Fast Fourier Transform (FFT) Implementation with Software and Hardware

2024-06-20

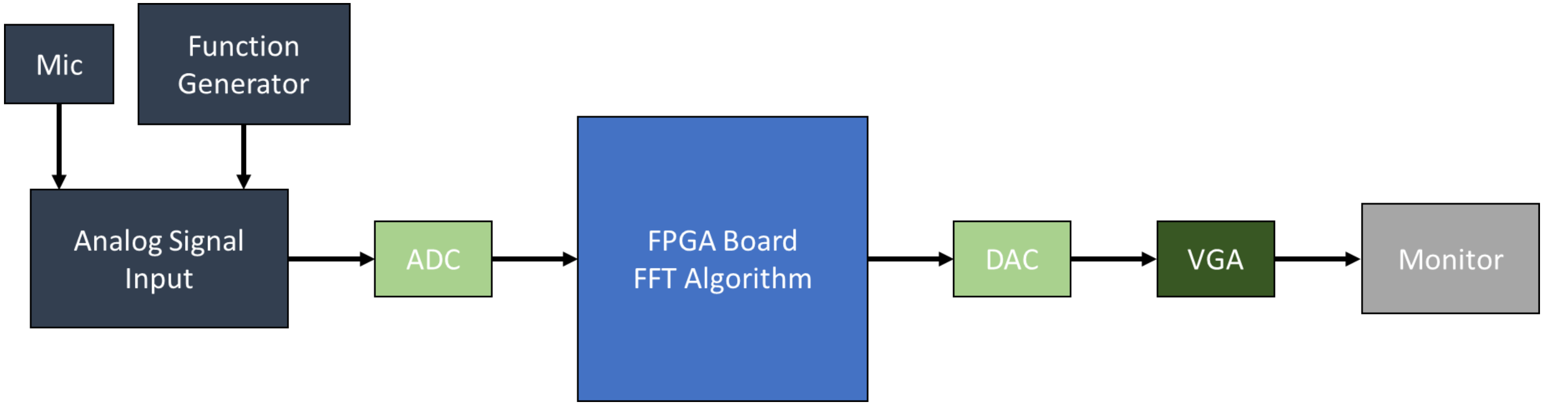
Jong Wan Ko (jongwanko@gmail.com)
Shin Dongho (dhooya99@daum.net)
Ahn Dongrin (ahndongring13@gmail.com)

2024 Capstone Design

Contents

- Overview
- Project Timeline
- Project Progress
- Further Improvements
- Q & A

Overview



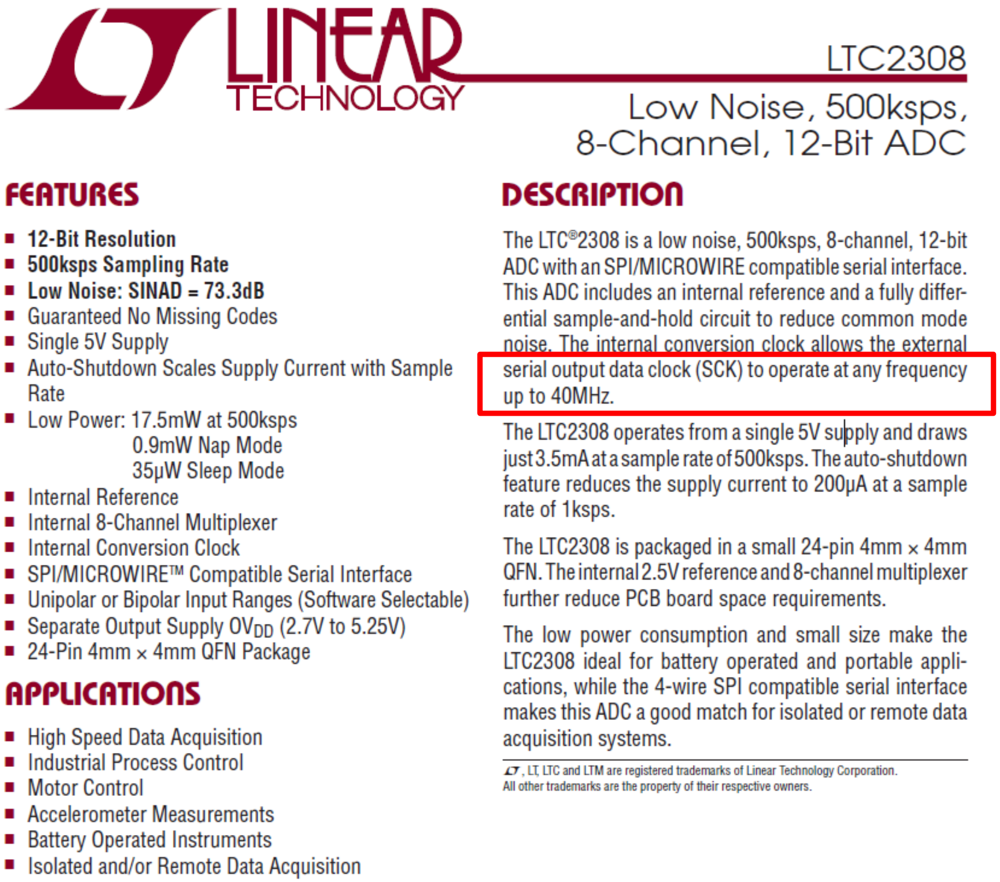
➢ Overall Project Diagram

Project Timeline

TIMELINE								
Phase	Details	2024 03/29	2024 04/05	2024 04/12	2024 04/19	2024 04/26	2024 05/03	2024 05/10
<Phase3> H/W Development	고종완	예비캡스톤 진행사항 리뷰	ADC Part-1 선행조사	ADC Part-2 개발 및 분석	Mid - Term		FFT 개발	
	신동호							
	안동린							
	기타							

TIMELINE						
Phase	Details	2024 05/17	2024 05/24	2024 05/31	2024 06/07	2024 06/20
<Phase3> H/W Development	고종완	Display 출력부 개발		통합 및 검증	Final Exams	Final Presentation
	신동호					
	안동린					
	기타					

Project Progress - ADC Controller



LINEAR TECHNOLOGY

LTC2308

Low Noise, 500ksps, 8-Channel, 12-Bit ADC

FEATURES

- **12-Bit Resolution**
- **500ksps Sampling Rate**
- **Low Noise: SINAD = 73.3dB**
- Guaranteed No Missing Codes
- Single 5V Supply
- Auto-Shutdown Scales Supply Current with Sample Rate
- Low Power: 17.5mW at 500ksps
  0.9mW Nap Mode
  35μW Sleep Mode
- Internal Reference
- Internal 8-Channel Multiplexer
- Internal Conversion Clock
- SPI/MICROWIRE™ Compatible Serial Interface
- Unipolar or Bipolar Input Ranges (Software Selectable)
- Separate Output Supply OV_{DD} (2.7V to 5.25V)
- 24-Pin 4mm × 4mm QFN Package

APPLICATIONS

- High Speed Data Acquisition
- Industrial Process Control
- Motor Control
- Accelerometer Measurements
- Battery Operated Instruments
- Isolated and/or Remote Data Acquisition

DESCRIPTION

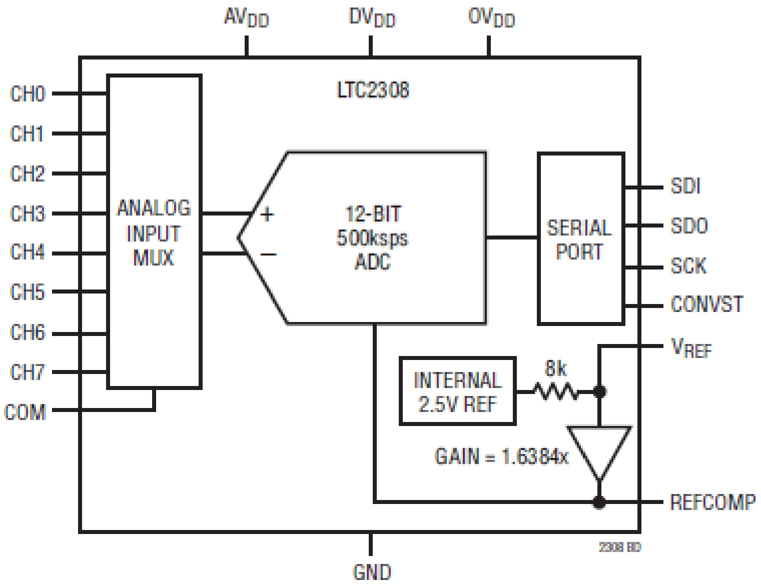
The LTC®2308 is a low noise, 500ksps, 8-channel, 12-bit ADC with an SPI/MICROWIRE compatible serial interface. This ADC includes an internal reference and a fully differential sample-and-hold circuit to reduce common mode noise. The internal conversion clock allows the external serial output data clock (SCK) to operate at any frequency up to 40MHz.

The LTC2308 operates from a single 5V supply and draws just 3.5mA at a sample rate of 500ksps. The auto-shutdown feature reduces the supply current to 200μA at a sample rate of 1ksps.

The LTC2308 is packaged in a small 24-pin 4mm × 4mm QFN. The internal 2.5V reference and 8-channel multiplexer further reduce PCB board space requirements.

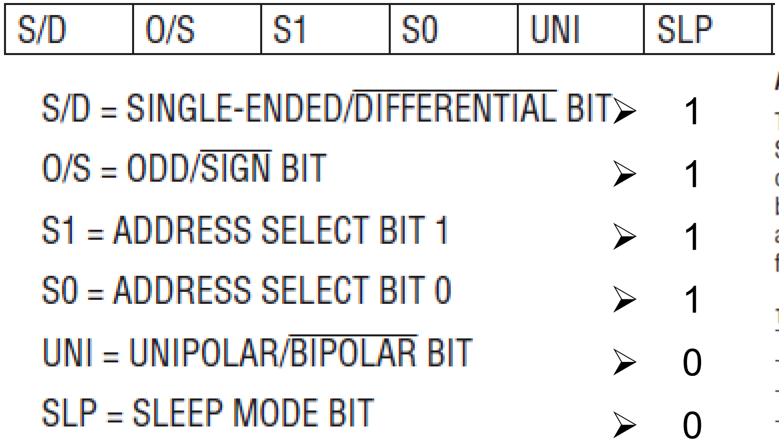
The low power consumption and small size make the LTC2308 ideal for battery operated and portable applications, while the 4-wire SPI compatible serial interface makes this ADC a good match for isolated or remote data acquisition systems.

LT, LTC and LTM are registered trademarks of Linear Technology Corporation. All other trademarks are the property of their respective owners.



- ➢ ADC Datasheet Summary

Project Progress – ADC Controller (Cont.)



S/D	O/S	S1	S0	UNI	SLP

S/D = SINGLE-ENDED/$\overline{\text{DIFFERENTIAL}}$ BIT ➢ 1

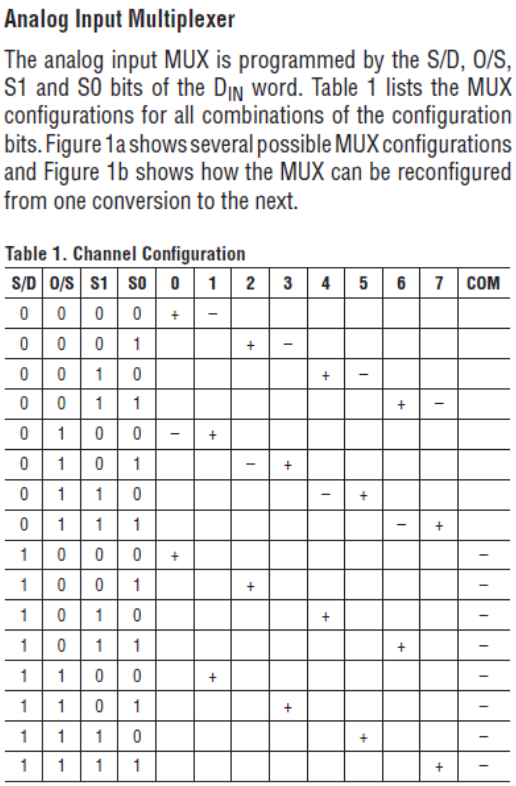
O/S = ODD/$\overline{\text{SIGN}}$ BIT ➢ 1

S1 = ADDRESS SELECT BIT 1 ➢ 1

S0 = ADDRESS SELECT BIT 0 ➢ 1

UNI = UNIPOLAR/$\overline{\text{BIPOLAR}}$ BIT ➢ 0

SLP = SLEEP MODE BIT ➢ 0



Analog Input Multiplexer

The analog input MUX is programmed by the S/D, O/S, S1 and S0 bits of the D_{IN} word. Table 1 lists the MUX configurations for all combinations of the configuration bits. Figure 1a shows several possible MUX configurations and Figure 1b shows how the MUX can be reconfigured from one conversion to the next.

Table 1. Channel Configuration

S/D	O/S	S1	S0	0	1	2	3	4	5	6	7	COM
0	0	0	0	+	–							
0	0	0	1			+	–					
0	0	1	0					+	–			
0	0	1	1							+	–	
0	1	0	0	–	+							
0	1	0	1			–	+					
0	1	1	0					–	+			
0	1	1	1							–	+	
1	0	0	0	+								–
1	0	0	1			+						–
1	0	1	0					+				–
1	0	1	1							+		–
1	1	0	0		+							–
1	1	0	1				+					–
1	1	1	0						+			–
1	1	1	1								+	–

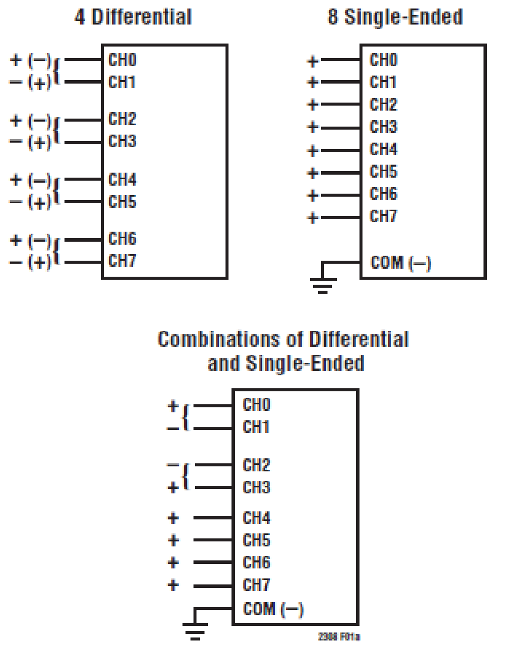


Figure 1a. Example MUX Configurations

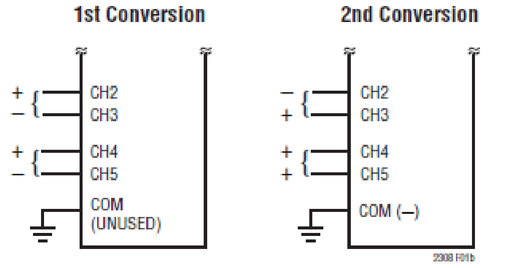


Figure 1b. Changing the MUX Assignment "On the Fly"

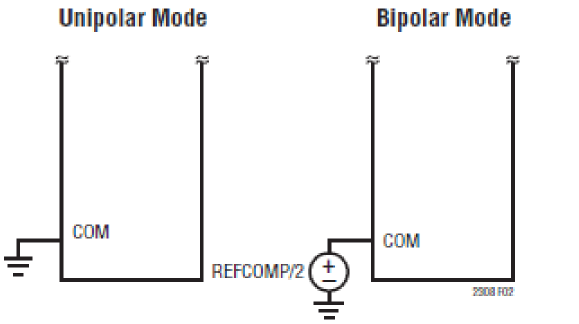


Figure 2. Driving COM in UNIPOLAR and BIPOLAR Modes

➢ ADC Datasheet Configuration

Project Progress – ADC Controller (Cont.)

