Experiment No 4

Vhdl fifo memory simulat

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

use IEEE.STD\_LOGIC\_ARITH.ALL;

use IEEE.STD\_LOGIC\_UNSIGNED.ALL;

---- Uncomment the following library declaration if instantiating

---- any Xilinx primitives in this code.

--library UNISIM;

--use UNISIM.VComponents.all;

entity FIFO is

Port ( DI : in std\_logic\_vector(3 downto 0);

DO : out std\_logic\_vector(3 downto 0);

RW : in std\_logic;

FULL : out std\_logic;

RS : in std\_logic;

CLK : in std\_logic);

end FIFO;

architecture Behavioral of FIFO is

begin

process(RS,CLK)

type memory is Array( 0 to 3)of std\_logic\_vector(3 downto 0);

variable mem: memory;

variable r\_p:integer range 0 to 3;

variable w\_p:integer range 0 to 3;

variable overwrite :boolean;

begin

if RS='1' then

DO<="0000";

elsif(CLK'event and CLK='1') then

if RW='1'then

if (overwrite=False OR w\_p/=r\_p) then

mem (w\_p):=DI;

if w\_p=3 then

w\_p:=0 ;

else

w\_p:=w\_p+1;

overwrite:=False;

end if;

end if;

elseif RW='0' then

if (overwrite=False OR w\_p/=r\_p) then

DO<=mem(r\_p);

if (r\_p=3 ) then

r\_p:=0 ;

overwrite:=true;

else

r\_p:=r\_p+1;

end if;

end if;

end if;

if w\_p=r\_p then

if overwrite=true then

FULL<='1';

else

FULL<='0';

end if;

else

FULL<='0';

end if;

end if;

end process;

end Behavioral;

ENTITY SSSA\_vhd IS

END SSSA\_vhd;

ARCHITECTURE behavior OF SSSA\_vhd IS

-- Component Declaration for the Unit Under Test (UUT)

COMPONENT fifo

PORT(

DI : IN std\_logic\_vector(3 downto 0);

RW : IN std\_logic;

RS : IN std\_logic;

CLK : IN std\_logic;

DO : OUT std\_logic\_vector(3 downto 0);

FULL : OUT std\_logic

);

END COMPONENT;

--Inputs

SIGNAL RW : std\_logic := '0';

SIGNAL RS : std\_logic := '0';

SIGNAL CLK : std\_logic := '0';

SIGNAL DI : std\_logic\_vector(3 downto 0) := (others=>'0');

--Outputs

SIGNAL DO : std\_logic\_vector(3 downto 0);

SIGNAL FULL : std\_logic;

constant CLK\_period : time := 10 ns;

BEGIN

-- Instantiate the Unit Under Test (UUT)

uut: fifo PORT MAP(

DI => DI,

DO => DO,

RW => RW,

FULL => FULL,

RS => RS,

CLK => CLK

);

process

begin

CLK<='0' ;

wait for CLK\_period/2;

CLK<='1' ;

wait for CLK\_period/2;

end process;

tb : PROCESS

BEGIN

RS<='1';

wait for 100 ns;

RS<='0';

RW<='1';

DI<="0011";

-- Wait 100 ns for global reset to finish

wait for 10 ns;

RW<='1';

DI<="0110";

-- Wait 100 ns for global reset to finish

wait for 10 ns;

RW<='1';

DI<="1100";

-- Wait 100 ns for global reset to finish

wait for 10 ns;

RW<='1';

DI<="1001";

-- Wait 100 ns for global reset to finish

wait for 10 ns;

-- Place stimulus here

RW<='0';

-- Wait 100 ns for global reset to finish

wait for 10 ns;

RW<='0';

-- Wait 100 ns for global reset to finish

wait for 10 ns;

RW<='0';

-- Wait 100 ns for global reset to finish

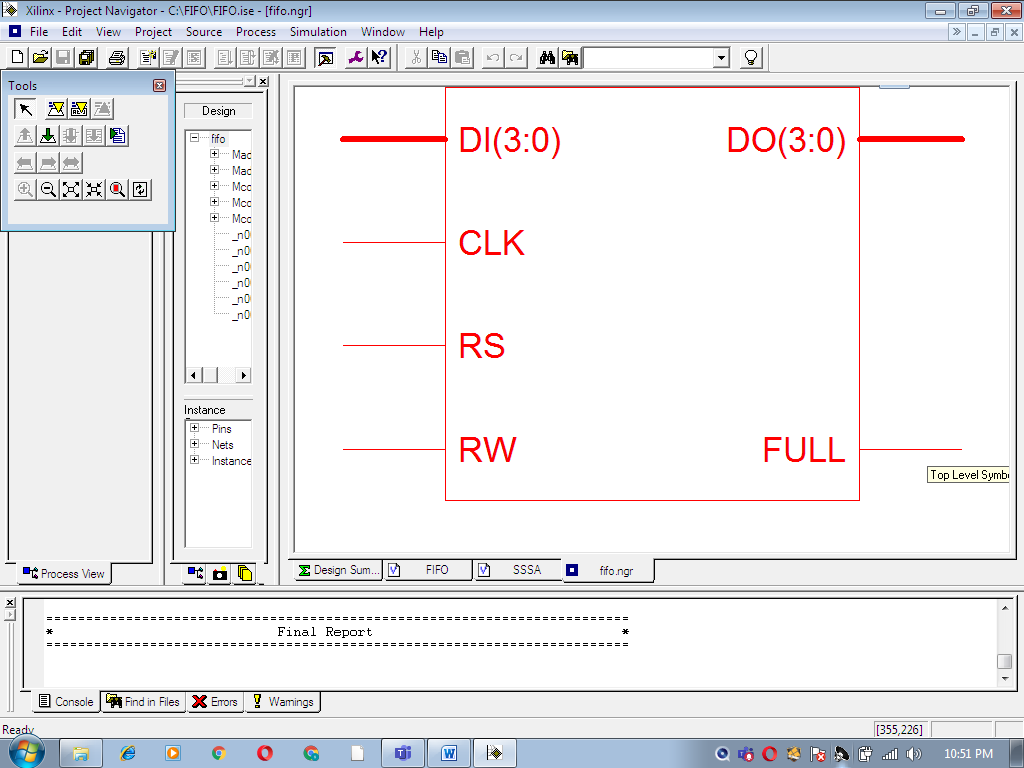
wait for 10 ns;

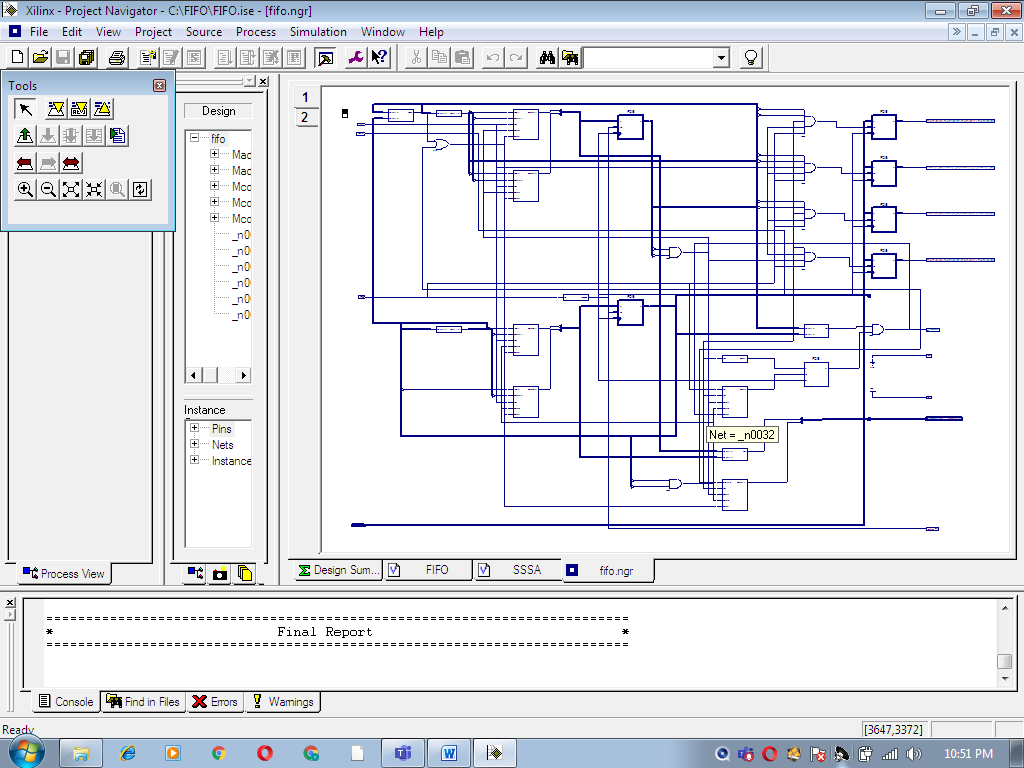
WAIT FOR CLK\_period\*10;

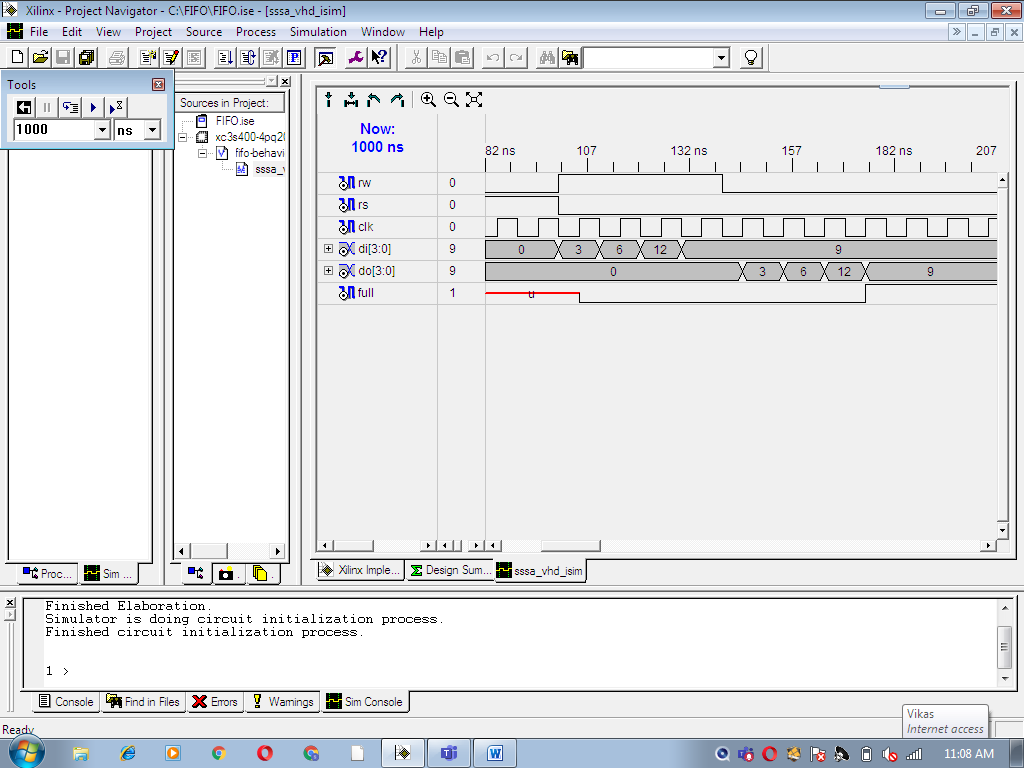
wait; **-- will wait forever**

END PROCESS;

END;







=========================================================================

\* Synthesis Options Summary \*

=========================================================================

---- Source Parameters

Input File Name : "fifo.prj"

Input Format : mixed

Ignore Synthesis Constraint File : NO

---- Target Parameters

Output File Name : "fifo"

Output Format : NGC

Target Device : xc3s400-4-pq208

---- Source Options

Top Module Name : fifo

Automatic FSM Extraction : YES

FSM Encoding Algorithm : Auto

FSM Style : lut

RAM Extraction : Yes

RAM Style : Auto

ROM Extraction : Yes

ROM Style : Auto

Mux Extraction : YES

Decoder Extraction : YES

Priority Encoder Extraction : YES

Shift Register Extraction : YES

Logical Shifter Extraction : YES

XOR Collapsing : YES

Resource Sharing : YES

Multiplier Style : auto

Automatic Register Balancing : No

**-**--- Target Options

Add IO Buffers : YES

Global Maximum Fanout : 500

Add Generic Clock Buffer(BUFG) : 8

Register Duplication : YES

Equivalent register Removal : YES

Slice Packing : YES

Pack IO Registers into IOBs : auto

---- General Options

Optimization Goal : Speed

Optimization Effort : 1

Keep Hierarchy : NO

Global Optimization : AllClockNets

RTL Output : ONLY

Write Timing Constraints : NO

Hierarchy Separator : /

Bus Delimiter : <>

Case Specifier : maintain

Slice Utilization Ratio : 100

Slice Utilization Ratio Delta : 5

---- Other Options

lso : fifo.lso

Read Cores : YES

cross\_clock\_analysis : NO

verilog2001 : YES

safe\_implementation : No

Optimize Instantiated Primitives : NO

use\_clock\_enable : Yes

use\_sync\_set : Yes

use\_sync\_reset : Yes

enable\_auto\_floorplanning : No

=========================================================================

=========================================================================

\* HDL Compilation \*

=========================================================================

Compiling vhdl file "C:/FIFO/FIFO.vhd" in Library work.

Entity <FIFO> compiled.

Entity <FIFO> (Architecture <Behavioral>) compiled.

=========================================================================

\* HDL Analysis \*

=========================================================================

Analyzing Entity <fifo> (Architecture <Behavioral>).

Entity <fifo> analyzed. Unit <fifo> generated.

=========================================================================

\* HDL Synthesis \*

=========================================================================

Synthesizing Unit <fifo>.

Related source file is "C:/FIFO/FIFO.vhd".

Found 4-bit register for signal <DO>.

Found 1-bit register for signal <FULL>.

Found 4-bit 4-to-1 multiplexer for signal <$n0015> created at line 69.

Found 1-bit 4-to-1 multiplexer for signal <$n0018>.

Found 2-bit adder for signal <$n0019> created at line 62.

Found 2-bit adder for signal <$n0020> created at line 75.

Found 2-bit comparator equal for signal <$n0021> created at line 81.

Found 2-bit comparator not equal for signal <$n0022> created at line 56.

Found 2-bit comparator equal for signal <$n0026> created at line 56.

Found 16-bit register for signal <mem>.

Found 1-bit register for signal <overwrite<0>>.

Found 2-bit register for signal <r\_p>.

Found 2-bit register for signal <w\_p>.

Summary:

inferred 26 D-type flip-flop(s).

inferred 2 Adder/Subtractor(s).

inferred 3 Comparator(s).

inferred 5 Multiplexer(s).

Unit <fifo> synthesized.

=========================================================================

\* Advanced HDL Synthesis \*

=========================================================================

Advanced RAM inference ...

Advanced multiplier inference ...

Advanced Registered AddSub inference ...

Dynamic shift register inference ...

=========================================================================

HDL Synthesis Report

Macro Statistics

# Adders/Subtractors : 2

2-bit adder : 2

# Registers : 9

1-bit register : 2

2-bit register : 2

4-bit register : 5

# Comparators : 3

2-bit comparator equal : 2

2-bit comparator not equal : 1

# Multiplexers : 2

1-bit 4-to-1 multiplexer : 1

4-bit 4-to-1 multiplexer : 1

=========================================================================

=========================================================================

\* Low Level Synthesis \*

=========================================================================

=========================================================================

\* Final Report \*

=========================================================================

Final Results

RTL Top Level Output File Name : fifo.ngr

Keep Hierarchy : NO

Design Statistics

# IOs : 12

Cell Usage :

# BELS : 2

# GND : 1

# VCC : 1

# FlipFlops/Latches : 26

# FDCE : 4

# FDE : 22

=========================================================================

CPU : 5.92 / 8.26 s | Elapsed : 6.00 / 8.00 s

-->

Total memory usage is 122500 kilobytes

Number of errors : 0 ( 0 filtered)

Number of warnings : 0 ( 0 filtered)

Number of infos : 0 ( 0 filtered)