Lab 3

Sara Frazer

11/3/23

CDA 3203 Computer Logic Design

Fall 2023

Dr. Maria Petrie

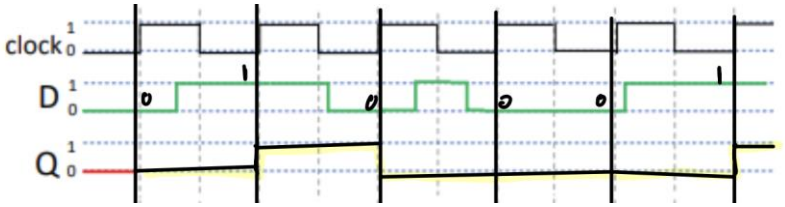
Florida Atlantic University

Hand Work

1.1 D Flip Flop

1.1 D Flip Flop

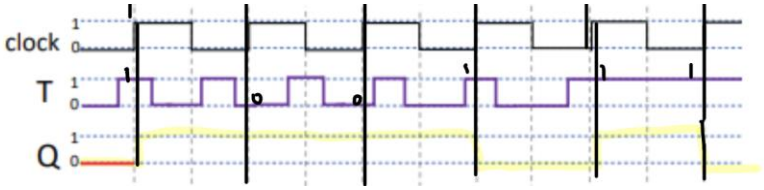
D	Q	Command	Q+
0	0	make Q=0	0
	1		0
1	0	make Q=1	1
	1		1



1.2 T Flip Flop

1.2 T Flip Flop

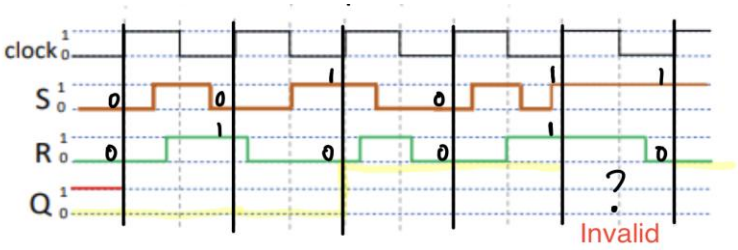
T	Q	Command	Q+
0	0	Hold	0
	1		1
1	0	Toggle	1
	1		0



1.3 S-R Flip Flop

1.3 S-R Flip Flop

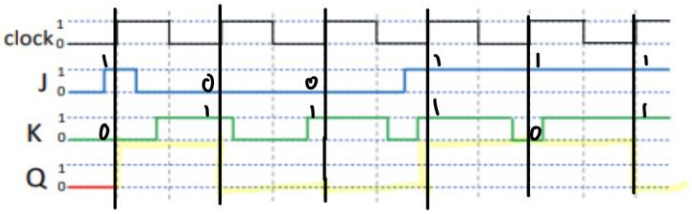
S	R	Q	Command	Q+
0	0	0	Hold	0
		1		1
0	1	0	Reset	0
		1		0
1	0	0	Set	1
		1		1
1	1	0	Invalid	?
		1		?



1.4 J-K Flip Flop

1.4 J-K Flip Flop

J	K	Q	Command	Q+
0	0	0	Hold	0
		1		1
0	1	0	Reset	0
		1		0
1	0	0	Set	1
		1		1
1	1	0	Toggle	1
		1		0



1.5 Excitation Tables

Q→Q*	Command	D
0→0	Q=0	0
0→1	Q=1	1
1→0	Q=0	0
1→1	Q=1	1

Q→Q*	Command	T
0→0	Hold	0
0→1	Toggle	1
1→0	Toggle	1
1→1	Hold	0

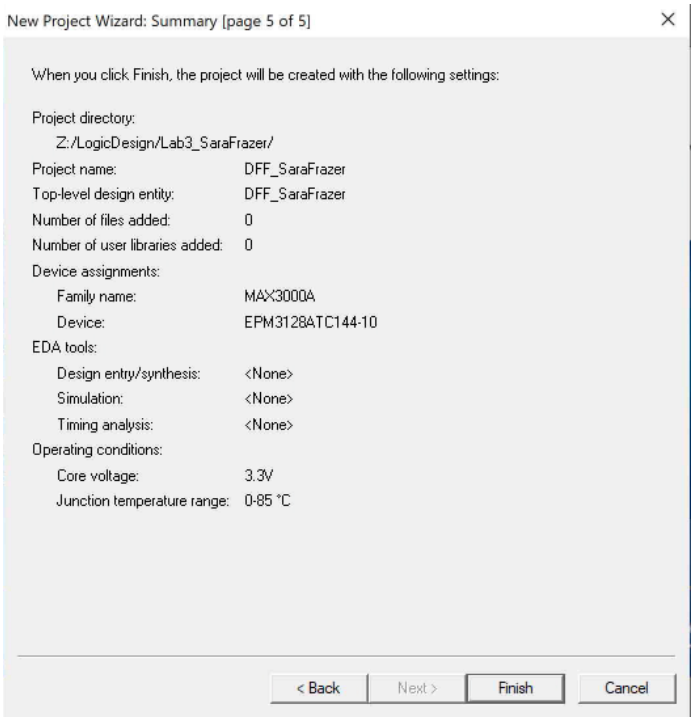
Q→Q*	Commands	S R
0→0	H 00 R 01	0x
0→1	S 10	10
1→0	R 01	01
1→1	H 00 S 10	x0

Q→Q*	Commands	J K
0→0	H 00 R 01	0x
0→1	T 11 S 10	1x
1→0	T 11 R 01	x1
1→1	H 00 S 10	x0

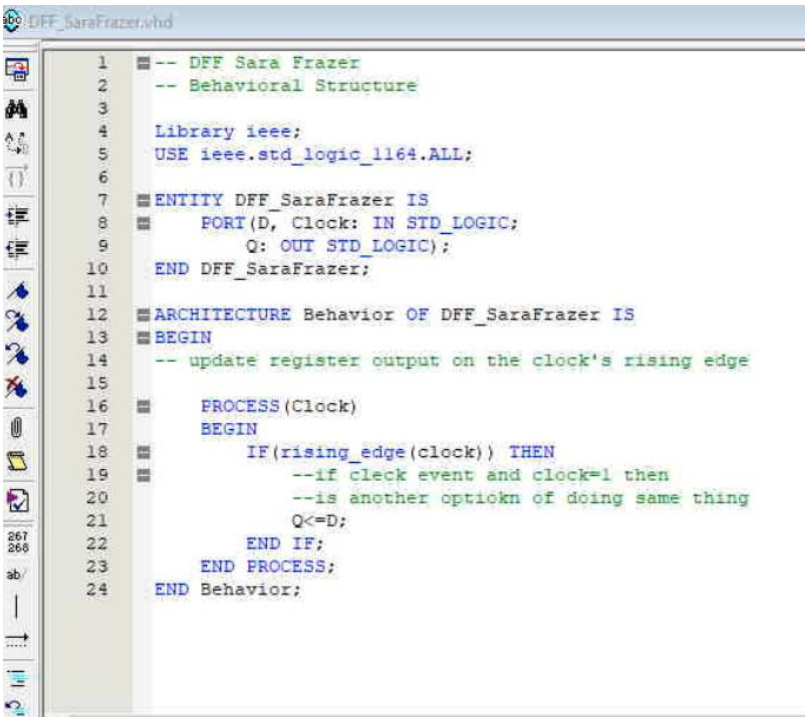
2.0 Design and Simulation of Sequential Components in Altera Quartus Using VHDL

2.1 Data Flip Flop (DFF)

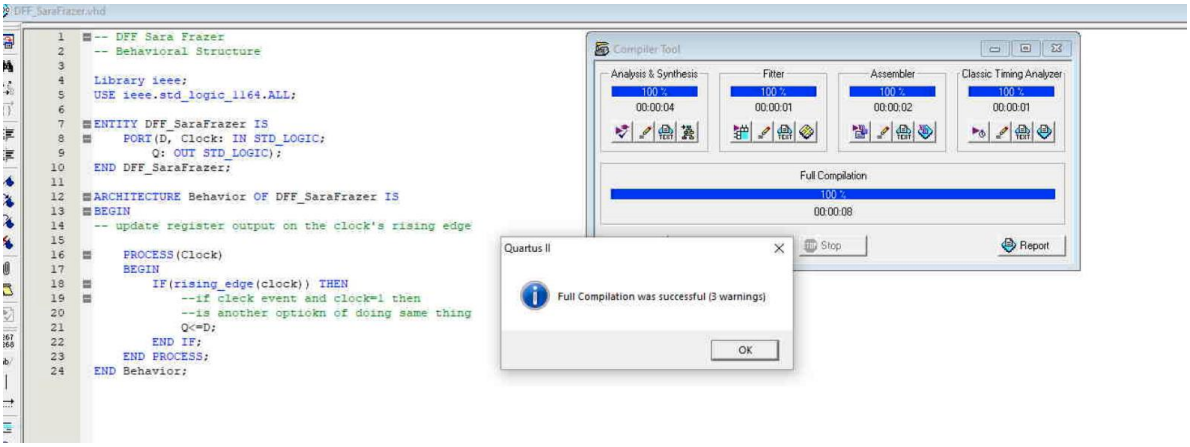
Project Wizard Settings



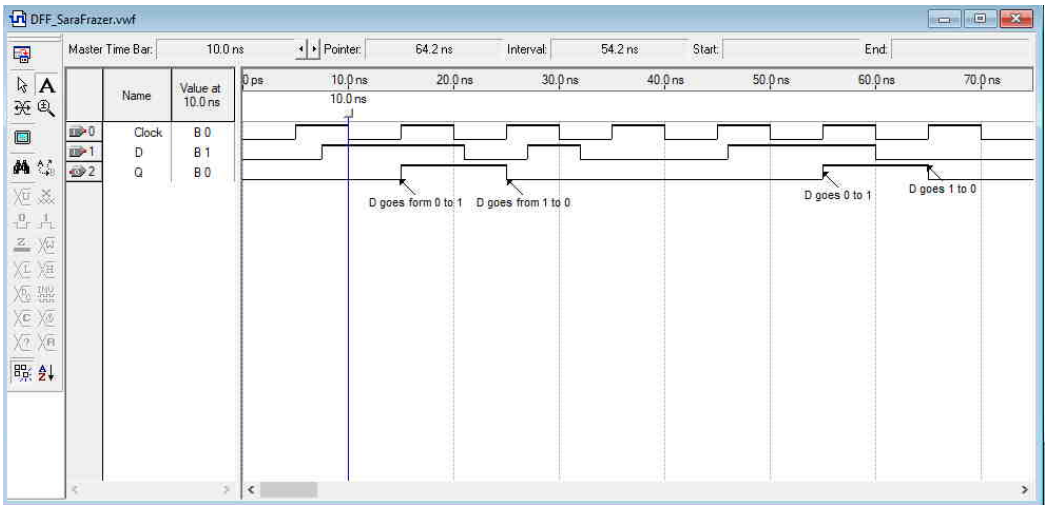
VHDL Code



Successful Compilation



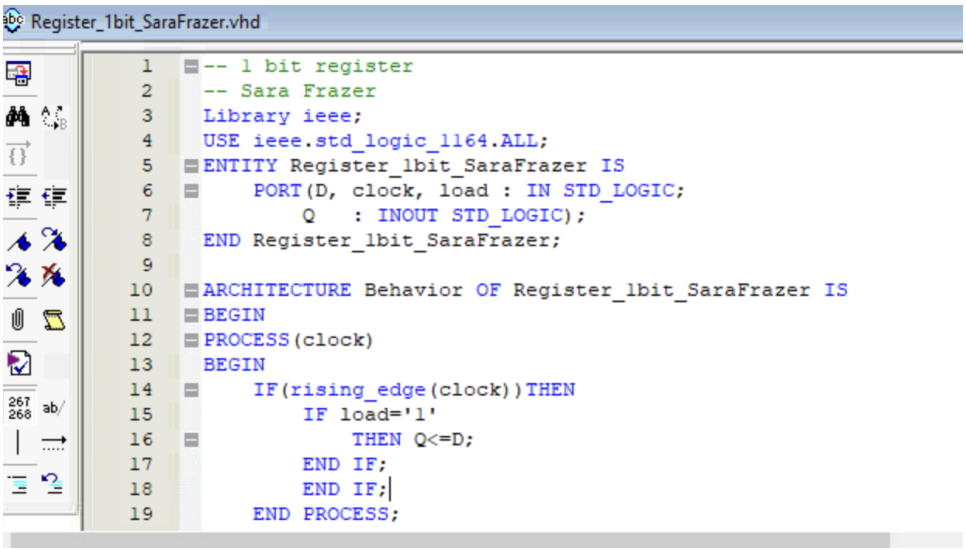
Timing Diagram



2.2-1 1-Bit Register

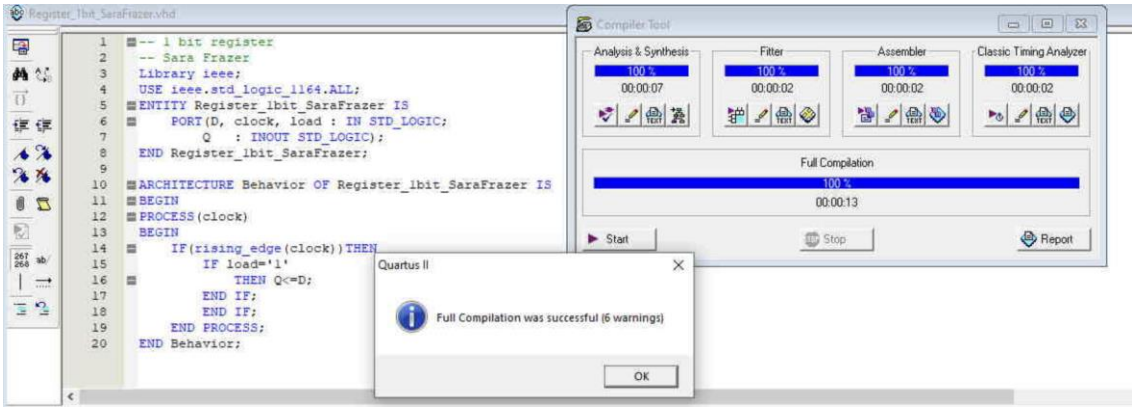
Project Settings

VHDL Code

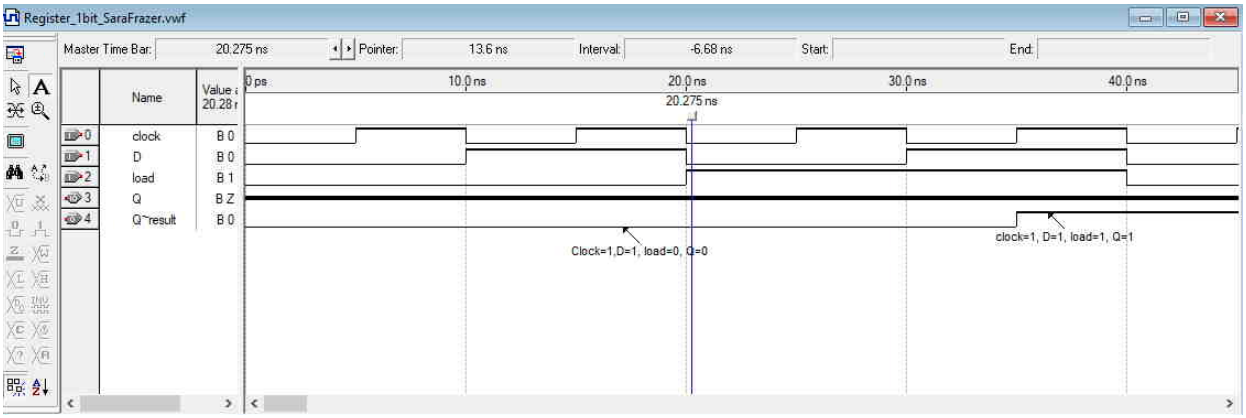


I accidentally cut off the “END Behavior;”

Successful Compilation

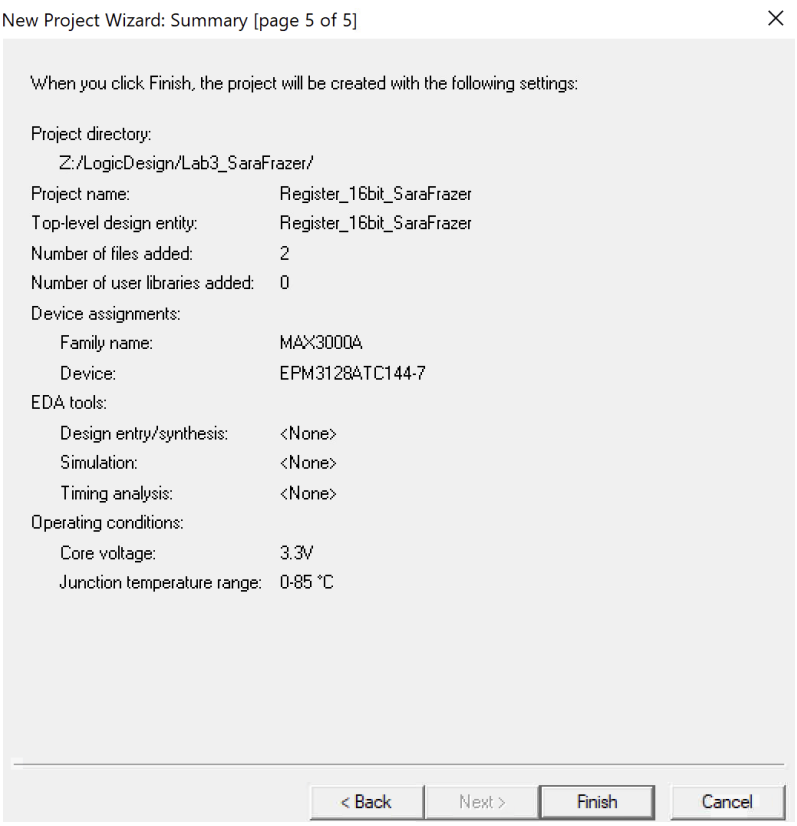


Timing Diagram

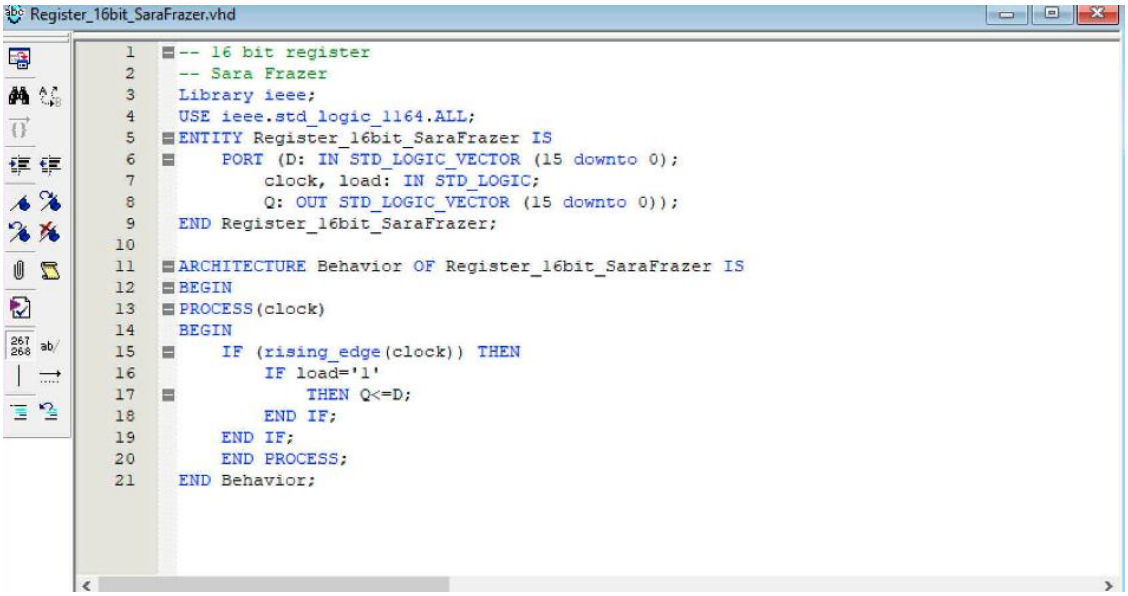


2.3 16-Bit Register

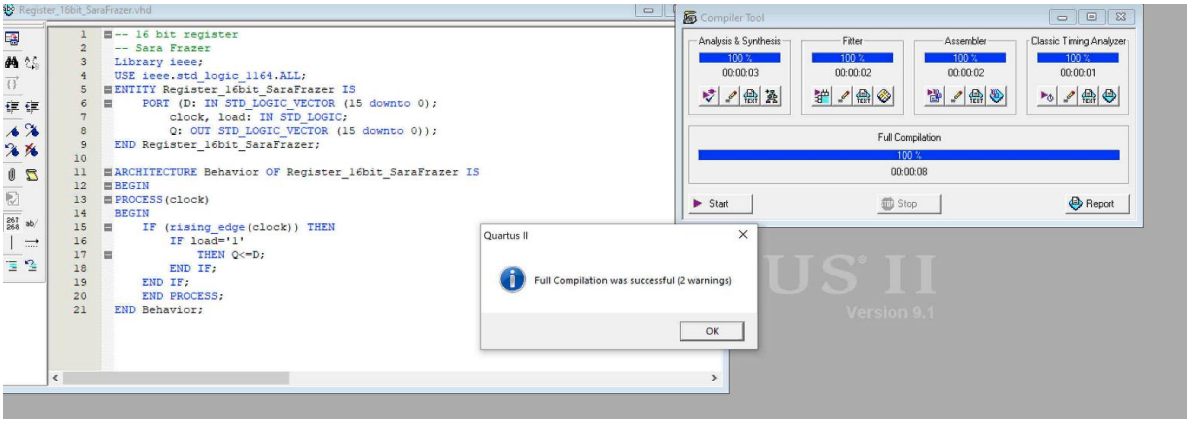
Project Settings



VHDL



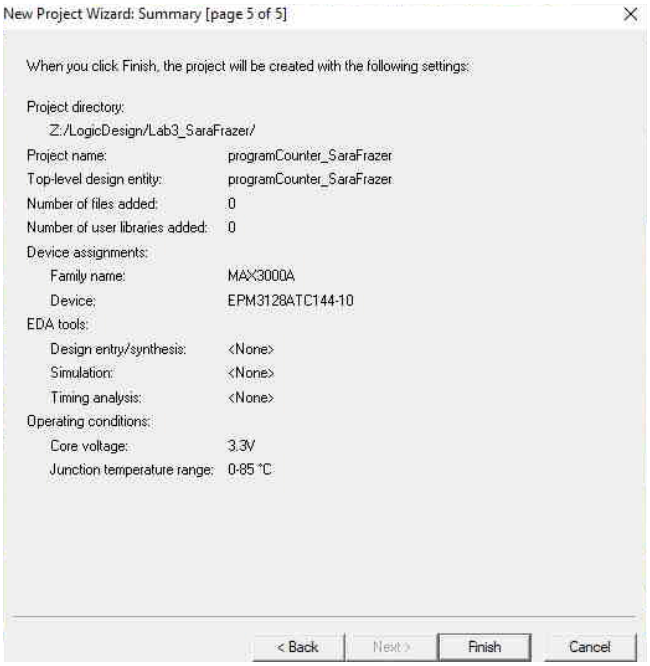
Successful Compilation



Timing Diagram

2.4 Program Counter Register

Project Settings

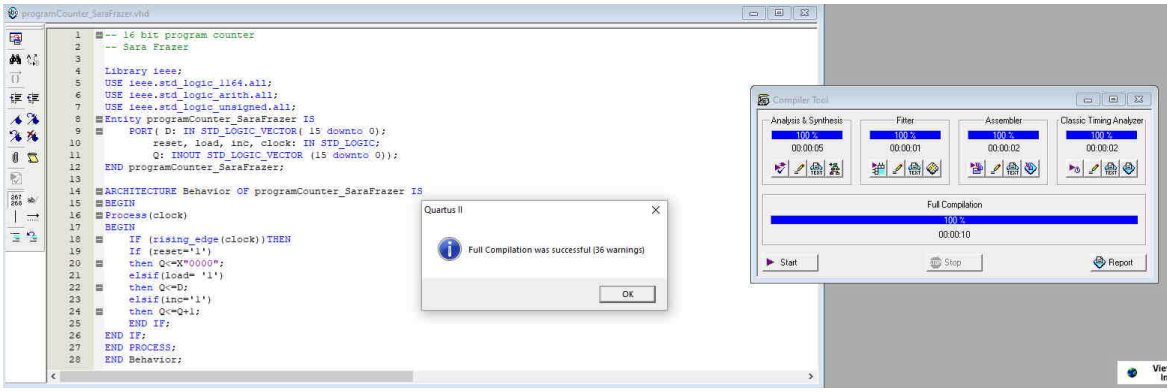


VHDL Code

