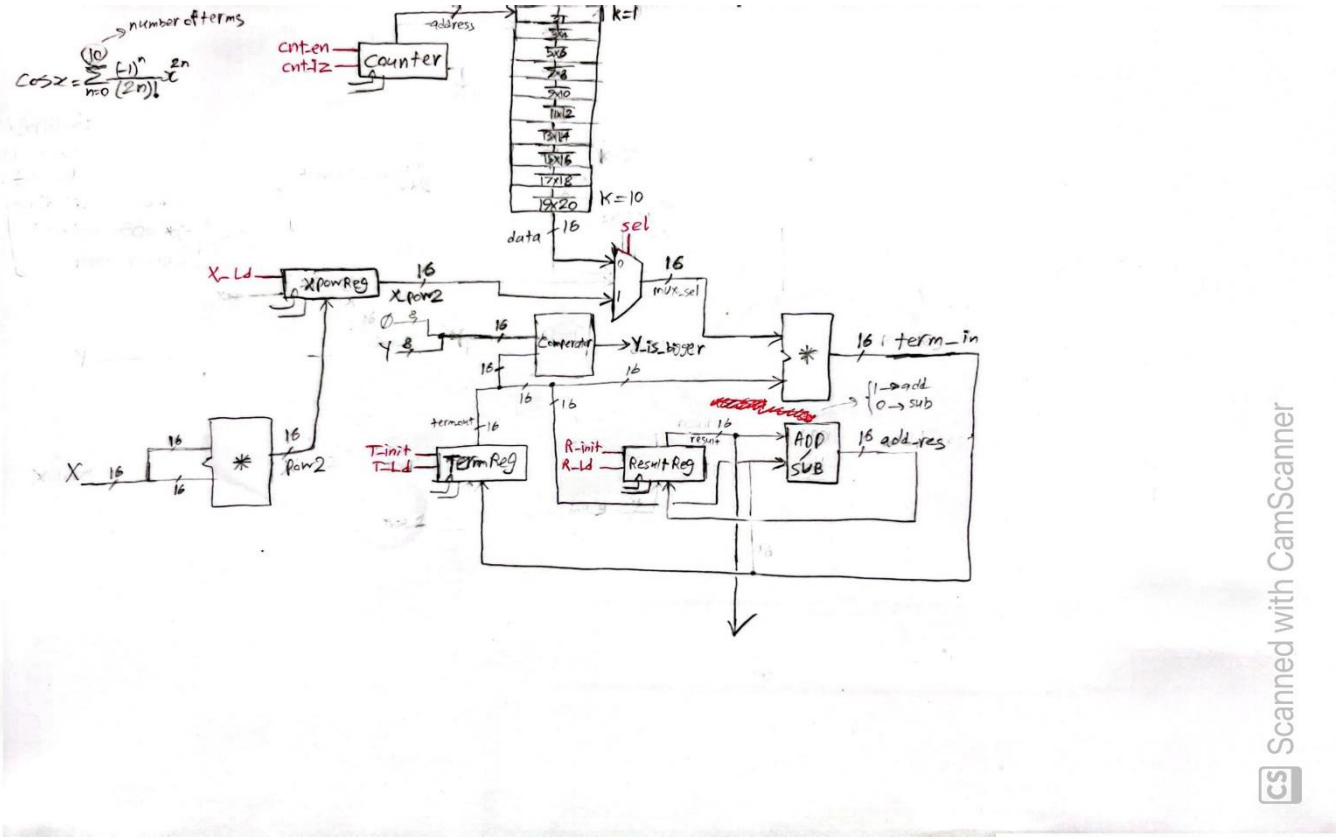
Amin Aghakasiri 810101381

CA6 Report

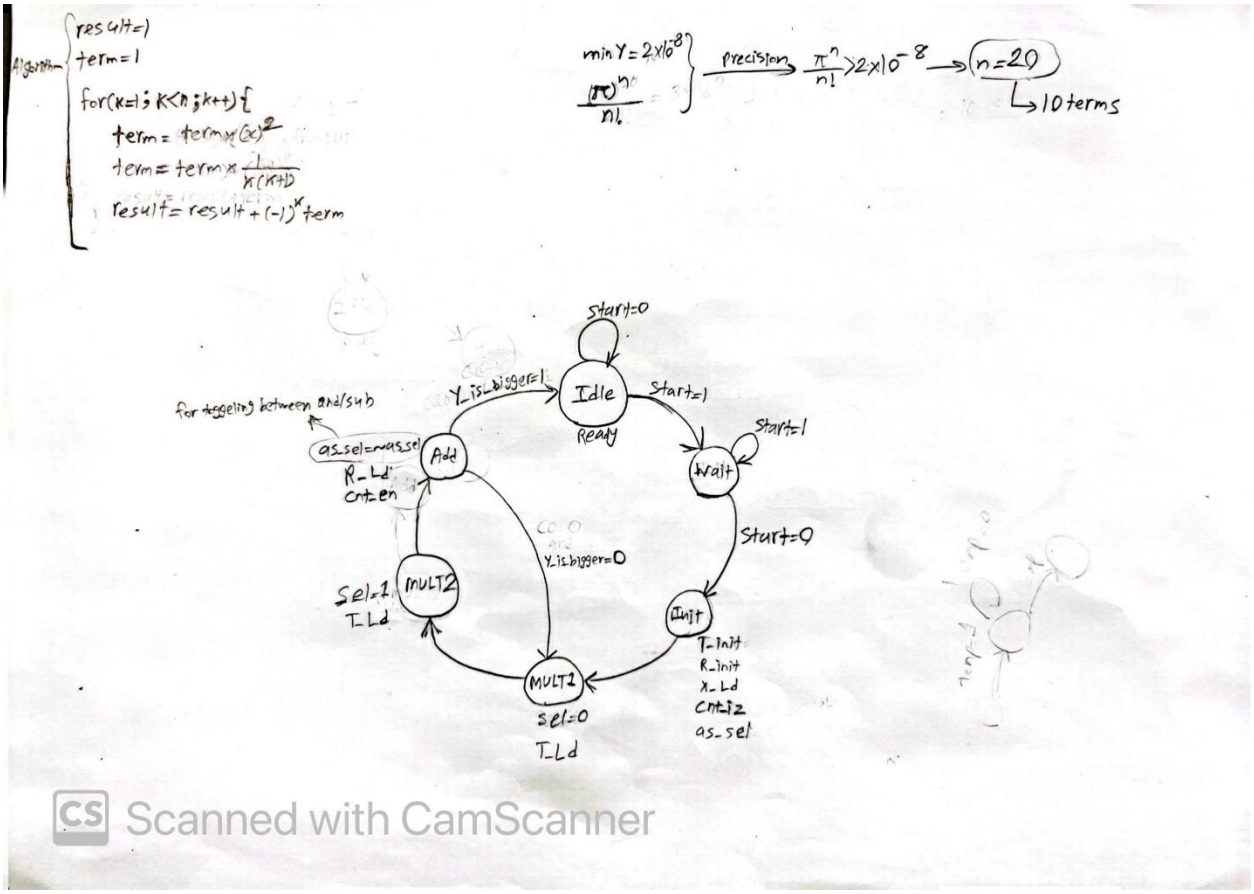
Design Phase:

Part 1 and 3:

Datapath design:



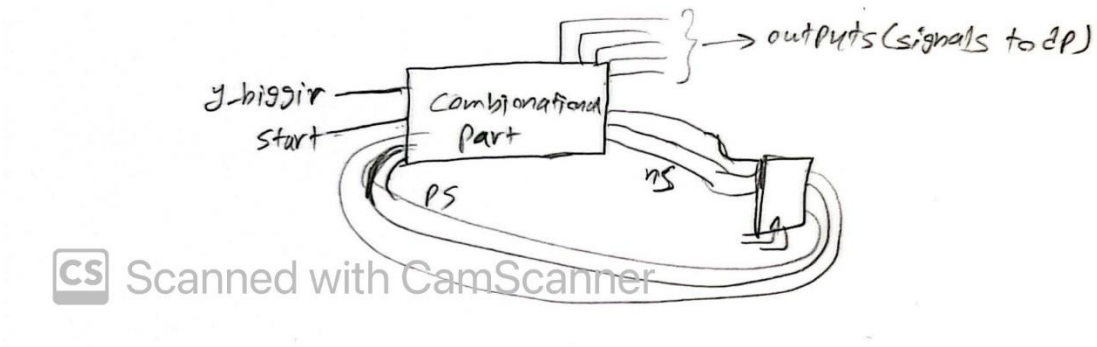
Part 2:


Algorithm
result=1
term=1
for(k=1; k<n; k++){
term = term*(x)^2
term = term*1/(k(k+1))
result = result + (-1)^k term
min Y = 2x10^-8
precision
n=20
10 terms
start=0
Idle
Ready
Y_is_bigger=1
start=1
for toggling between add/sub
as_sel=mass
R_Ld
cnt_en
Add
wait
start=1
start=0
Y_is_bigger=0
sel=1
T_Ld
MULT2
Init
T_init
R_init
x_Ld
cnt_12
as_sel
MULT2
sel=0
T_Ld
Scanned with CamScanner

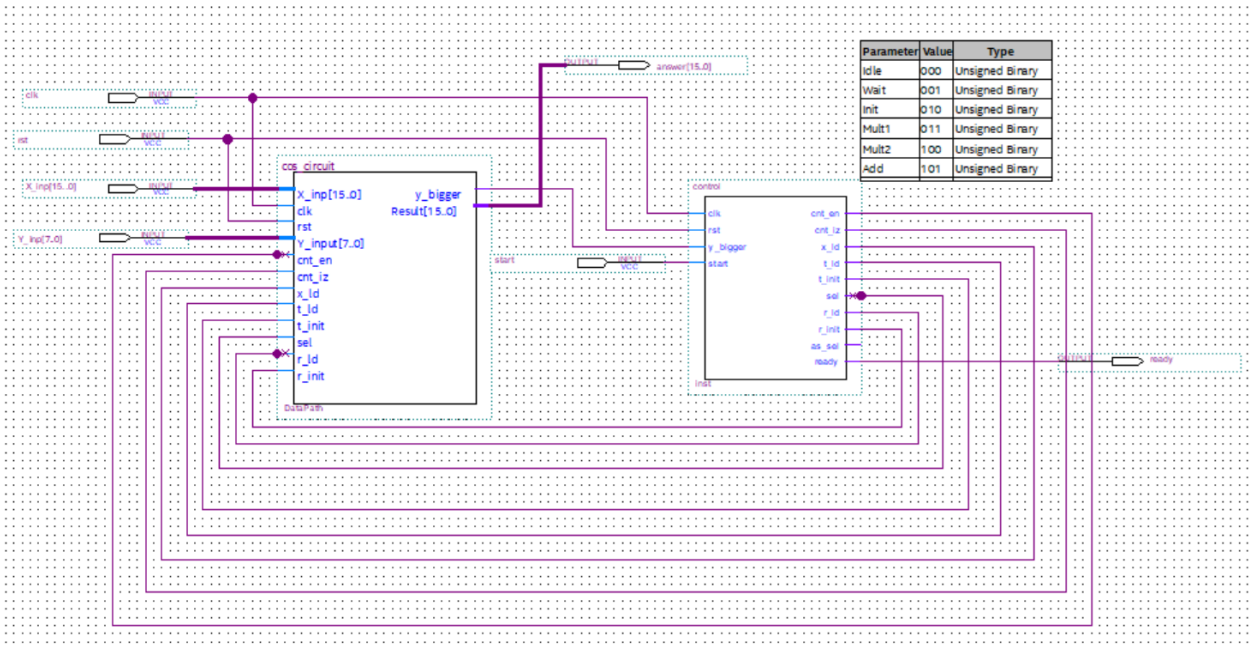
Part 4:

```
1  module controller (input clk,rst,y_bigger,start, output reg cnt_en,cnt_iz,x_ld,t_ld,t_init,sel,r_ld,r_init,as_sel,ready);
2      reg[2:0] ns,ps;
3      parameter [2:0]  Idle = 0, Wait = 1,  Init = 2, Mult1 = 3, Mult2 = 4, Add = 5;
4      always @(ps,y_bigger) begin
5          {cnt_en,cnt_iz,x_ld,t_ld,t_init,sel,r_ld,r_init,as_sel,ready} = 10'b0;
6          case(ps)
7          Idle : begin ns = start ? Wait : Idle; ready = 1'b1; end
8          Wait : begin ns = start ? Wait : Init; end
9          Init : begin ns = Mult1; t_init = 1'b1; r_init = 1'b1; x_ld = 1'b1; cnt_iz = 1'b1; end
10         Mult1 : begin ns = Mult2; sel = 1'b0; t_ld = 1'b1; end
11         Mult2 : begin ns = Add; sel = 1'b1; t_ld = 1'b1; end
12         Add : begin ns = y_bigger ? Idle : Mult1; r_ld = 1'b1; cnt_en = 1'b1; end
13         endcase
14     end
15     always @(posedge clk,posedge rst) begin
16         if(rst)
17             ps <= Idle;
18         else
19             ps <= ns;
20     end
21 endmodule
```

From line 4 to 14, we have the combinational part of the controller and from line 15 to 20, we have the sequential part of the controller. Here is the Hoffman model of the controller:



Part 5:



Implementation Phase:

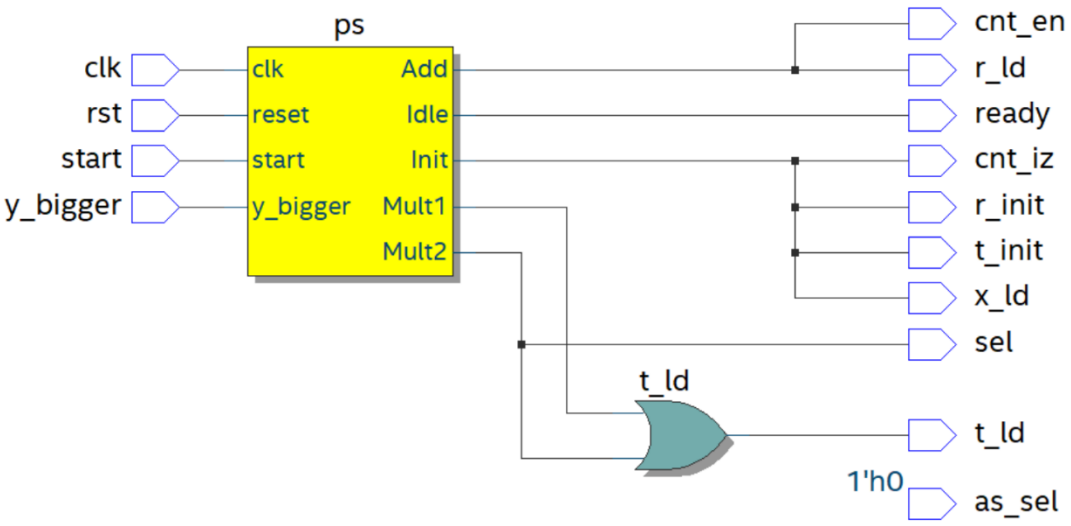
Controller part:

Code:

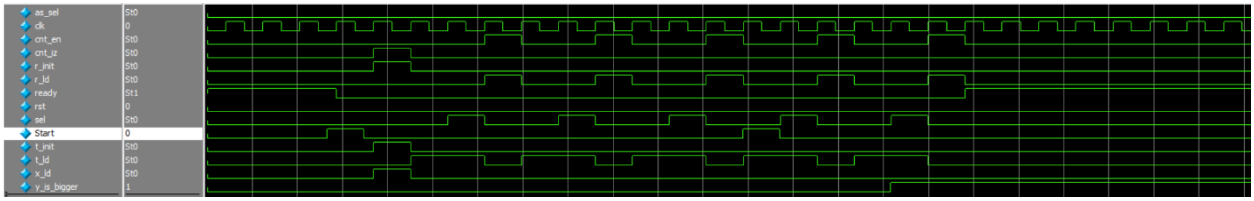
```
module controller (input clk,rst,y_bigger,start, output reg cnt_en,cnt_iz,x_ld,t_ld,t_init,sel,r_ld,r_init,as_sel,ready);
    reg[2:0] ns,ps;
    parameter [2:0]  Idle = 0, Wait = 1,  Init = 2, Mult1 = 3, Mult2 = 4, Add = 5;
    always @(ps,y_bigger) begin
        {cnt_en,cnt_iz,x_ld,t_ld,t_init,sel,r_ld,r_init,as_sel,ready} = 10'b0;
        case(ps)
        Idle : begin ns = start ? Wait : Idle; ready = 1'b1; end
        Wait : begin ns = start ? Wait : Init; end
        Init : begin ns = Mult1; t_init = 1'b1; r_init = 1'b1; x_ld = 1'b1; cnt_iz = 1'b1; end
        Mult1 : begin ns = Mult2; sel = 1'b0; t_ld = 1'b1; end
        Mult2 : begin ns = Add; sel = 1'b1; t_ld = 1'b1; end
        Add : begin ns = y_bigger ? Idle : Mult1; r_ld = 1'b1; cnt_en = 1'b1; end
        endcase
    end
    always @(posedge clk,posedge rst) begin
        if(rst)
            ps <= Idle;
        else
            ps <= ns;
    end
endmodule
```

Number of used elements for building the circuit:

Flow Status	Successful - Mon Jan 08 18:39:13 2024
Quartus Prime Version	20.1.0 Build 711 06/05/2020 SJ Lite Edition
Revision Name	control
Top-level Entity Name	control
Family	Cyclone IV GX
Device	EP4CGX15BF14A7
Timing Models	Final
Total logic elements	7 / 14,400 (< 1 %)
Total registers	6
Total pins	14 / 81 (17 %)
Total virtual pins	0
Total memory bits	0 / 552,960 (0 %)
Embedded Multiplier 9-bit elements	0
Total GXB Receiver Channel PCS	0 / 2 (0 %)
Total GXB Receiver Channel PMA	0 / 2 (0 %)
Total GXB Transmitter Channel PCS	0 / 2 (0 %)
Total GXB Transmitter Channel PMA	0 / 2 (0 %)
Total PLLs	0 / 3 (0 %)

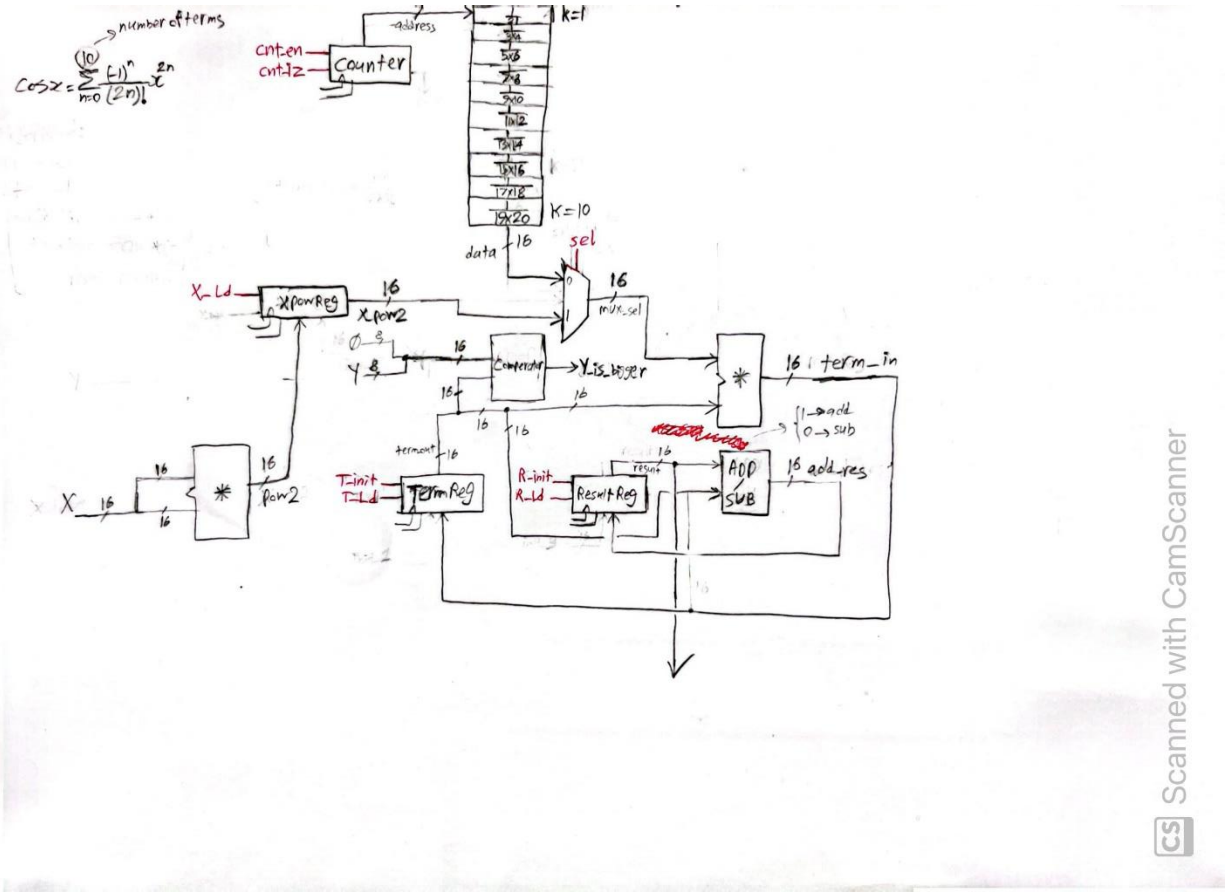


Controller Testbench:

