMING PARAMETER

Parameter	Value	Description	Comments
tRC	76	Row cycle time tRC=tRAS + tRP	Precharge to Precharge
tRAS	52	Row Address strobe	Activate to Precharge
tRRD_L	6	Row to Row delay long	Applies for accesses to same bank groups
tRRD_S	4	Row to Row delay Short	Applies for accesses to different bank groups
tRP	24	Row Precharge delay	Precharge to Activate
tRFC	350ns	Refresh Cycle Time	Time for a single refresh command
CWL(tWCD)	20	CAS Write Latency	Write to Data out
tCAS(CL)	24	CAS Latency	Read to Data out
tRCD	24	Row Command delay	Activate to Read
tWR	20	Write Recovery time	Time between the last write command to a row and precharging it. tRAS=tRCD+tWR
tRTP	12	Read to Pre-charge Delay	Read to Precharge
tCCD_L	8	Column to Column Delay long	Applies for accesses to same bank groups
tCCD_S	4	Column to Column Delay short	Applies for accesses to different bank groups
tBURST	4	Data Read Burst	Time for burst data to be read completely
tWTR_L	12	Write to Read Delay long	Applies for accesses to same bank groups
tWTR_S	4	Write to Read Delay short	Applies for accesses to different bank groups
REFI	7.8µs	Refresh Interval	Interval between two refresh commands

TABLE 4: Timing parameters

OUR APPROACH-(ARCHITECTURE)

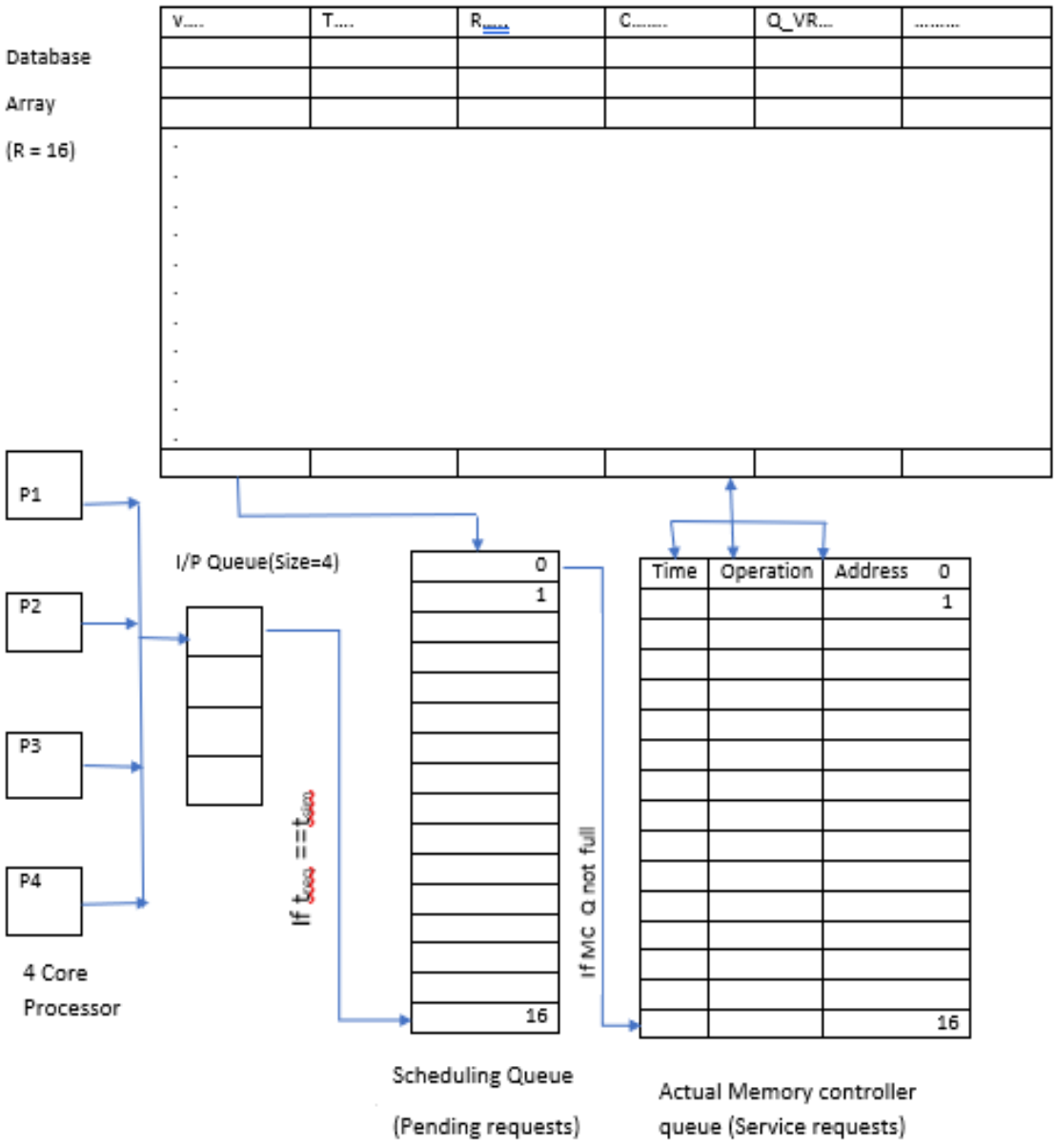

Figure 1: Code approach

We have four processors which gives out requests. We take those processors in an input queue which is of size four, to check for requests coming out at the same time and we will give those requests to a scheduling queue which is of size 16 to handle pending requests. Here we will order requests to be scheduled which uses bank parallelism and gives requests to the actual memory controller queue (which is also of size 16) when the actual queue is not full.