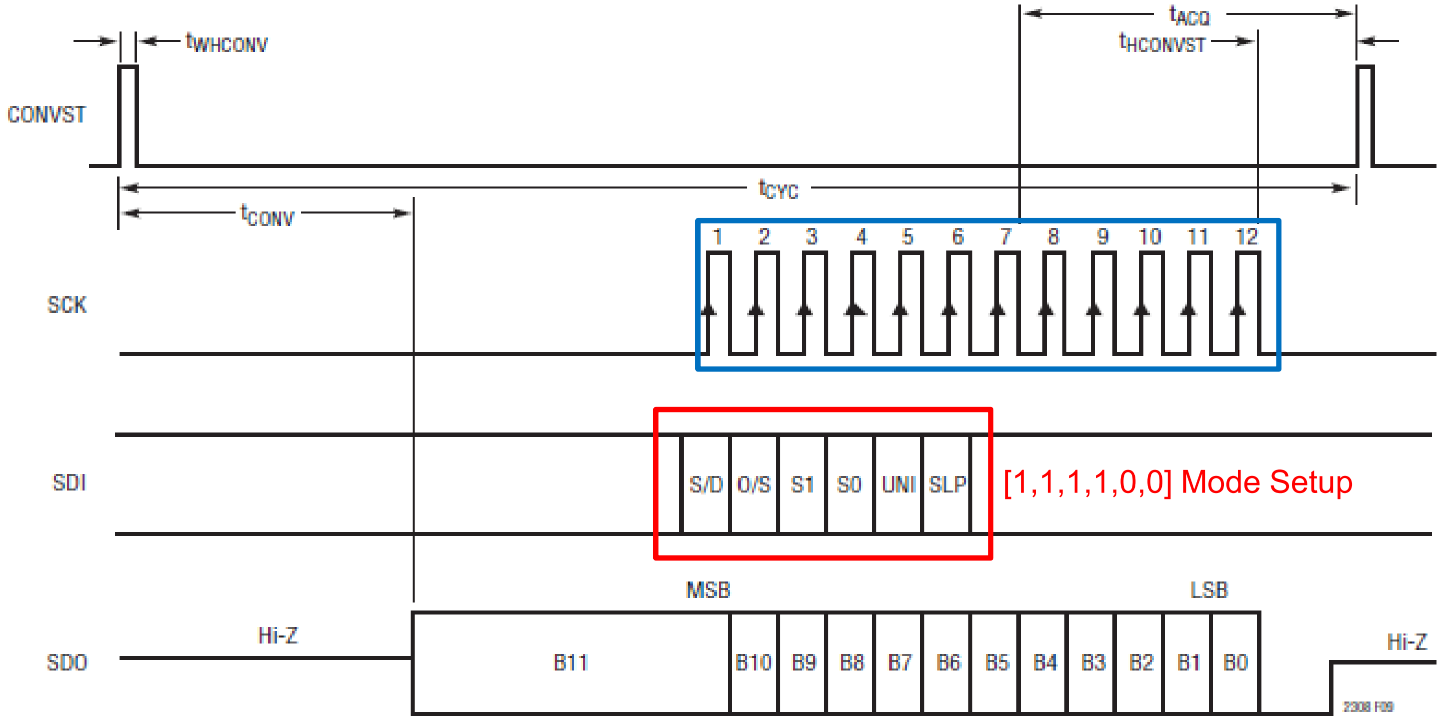
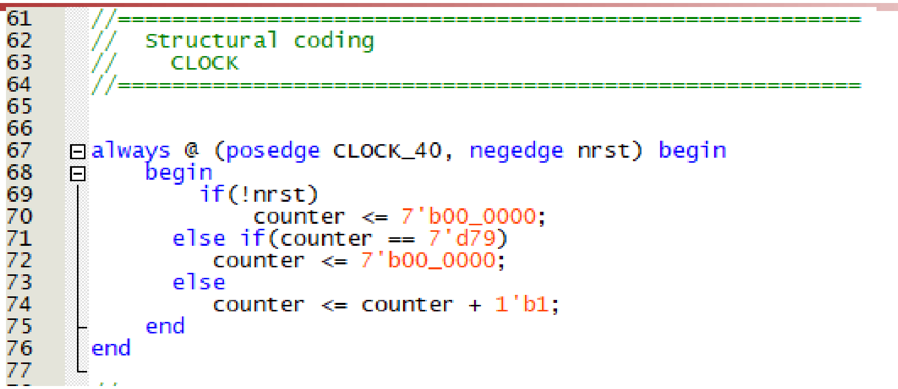


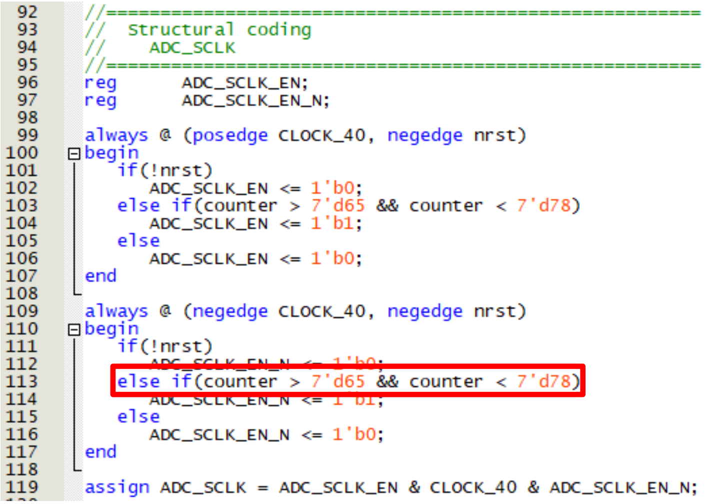
Figure 9. LTC2308 Timing with a Short CONVST Pulse

- ADC Datasheet Timing Diagram

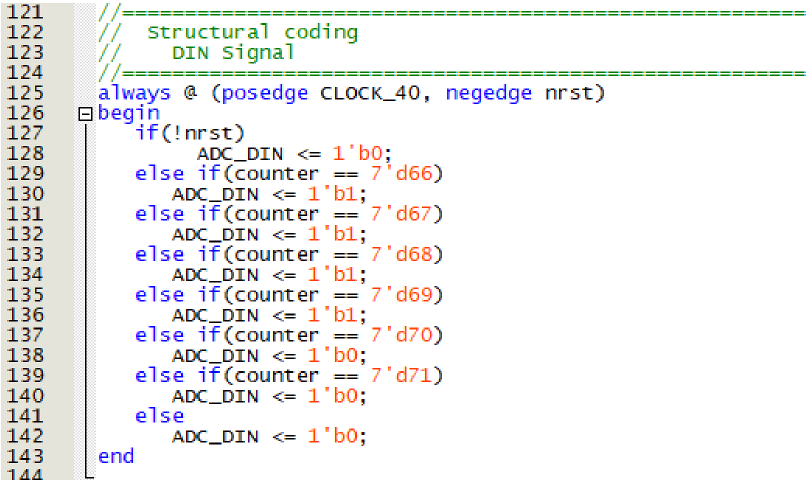
Project Progress – ADC Controller (Cont.)



```
//=========================================================
//  Structural coding
//    CLOCK
//=========================================================


always @ (posedge CLOCK_40, negedge nrst) begin
    begin
        if(!nrst)
            counter <= 7'b00_0000;
      else if(counter == 7'd79)
         counter <= 7'b00_0000;
      else
         counter <= counter + 1'b1;
    end
end
```



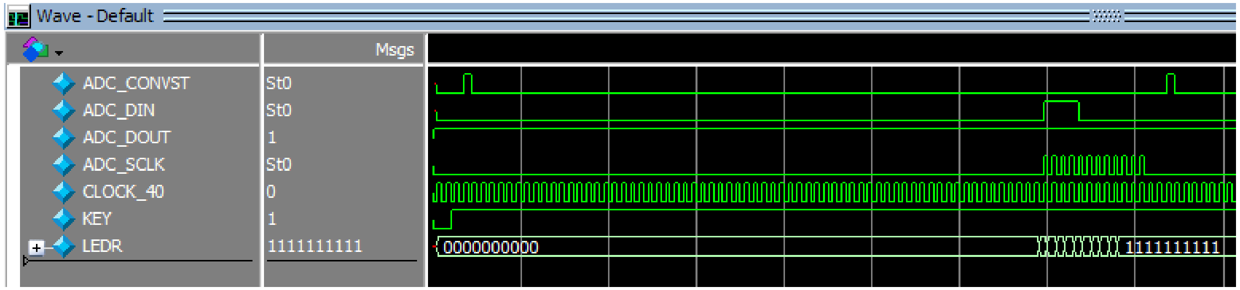
```
//=========================================================
//  Structural coding
//    ADC_SCLK
//=========================================================
reg       ADC_SCLK_EN;
reg       ADC_SCLK_EN_N;

always @ (posedge CLOCK_40, negedge nrst)
begin
   if(!nrst)
      ADC_SCLK_EN <= 1'b0;
   else if(counter > 7'd65 && counter < 7'd78)
      ADC_SCLK_EN <= 1'b1;
   else
      ADC_SCLK_EN <= 1'b0;
end

always @ (negedge CLOCK_40, negedge nrst)
begin
   if(!nrst)
      ADC_SCLK_EN_N <= 1'b0;
   else if(counter > 7'd65 && counter < 7'd78)
      ADC_SCLK_EN_N <= 1'b1;
   else
      ADC_SCLK_EN_N <= 1'b0;
end

assign ADC_SCLK = ADC_SCLK_EN & CLOCK_40 & ADC_SCLK_EN_N;
```



```
//=========================================================
//  Structural coding
//    DIN Signal
//=========================================================
always @ (posedge CLOCK_40, negedge nrst)
begin
   if(!nrst)
        ADC_DIN <= 1'b0;
   else if(counter == 7'd66)
      ADC_DIN <= 1'b1;
   else if(counter == 7'd67)
      ADC_DIN <= 1'b1;
   else if(counter == 7'd68)
      ADC_DIN <= 1'b1;
   else if(counter == 7'd69)
      ADC_DIN <= 1'b1;
   else if(counter == 7'd70)
      ADC_DIN <= 1'b0;
   else if(counter == 7'd71)
      ADC_DIN <= 1'b0;
   else
      ADC_DIN <= 1'b0;
end
```

- ➢ ADC Datasheet Timing HDL

Project Progress – ADC Controller (Cont.)



Wave - Default

	Msgs
ADC_CONVST	St0
ADC_DIN	St0
ADC_DOUT	1
ADC_SCLK	St0
CLOCK_40	0
KEY	1
LEDR	1111111111

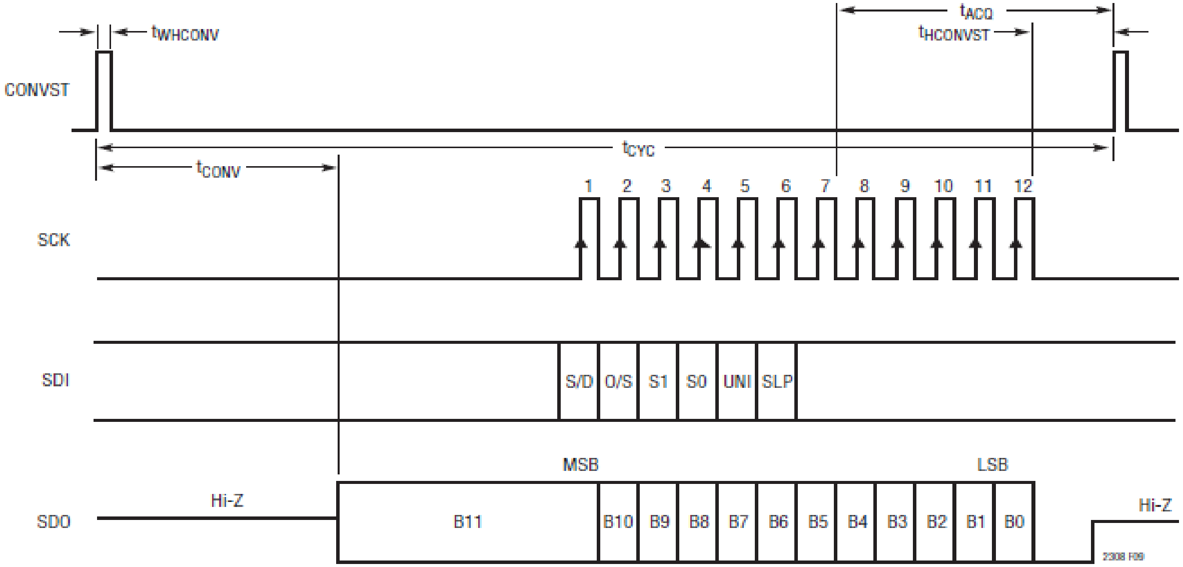
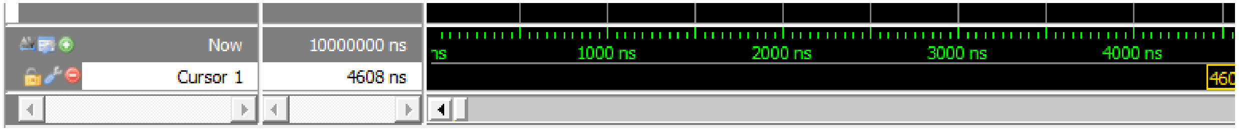


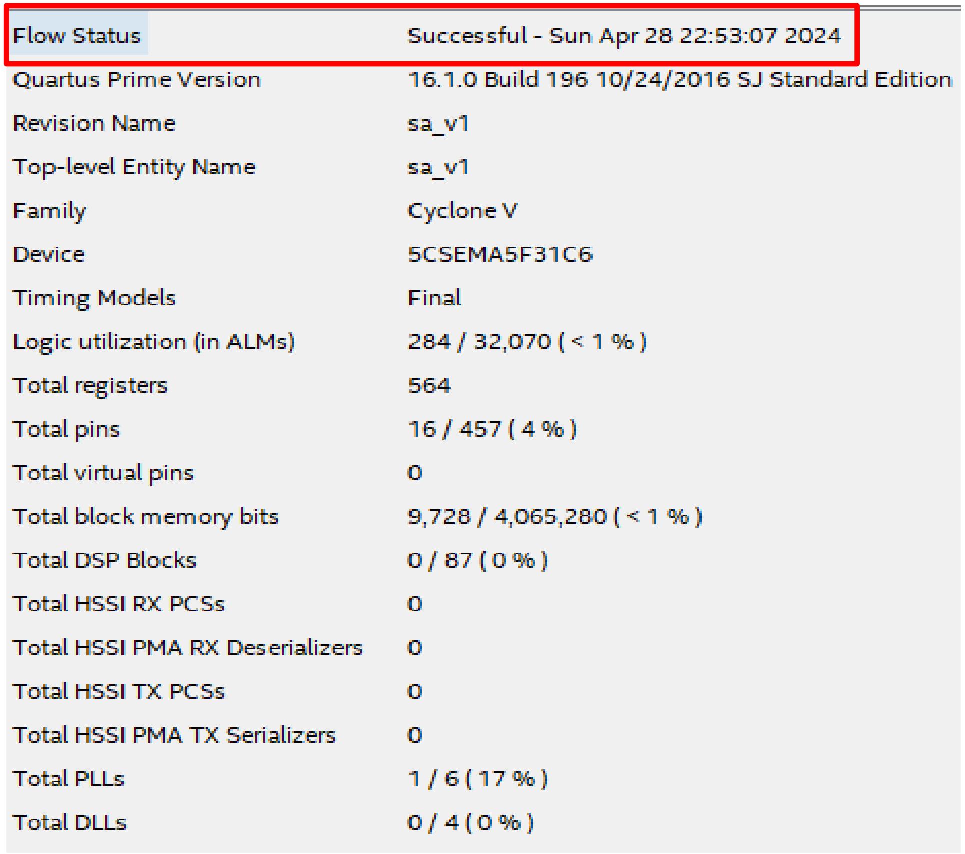
Figure 9. LTC2308 Timing with a Short CONVST Pulse



Now	10000000 ns
Cursor 1	4608 ns

- ➢ ADC Simulation and Datasheet Comparison

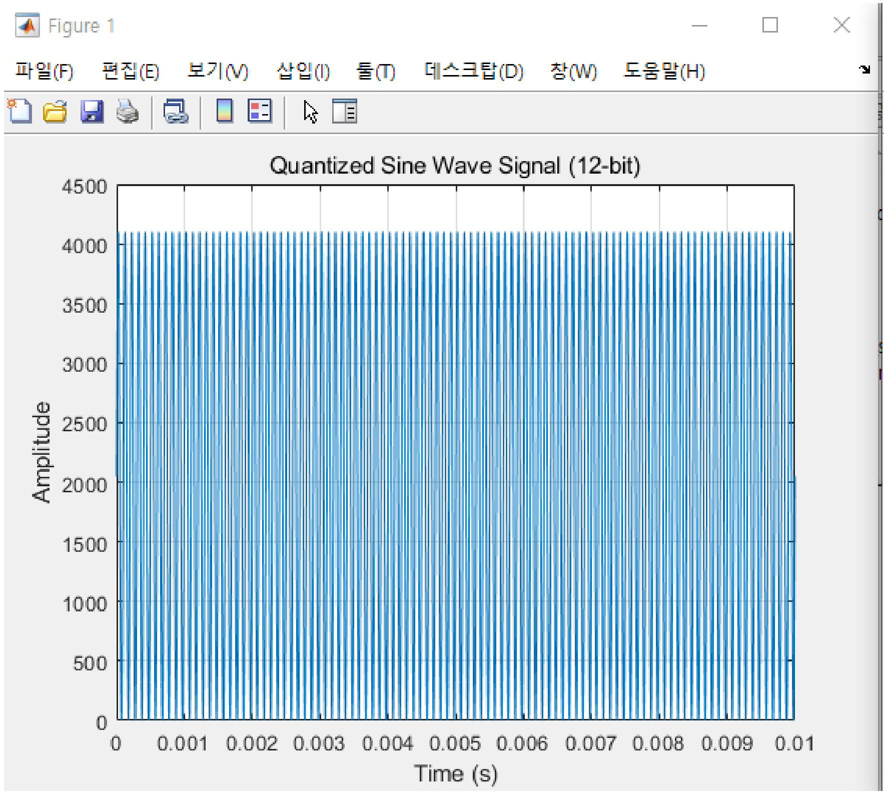
Project Progress – ADC Controller (Cont.)



Flow Status	Successful - Sun Apr 28 22:53:07 2024
Quartus Prime Version	16.1.0 Build 196 10/24/2016 SJ Standard Edition
Revision Name	sa_v1
Top-level Entity Name	sa_v1
Family	Cyclone V
Device	5CSEMA5F31C6
Timing Models	Final
Logic utilization (in ALMs)	284 / 32,070 (< 1 %)
Total registers	564
Total pins	16 / 457 (4 %)
Total virtual pins	0
Total block memory bits	9,728 / 4,065,280 (< 1 %)
Total DSP Blocks	0 / 87 (0 %)
Total HSSI RX PCSs	0
Total HSSI PMA RX Deserializers	0
Total HSSI TX PCSs	0
Total HSSI PMA TX Serializers	0
Total PLLs	1 / 6 (17 %)
Total DLLs	0 / 4 (0 %)

- ADC Synthesis Result

Project Progress – ADC Controller (Cont.)



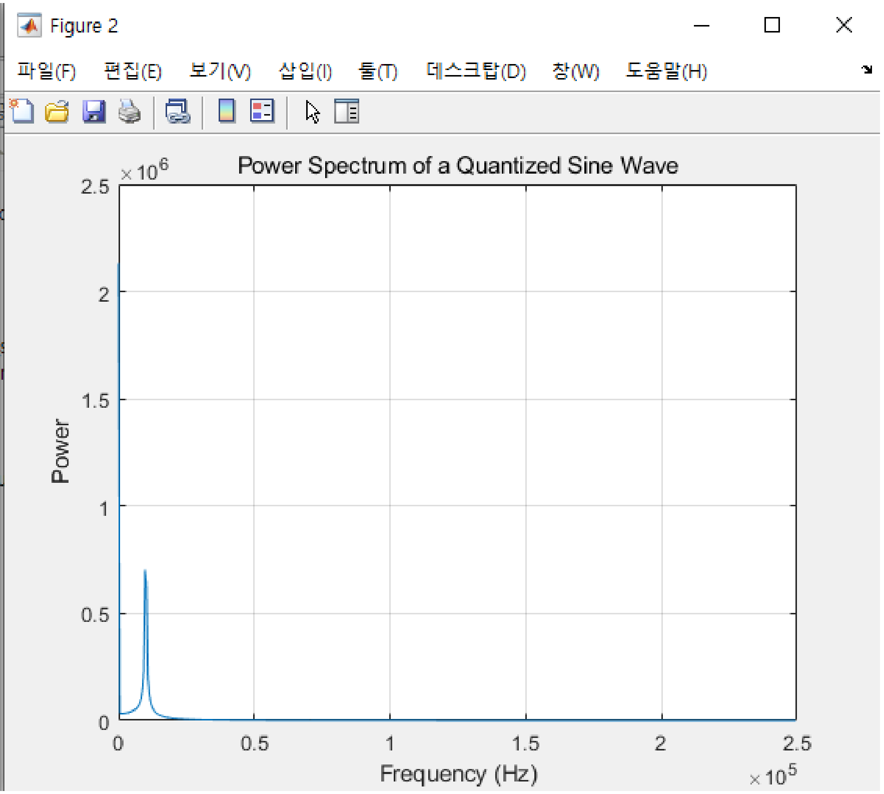
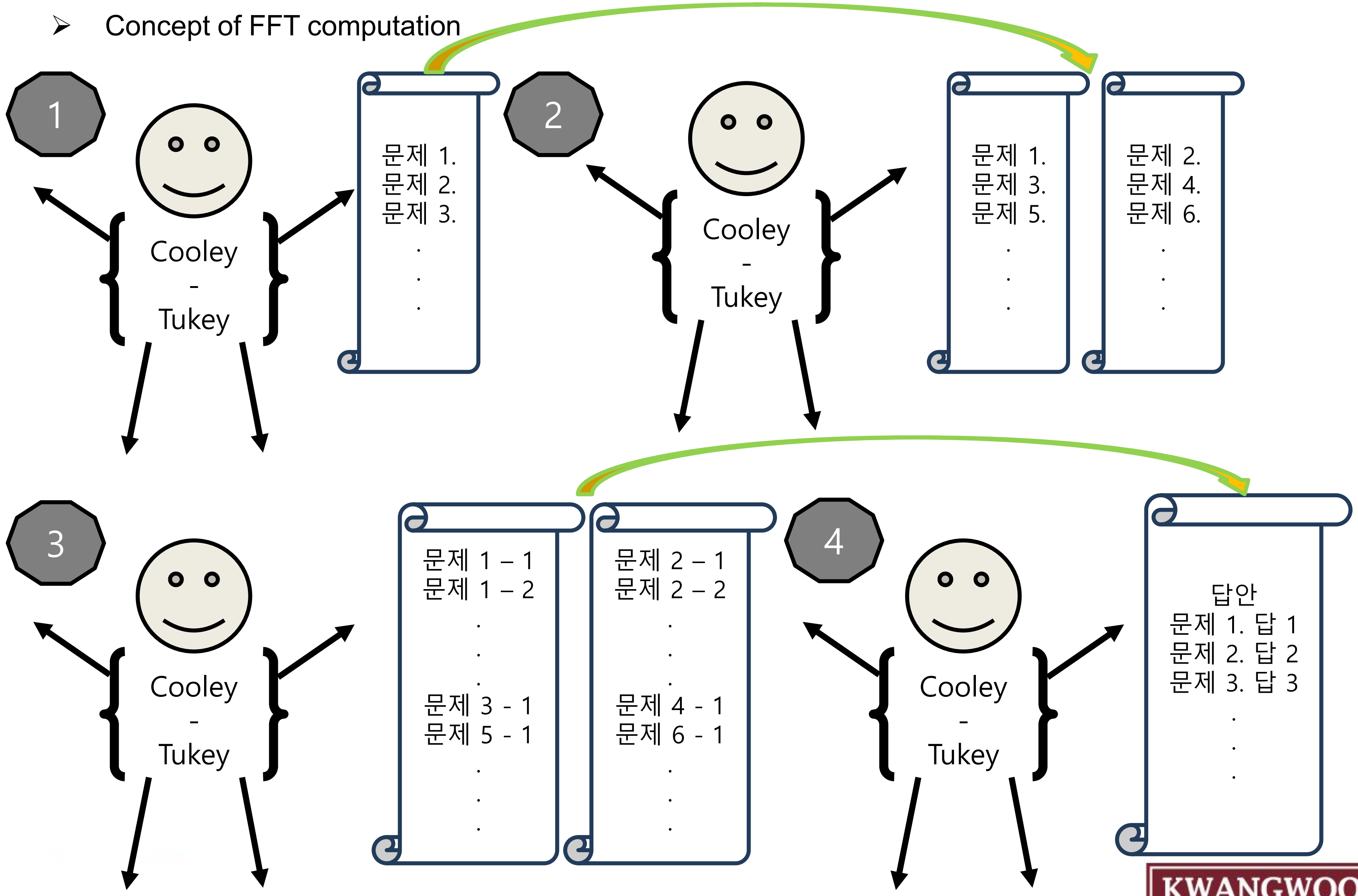


Figure 2
파일(F) 편집(E) 보기(V) 삽입(I) 툴(T) 데스크탑(D) 창(W) 도움말(H)
Power Spectrum of a Quantized Sine Wave
×10^6
Power
2.5
2
1.5
1
0.5
0
0.5
1
1.5
2
2.5
Frequency (Hz)
×10^5

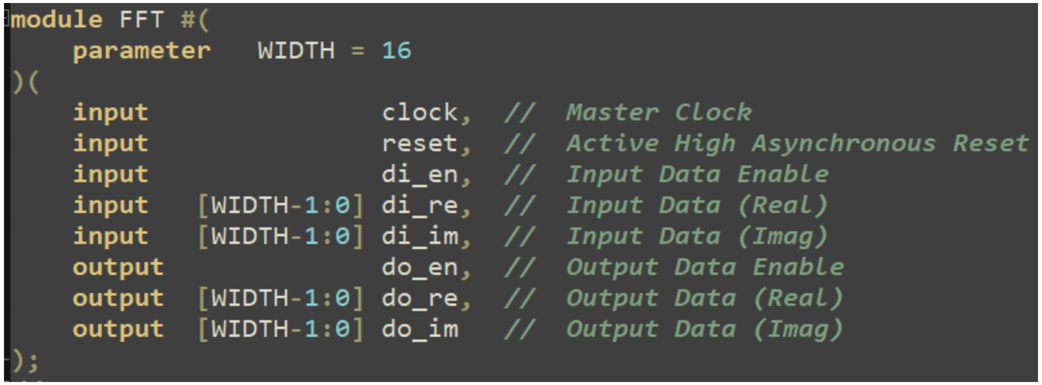
- ➢ Matlab ADC Verification

Project Progress – FFT Computation

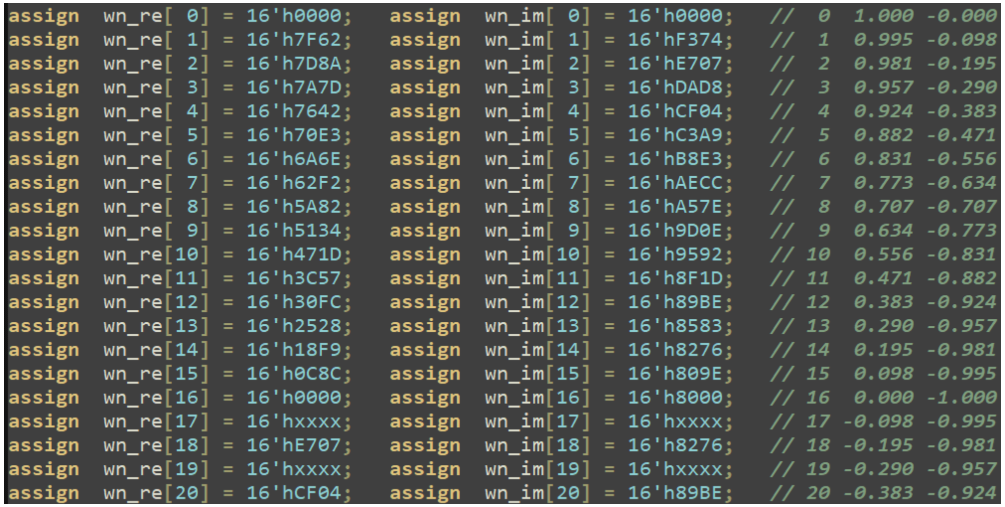
- Concept of FFT computation



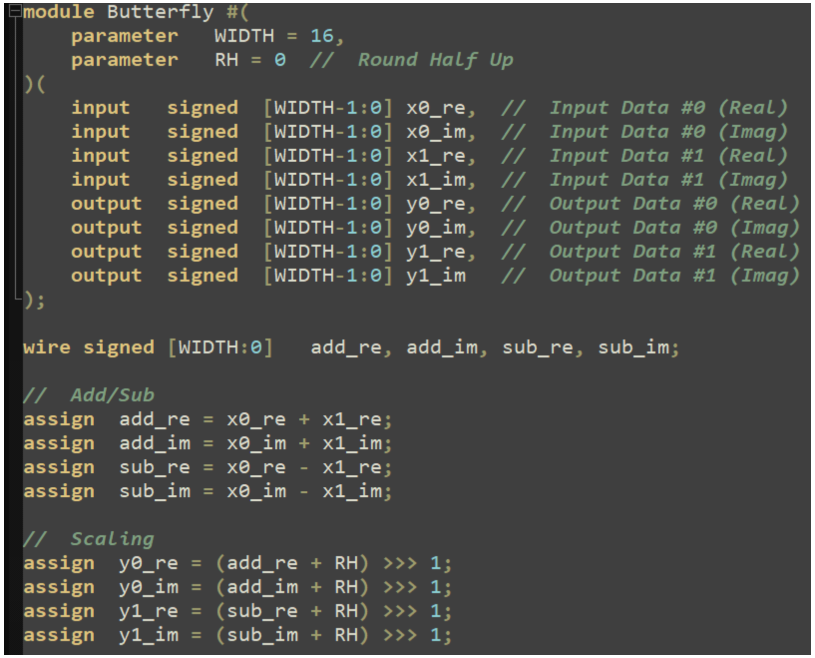
Project Progress – FFT



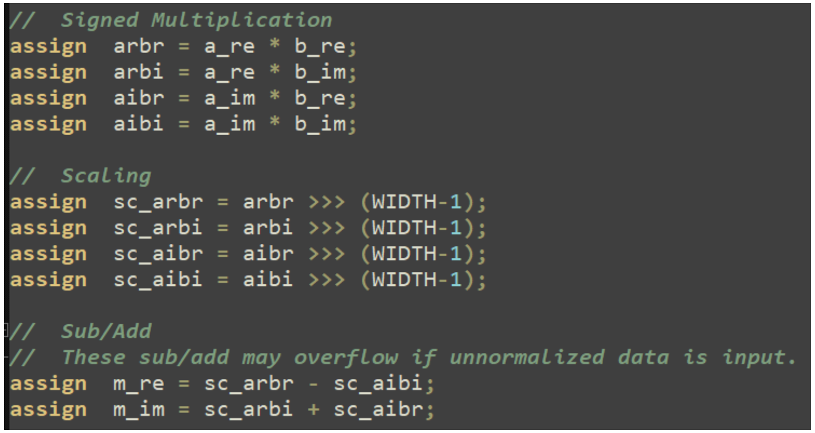
```
module FFT #(
    parameter   WIDTH = 16
)(
    input                   clock,  //  Master Clock
    input                   reset,  //  Active High Asynchronous Reset
    input                   di_en,  //  Input Data Enable
    input    [WIDTH-1:0]    di_re,  //  Input Data (Real)
    input    [WIDTH-1:0]    di_im,  //  Input Data (Imag)
    output                  do_en,  //  Output Data Enable
    output   [WIDTH-1:0]    do_re,  //  Output Data (Real)
    output   [WIDTH-1:0]    do_im   //  Output Data (Imag)
);
```



```
assign  wn_re[ 0] = 16'h0000;   assign  wn_im[ 0] = 16'h0000;   //  0  1.000 -0.000
assign  wn_re[ 1] = 16'h7F62;   assign  wn_im[ 1] = 16'hF374;   //  1  0.995 -0.098
assign  wn_re[ 2] = 16'h7D8A;   assign  wn_im[ 2] = 16'hE707;   //  2  0.981 -0.195
assign  wn_re[ 3] = 16'h7A7D;   assign  wn_im[ 3] = 16'hDAD8;   //  3  0.957 -0.290
assign  wn_re[ 4] = 16'h7642;   assign  wn_im[ 4] = 16'hCF04;   //  4  0.924 -0.383
assign  wn_re[ 5] = 16'h70E3;   assign  wn_im[ 5] = 16'hC3A9;   //  5  0.882 -0.471
assign  wn_re[ 6] = 16'h6A6E;   assign  wn_im[ 6] = 16'hB8E3;   //  6  0.831 -0.556
assign  wn_re[ 7] = 16'h62F2;   assign  wn_im[ 7] = 16'hAECC;   //  7  0.773 -0.634
assign  wn_re[ 8] = 16'h5A82;   assign  wn_im[ 8] = 16'hA57E;   //  8  0.707 -0.707
assign  wn_re[ 9] = 16'h5134;   assign  wn_im[ 9] = 16'h9D0E;   //  9  0.634 -0.773
assign  wn_re[10] = 16'h471D;   assign  wn_im[10] = 16'h9592;   // 10  0.556 -0.831
assign  wn_re[11] = 16'h3C57;   assign  wn_im[11] = 16'h8F1D;   // 11  0.471 -0.882
assign  wn_re[12] = 16'h30FC;   assign  wn_im[12] = 16'h89BE;   // 12  0.383 -0.924
assign  wn_re[13] = 16'h2528;   assign  wn_im[13] = 16'h8583;   // 13  0.290 -0.957
assign  wn_re[14] = 16'h18F9;   assign  wn_im[14] = 16'h8276;   // 14  0.195 -0.981
assign  wn_re[15] = 16'h0C8C;   assign  wn_im[15] = 16'h809E;   // 15  0.098 -0.995
assign  wn_re[16] = 16'h0000;   assign  wn_im[16] = 16'h8000;   // 16  0.000 -1.000
assign  wn_re[17] = 16'hxxxx;   assign  wn_im[17] = 16'hxxxx;   // 17 -0.098 -0.995
assign  wn_re[18] = 16'hE707;   assign  wn_im[18] = 16'h8276;   // 18 -0.195 -0.981
assign  wn_re[19] = 16'hxxxx;   assign  wn_im[19] = 16'hxxxx;   // 19 -0.290 -0.957
assign  wn_re[20] = 16'hCF04;   assign  wn_im[20] = 16'h89BE;   // 20 -0.383 -0.924
```



```
module Butterfly #(
    parameter   WIDTH = 16,
    parameter   RH = 0  //  Round Half Up
)(
    input   signed  [WIDTH-1:0] x0_re,  //  Input Data #0 (Real)
    input   signed  [WIDTH-1:0] x0_im,  //  Input Data #0 (Imag)
    input   signed  [WIDTH-1:0] x1_re,  //  Input Data #1 (Real)
    input   signed  [WIDTH-1:0] x1_im,  //  Input Data #1 (Imag)
    output  signed  [WIDTH-1:0] y0_re,  //  Output Data #0 (Real)
    output  signed  [WIDTH-1:0] y0_im,  //  Output Data #0 (Imag)
    output  signed  [WIDTH-1:0] y1_re,  //  Output Data #1 (Real)
    output  signed  [WIDTH-1:0] y1_im   //  Output Data #1 (Imag)
);

wire signed [WIDTH:0]   add_re, add_im, sub_re, sub_im;

//  Add/Sub
assign  add_re = x0_re + x1_re;
assign  add_im = x0_im + x1_im;
assign  sub_re = x0_re - x1_re;
assign  sub_im = x0_im - x1_im;

//  Scaling
assign  y0_re = (add_re + RH) >>> 1;
assign  y0_im = (add_im + RH) >>> 1;
assign  y1_re = (sub_re + RH) >>> 1;
assign  y1_im = (sub_im + RH) >>> 1;
```

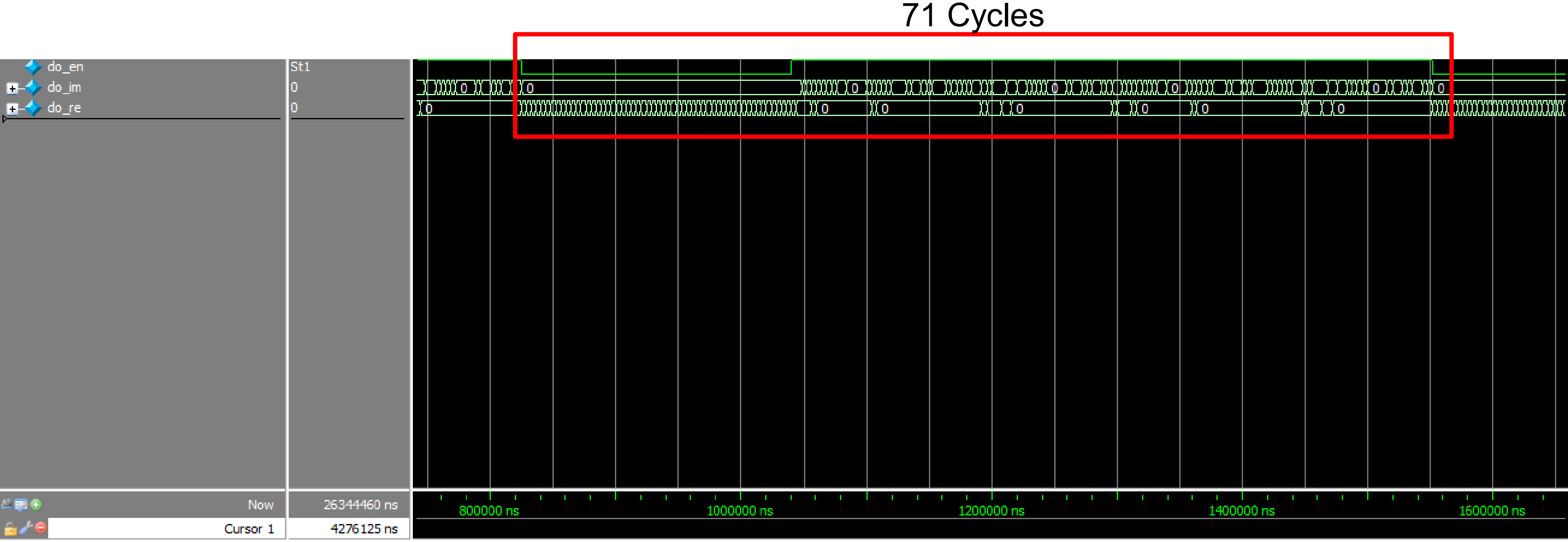


```
//  Signed Multiplication
assign  arbr = a_re * b_re;
assign  arbi = a_re * b_im;
assign  aibr = a_im * b_re;
assign  aibi = a_im * b_im;

//  Scaling
assign  sc_arbr = arbr >>> (WIDTH-1);
assign  sc_arbi = arbi >>> (WIDTH-1);
assign  sc_aibr = aibr >>> (WIDTH-1);
assign  sc_aibi = aibi >>> (WIDTH-1);

//  Sub/Add
//  These sub/add may overflow if unnormalized data is input.
assign  m_re = sc_arbr - sc_aibi;
assign  m_im = sc_arbi + sc_aibr;
```

FFT Modules (Top, Butterfly, Twiddle, Multiply)

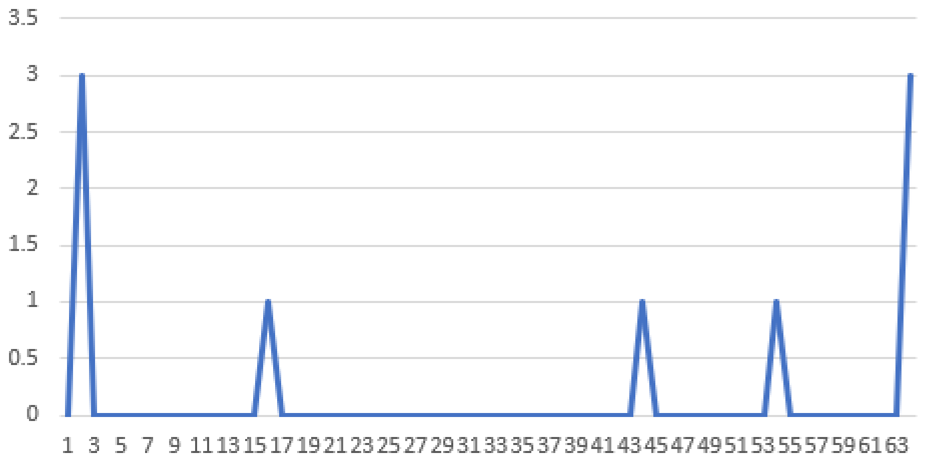
Project Progress – FFT (Cont.)

71 Cycles



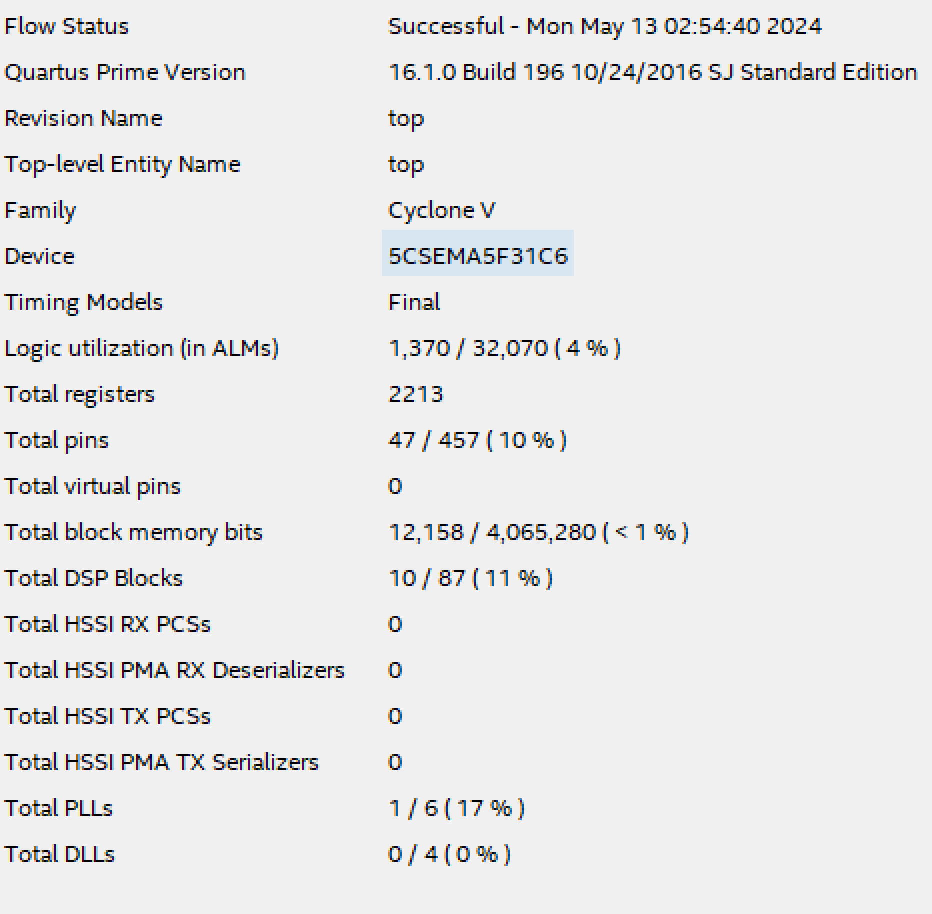
- FFT Simulation

Project Progress – FFT (Cont.)



- FFT Result on Excel

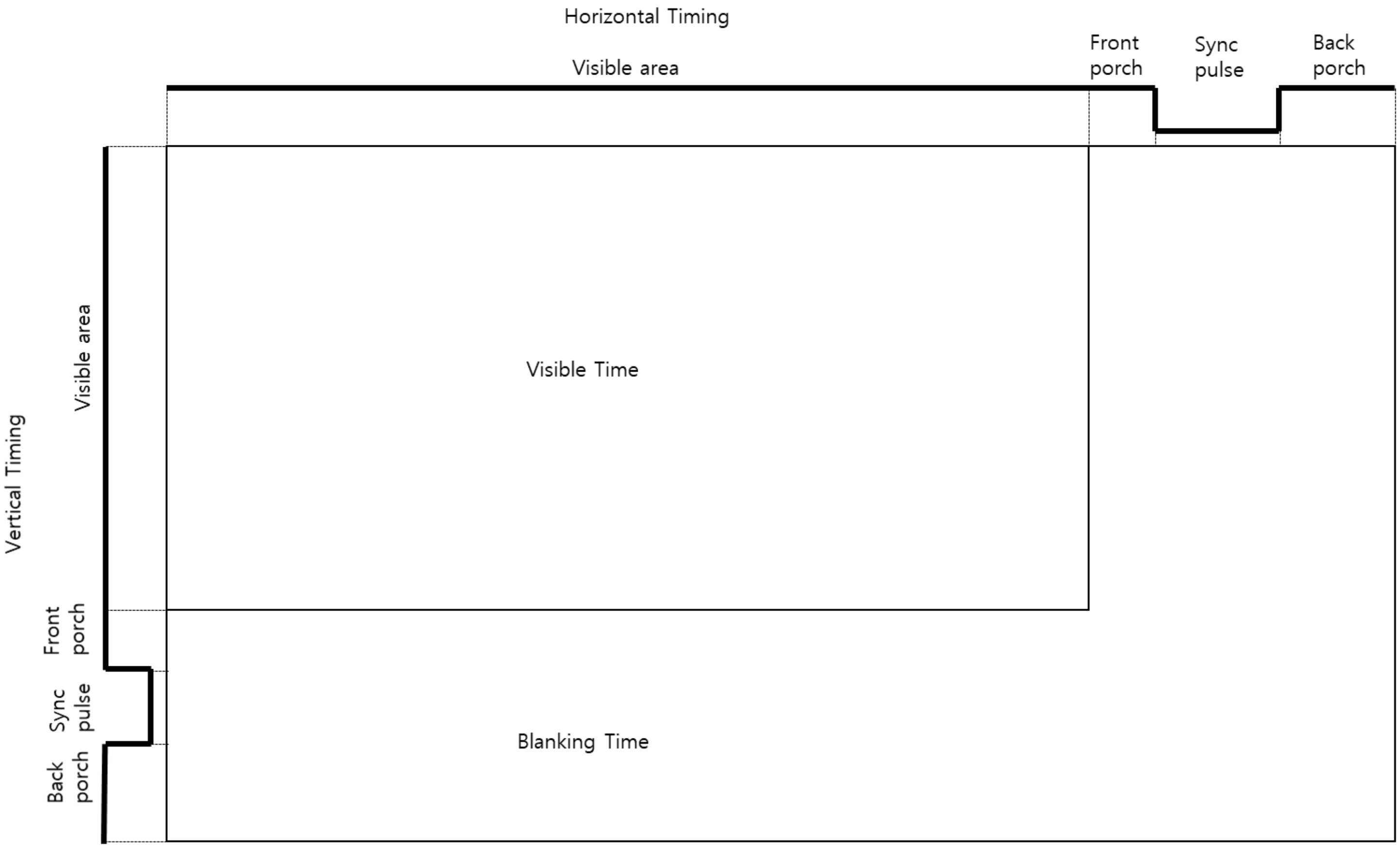
Project Progress – FFT (Cont.)



Flow Status	Successful - Mon May 13 02:54:40 2024
Quartus Prime Version	16.1.0 Build 196 10/24/2016 SJ Standard Edition
Revision Name	top
Top-level Entity Name	top
Family	Cyclone V
Device	5CSEMA5F31C6
Timing Models	Final
Logic utilization (in ALMs)	1,370 / 32,070 (4 %)
Total registers	2213
Total pins	47 / 457 (10 %)
Total virtual pins	0
Total block memory bits	12,158 / 4,065,280 (< 1 %)
Total DSP Blocks	10 / 87 (11 %)
Total HSSI RX PCSs	0
Total HSSI PMA RX Deserializers	0
Total HSSI TX PCSs	0
Total HSSI PMA TX Serializers	0
Total PLLs	1 / 6 (17 %)
Total DLLs	0 / 4 (0 %)

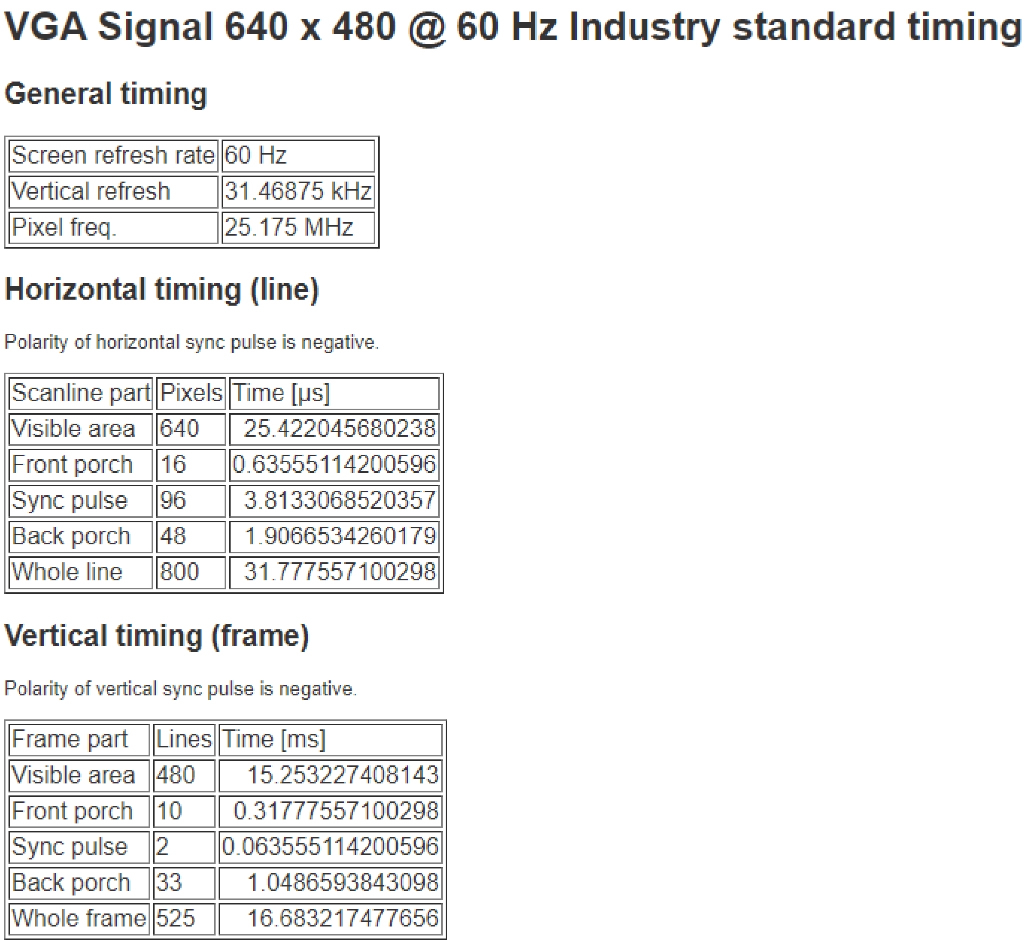
- ADC and FFT Synthesis

Project Progress – VGA



- VGA Speciation

Project Progress – VGA (Cont.)



VGA Signal 640 x 480 @ 60 Hz Industry standard timing

General timing

Screen refresh rate	60 Hz
Vertical refresh	31.46875 kHz
Pixel freq.	25.175 MHz

Horizontal timing (line)

Polarity of horizontal sync pulse is negative.

Scanline part	Pixels	Time [µs]
Visible area	640	25.422045680238
Front porch	16	0.63555114200596
Sync pulse	96	3.8133068520357
Back porch	48	1.9066534260179
Whole line	800	31.777557100298

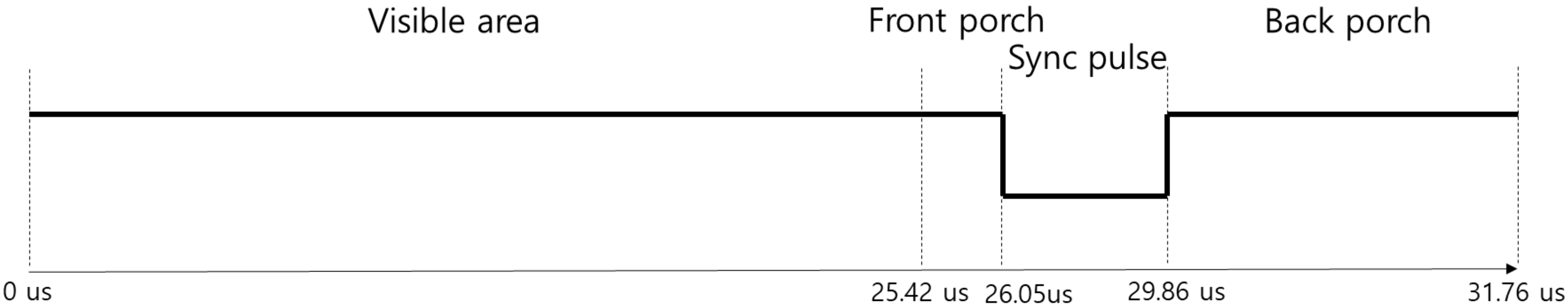
Vertical timing (frame)

Polarity of vertical sync pulse is negative.

Frame part	Lines	Time [ms]
Visible area	480	15.253227408143
Front porch	10	0.31777557100298
Sync pulse	2	0.063555114200596
Back porch	33	1.0486593843098
Whole frame	525	16.683217477656

- VGA Speciation

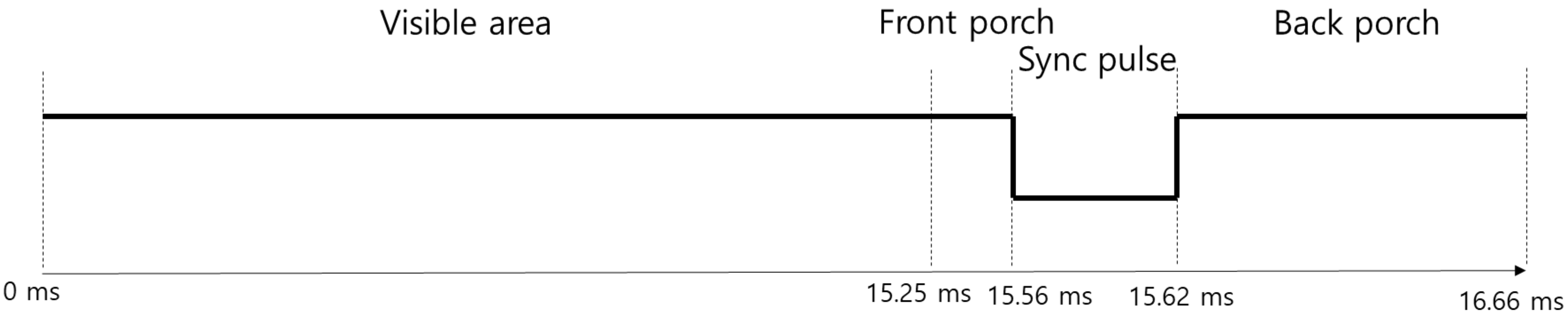
Project Progress – VGA (Cont.)



Horizontal timing (Line)		
Scanline part	Pixels	Time [us]
Visible area	640	25.42
Front porch	16	0.63
Sync pulse	96	3.81
Back porch	48	1.90
Whole line	800	31.77

- Horizontal VGA Speciation

Project Progress – VGA (Cont.)



Vertical timing (Frame)		
Frame part	Lines	Time [ms]
Visible area	480	15.25
Front porch	10	0.31
Sync pulse	2	0.06
Back porch	33	1.04
Whole frame	525	16.68

- Vertical VGA Speciation

Project Progress – VGA (Cont.)

FUNCTIONAL BLOCK DIAGRAM

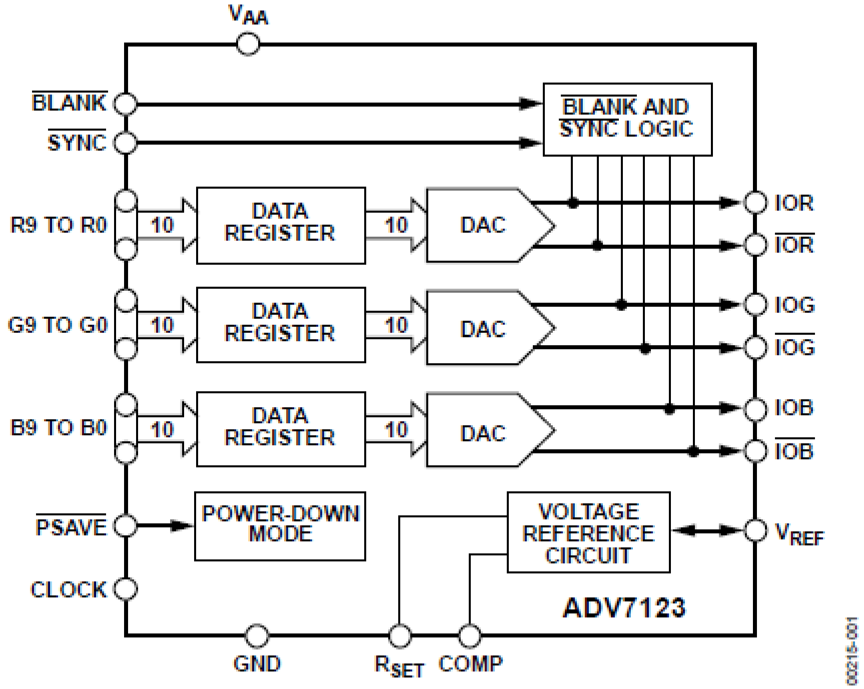


Figure 1.

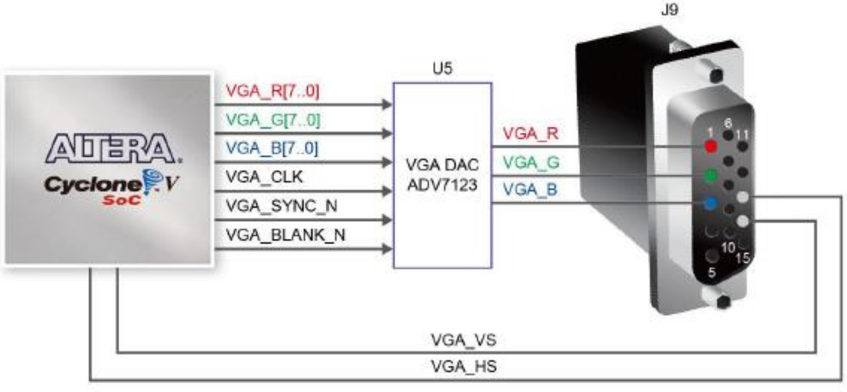


Figure 3-22 Connections between the FPGA and VGA

- ➢ VGA Manual Speciation

Project Progress – VGA (Cont.)

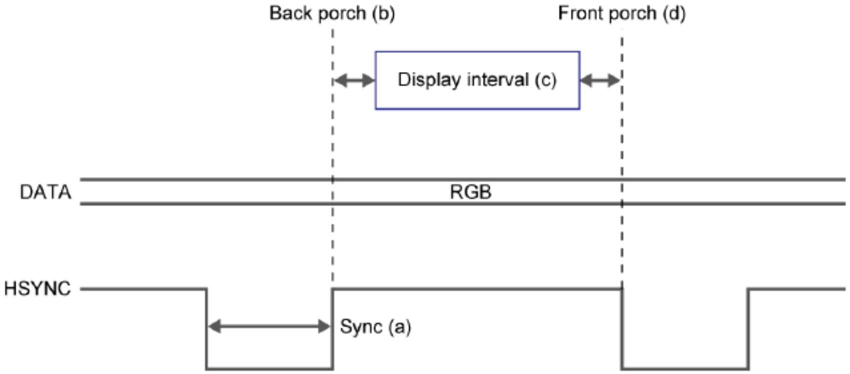
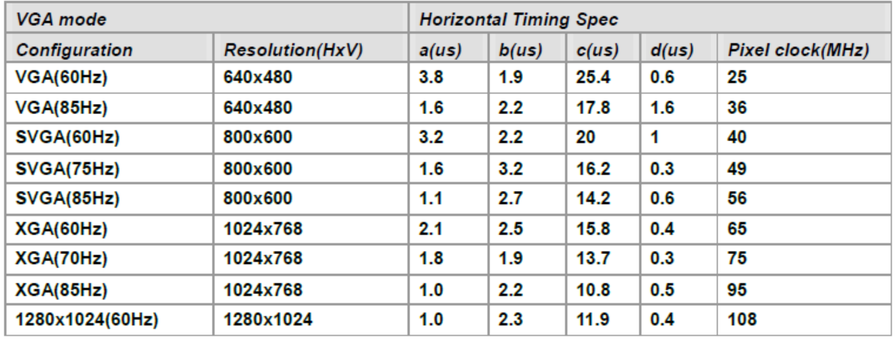


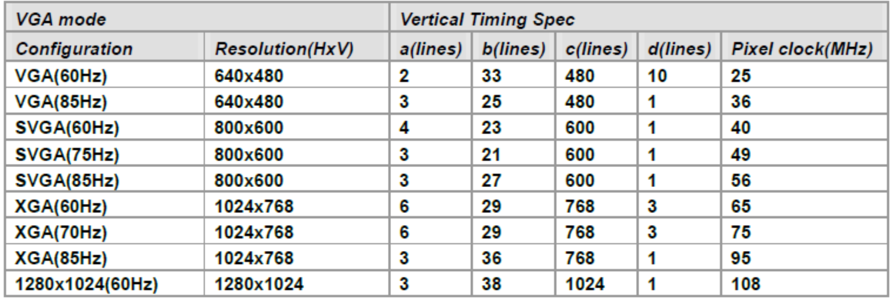
Figure 3-23 VGA horizontal timing specification

Table 3-14 VGA Horizontal Timing Specification



VGA mode		*Horizontal Timing Spec*				
Configuration	*Resolution(HxV)*	*a(us)*	*b(us)*	*c(us)*	*d(us)*	*Pixel clock(MHz)*
VGA(60Hz)	640x480	3.8	1.9	25.4	0.6	25
VGA(85Hz)	640x480	1.6	2.2	17.8	1.6	36
SVGA(60Hz)	800x600	3.2	2.2	20	1	40
SVGA(75Hz)	800x600	1.6	3.2	16.2	0.3	49
SVGA(85Hz)	800x600	1.1	2.7	14.2	0.6	56
XGA(60Hz)	1024x768	2.1	2.5	15.8	0.4	65
XGA(70Hz)	1024x768	1.8	1.9	13.7	0.3	75
XGA(85Hz)	1024x768	1.0	2.2	10.8	0.5	95
1280x1024(60Hz)	1280x1024	1.0	2.3	11.9	0.4	108

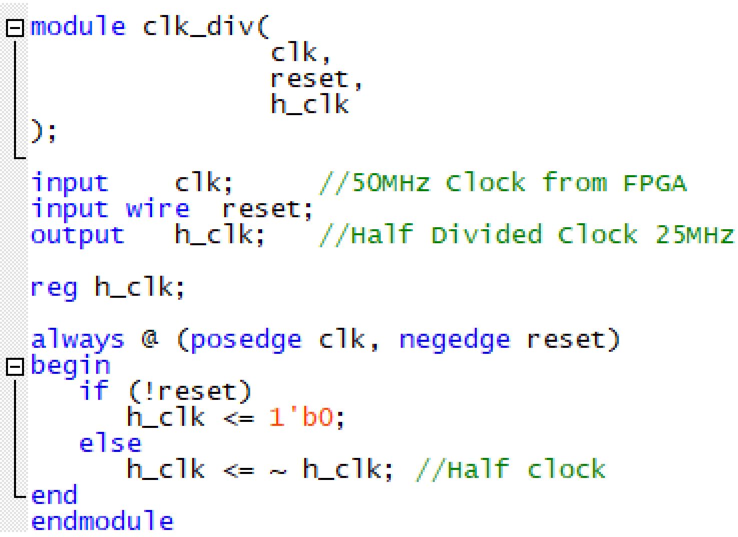
Table 3-15 VGA Vertical Timing Specification



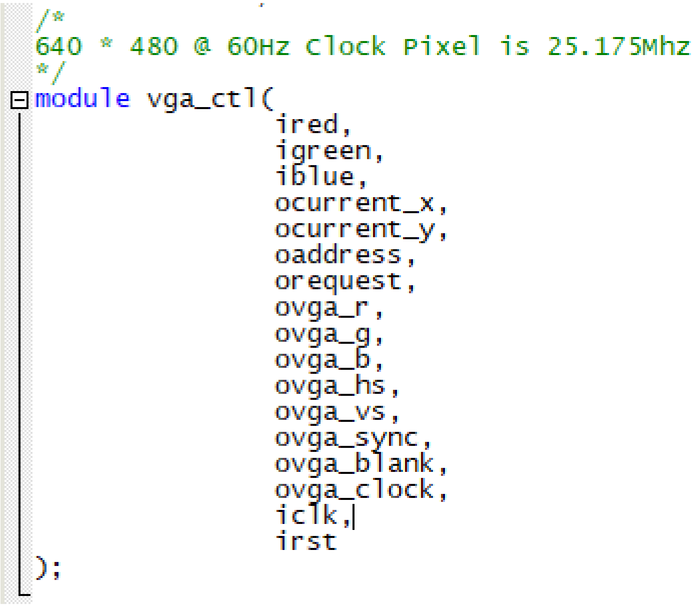
VGA mode		*Vertical Timing Spec*				
Configuration	*Resolution(HxV)*	*a(lines)*	*b(lines)*	*c(lines)*	*d(lines)*	*Pixel clock(MHz)*
VGA(60Hz)	640x480	2	33	480	10	25
VGA(85Hz)	640x480	3	25	480	1	36
SVGA(60Hz)	800x600	4	23	600	1	40
SVGA(75Hz)	800x600	3	21	600	1	49
SVGA(85Hz)	800x600	3	27	600	1	56
XGA(60Hz)	1024x768	6	29	768	3	65
XGA(70Hz)	1024x768	6	29	768	3	75
XGA(85Hz)	1024x768	3	36	768	1	95
1280x1024(60Hz)	1280x1024	3	38	1024	1	108

- VGA Manual Speciation

Project Progress – VGA (Cont.)