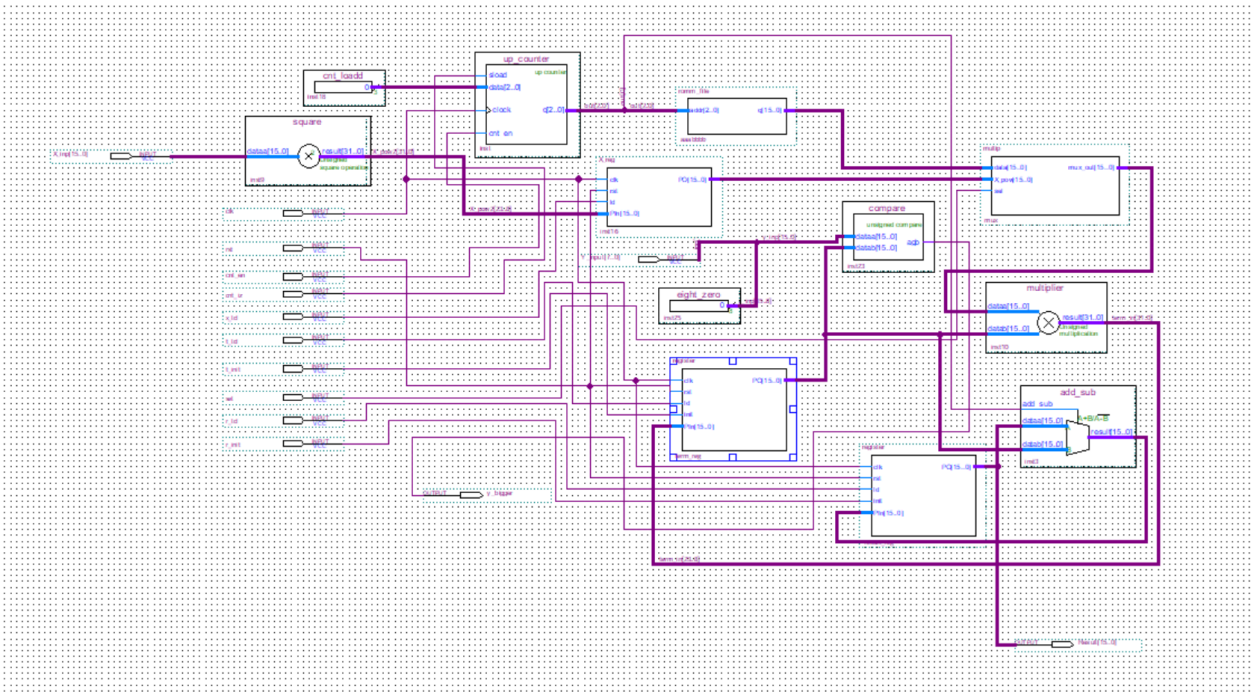


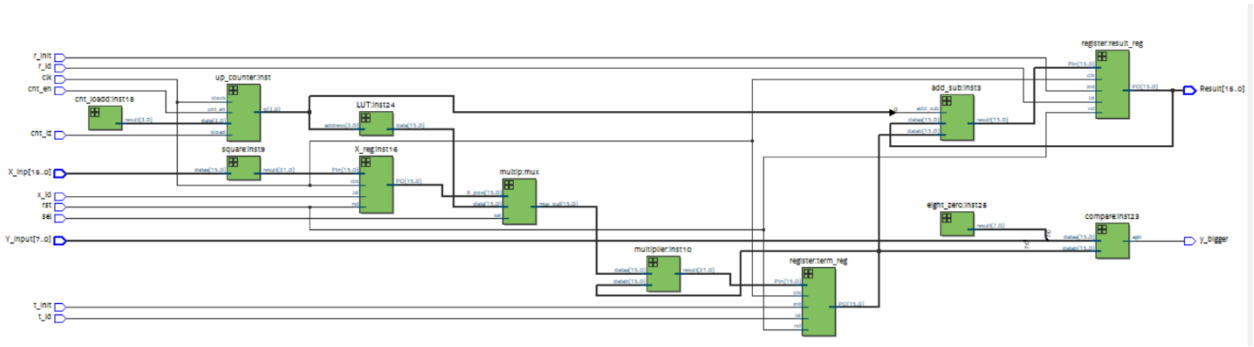
When y becomes bigger than term, we will be back to the Idle state and Ready will be 1.

