CS147 HW1

Keonwoong Min

1.

a. the size of shamt is determined by the size of the memory over the number of registers.

Since shamt is 6 bits, shamt = log2(x) = 6 bits, x = 64

So, 2KB / number of registers = 64bits, 16384 / 64 = 256

Number of registers = **256**

b. For R-type, rs + rd + rt = 64bits – (opcode + shamt + fucnt) = 64 – ( 10 + 6 + 24) = 24

So, It has 24 / 3 = 8 bit for each, rs, rt, rd.

For I-type, there are opcode, rs, rt, and immediate.

So I-type has 38 bits for immediate.

Since 2’s complement form is used, the range is **[ -237, 237-1 ]**

c. To get the maximum number of instructions, it needs to add up all three R-type, I-type, J-type instructions.

opcode is 0X0 for R-type and fucnt has 24 bit on the graph, so it has 224 = 16,777,216 instructions.

For J and I type has 10bit for opcode 210 - 1 =1023 instructions

So, **16,778,239** instructions in total.

d. Since memory has 32GB, (1GB = 230 byte)

32 GB = 32 \* 230 byte = 34359738368 byte \* (8 bit/1byte) = 274877906944 bit

274877906944 /32 = 8589934592 address buckets.

log2(8589934592) = 33

It has **33** address ports.

e. Since the memory is running with 1.2 GHz clock, the processor’s clock period T = 1/frequency= 1 / 1.2GHz = 0.833\*10-9 sec= 0.833 ns.

f. Since Write transaction is 25% with 10 % of write operation right after read operation,

4 million \* 0.75 = 3 million of read transaction

4 million \* 0.25 \* 0.9 = 0.9 million of write transaction

4 million \* 0.25 \* 0.1 = 0.1 million of write/read transaction

Since this memory need one cycle to complete read and write request and need another one more cycle for read/write request,

3 million \* 1 cycle = 3 million cycle

0.9 million \* 1 cycle = 0.9 million cycle

0.1 million \* 2 cycle = 0.2 million cycle (RWRW for 32 but RRWW for 64 bit, and only care RW part in the 64bit so dont need to times by 2) so its just 0.1 million cycle

Total clock cycle is 4.1 million cycle

4.2 million clock cycle \* (0.833 ns / 1 clock cycle) \* (1\*10-6 ms / 1 ns) = 3.4153ms

But, since 4 million 64 bit data transaction

3.4153ms \* 2 = **6.8306ms**

2.

a. TiLaSoDoReMiFa = (6540123)7

= (6\*76 + 5\*75 + 4\*74 + 0\*73 + 1\*72 + 2\*71 + 3\*70) = (799599)10

b. (1546781)10 = LaTiFaReDoReLaRe

|  |  |  |
| --- | --- | --- |
| Divide by | Quotient | Remainder |
| 7 | 1546781 |  |
| 7 | 220968 | 5(La) |
| 7 | 31566 | 6(Ti) |
| 7 | 4509 | 3(Fa) |
| 7 | 644 | 1(Re) |
| 7 | 92 | 0(Do) |
| 7 | 13 | 1(Re) |
| 7 | 1 | 5(La) |
|  | 0 | 1(Re) |

3.

a. F(x, y, z) = x’z + xy = x’z(y + y’) + xy

= x’y’z + x’yz + xy

= x’y’z + y(x’z + x)

= x’y’z + y((x+x’)(x+z)

= x’y’z + y(x+z)

= x’y’z + yz + xy

b. F(a, b, c, d) = a’b’c’d’ + a’b’cd + a’b’cd’ + ab’c’d’ + ab’cd’ + ab’cd

= b’(a’c’d’+a’cd+a’cd’+ac’d’+acd’+acd)

= b’( c’d’(a’+a) + c(a’d + a’d’ + ad’ + ad) )

= b’(c’d’ + c)

= b’( (c’+c)(d’+c) )

= b’(c+d’)

c.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **X** | **Y** | **Z** | **x’z + xy** | **x’y’z + yz + xy** |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 0 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 |

d.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **a** | **b** | **c** | **d** | **a’b’c’d’ + a’b’cd + a’b’cd’ + ab’c’d’ + ab’cd’ + ab’cd** | **b’(c+d’)** |
| 0 | 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 1 | 1 |
| 0 | 0 | 1 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 | 1 |
| 1 | 0 | 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 | 1 |
| 1 | 0 | 1 | 1 | 1 | 1 |
| 1 | 1 | 0 | 0 | 0 | 0 |
| 1 | 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 1 | 0 | 0 | 0 |
| 1 | 1 | 1 | 1 | 0 | 0 |

4.

a. f(A,B,C,D)=∑m(0,5,7,8,10,12,14,15)

* Prime implicants = ad’ + a’bd + bc’d’ + bcd + abc
* Essential prime implicants = ad’ + a’bd + bc’d’ + bcd

OR

ad’ + a’bd + bc’d’ + abc

b. f(w, x, y, z) = ∑ m (1,3,4,7,11) + d(5, 12, 13, 14, 15)

* Prime implicants = yz + w`z + xy`
* Essential prime implicants = yz + w`z + xy`

5.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **W** | **X** | **Y** | **Z** | **Decimal** | **output** |
| 0 | 0 | 0 | 0 | 0 | **1** |
| 0 | 0 | 0 | 1 | 1 | **1** |
| 0 | 0 | 1 | 0 | 2 | **1** |
| 0 | 0 | 1 | 1 | 3 | **1** |
| 0 | 1 | 0 | 0 | 4 | 0 |
| 0 | 1 | 0 | 1 | 5 | 0 |
| 0 | 1 | 1 | 0 | 6 | **1** |
| 0 | 1 | 1 | 1 | 7 | 0 |
| 1 | 0 | 0 | 0 | 8 | 0 |
| 1 | 0 | 0 | 1 | 9 | **1** |
| 1 | 0 | 1 | 0 | 10 | 0 |
| 1 | 0 | 1 | 1 | 11 | 0 |
| 1 | 1 | 0 | 0 | 12 | **1** |
| 1 | 1 | 0 | 1 | 13 | **1** |
| 1 | 1 | 1 | 0 | 14 | **1** |
| 1 | 1 | 1 | 1 | 15 | **1** |

So, f(w, x, y, z) = ∑m(0, 1, 2, 3, 6, 9, 12, 13, 14, 15) = w’x’ + w’yz’ + wx + wy’z