The top request in the actual queue will be serviced first and will be referenced with a database array, which contains validity bit, time, row, column and Q-reference values. When the next reference is for a different bank or bank group, it will be scheduled parallelly at the request time or at the next clock tick (if the request times are same). If the next reference is for the same bank which is being serviced, we will till the earlier request to be completed, to schedule the next request. Once the queue is full, the pending requests pile up in the scheduling queue and will be put in the actual queue when any request in the MC queue gets serviced completely.

SCHEDULING POLICY

FIRST READY FIRST ACCESS:

We employed first ready first access scheduling algorithm which processes requests in the order received but schedules DRAM commands such that after issuing an activate command for one request, it can issue another activate or pre-charge command for another bank/bank group of another request before issuing the read command for the earlier request.

OPEN PAGE POLICY:

We also use open page policy, so that the page is open for the next read or write request to the same row without precharging it.

ADAPTIVE SCHEDULING:

We will reorder the request in the scheduling queue so that the request In the open page will be scheduled first rather than request to an another page.

REFRESH:

The refresh command needs to be included in the interval of 24960 clock cycles and the refresh command will execute for 1120 clock cycles which refreshes all the banks.

TESTING STRATEGY

Testcase	Description	Bank group	Bank	Row	Column	Instruction	Time
Scenario 1	Accesses to same row in same bank and bankgroup	0	1	100	100	Read	10
		0	1	100	128	Read	11
		0	1	100	255	Read	12
Scenario 2	Accesses to different row in same bank and bankgroup	0	1	100	100	Read	10
		0	1	102	128	Write	11
		0	1	100	255	Read	12
Scenario 3	Accesses to same row in same bank and bankgroup at the same time	0	1	100	128	Read	10
		0	1	100	128	Read	10
		0	1	100	255	Read	10
Scenario 4	Accesses to different row in same bank and bankgroup at the same time	0	1	100	128	Read	10
		0	1	102	128	Read	10
		0	1	100	255	Read	10
Scenario 5	Accesses to same bank group and different banks	0	1	100	128	Read	10
		0	2	100	128	Write	11
		0	3	100	255	Read	12
Scenario 6	Accesses to different bank group	0	1	100	128	Read	10
		1	1	100	128	Read	11
		2	1	100	255	Read	12
Scenario 7	Accesses to same bank group and different banks at the same time	0	1	100	128	Read	10
		0	2	100	128	Write	10
		0	3	100	255	Read	10
Scenario 8	Accesses to different bank group at the same time	0	1	100	128	Read	10
		1	1	100	128	Read	10
		2	1	100	255	Read	10
Scenario 9	Reading the same column	0	1	100	100	Read	10
		0	1	100	100	Read	11
		0	1	100	100	Read	12
Scenario 10	Writing back to back and reading to check last updated value is read	0	1	100	100	Write	10
		0	1	100	100	Write	11
		0	1	100	100	Read	12
Scenario 11	Reading the same column at the same time	0	1	100	100	Read	10
		0	1	100	100	Read	10
		0	1	100	100	Read	10
Scenario 12	Writing to the same column at the same time	0	1	100	100	Write	10
		0	1	100	100	Write	10
		0	1	100	100	Write	10
Scenario 13	Prioritizing read overwrites for different addresses	0	1	100	100	Write	10
		0	1	105	128	Write	11

		0	1	110	156	Read	12
Scenario 14	Prioritizing read overwrites for different addresses at the same time	0	1	100	100	Write	10
		0	1	105	128	Write	10
		0	1	110	156	Read	10
Scenario 15	Queue is full and instructions are pending	0	1	100	128	Read	20
		0	1	102	128	Write	21
		0	1	100	255	Read	22
Scenario 16	Queue has one place left and multiple instructions are pending	0	1	100	128	Read	20
		0	1	102	128	Write	21
		0	1	100	255	Read	22
Scenario 17	Queue has one place left and pending instructions are at the same time	0	1	100	128	Read	20
		0	1	102	128	Write	20
		0	1	100	255	Read	20
Scenario 18	Accesses to same row in same bank and bank group(5th and 21st) and no request has happened to this bank in-between	0	1	100	128	Read	10
		0	2	100	128	Write	11
		0	1	100	128	Read	32
Scenario 19	Write to same row in same bank and bank group(5th and 21st) and no request has happened to this bank inbetween+C51	0	1	100	128	Write	10
		0	2	100	128	Write	11
		0	1	100	128	Write	32
Scenario 20	When request time is very higher (1000) compared to simulation time(50) and queue is not full(next request comes at a high request time)	0	1	100	128	Read	50
		0	2	100	128	Write	1000
		0	1	100	128	Write	1002
Scenario 21	Consecutive requests for the same bank and bank group will wait till the first request is completed	0	1	100	128	Read	20
		0	1	102	128	Write	21
		0	2	100	255	Read	22
Scenario 22	Write and read rows in same bank group and bank	0	1	100	128	Write	10
		0	1	102	128	Read	21
		0	1	104	128	Write	22
Scenario 23	Write and read rows in different bank groups and bank	0	1	100	128	Write	10
		1	1	102	128	Read	21
		2	1	104	128	Write	22