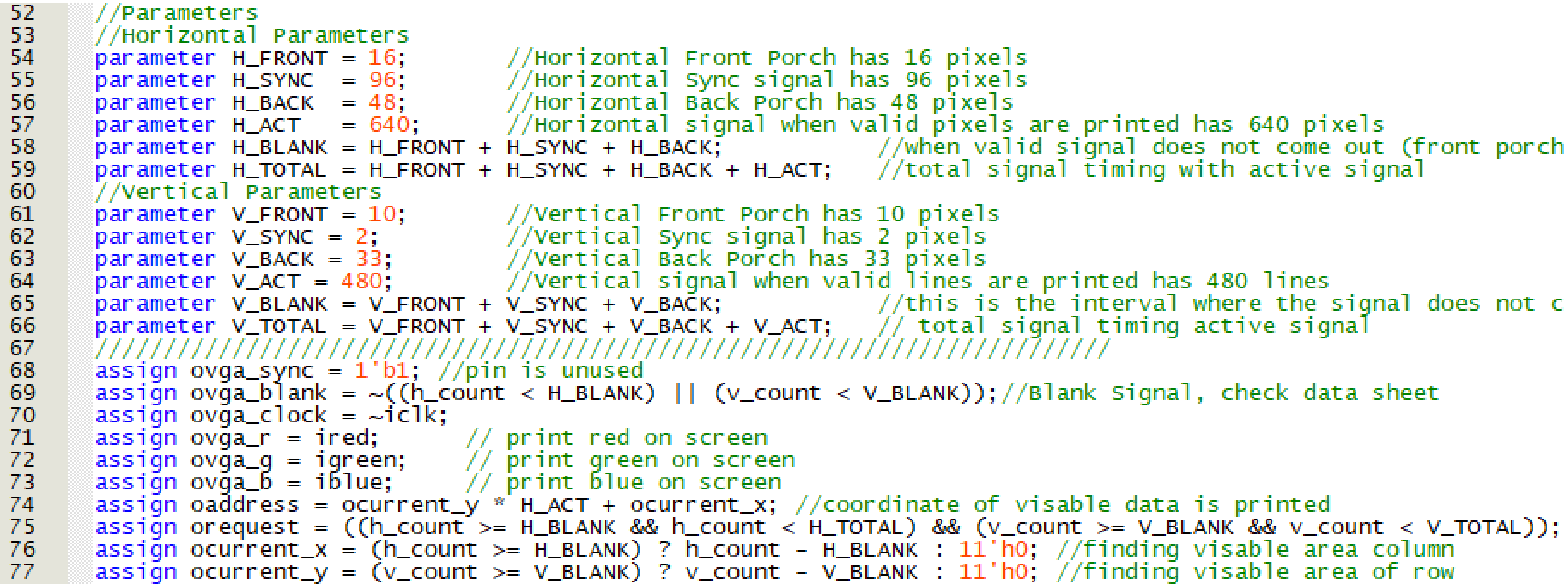
```
module clk_div(
                clk,
                reset,
                h_clk
);

input    clk;     //50MHz Clock from FPGA
input wire  reset;
output   h_clk;   //Half Divided Clock 25MHz

reg h_clk;

always @ (posedge clk, negedge reset)
begin
   if (!reset)
      h_clk <= 1'b0;
   else
      h_clk <= ~ h_clk; //Half clock
end
endmodule
```



```
/*
640 * 480 @ 60Hz Clock Pixel is 25.175Mhz
*/
module vga_ctl(
               ired,
               igreen,
               iblue,
               ocurrent_x,
               ocurrent_y,
               oaddress,
               orequest,
               ovga_r,
               ovga_g,
               ovga_b,
               ovga_hs,
               ovga_vs,
               ovga_sync,
               ovga_blank,
               ovga_clock,
               iclk,
               irst
);
```

- VGA HDL Top

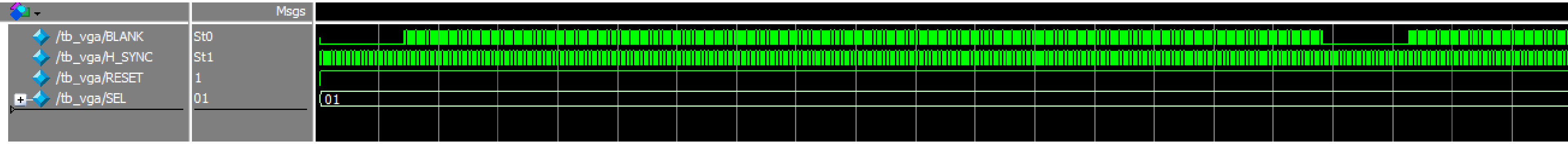
Project Progress – VGA (Cont.)



```
//Parameters
//Horizontal Parameters
parameter H_FRONT = 16;       //Horizontal Front Porch has 16 pixels
parameter H_SYNC  = 96;       //Horizontal Sync signal has 96 pixels
parameter H_BACK  = 48;       //Horizontal Back Porch has 48 pixels
parameter H_ACT   = 640;      //Horizontal signal when valid pixels are printed has 640 pixels
parameter H_BLANK = H_FRONT + H_SYNC + H_BACK;           //when valid signal does not come out (front porch
parameter H_TOTAL = H_FRONT + H_SYNC + H_BACK + H_ACT;   //total signal timing with active signal
//Vertical Parameters
parameter V_FRONT = 10;       //Vertical Front Porch has 10 pixels
parameter V_SYNC = 2;         //Vertical Sync signal has 2 pixels
parameter V_BACK = 33;        //Vertical Back Porch has 33 pixels
parameter V_ACT = 480;        //Vertical signal when valid lines are printed has 480 lines
parameter V_BLANK = V_FRONT + V_SYNC + V_BACK;           //this is the interval where the signal does not c
parameter V_TOTAL = V_FRONT + V_SYNC + V_BACK + V_ACT;   // total signal timing active signal
//////////////////////////////////////////////////////////////////////////
assign ovga_sync = 1'b1; //pin is unused
assign ovga_blank = ~((h_count < H_BLANK) || (v_count < V_BLANK));//Blank Signal, check data sheet
assign ovga_clock = ~iclk;
assign ovga_r = ired;      // print red on screen
assign ovga_g = igreen;    // print green on screen
assign ovga_b = iblue;     // print blue on screen
assign oaddress = ocurrent_y * H_ACT + ocurrent_x; //coordinate of visable data is printed
assign orequest = ((h_count >= H_BLANK && h_count < H_TOTAL) && (v_count >= V_BLANK && v_count < V_TOTAL));
assign ocurrent_x = (h_count >= H_BLANK) ? h_count - H_BLANK : 11'h0; //finding visable area column
assign ocurrent_y = (v_count >= V_BLANK) ? v_count - V_BLANK : 11'h0; //finding visable area of row
```

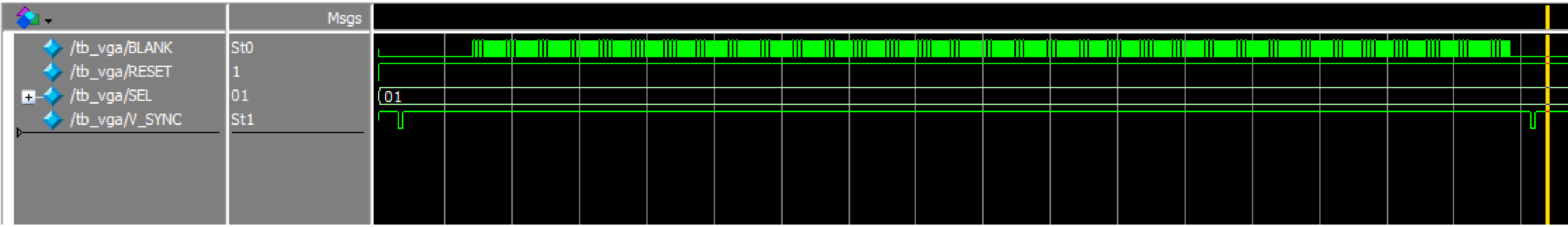
- ➢ VGA HDL Timing parameters

Project Progress – VGA (Cont.)



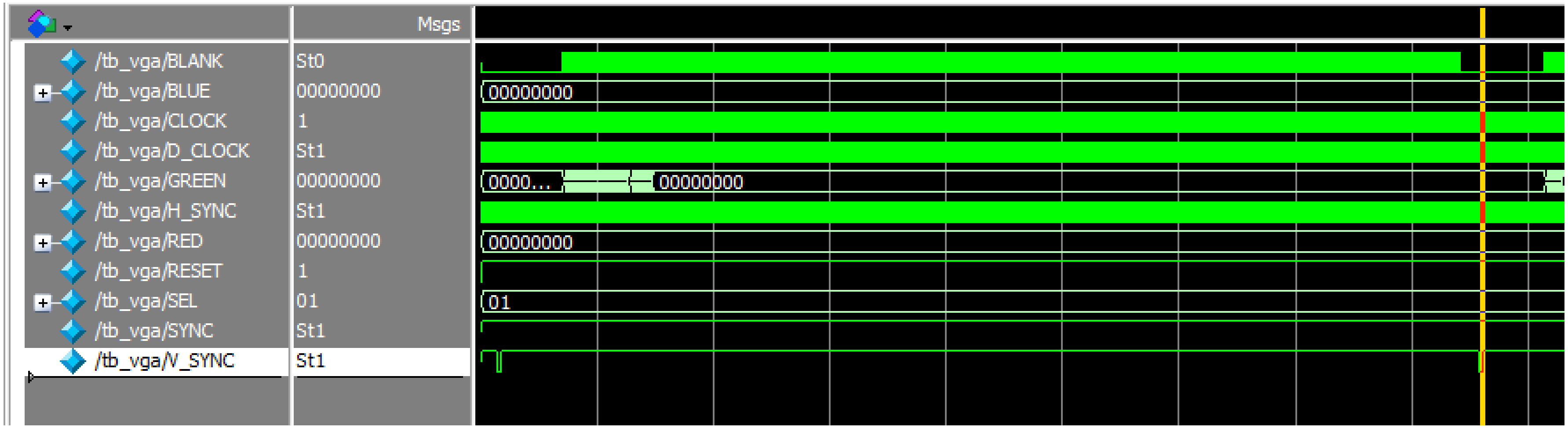
	Msgs
/tb_vga/BLANK	St0
/tb_vga/H_SYNC	St1
/tb_vga/RESET	1
/tb_vga/SEL	01

- Horizontal VGA Timing Simulation



	Msgs
/tb_vga/BLANK	St0
/tb_vga/RESET	1
/tb_vga/SEL	01
/tb_vga/V_SYNC	St1

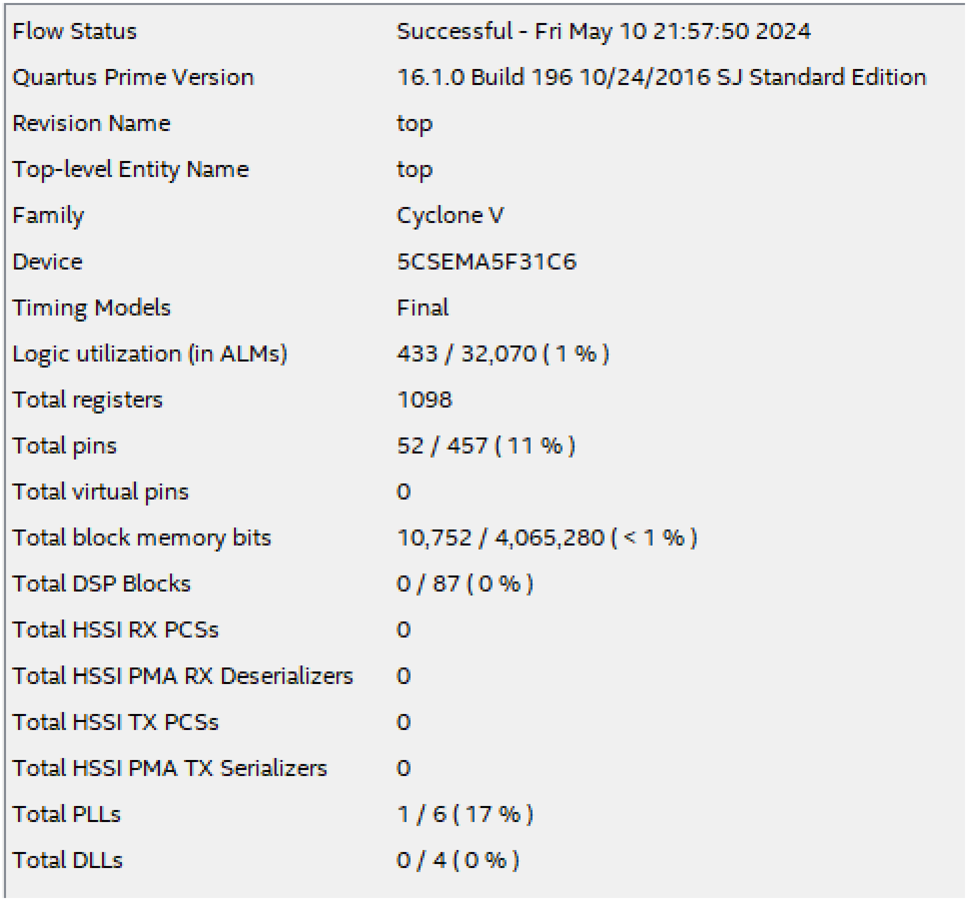
- Vertical VGA Timing Simulation



	Msgs
/tb_vga/BLANK	St0
/tb_vga/BLUE	00000000
/tb_vga/CLOCK	1
/tb_vga/D_CLOCK	St1
/tb_vga/GREEN	00000000
/tb_vga/H_SYNC	St1
/tb_vga/RED	00000000
/tb_vga/RESET	1
/tb_vga/SEL	01
/tb_vga/SYNC	St1
/tb_vga/V_SYNC	St1

- VGA HDL Timing Simulation

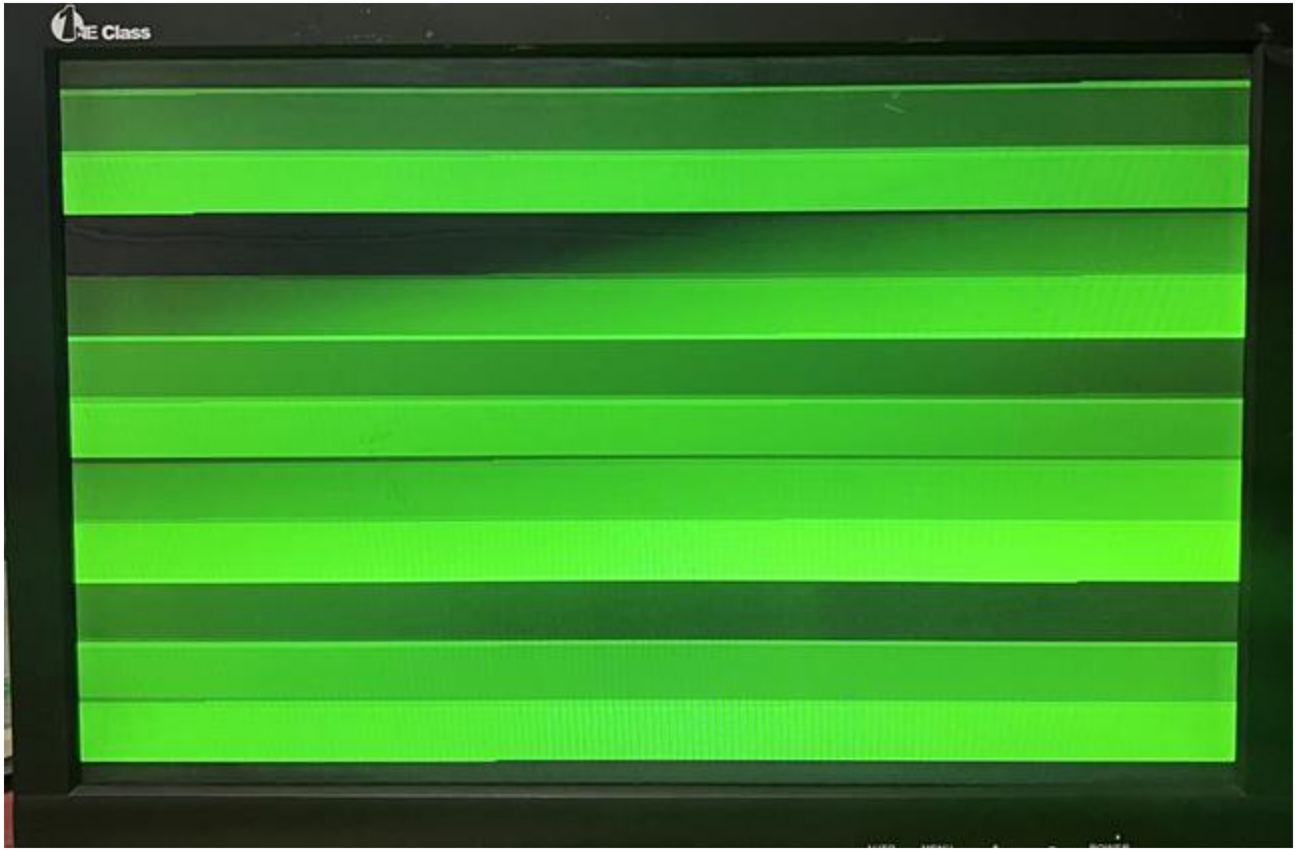
Project Progress – VGA (Cont.)



Flow Status	Successful - Fri May 10 21:57:50 2024
Quartus Prime Version	16.1.0 Build 196 10/24/2016 SJ Standard Edition
Revision Name	top
Top-level Entity Name	top
Family	Cyclone V
Device	5CSEMA5F31C6
Timing Models	Final
Logic utilization (in ALMs)	433 / 32,070 (1 %)
Total registers	1098
Total pins	52 / 457 (11 %)
Total virtual pins	0
Total block memory bits	10,752 / 4,065,280 (< 1 %)
Total DSP Blocks	0 / 87 (0 %)
Total HSSI RX PCSs	0
Total HSSI PMA RX Deserializers	0
Total HSSI TX PCSs	0
Total HSSI PMA TX Serializers	0
Total PLLs	1 / 6 (17 %)
Total DLLs	0 / 4 (0 %)

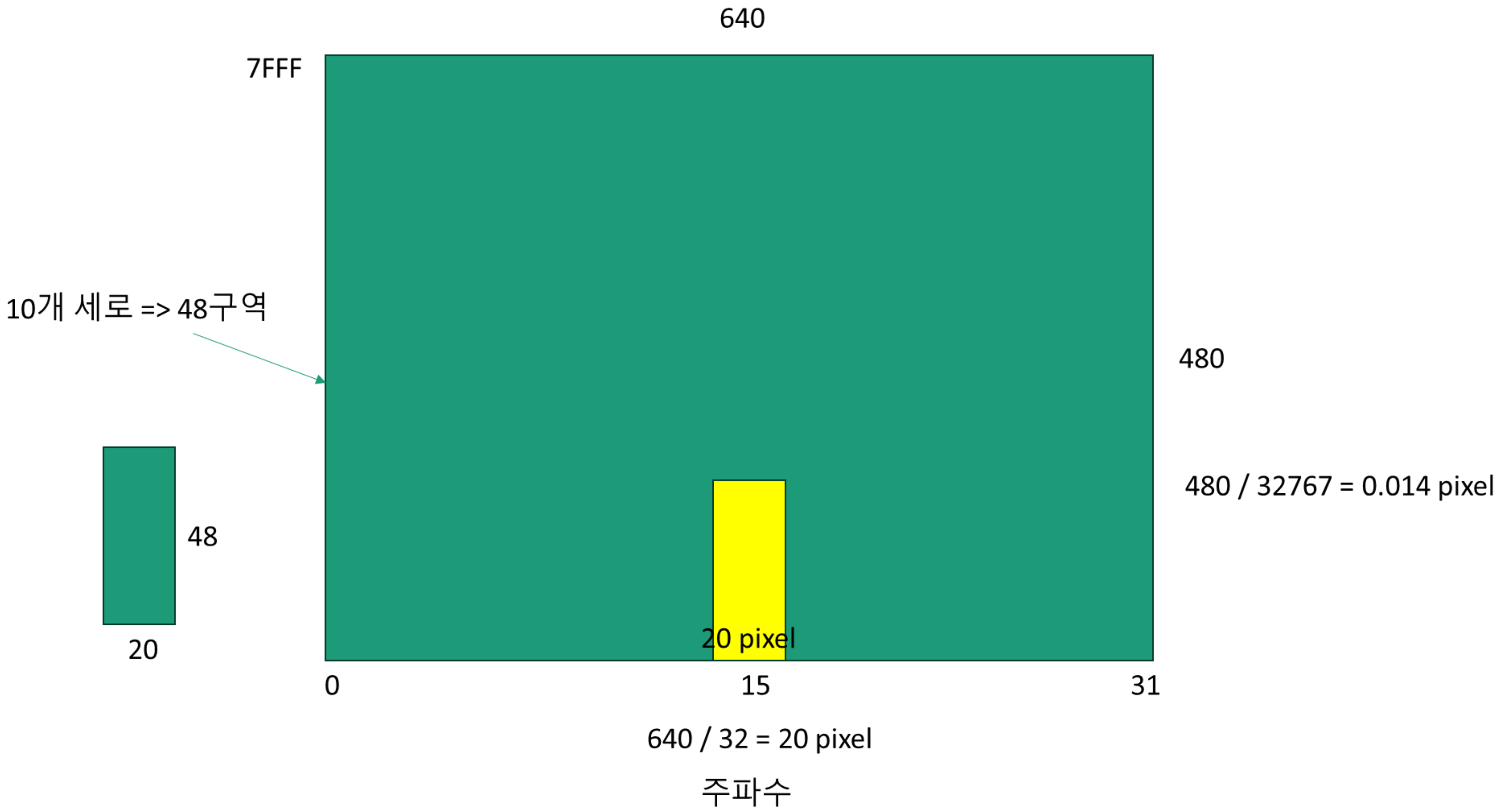
➢ VGA Synthesis

Project Progress – VGA (Cont.)



- VGA Test

Project Progress – VGA (Cont.)



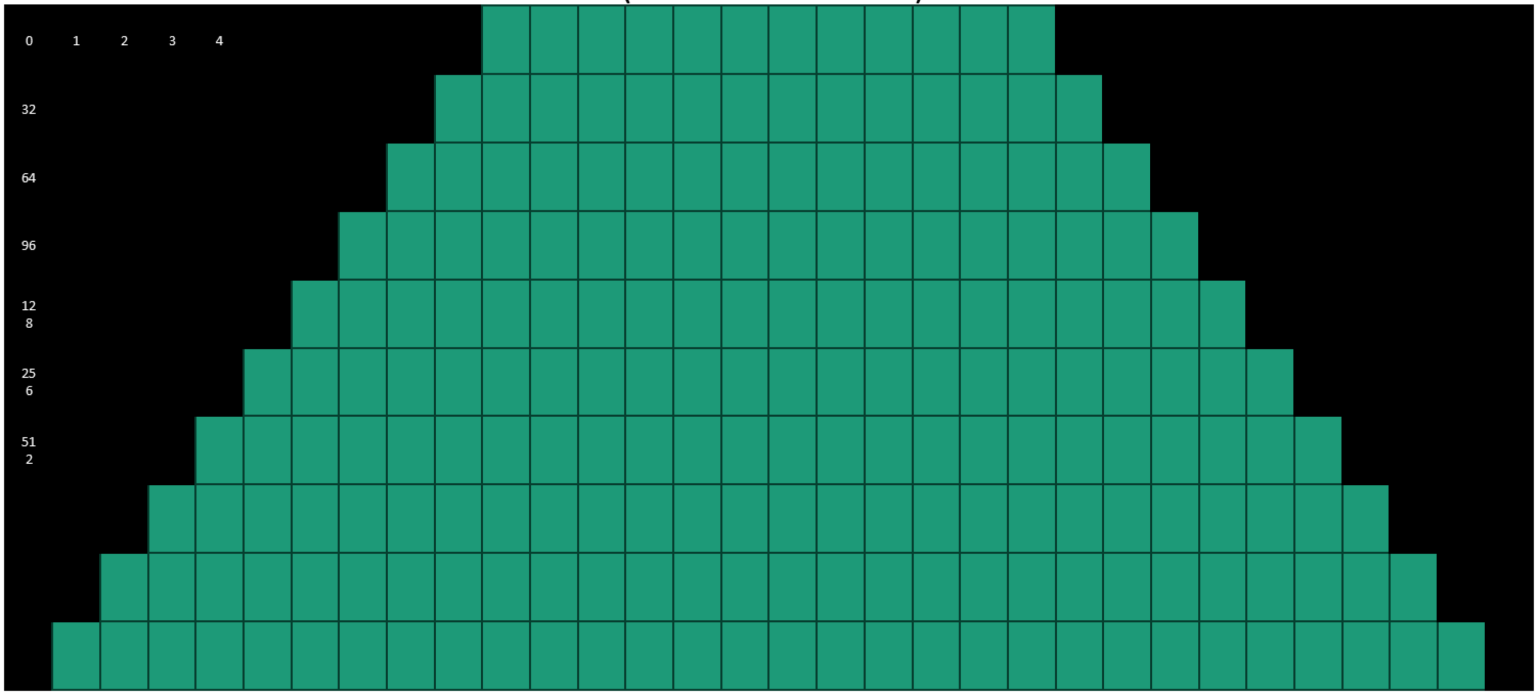
Project Progress – VGA (Cont.)

640 (640 x RGB x 8bits)

640 (32 x 20 x G x 8bits)

480
(480)

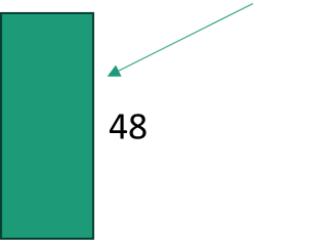
480
(48 * 10)



10

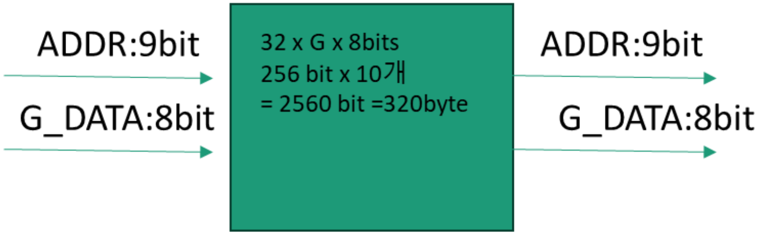
32

8bit



48

DPRAM



ADDR:9bit

G_DATA:8bit

32 x G x 8bits
256 bit x 10개
= 2560 bit =320byte

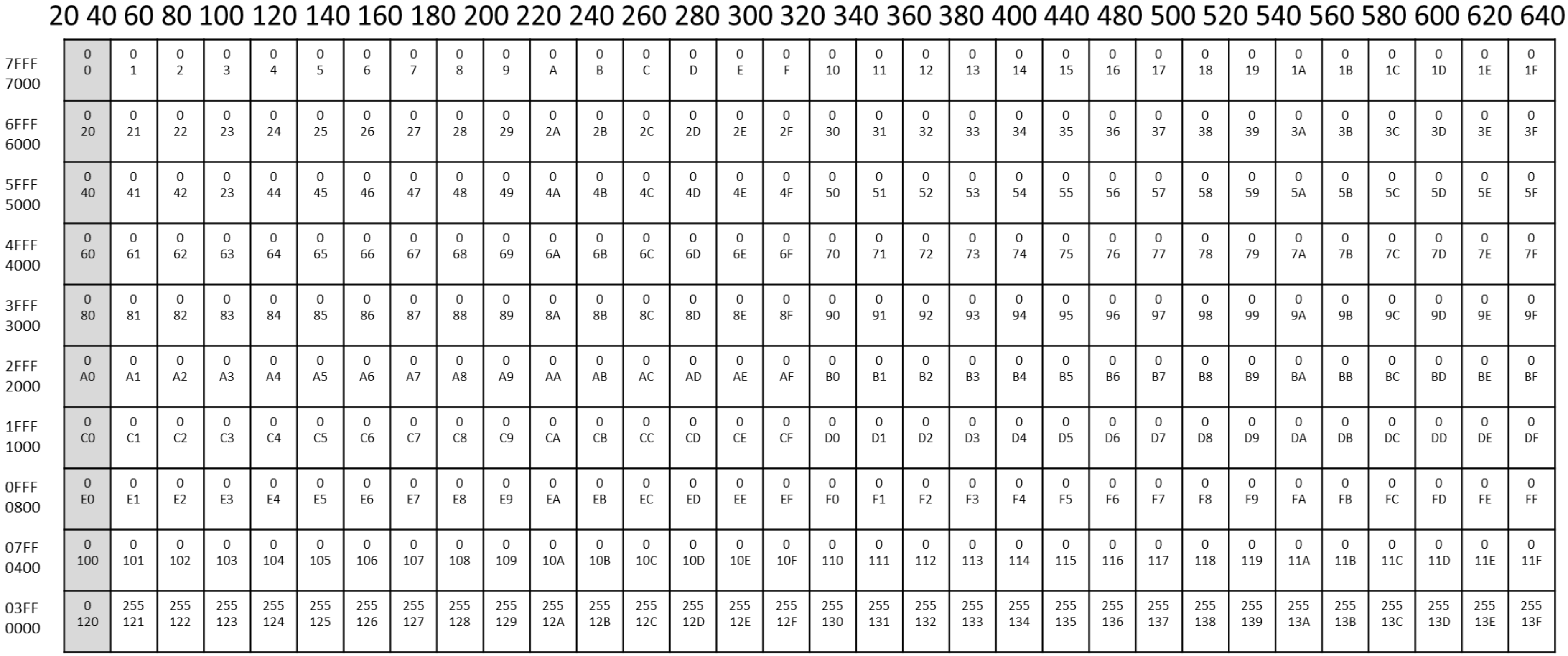
ADDR:9bit

G_DATA:8bit

0:
black

1:
BLue

Project Progress – VGA (Cont.)

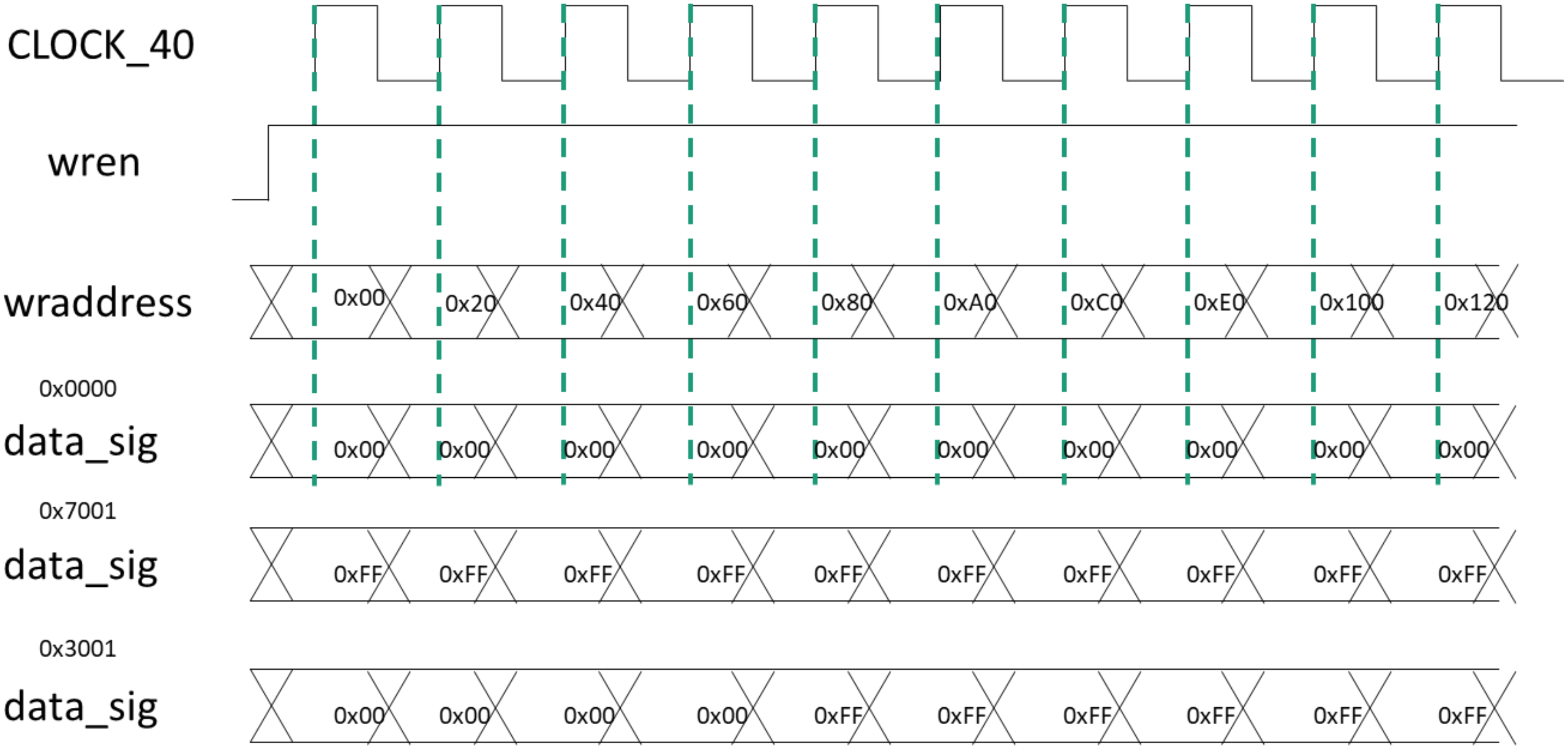


20 40 60 80 100 120 140 160 180 200 220 240 260 280 300 320 340 360 380 400 440 480 500 520 540 560 580 600 620 640

7FFF 7000	0 0	0 1	0 2	0 3	0 4	0 5	0 6	0 7	0 8	0 9	0 A	0 B	0 C	0 D	0 E	0 F	0 10	0 11	0 12	0 13	0 14	0 15	0 16	0 17	0 18	0 19	0 1A	0 1B	0 1C	0 1D	0 1E	0 1F
6FFF 6000	0 20	0 21	0 22	0 23	0 24	0 25	0 26	0 27	0 28	0 29	0 2A	0 2B	0 2C	0 2D	0 2E	0 2F	0 30	0 31	0 32	0 33	0 34	0 35	0 36	0 37	0 38	0 39	0 3A	0 3B	0 3C	0 3D	0 3E	0 3F
5FFF 5000	0 40	0 41	0 42	0 23	0 44	0 45	0 46	0 47	0 48	0 49	0 4A	0 4B	0 4C	0 4D	0 4E	0 4F	0 50	0 51	0 52	0 53	0 54	0 55	0 56	0 57	0 58	0 59	0 5A	0 5B	0 5C	0 5D	0 5E	0 5F
4FFF 4000	0 60	0 61	0 62	0 63	0 64	0 65	0 66	0 67	0 68	0 69	0 6A	0 6B	0 6C	0 6D	0 6E	0 6F	0 70	0 71	0 72	0 73	0 74	0 75	0 76	0 77	0 78	0 79	0 7A	0 7B	0 7C	0 7D	0 7E	0 7F
3FFF 3000	0 80	0 81	0 82	0 83	0 84	0 85	0 86	0 87	0 88	0 89	0 8A	0 8B	0 8C	0 8D	0 8E	0 8F	0 90	0 91	0 92	0 93	0 94	0 95	0 96	0 97	0 98	0 99	0 9A	0 9B	0 9C	0 9D	0 9E	0 9F
2FFF 2000	0 A0	0 A1	0 A2	0 A3	0 A4	0 A5	0 A6	0 A7	0 A8	0 A9	0 AA	0 AB	0 AC	0 AD	0 AE	0 AF	0 B0	0 B1	0 B2	0 B3	0 B4	0 B5	0 B6	0 B7	0 B8	0 B9	0 BA	0 BB	0 BC	0 BD	0 BE	0 BF
1FFF 1000	0 C0	0 C1	0 C2	0 C3	0 C4	0 C5	0 C6	0 C7	0 C8	0 C9	0 CA	0 CB	0 CC	0 CD	0 CE	0 CF	0 D0	0 D1	0 D2	0 D3	0 D4	0 D5	0 D6	0 D7	0 D8	0 D9	0 DA	0 DB	0 DC	0 DD	0 DE	0 DF
0FFF 0800	0 E0	0 E1	0 E2	0 E3	0 E4	0 E5	0 E6	0 E7	0 E8	0 E9	0 EA	0 EB	0 EC	0 ED	0 EE	0 EF	0 F0	0 F1	0 F2	0 F3	0 F4	0 F5	0 F6	0 F7	0 F8	0 F9	0 FA	0 FB	0 FC	0 FD	0 FE	0 FF
07FF 0400	0 100	0 101	0 102	0 103	0 104	0 105	0 106	0 107	0 108	0 109	0 10A	0 10B	0 10C	0 10D	0 10E	0 10F	0 110	0 111	0 112	0 113	0 114	0 115	0 116	0 117	0 118	0 119	0 11A	0 11B	0 11C	0 11D	0 11E	0 11F
03FF 0000	0 120	255 121	255 122	255 123	255 124	255 125	255 126	255 127	255 128	255 129	255 12A	255 12B	255 12C	255 12D	255 12E	255 12F	255 130	255 131	255 132	255 133	255 134	255 135	255 136	255 137	255 138	255 139	255 13A	255 13B	255 13C	255 13D	255 13E	255 13F

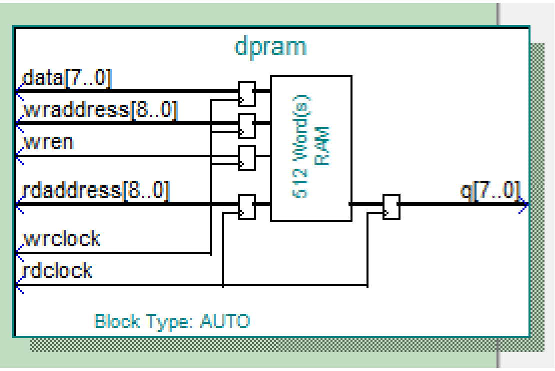
- Defined each Pixel

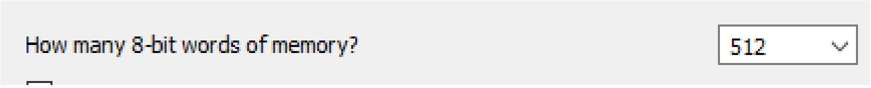
Project Progress – VGA (Cont.)

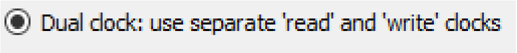

CLOCK_40
wren
wraddress
0x00 0x20 0x40 0x60 0x80 0xA0 0xC0 0xE0 0x100 0x120
0x0000
data_sig
0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
0x7001
data_sig
0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF
0x3001
data_sig
0x00 0x00 0x00 0x00 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF

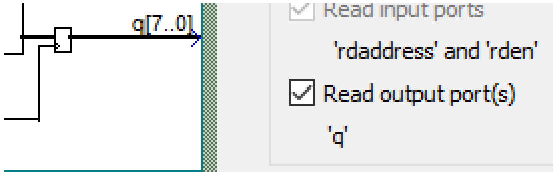
- Defined each Pixel

Project Progress – Dual Port RAM



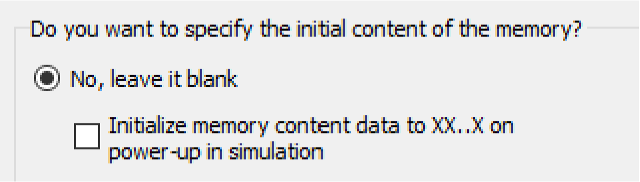

How many 8-bit words of memory? 512


● Dual clock: use separate 'read' and 'write' clocks



☑ Read input ports 'rdaddress' and 'rden'

☑ Read output port(s) 'q'

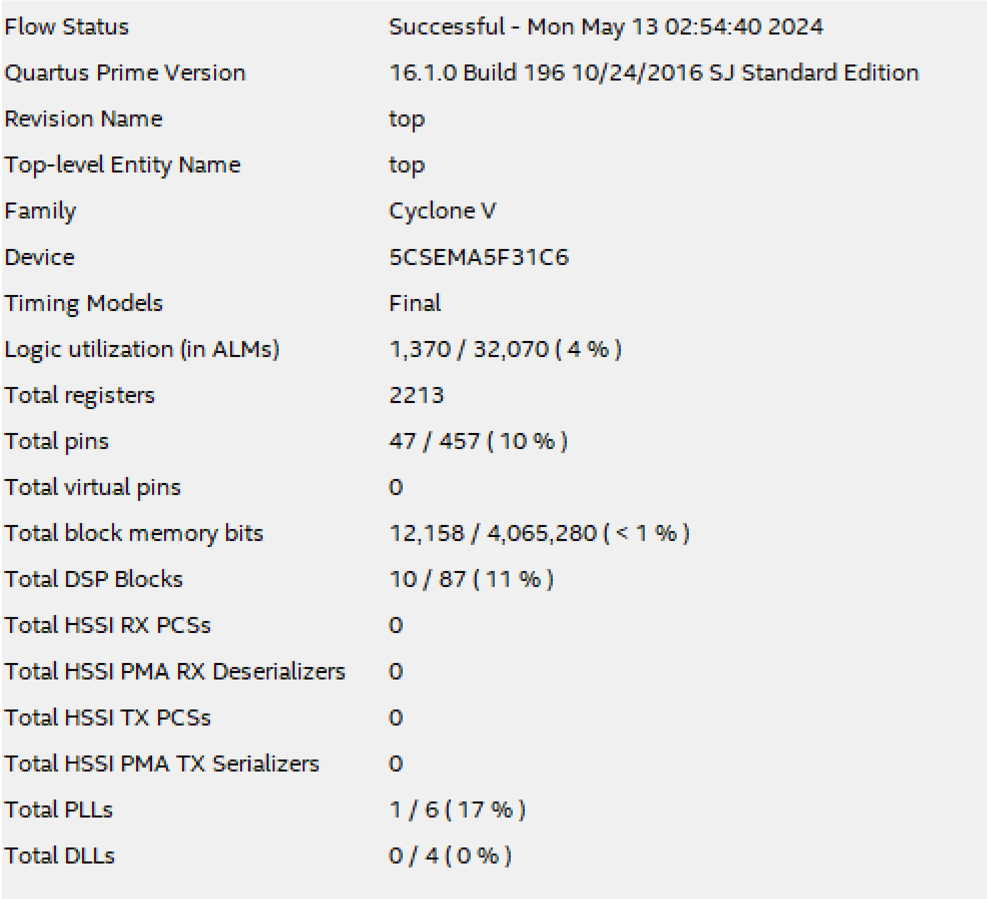

Do you want to specify the initial content of the memory?

● No, leave it blank

□ Initialize memory content data to XX..X on power-up in simulation

- Dual Port RAM IP

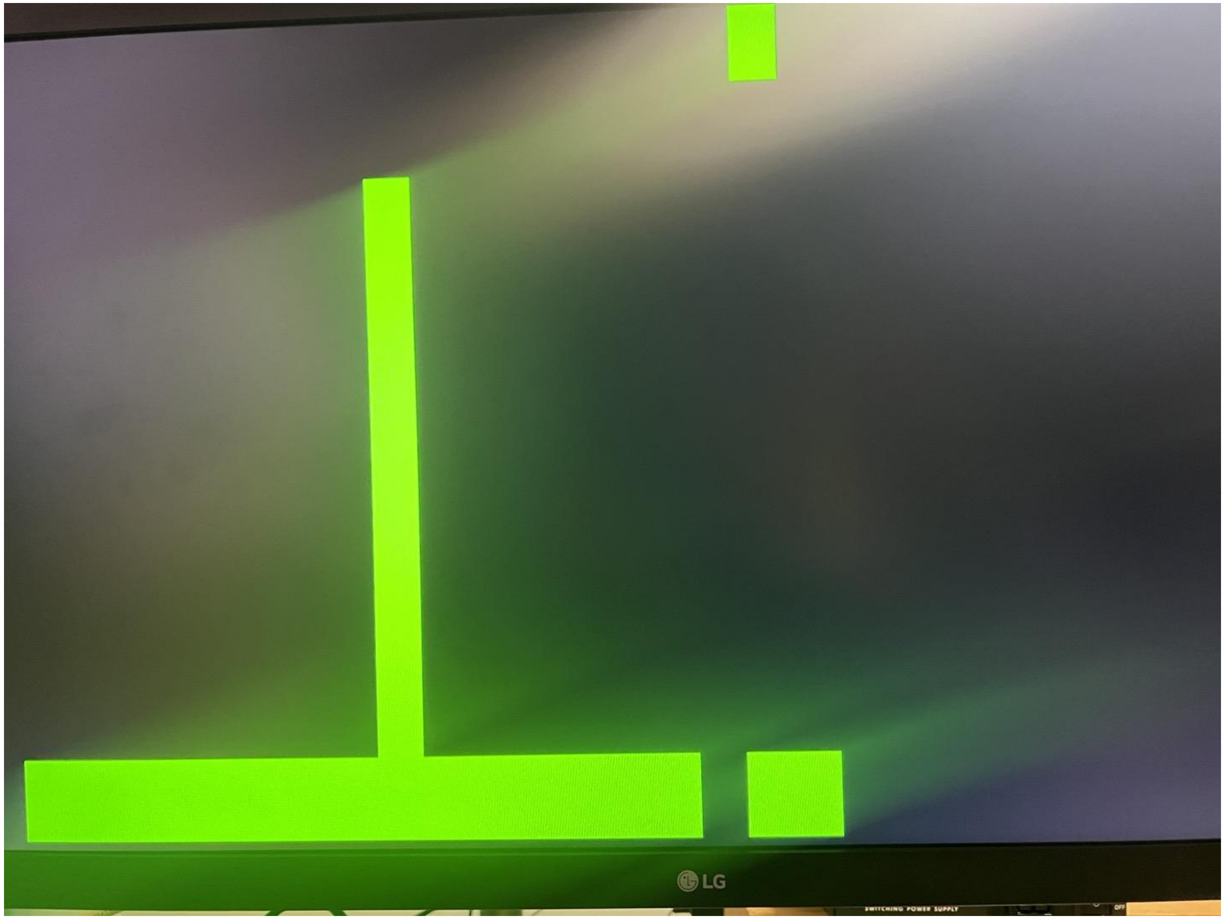
Project Progress – Top Synthesis



Flow Status	Successful - Mon May 13 02:54:40 2024
Quartus Prime Version	16.1.0 Build 196 10/24/2016 SJ Standard Edition
Revision Name	top
Top-level Entity Name	top
Family	Cyclone V
Device	5CSEMA5F31C6
Timing Models	Final
Logic utilization (in ALMs)	1,370 / 32,070 (4 %)
Total registers	2213
Total pins	47 / 457 (10 %)
Total virtual pins	0
Total block memory bits	12,158 / 4,065,280 (< 1 %)
Total DSP Blocks	10 / 87 (11 %)
Total HSSI RX PCSs	0
Total HSSI PMA RX Deserializers	0
Total HSSI TX PCSs	0
Total HSSI PMA TX Serializers	0
Total PLLs	1 / 6 (17 %)
Total DLLs	0 / 4 (0 %)

- ➢ Top Synthesis

Project Progress – Top Synthesis



- Result : 4Vp-p Sine wave @100Khz

Further Improvements

- 디스플레이 출력 오류.
- 프레임이 너무 빠르게 업데이트 됨.
- 스펙트럼 분석기의 화면의 분해능과 해상도 개선.

Q & A