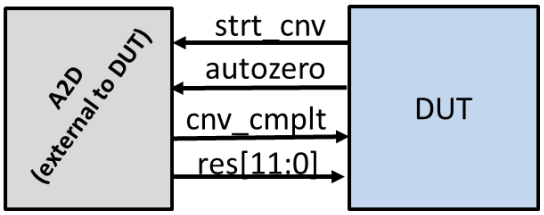
Name: _____

FINAL EXAM

Closed book except for one 8½ x 11 sheet. No calculators or electronics devices.

1. **(10pts)** Circle True of False for each statement
   a. **(True/False)** Within a **function**: event, delay, or timing control statements are not permitted
   b. **(True/False)** For a primary input Synopsys has a decent estimate of the drive strength of the signal, but needs constraint information about the capacitive loading.
   c. **(True/False)** Concurrent assertions can be useful for checking adherence to protocols
   d. **(True/False)** Bottom up compile strategy is only useful for large designs.
   e. **(True/False)** The TSMC wireload model we used in this course was incomplete because it did not estimate area impact, only timing.
   f. **(True/False)** The RC delay of chip interconnect is less of an issue in modern processes because the chips are smaller, hence the routes are shorter.
   g. **(True/False)** A SDF file models the delay of every gate in the design, as well as providing flight time information for the interconnect.
   h. **(True/False)** Architectural optimizations/choices have a bigger impact on area/speed than gate level optimizations.
   i. **(True/False)** A FPGA implementation of the given function will be lower power than an ASIC implementation.
   j. **(True/False)** If the person running your APR tool is good, they will ensure the layout of your design has no parasitic capacitances.

2. **(7pts)** Your DUT is getting conversion results from an A2D converter. The signal ***autozero*** is supposed to be asserted for the first 100 samples after reset. Either write a **task** called **checkAZ()** that is called after reset deasserts, or show how you could do this with a concurrent assertion. If you choose a task the task must employ **fork/join**



3. (6pts) Match the unit name to the description (draw arrows).

Auth_blk	Made use of load cell data to perform its main function.
inertial_integrator	Produced an 11-bit vector that determined both left & right motor speeds.
balance_cntrl	Contained an instance of **UART_rcv**
steer_en	Contained an instance of **PWM11**
A2D_intf	Acquired load cell data
mtr_drv	Performed sensor fusion
piezo	

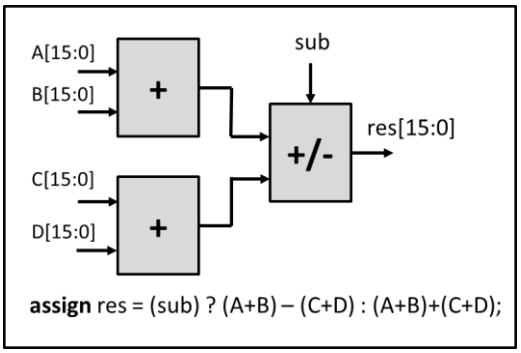
4. (7 pts) draw how you think this code would Synthesize. This code is not a preferred style in my opinion, but it does synthesize.

```
module ssum
  (output logic [5:0] sum,
  input wire [2:0] mask,
  input wire [3:0] a[0:2] );  // array of 3 vectors

  always @* begin
    sum = 0;
    // Hint: for loop implies replication
    // of hardware
    for ( int i=0; i<3; i++ )
      if ( mask[i] )
        sum += a[i];
  end
endmodule
```

5. (4 pts) Describe something that a parameter can be used for that an ordinary input port cannot and something that an input port can be used for that a parameter cannot.

6. (10 pts) The code shown is likely to synthesize as shown. Draw how you would want it to synthesize if ***sub*** was a late arriving signal. Draw how you would want it to synthesize if ***A[15:0]*** was the late arriving signal. For each case show how you would change the coding to make that occur.



7. **(9 pts)** Create a testbench to read and write to a sensor that uses a SPI protocol very similar to the inertial sensor we used on the project. Complete the testbench to write desired value of 0x5A to address 0x1B. Then read from address 0x1B and check that it has the correct value. You do not need to test functionality of SPI_mnrch. You do not have to show the full instantiation of SPI_mnrch (can just use … to indicate you are connecting all the signals). Write small.

Sensor SPI Interface Protocol:	**SPI_mstr16 Interface:**
First bit sent through SPI is the R/~W bit (1 is read, 0 is write)	**Inputs:** clk, rst_n, MISO, cmd[15:0], send_cmd
Next 7 bits are address	**Outputs:** MOSI, SCLK, SS_n, done, rd_data[15:0]
Next 8 bits are data to be written when a write, and of no consequence if it is a read.	

```
initial begin
  rst_n = 0;
  clk = 0;
  send_cmd = 0;
  @(posedge clk);
  @(negedge clk);
  rst_n = 1;      // deassert reset









end
```

8. If clock gating is successfully employed on a functional block the resulting version is: (circle one in each row) **(4.5pts)**

i.	Lower power	Higher power	Same Power
ii.	Lower area	Higher Area	Same Area
iii.	Lower Latency	Higher Latency	Same Latency

9. (4pts) Comment on the speed path that was discovered when you synthesized the project, and how you solved it.