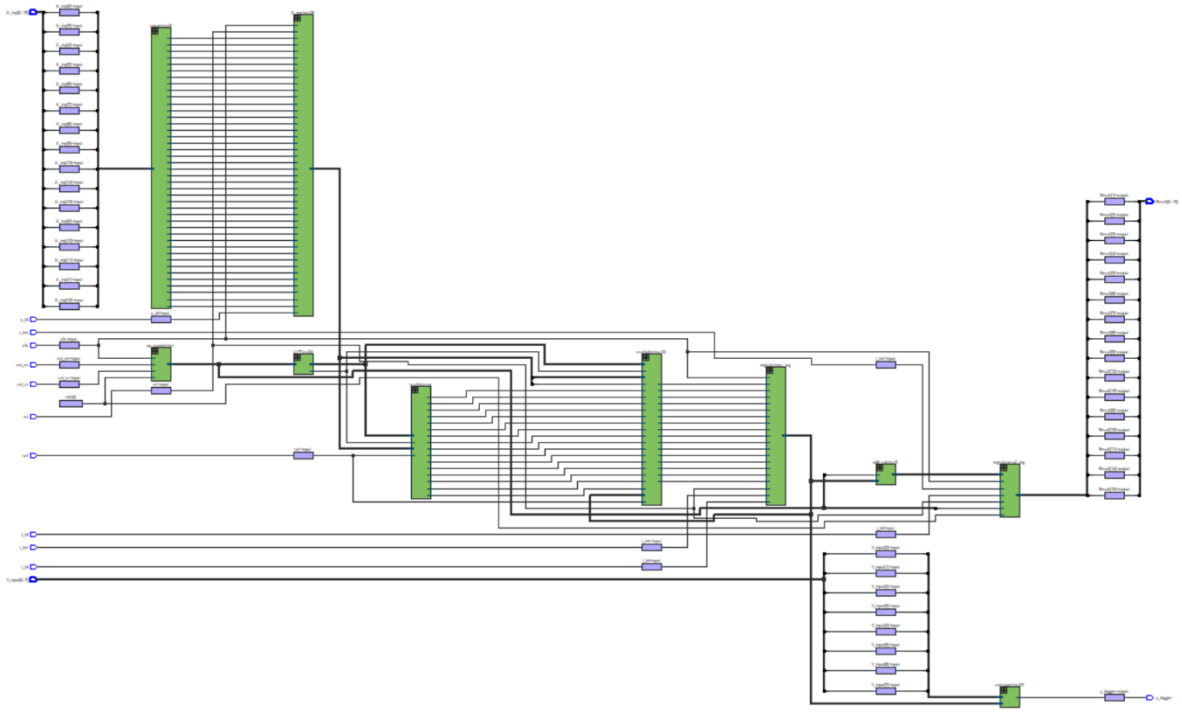
Datapath part:



Block diagram:




r_init
r_id
clk
cnt_en
cnt_load:inst11
cnt_ld
X_inp[15..0]
x_id
rst
sel
Y_input[7..0]
t_init
t_id
up_counter:inst
square:inst9
LUT:inst24
X_reg:inst16
multip:mux
multiplier:inst10
register:term_reg
add_sub:inst3
register:result_reg
Result[15..0]
eight_zero:inst28
compare:inst23
y_bigger



Number of used elements for building datapath:

Flow Status	Successful - Mon Jan 08 18:51:16 2024
Quartus Prime Version	20.1.0 Build 711 06/05/2020 SJ Lite Edition
Revision Name	cos_circuit
Top-level Entity Name	cos_circuit
Family	Cyclone IV GX
Device	EP4CGX15BF14A7
Timing Models	Final
Total logic elements	593 / 14,400 (4 %)
Total registers	52
Total pins	51 / 81 (63 %)
Total virtual pins	0
Total memory bits	0 / 552,960 (0 %)
Embedded Multiplier 9-bit elements	0
Total GXB Receiver Channel PCS	0 / 2 (0 %)
Total GXB Receiver Channel PMA	0 / 2 (0 %)
Total GXB Transmitter Channel PCS	0 / 2 (0 %)
Total GXB Transmitter Channel PMA	0 / 2 (0 %)
Total PLLs	0 / 3 (0 %)

```
module X_reg(input clk,rst,ld,input[15:0] PIn, output reg [15:0] PO);
    always @(posedge clk,posedge rst) begin
        if(rst)
            PO <= 16'b0;
        else
            PO <= (ld) ? PIn : PO;
    end
endmodule
```

```
module register(input clk,rst,ld,init,input[15:0] PIn, output reg [15:0] PO);
    always @(posedge clk,posedge rst) begin
        if(rst)
            PO <= 16'b0;
        else if(init)
            PO <= 16'b0000000011111111;
        else
            PO <= (ld) ? PIn : PO;
    end
endmodule
```

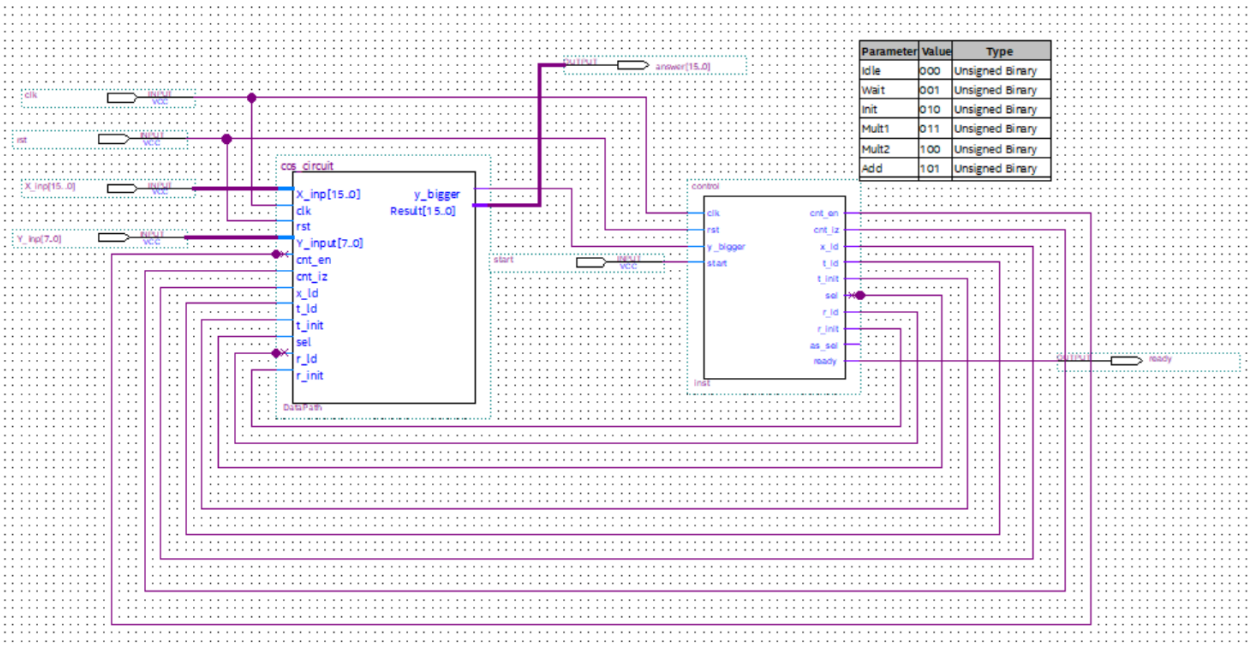
Look Up Table (LUT) Verilog code:

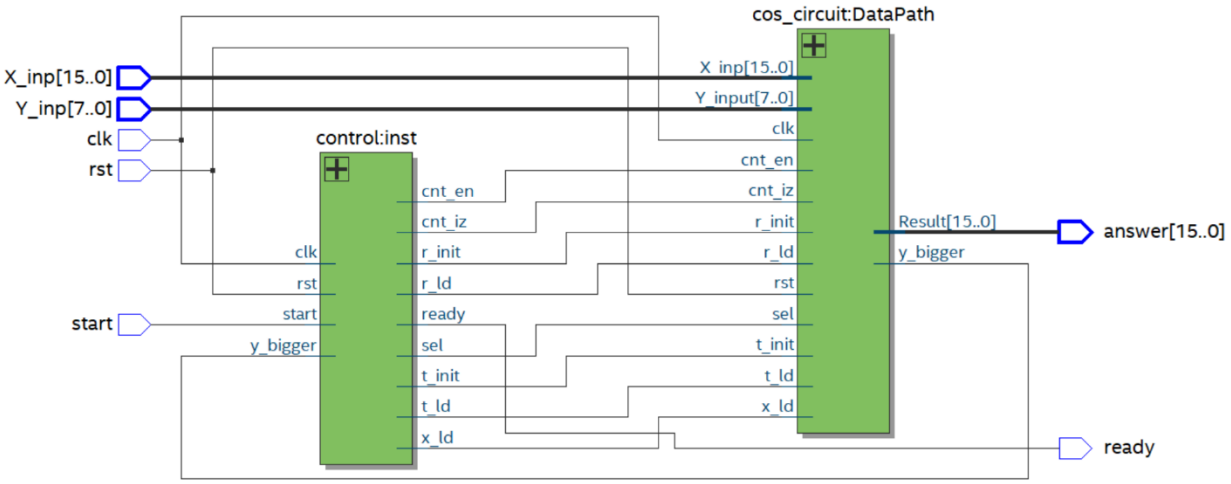
```
1   module  romm_file(
2   addr,
3   q
4   );
5   //-----------Input Ports---------------
6   input [2:0] addr ;
7   output reg [15:0] q ;
8
9   (* romstyle = "M9K" *)(* ram_init_file = "romm_file.mif" *)  reg  [15:0] rom [2:0];
10
11      always @ (addr)
12      begin
13         q <= rom[addr];
14      end
15
16  endmodule
```

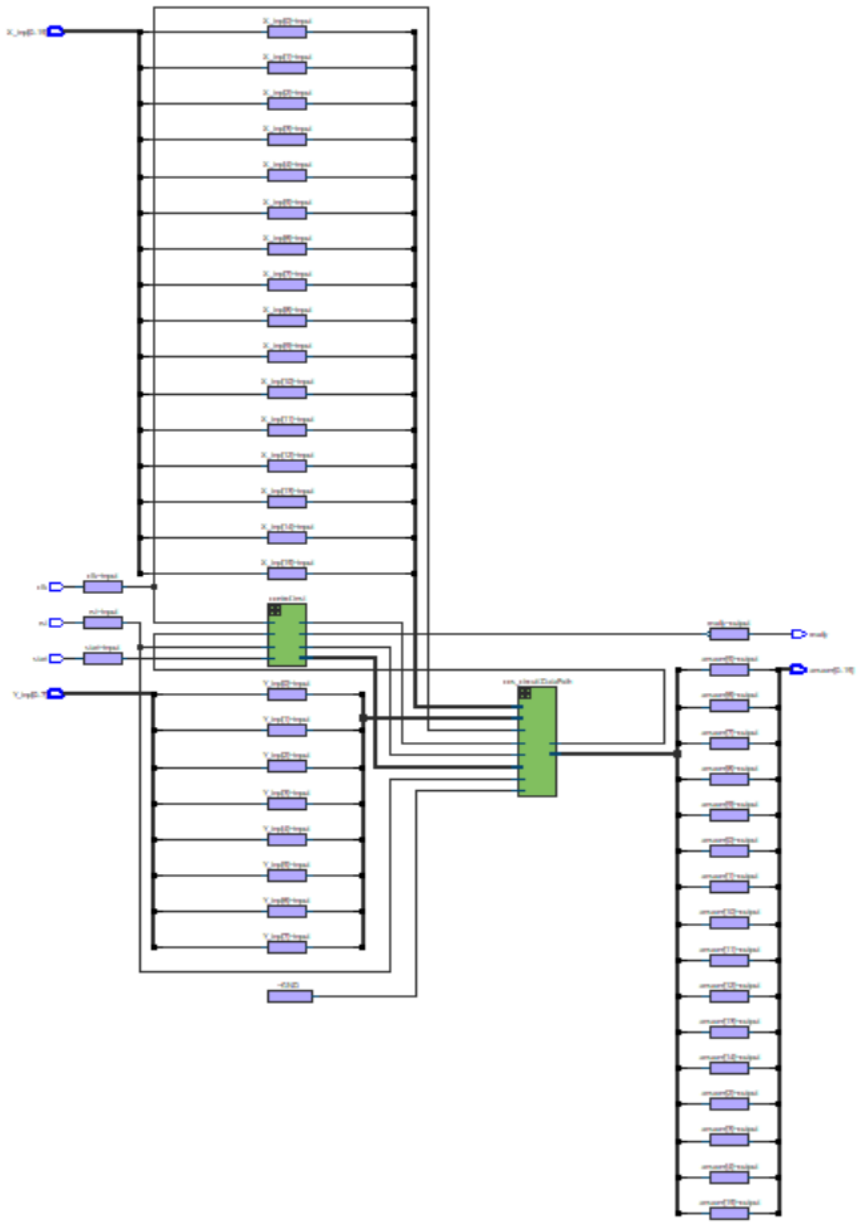
Addr	+0	+1	+2	+3	+4	+5	+6	+7	ASCII
0	0080	0015	0008	0004	0002	0001	0001	0001	.....

Final part:

Block diagram:




cos_circuit:DataPath
X_inp[15..0]
Y_inp[7..0]
clk
rst
start
control:inst
cnt_en
cnt_iz
r_init
r_ld
ready
sel
t_init
t_ld
x_ld
y_bigger
X_inp[15..0]
Y_input[7..0]
Result[15..0]
answer[15..0]

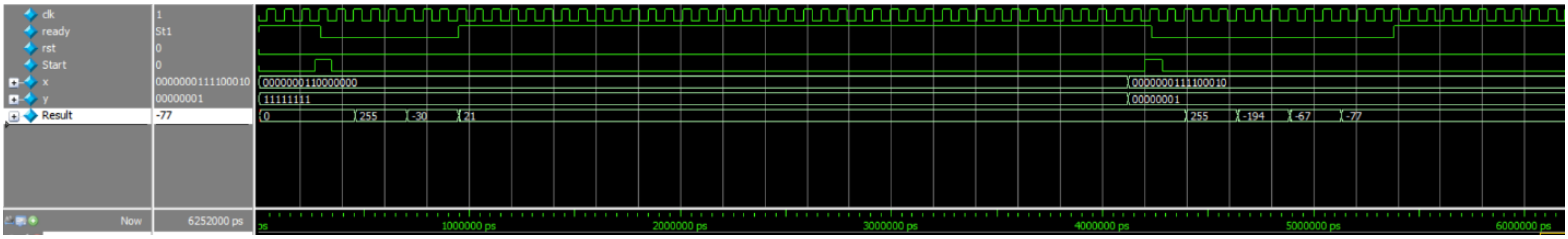


Number of used elements for building the whole circuit:

Flow Status	Successful - Mon Jan 08 19:13:24 2024
Quartus Prime Version	20.1.0 Build 711 06/05/2020 SJ Lite Edition
Revision Name	cosx
Top-level Entity Name	cosx
Family	Cyclone IV GX
Device	EP4CGX15BF14A7
Timing Models	Final
Total logic elements	593 / 14,400 (4 %)
Total registers	58
Total pins	44 / 81 (54 %)
Total virtual pins	0
Total memory bits	0 / 552,960 (0 %)
Embedded Multiplier 9-bit elements	0
Total GXB Receiver Channel PCS	0 / 2 (0 %)
Total GXB Receiver Channel PMA	0 / 2 (0 %)
Total GXB Transmitter Channel PCS	0 / 2 (0 %)
Total GXB Transmitter Channel PMA	0 / 2 (0 %)
Total PLLs	0 / 3 (0 %)

Testbench and waveform:

```
`timescale 1ns/1ns
module cosx_tb();
    reg Start=0,clk=0,rst=0;
    reg [7:0] y=8'b11111111;
    reg [15:0] x=16'b0000000110000000;//cos(1.5) -> exact answer: 0.0707 -> my answer: 0.082 (21/255 = 0.082)
    wire [15:0] Result;
    wire ready;
    cos_circuit UUT(ready,clk,rst,y,x,Start,Result);
    always begin #41;clk=~clk;end
    initial begin
        #82;
        #82;
        #82;
        #20; Start = 1;
        #82; Start=0;
        #1640;
        #82;
        #82;
        #1640;
        #82;
        #82;
        #82;#82; y=8'b00000001; x=16'b0000000111100010;//cos(1.882) -> exact answer: -0.306 -> my answer: -0.301 (The precision has increased with decreasing y)(-77/255 = -0.301)
        #82; Start = 1;
        #82; Start=0;
        #1640;
        #82;
        #82;
        #82;
        #82;
        $stop;
    end
endmodule
```



Chip planner: