**Q5.** If data = 0xA543, what is the value of result after the following System Verilog statement?

**result = | (data[15:0] & 16'hC820);**

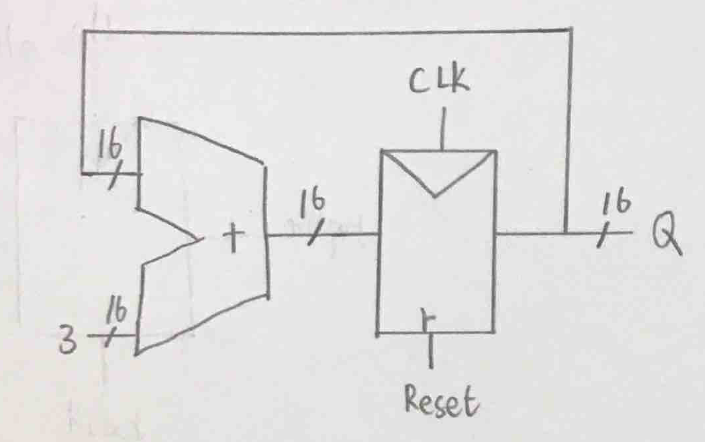
data  =  1010 0101 0100 0011

C820 = 1100 1000 0010 0000

result = 1000 0000 0000 0000

result = 0x8000

**Q7.** Design a 16-bit counter that adds 3 at each clock edge. The counter has reset and clock inputs. Upon reset, the counter output is all 0.



module counter16bit\_add3 ( input logic clk, logic rst,

output logic[15:0]):

always\_ff@(posedge clk, posedge rst)

if(rst) q <= 0;

else q <= q +3;

endmodule

Q8. Write a test bench code to test the following BCD Counter:

 module bcdadd\_8(input logic [7:0] a, b,  
input logic cin,  
output logic [7:0] s,  
output logic cout);

logic c0;

bcdadd\_4 bcd0(a[3:0], b[3:0], cin, s[3:0], c0);  
bcdadd\_4 bcd1(a[7:4], b[7:4], c0, s[7:4], cout);

endmodule

module bcdadd\_4(input logic [3:0] a, b,  
input logic cin,  
output logic [3:0] s,  
output logic cout);

logic [4:0] result, sub10;

assign result = a + b + cin;  
assign sub10 = result - 10;  
assign cout = ~sub10[4];  
assign s = sub10[4] ? result[3:0] : sub10[3:0];

endmodule

module testbench8bcd();

logic clk;  
logic [7:0] a, b, s, s\_expected;  
logic cin,cout,cout\_expected;

logic [31:0] vectornum;  
logic [25:0] testvectors[10000:0];

bcdadd\_8 dut(a,b,cin,s,cout);

always  
begin  
clk = 1; #5; clk = 0; #5;  
end

initial  
begin  
$readmemb("bcd8test.tv", testvectors);  
vectornum = 0;  
end

always @(posedge clk)  
begin  
#1; {a, b, cin, s\_expected, cout\_expected} = testvectors[vectornum];  
end

//check results  
always @(negedge clk)  
if (s ! == s\_expected or cout ! == cout\_expected) begin  
$display("Error: inputs = %b", {a, b, cin});

end  
vectornum = vectornum + 1;  
if (testvectors[vectornum] === 26'bx) begin  
$display("tests completed");  
$finish;  
end

endmodule

//////////////////////////////bcd8test.tv///////////////////////////////////

00000000\_00000000\_0\_00000000\_0

00000000\_00000001\_0\_00000001\_0

.

.

10011001\_10011001\_1\_10011001\_1

Q11. Consider memory storage of a 32-bit word stored at memory word 15 in a byte-addressable memory.  
(a) What is the byte address of memory word 15?  
(b) What are the byte addresses that memory word 15 spans?

a) 15\*4 = 60 = 0011\_1100 = 0x3C

b) 0x3C, 0x3D, 0x3E, 0x3F

**Q12.** Convert the following MIPS assembly code into machine language.  Write the instructions in hexadecimal.

add $t0, $s0, $s1  
lw $t0, 0x20($t7)  
addi $s0, $0, −10

add $t0, $s0, $s1 : 0x02114020

lw $t0, 0x20($t7): 0x8DE80020

addi $s0, $0, −10: 0x2010fff6

Q13. Implement the following high-level code segments using the slt instruction. Assume the integer variables g and h are in registers $s0 and $s1, respectively.

if (g >= h)  
g = g + 1;  
else  
h = h − 1;

slt $t0, $s0, $s1 # if g < h, $t0 = 1

bne $t0, $0, else # if $t0 != 0, do else

addi $s0, $s0, 1 # g = g + 1

j done # jump past else block

else: addi $s1, $s1, -1 # h = h - 1

done: