

ECE -585 - MICROPROCESSOR SYSTEM DESIGN

Test Plan-GROUP 10

Simulation of the Scheduler portion of a DDR5 Memory Controller

Team Members:

Gagana Hirenallur Mohan Kumar PSU ID: 939272712 Mail id: gagana@pdx.edu	Raksha Mairpady PSU ID: 965104471 Mail id: raksham@pdx.edu
Rahul Jnanesh Gowda PSU ID:979476354 Mail id: rahulj@pdx.edu	Shashi Kirana Chinchandlahalli Ravanappa PSU ID: 910530955 Mail id: shashiki@pdx.edu

1. Successive read request in the same row of the same bank group, bank but different column address

Input format: 1 6 0 00012050A
2 6 0 00013F50A

For address : 0x 00012050A		For address:00013F50A	
Byte select	2	Byte select	2
Lower column	2	Lower column	2
Channel	0	Channel	0
Bank group	2	Bank group	2
Bank	1	Bank	1
Upper column	20	Upper column	3f
Row	4	Row	4

Obtained output:

2 0 ACT0 2 1 0004
4 0 ACT1 2 1 0004

82	0 RD0 2 1 202
84	0 RD1 2 1 202
180	0 PRE 2 1
260	0 ACT0 2 1 0004
262	0 ACT1 2 1 0004
340	0 RD0 2 1 3f2
342	0 RD1 2 1 3f2
438	0 PRE 2 1

2. Successive read requests in the different rows of the same bank group.

Input format: 38 7 0 00019050A
39 7 0 000E2050A

For address: 0x00019050A	
Byte select	2
Lower column	2
Channel	0
Bank group	2
Bank	1
Upper column	10
Row	6

For address:0X000E2050A	
Byte select	2
Lower column	2
Channel	0
Bank group	2
Bank	1
Upper column	20
Row	38

Obtained output:

518	0 ACT0 2 1 0006
520	0 ACT1 2 1 0006
598	0 RD0 2 1 102
600	0 RD1 2 1 102
696	0 PRE 2 1
776	0 ACT0 2 1 0038
778	0 ACT1 2 1 0038
856	0 RD0 2 1 202
858	0 RD1 2 1 202
954	0 PRE 2 1

3. Successive read requests to the same bank group, but different banks

Input format: 40 8 0 00012090A
41 8 0 000120D0A

For address : 0x 00012090A	
Byte select	2
Lower column	2
Channel	0
Bank group	2
Bank	2
Upper column	20
Row	4

For address: 0x000120D0A	
Byte select	2
Lower column	2
Channel	0
Bank group	2
Bank	3
Upper column	20
Row	4

Obtained output:

```

1034    0 ACT0 2 2 0004
1036    0 ACT1 2 2 0004
1114    0 RD0 2 2 202
1116    0 RD1 2 2 202
1212    0 PRE 2 2
1292    0 ACT0 2 3 0004
1294    0 ACT1 2 3 0004
1372    0 RD0 2 3 202
1374    0 RD1 2 3 202
1470    0 PRE 2 3

```

4. Successive read requests to the different bank groups, but the same bank

Input format: 42 9 0 00010C200
43 9 0 00010C380

For address : 0x 00010C200	
Byte select	0
Lower column	0
Channel	0
Bank group	4
Bank	0
Upper column	C
Row	4

For address: 0x00010C380	
Byte select	0
Lower column	0
Channel	0
Bank group	7
Bank	0
Upper column	C
Row	4

Obtained output:

```

1550    0 ACT0 4 0 0004
1552    0 ACT1 4 0 0004
1630    0 RD0 4 0 0c0
1632    0 RD1 4 0 0c0
1728    0 PRE 4 0
1808    0 ACT0 7 0 0004
1810    0 ACT1 7 0 0004
1888    0 RD0 7 0 0c0
1890    0 RD1 7 0 0c0
1986    0 PRE 7 0

```

5. Read requests to the row of the same bank group and bank, followed by write to the different row of the same bank groups and bank

Input format: 44 4 0 000817084
45 4 1 000C97084

For address : 0x 000817084	
Byte select	0
Lower column	1
Channel	0
Bank group	1
Bank	0
Upper column	17
Row	20

For address: 000C97084	
Byte select	0
Lower column	1
Channel	0
Bank group	1
Bank	0
Upper column	17
Row	32

Obtained output:

```

2066    0 ACT0 1 0 0020
2068    0 ACT1 1 0 0020
2146    0 RD0 1 0 171

```

```

2148    0 RD1 1 0 171
2244    0 PRE 1 0
2324    0 ACT0 1 0 0032
2326    0 ACT1 1 0 0032
2404    0 WR0 1 0 171
2406    0 WR1 1 0 171
2558    0 PRE 1 0

```

6. Write to a row of the same bank group and bank followed by a request to read in the same row of the same bank and bank group

Input format: 40 4 1 00016A000
41 4 0 00016A000

For address : 0x 00016A000	
Byte select	0
Lower column	0
Channel	0
Bank group	0
Bank	0
Upper column	2A
Row	5

Obtained output:

```

42      0 ACT0 0 0 0005
44      0 ACT1 0 0 0005
122     0 WR0 0 0 2a0
124     0 WR1 0 0 2a0
276     0 PRE 0 0
356     0 ACT0 0 0 0005
358     0 ACT1 0 0 0005
436     0 RD0 0 0 2a0
438     0 RD1 0 0 2a0
534     0 PRE 0 0

```

7. Write to a row within a specific bank and bank group followed by write to a distinct row within the same bank group and bank.

Input format: 42 5 1 000301210
43 5 1 000FC1210

For address: 0x 000301210	
Byte select	0
Lower column	4
Channel	0
Bank group	4
Bank	0
Upper column	1
Row	C

For address: 000FC1210	
Byte select	0
Lower column	4
Channel	0
Bank group	4
Bank	0
Upper column	1
Row	3f

Obtained output:

```

614      0 ACT0 4 0 000c
616      0 ACT1 4 0 000c
694      0 WR0 4 0 014
696      0 WR1 4 0 014
848      0 PRE 4 0
928      0 ACT0 4 0 003f
930      0 ACT1 4 0 003f
1008     0 WR0 4 0 014
1010     0 WR1 4 0 014
1162     0 PRE 4 0

```

8. Write from a row within the same bank group and bank followed by read to distinct bank groups and bank.

Input format: 44 3 1 000510230
45 3 0 000510EB0

For address : 0x 000510230	
Byte select	0
Lower column	C
Channel	0
Bank group	4
Bank	0
Upper column	10
Row	14

For address: 000510EB0	
Byte select	0
Lower column	C
Channel	0
Bank group	5
Bank	3
Upper column	10
Row	14

Obtained output:

```
1242    0 ACT0 4 0 0014
1244    0 ACT1 4 0 0014
1322    0 WR0 4 0 10c
1324    0 WR1 4 0 10c
1476    0 PRE 4 0
1556    0 ACT0 5 3 0014
1558    0 ACT1 5 3 0014
1636    0 RD0 5 3 10c
1638    0 RD1 5 3 10c
1734    0 PRE 5 3
```

9. Write from a row within a bank of a bank group followed by request to write to a distinct row and column within the same bank and bank group.

Input format: 46 5 1 00064049F
47 5 1 0009C2493

For address: 0x00064049F	
Byte select	3
Lower column	7
Channel	0
Bank group	1
Bank	1
Upper column	0
Row	19

For address: 0X0009C2493	
Byte select	3
Lower column	4
Channel	0
Bank group	1
Bank	1
Upper column	2
Row	27

Obtained output:

```
1814    0 ACT0 1 1 0019
1816    0 ACT1 1 1 0019
1894    0 WR0 1 1 007
1896    0 WR1 1 1 007
2048    0 PRE 1 1
2128    0 ACT0 1 1 0027
2130    0 ACT1 1 1 0027
2208    0 WR0 1 1 024
2210    0 WR1 1 1 024
2362    0 PRE 1 1
```

10. Consecutive reads to different banks in the same bank group.

Input format: 48 4 0 00084789A
 48 4 0 000847C9A
 48 4 0 00084709A

For address: 0x00084789A	
Byte select	2
Lower column	6
Channel	0
Bank group	1
Bank	2
Upper column	7
Row	21

For address: 0X000847C9A	
Byte select	2
Lower column	6
Channel	0
Bank group	1
Bank	3
Upper column	7
Row	21

For address: 0X00084709A	
Byte select	2
Lower column	6
Channel	0
Bank group	1
Bank	0
Upper column	7
Row	21

Obtained output:

2442	0 ACT0 1 2 0021
2444	0 ACT1 1 2 0021
2522	0 RD0 1 2 076
2524	0 RD1 1 2 076
2620	0 PRE 1 2
2700	0 ACT0 1 3 0021
2702	0 ACT1 1 3 0021
2780	0 RD0 1 3 076
2782	0 RD1 1 3 076
2878	0 PRE 1 3
2958	0 ACT0 1 0 0021
2960	0 ACT1 1 0 0021
3038	0 RD0 1 0 076
3040	0 RD1 1 0 076
3136	0 PRE 1 0

11. Access to same row in same bank and same bank group at the same time.

Input format: 1 6 0 00012050A
 1 6 0 00013F50A

For address : 0x 00012050A	
Byte select	2
Lower column	2
Channel	0
Bank group	2
Bank	1
Upper column	20
Row	4

For address:00013F50A	
Byte select	2
Lower column	2
Channel	0
Bank group	2
Bank	1
Upper column	3f
Row	4

Obtained output:

```

2      0 ACT0 2 1 0004
4      0 ACT1 2 1 0004
82     0 RD0 2 1 202
84     0 RD1 2 1 202
180    0 PRE 2 1
260    0 ACT0 2 1 0004
262    0 ACT1 2 1 0004
340    0 RD0 2 1 3f2
342    0 RD1 2 1 3f2
438    0 PRE 2 1

```

12. Access to different row in same bank and bank group at the same time.

Input format: 38 7 0 00019050A

38 7 0 000E2050A

For address: 0x00019050A	
Byte select	2
Lower column	2
Channel	0
Bank group	2
Bank	1
Upper column	10
Row	6

For address:0X000E2050A	
Byte select	2
Lower column	2
Channel	0
Bank group	2
Bank	1
Upper column	20
Row	38

Obtained output:

```

518      0 ACT0 2 1 0006
520      0 ACT1 2 1 0006
598      0 RD0 2 1 102
600      0 RD1 2 1 102
696      0 PRE 2 1
776      0 ACT0 2 1 0038
778      0 ACT1 2 1 0038
856      0 RD0 2 1 202
858      0 RD1 2 1 202
954      0 PRE 2 1

```

13. Access to same bank group and same bank at the same time.

Input format: 38 7 0 00015A240
38 7 0 00015A240

For address: 0x00015A200	
Byte select	0
Lower column	0
Channel	0
Bank group	4
Bank	0
Upper column	1a
Row	5

For address:0X00015A200	
Byte select	0
Lower column	0
Channel	0
Bank group	4
Bank	0
Upper column	1a
Row	5

Obtained output:

```

1034      1 ACT0 4 0 0005
1036      1 ACT1 4 0 0005
1114      1 RD0 4 0 1a0
1116      1 RD1 4 0 1a0
1212      1 PRE 4 0
1292      1 ACT0 4 0 0005
1294      1 ACT1 4 0 0005
1372      1 RD0 4 0 1a0
1374      1 RD1 4 0 1a0
1470      1 PRE 4 0

```

14. Access to different bank groups at the same time.

Input format: 40 8 0 00012090A

40 8 0 000120D0A

For address : 0x 00012090A	
Byte select	2
Lower column	2
Channel	0
Bank group	2
Bank	2
Upper column	20
Row	4

For address: 0x000120D0A	
Byte select	2
Lower column	2
Channel	0
Bank group	2
Bank	3
Upper column	20
Row	4

Obtained output:

```

1550      0 ACT0 2 2 0004
1552      0 ACT1 2 2 0004
1630      0 RD0 2 2 202
1632      0 RD1 2 2 202
1728      0 PRE 2 2
1808      0 ACT0 2 3 0004
1810      0 ACT1 2 3 0004
1888      0 RD0 2 3 202
1890      0 RD1 2 3 202
1986      0 PRE 2 3

```

15. Reading the same column at different time.

Input format: 1 6 0 00012050A

2 6 0 00012050A

For address : 0x 00012050A	
Byte select	2
Lower column	2
Channel	0
Bank group	2
Bank	1
Upper column	20
Row	4

For address:00012050A	
Byte select	2
Lower column	2
Channel	0
Bank group	2
Bank	1
Upper column	20
Row	4

Obtained output:

```
2066      0 ACT0 2 1 0004
2068      0 ACT1 2 1 0004
2146      0 RD0 2 1 202
2148      0 RD1 2 1 202
2244      0 PRE 2 1
2324      0 ACT0 2 1 0004
2326      0 ACT1 2 1 0004
2404      0 RD0 2 1 202
2406      0 RD1 2 1 202
2502      0 PRE 2 1
```

16. Writing to same column at the same time

Input format: 1 6 1 00012050A

1 6 1 00012050A

For address : 0x 00012050A	
Byte select	2
Lower column	2
Channel	0
Bank group	2
Bank	1
Upper column	20
Row	4

For address:00012050A	
Byte select	2
Lower column	2
Channel	0
Bank group	2
Bank	1
Upper column	20
Row	4

Obtained output:

```

2582    0 ACT0 2 1 0004
2584    0 ACT1 2 1 0004
2662    0 WR0 2 1 202
2664    0 WR1 2 1 202
2816    0 PRE 2 1
2896    0 ACT0 2 1 0004
2898    0 ACT1 2 1 0004
2976    0 WR0 2 1 202
2978    0 WR1 2 1 202
3130    0 PRE 2 1

```

17. Prioritizing read over write for different address at the same time.

Input format: 40 4 0 000817084
40 4 1 000C97084

For address : 0x 000817084	
Byte select	0
Lower column	1
Channel	0
Bank group	3
Bank	2
Upper column	17
Row	20

For address: 000C97084	
Byte select	0
Lower column	1
Channel	0
Bank group	1
Bank	0
Upper column	17
Row	32

Obtained output:

```

3210    0 ACT0 3 2 0020

```

3212	0 ACT1 3 2 0020
3290	0 RD0 3 2 171
3292	0 RD1 3 2 171
3388	0 PRE 3 2
3468	0 ACT0 1 0 0032
3470	0 ACT1 1 0 0032
3548	0 WR0 1 0 171
3550	0 WR1 1 0 171
3702	0 PRE 1 0

18. Out of bound checking:

Check whether the address, bank group, bank, row, channel, column is within the boundary value specified in the spec.

Take the value of the operations greater than 2 and check whether the condition fails or exceeds the limit specified in the spec. Similarly, testing for bank group, bank, channel, row and column.

Obtained output: Trace file has out of bound values.

19. Testing full and empty conditions of the queue:

- i. Check for the condition by giving the values for input CPU clocks less than the DIMM clocks. So, the queue would be empty
- ii. Check for the condition by giving values for input CPU clocks continuously up to a particular range where the frequency of the popping requests will be less than the frequency of the inputs at CPU clock cycles. Hence, Queue would be full.
- iii. Check whether the queue is full by adding and removing the request at the same time.
- iv. Adding and removing the request at the same time when the queue is not full.