10. (6pts) When passing a signal to a task you can pass it as simply **input**, or as **ref**. Describe the difference between the two and when you would choose **ref** vs **input**.

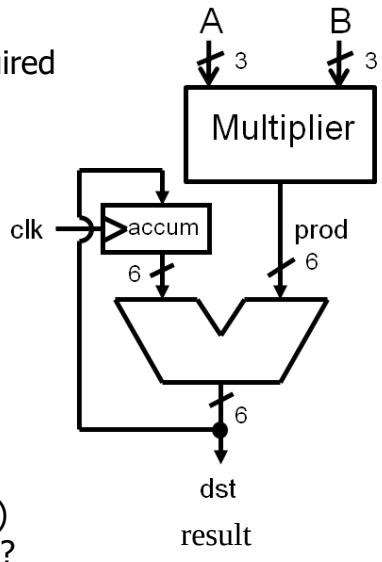
11. (5pts) (4pts) For the following Verilog code draw the circuit synthesis will create. Do you think this was the intended result?

```
module back2back (output reg q2,
input d, clk, rst_n);
  reg q1,q2;
  always @(posedge clk)
    if (!rst_n)
      q1 <= 1'b0;
    else begin
      q1 <= d;
      q2 <= q1;
    end
endmodule
```

12. (4pts) How many arithmetic blocks will be inferred after doing a: "`read_file -format verilog`" on the following code snippet? How many arithmetic blocks do you think will be utilized in the final synthesis results?

```
always_comb
  if(op==2'b00)
    C = A + B;
  else if (op==2'b01)
    C = A - B;
  else
    C = B - A;
```

13. (8 pts) Below two timing reports are given (max_delay & min_delay) for the datapath circuit shown. Answer the following questions:



a. How many ns after the rising edge of the clock is required before
we know input **A** is valid.

b. As specified can this design run at 500MHz?

c. What is the best case (fastest) clk2q timing of accum_reg[0]

d. What would be the max and min delay margins (slack) If the magnitude of clock skew was increased by 0.2ns?

max_delay

Point	Incr	Path
clock clk (rise edge)	0.00	0.00
clock network delay (ideal)	0.00	0.00
input external delay	0.45	0.45
A[1] (in)	0.00	0.45
mult_17/a[1] (mac_DW_mult_uns_0)	0.00	0.45
mult_17/U33/Z (N1M1P)	0.03	0.48
mult_17/U31/Z (NR2M1P)	0.03	0.51
mult_17/U7/S (HA1M1P)	0.14	0.65
mult_17/U4/S (FA1AM1P)	0.14	0.79
mult_17/product[2] (mac_DW_mult_uns_0)	0.00	0.79
add_18/A[2] (mac_DW01_add_0)	0.00	0.79
add_18/U1_2/CO (FA1AM1P)	0.11	0.89
add_18/U1_3/CO (FA1AM1P)	0.10	0.99
add_18/U1_4/CO (FA1AM1P)	0.10	1.08
add_18/U1_5/S (FA1AM1P)	0.13	1.21
add_18/SUM[5] (mac_DW01_add_0)	0.00	1.21
result[5] (out)	0.00	1.21
data arrival time		1.21
clock clk (rise edge)	2.00	2.00
clock network delay (ideal)	0.00	2.00
clock uncertainty	-0.01	1.99
output external delay	-0.30	1.69
data required time		1.69
data required time		1.69
data arrival time		-1.21
slack (MET)		0.48

min_delay

```
Startpoint: accum_reg[0]
            (rising edge-triggered flip-flop clocked by clk
Endpoint: accum_reg[0]
          (rising edge-triggered flip-flop clocked by clk)
Path Group: clk
Path Type: min

Des/Clust/Port     Wire Load Model       Library
------------------------------------------------
mac                B0.1X0.1              gflxp

Point                                    Incr       Path
-----------------------------------------------------------
clock clk (rise edge)                    0.00       0.00
clock network delay (ideal)              0.00       0.00
accum_reg[0]/CP (FD2QM1P)                0.00       0.00 r
accum_reg[0]/Q (FD2QM1P)                 0.11       0.11 f
add_18/B[0] (mac_DW01_add_0)             0.00       0.11 f
add_18/U2/Z (EOFM1P)                     0.04       0.14 r
add_18/SUM[0] (mac_DW01_add_0)           0.00       0.14 r
accum_reg[0]/D (FD2QM1P)                 0.00       0.14 r
data arrival time                                   0.14

clock clk (rise edge)                    0.00       0.00
clock network delay (ideal)              0.00       0.00
clock uncertainty                        0.01       0.01
accum_reg[0]/CP (FD2QM1P)                0.00       0.01 r
library hold time                        0.10       0.11
data required time                                  0.11
-----------------------------------------------------------
data required time                                  0.11
data arrival time                                  -0.14
-----------------------------------------------------------
slack (MET)                                         0.03
```

14. Match each of the original circuits to their optimized equivalent. Then for select ones (not gray) describe what the optimization was (why is it better?)

Original	Arrow	Optimized	Optimization Desscription: