INSTITUTO POLITÉCNICO NACIONAL

ESCUELA SUPERIOR DE COMPUTO

Std Logic Arith & signed

Practica 8

Materia:

Arquitectura de computadoras

Profesor:

Castillo Cabrera Gelacio

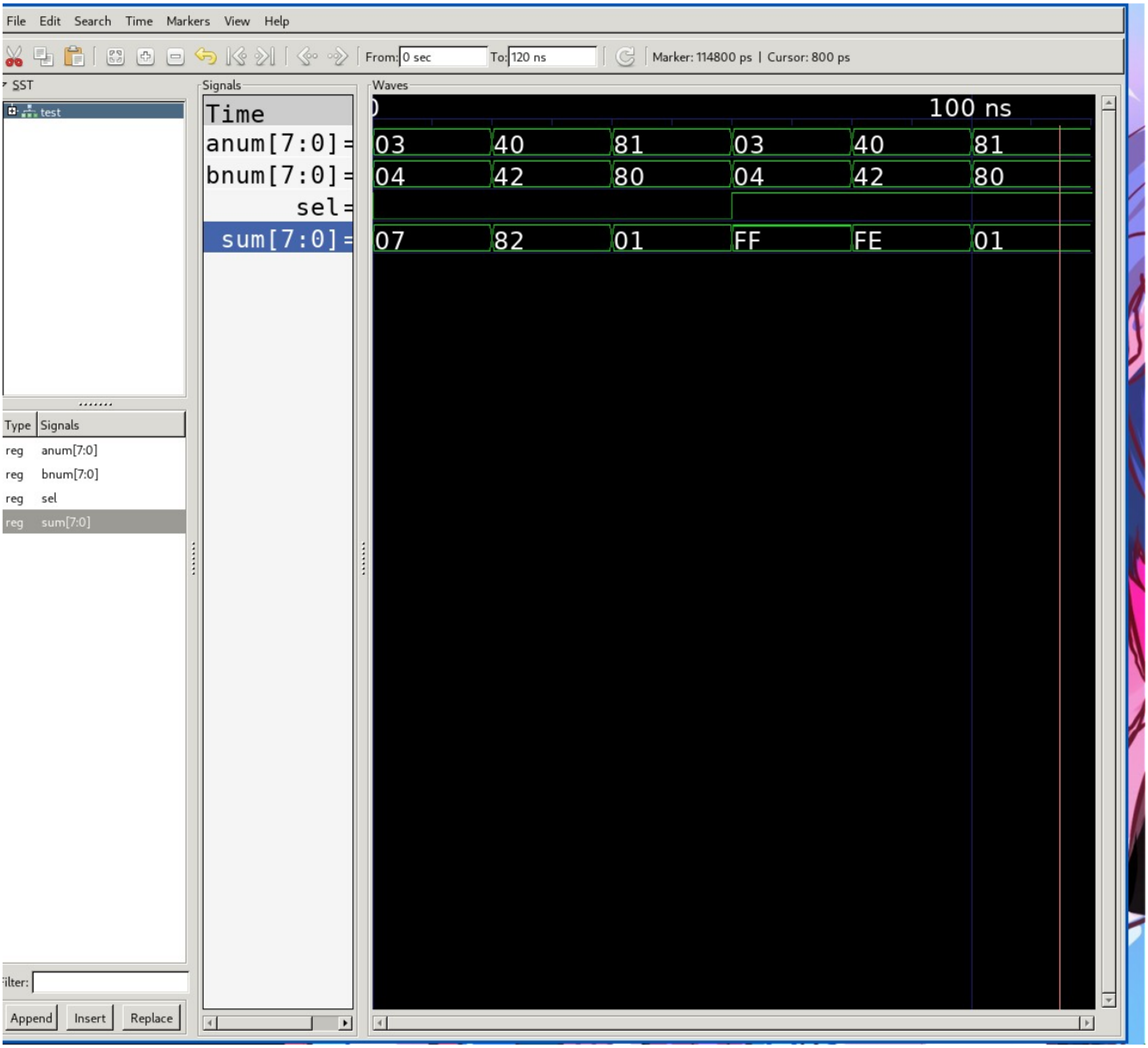
Alumno:

Cortés Piña Oziel

Grupo:

3CM12

Simulación en GHDL y GTLWAVE



```
../../src/synopsys/std_logic_arith.vhdl:255:20:@0ms:(assertion warning): There is
../../src/synopsys/std_logic_arith.vhdl:255:20:@0ms:(assertion warning): There is
gtkwave sim.vcd --rcvar 'fontname_waves Monospac 22' --rcvar 'fontname_signals Mon

GTKWave Analyzer v3.3.109 (w)1999-2020 BSI

RCVAR  | 'fontname_waves Monospac 22' FOUND
RCVAR  | 'fontname_signals Monospace 22' FOUND
[0] start time.
[120000000] end time.
WM Destroy
make[1]: se sale del directorio '/home/oz/docs/school/ESCOM/periods/6/arqui/repo/7
[1]+  Hecho                   libreoffice report.odt 2> /dev/null
 • 7 » cp report.odt ../8/report.odt
 • 7 » cd ../8/
 • 8 » libreoffice report.odt 2>/dev/null &
[1] 23439
 • 8 » m
make validate
make[1]: se entra en el directorio '/home/oz/docs/school/ESCOM/periods/6/arqui/rep
ghdl -s -fsynopsys *.vhdl
make[1]: se sale del directorio '/home/oz/docs/school/ESCOM/periods/6/arqui/repo/8
make build
make[1]: se entra en el directorio '/home/oz/docs/school/ESCOM/periods/6/arqui/rep
ghdl -a -fsynopsys *.vhdl
cat work-obj*
v 4
file . "test.vhdl" "a98d7d018c3ce0d6e5f5907f63544b6b22f5ca3a" "20210928223525.516"
  entity test at 1( 0) + 0 on 13;
  architecture arch of test at 9( 131) + 0 on 14;
file . "adder8bit.vhdl" "b4288d44e171a7536f29f84969879c8ee22e348a" "20210928223525
  entity adder8bit at 1( 0) + 0 on 11;
  architecture adder8bit_arch of adder8bit at 15( 265) + 0 on 12;
ghdl -e -fsynopsys test
make[1]: se sale del directorio '/home/oz/docs/school/ESCOM/periods/6/arqui/repo/8
make simulate
make[1]: se entra en el directorio '/home/oz/docs/school/ESCOM/periods/6/arqui/rep
ghdl -r -fsynopsys  test --vcd=sim.vcd
../../src/synopsys/std_logic_arith.vhdl:315:20:@0ms:(assertion warning): There is
../../src/synopsys/std_logic_arith.vhdl:315:20:@0ms:(assertion warning): There is
gtkwave sim.vcd --rcvar 'fontname_waves Monospac 22' --rcvar 'fontname_signals Mon

GTKWave Analyzer v3.3.109 (w)1999-2020 BSI

RCVAR  | 'fontname_waves Monospac 22' FOUND
RCVAR  | 'fontname_signals Monospace 22' FOUND
[0] start time.
[120000000] end time.
```

Código VHDL

Adder8bit

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_arith.all;
use ieee.std_logic_signed.all;

entity adder8bit is
      port(
            sel:  in    std_logic;
            anum: in    signed(7 downto 0);
            bnum: in    signed(7 downto 0);
            sum:  out   signed(7 downto 0)
      );
end adder8bit;

architecture adder8bit_arch of adder8bit is
begin
      sum <= anum + bnum when sel = '0'
               else anum - bnum;
end adder8bit_arch;
```

Test

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_arith.all;
use ieee.std_logic_signed.all;

entity test is
end test;

architecture arch of test is
      component adder8bit is
            port(
                  sel:   in    std_logic;
                  anum:  in    signed(7 downto 0);
                  bnum:  in    signed(7 downto 0);
                  sum:   out   signed(7 downto 0)
            );
      end component;
      signal sel:   std_logic;
      signal anum:  signed(7 downto 0);
      signal bnum:  signed(7 downto 0);
      signal sum:   signed(7 downto 0);
begin
      adder: adder8bit port map( sel => sel, anum => anum, bnum => bnum, sum => sum );
      process begin
            sel <= '0';
            anum <= "00000011";
            bnum <= "00000100";
            wait for 20 ns;
            anum <= "01000000";
            bnum <= "01000010";
            wait for 20 ns;
            anum <= "10000001";
            bnum <= "10000000";
```

```
            wait for 20 ns;
            sel <= '1';
            anum <= "00000011";
            bnum <= "00000100";
            wait for 20 ns;
            anum <= "01000000";
            bnum <= "01000010";
            wait for 20 ns;
            anum <= "10000001";
            bnum <= "10000000";
            wait for 20 ns;
            wait;
      end process;
end arch;
```

Análisis de vectores

Practica 6

Vectores de entrada

A		B	
Binario	Hexadecimal	Binario	Hexadecimal
00000011	3	00000100	4
01000000	40	01000010	42
10000001	81	10000000	80

Valores de salida

A		B		Selector		
Binario	Hexa	Binario	Hexa	Binario	Binario	Hexa
00000011	3	00000100	4	0	00000111	7
01000000	40	01000010	42	0	00000000	0
10000001	81	10000000	80	0	00000000	0
00000011	3	00000100	4	1	11111111	FF
01000000	40	01000010	42	1	11111110	FE
10000001	81	10000000	80	1	00000001	1

Practica 7

Vectores de entrada

A		B	
Binario	Hexadecimal	Binario	Hexadecimal
00000011	3	00000100	4
01000000	40	01000010	42
10000001	81	10000000	80

Valores de salida

A		B		Selector		
Binario	Hexa	Binario	Hexa	Binario	Binario	Hexa
00000011	3	00000100	4	0	00000111	7
01000000	40	01000010	42	0	00000000	82
10000001	81	10000000	80	0	00000000	01
00000011	3	00000100	4	1	11111111	FF
01000000	40	01000010	42	1	11111110	FE
10000001	81	10000000	80	1	00000001	1

Practica 8

Vectores de entrada

A		B	
Binario	Hexadecimal	Binario	Hexadecimal
00000011	3	00000100	4
01000000	40	01000010	42
10000001	81	10000000	80

Valores de salida

A		B		Selector		
Binario	Hexa	Binario	Hexa	Binario	Binario	Hexa
00000011	3	00000100	4	0	00000111	7
01000000	40	01000010	42	0	00000000	82
10000001	81	10000000	80	0	00000000	01
00000011	3	00000100	4	1	11111111	FF
01000000	40	01000010	42	1	11111110	FE
10000001	81	10000000	80	1	00000001	01

Conclusión

La biblioteca de arith en VHDL permite realizar circuitos más fáciles y ahorra mucho tiempo más sin en-cambio arroja valores distintos al momento de realizar las operaciones de suma y resta haciendo que se aproveche el número a sumarse y restarte a diferencia del circuito anterior en el que teníamos desbordes. Pareciera que usa los positivos y negativos a su favor del número para representarlo. En la practica 8 no parece haber un cambio en los números .