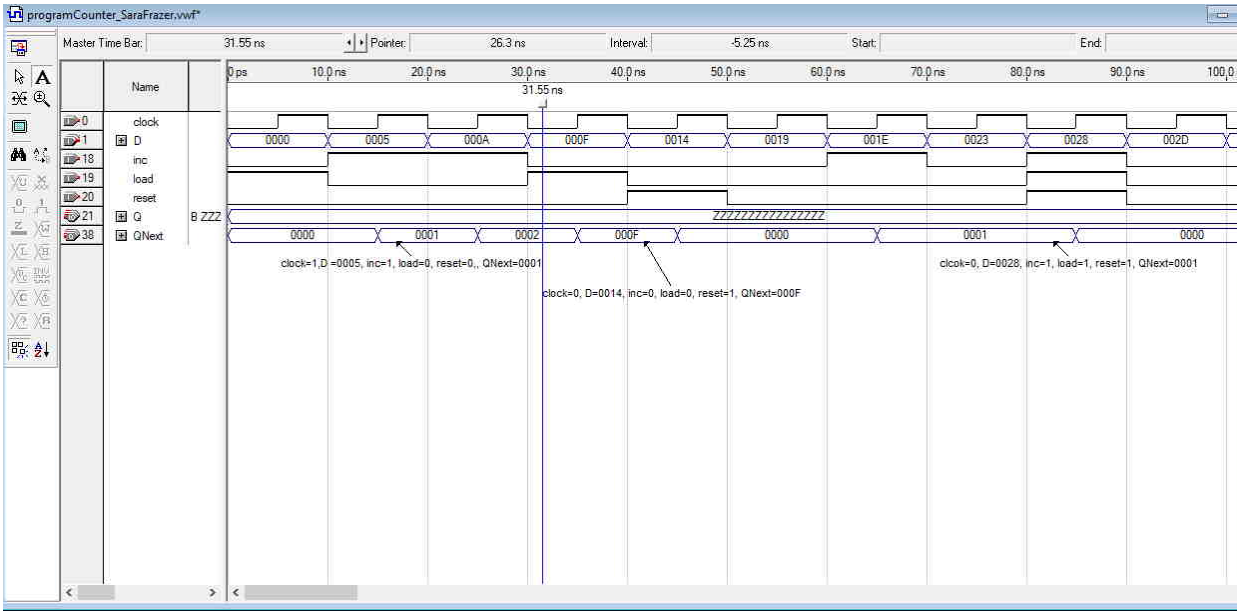
programCounter_SaraFrazer.vhd

```
1  -- 16 bit program counter
2  -- Sara Frazer
3
4   Library ieee;
5   USE ieee.std_logic_1164.all;
6   USE ieee.std_logic_arith.all;
7   USE ieee.std_logic_unsigned.all;
8  Entity programCounter_SaraFrazer IS
9      PORT( D: IN STD_LOGIC_VECTOR( 15 downto 0);
10         reset, load, inc, clock: IN STD_LOGIC;
11         Q: INOUT STD_LOGIC_VECTOR (15 downto 0));
12  END programCounter_SaraFrazer;
13
14 ARCHITECTURE Behavior OF programCounter_SaraFrazer IS
15 BEGIN
16 Process(clock)
17  BEGIN
18     IF (rising_edge(clock))THEN
19     If (reset='1')
20     then Q<=X"0000";
21     elsif(load= '1')
22     then Q<=D;
23     elsif(inc='1')
24     then Q<=Q+1;
25     END IF;
26  END IF;
27  END PROCESS;
28  END Behavior;
```

Successful Compilation

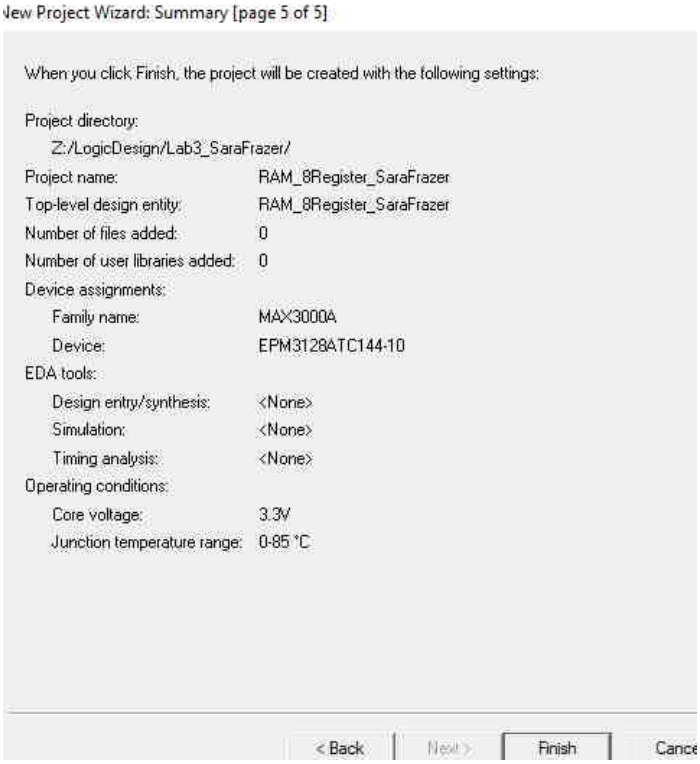


Timing Diagram

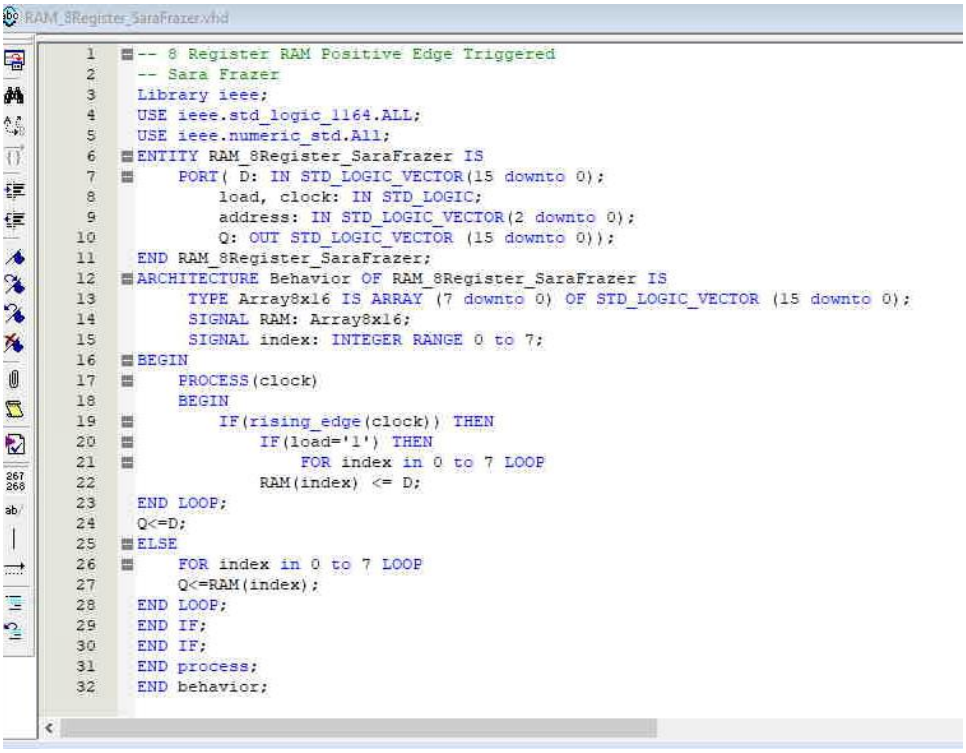


2.5 8 Register RAM

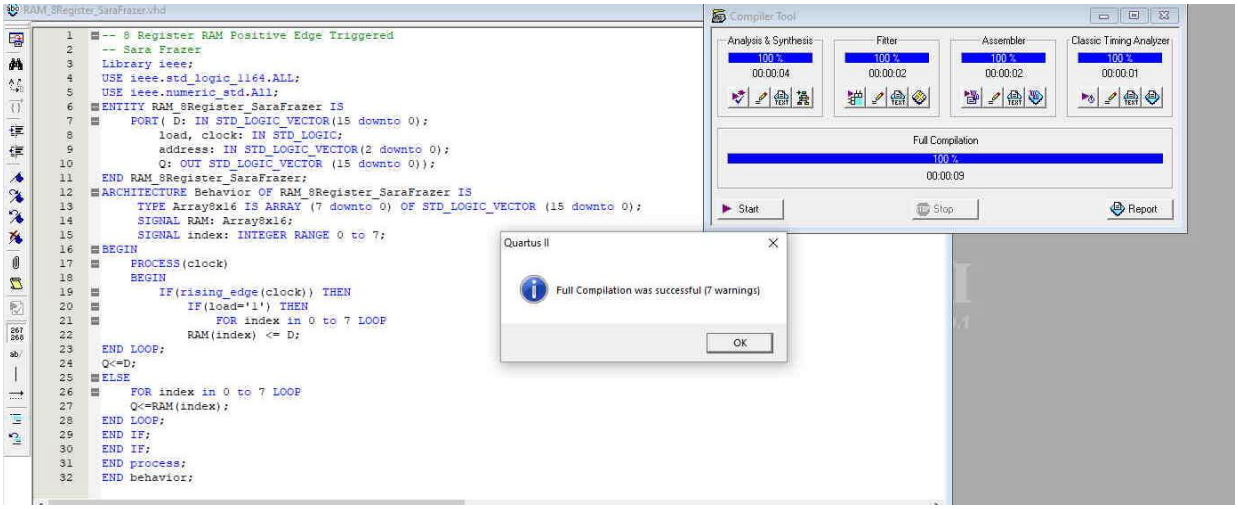
Project Settings



VHDL Code



Successful Compilation



Timing Diagram