EE 443
Digital and Computer Analysis and Design Laboratory

Lab *#6*: *Instruction and data memory; basic components and datapath for I- and J- format instructions*
8/8/13

Prepared by:
Matt Van Veldhuizen

Lab Participation
Provide the percent participation in lab by each lab partner (normally, 50%, 50% for 2 partners):

Lab member name	**Percent participation**
Matt Van Veldhuizen	100%

Does your solution work the way it's supposed to work? ☐YES ☐NO[1]

[1]If your answer is NO, please explain in your report.

Instructor/TA comments and grading:

1. Objective and Background

The main goal of this lab was to continue to construct the 16-bit microprocessor, specifically the instruction memory, and the data memory components. With these main components, others were needed to make the instruction and data memory work properly, these included a component that will increment a 16-bit number by 2, sign extend a 6-bit number to a 16-bit number, a 16-bit adder component, a 3-input, 16-bit multiplexer, and a component that shifts a 16-bit number to the left by one.

2. Equipment

Altera Quartus II

3. Procedure

The first two components are involved with the Instruction Fetch and PC Increment components of the microprocessor. The first of these components was designed to increments a 16-bit number by 2, this was called INCTWO which can be seen in Figure 1. The last component was designed to be the instruction memory, which has a 16-bit input and output. The instruction memory was to have hard coded instructions to be used with, these instructions where to load some data from the lowest memory into R1, loads data from the next address into R2, then add R1 and R2, and places the result into R3. Then it was to store the result in the next memory address, and lastly uses an infinite loop to halt the program. This was called INS_MEM which can be seen in Figure 2. The next two components were created to do the memory access portion of the microprocessor. The first component that was created was a sign extender, where it would take a 6-bit number and sign extend it to a 16-bit number, this was called SGNEXT6x16 which can be seen in Figure 3. The last component that was created was the data memory. The data memory was to take two 16-bit inputs, the address and the value, and two 1-bit inputs to enable read or write. This was called DAT_MEM and can be seen in Figure 4. The next two components were created to handle the branch instructions. The first component that was to be able to take a 16-bit number and shift that number left by one. This component was called SHLONE and can be seen in Figure 5. The last component needed for the branch instruction is a component that is a 16-bit adding component. This component was called ADD16, and can be seen in Figure 6. The last component was created to handle the jump instructions, and this was a 3-input, 16-bit multiplexer that was called MUX3x16 and can be seen in Figure 7.

4. Results

When designing each of the components, the timing delays for each component was recorded along with the chip resources. The max delay for INCTWO was 13.126ns between DIN[5] and DOUT[15] which can be seen in Table 1 and used less than 1% of the total logic elements, and 7% of the total pins, which can be seen in Figure 8. INS_MEM had a max delay of 10.399ns which can be seen in Table 2 and used 1% of the total logic and 7% of the total pins, as seen in Figure 9. SGNEXT6x16 had a max delay of 9.135ns between DIN[5] and DOUT[8] which can be seen in Table 3 and used about 5% of the total pins, as seen in Figure 10. DAT_MEN had a max delay of 9.988ns between RE and DOUT[11], which can be seen in Table 4 and used 0% of the total logic elements, 11% of the total pins and less than 1% of the total memory bits, as seen in Figure 11.SHLONE had a max delay of 8.918ns between DIN[13] and DOUT[14] which can be seen in Table 5 and used about 7% if the total pins, as seen in Figure 12. ADD16 had a max delay of 25.065ns between X[1] and Z[15], which can be seen in Table 6 and used 1% of the

total logic elements, 0 registers, and 10% of the total pins, as seen in Figure 13. MUX3x16 had a max delay of 11.890ns between S[0] and O[0] which can be seen in Table 7 and used less than 1% of the total logic elements, and 14% of the total pins, as seen in Figure 14. Along with this each component was tested and the simulation for INCTWO can be seen in Figure 15, INS_MEM can be seen in Figure 16, SGNEXT6x16 can be seen in Figure 17, DAT_MEM can be seen in Figure 18, SHLONE can be seen in Figure 19, ADD16 can be seen in Figure 20, and lastly MUX3x16 can be seen in Figure 21. In addition one of the simulations displays the third instruction in the program, and this can be seen in Figure 22. And in Table 8, shows the binary code of the program.

This lab worked as expected, the hardest part was making the instruction memory and the data memory components. I had a few design issues with the instruction memory component because I was not sure how to implement it so that the memory is byte addressable, and I am still not sure if it is. However what I do have works, but I do not know if it is the correct way of doing this.

5. Conclusion

This lab was to create the components necessary to execute I and J format instructions, and to incorporate the instruction and data memory components of the microprocessor.

6. Attachments

```
1   library ieee;
2   use ieee.std_logic_1164.all;
3   use work.lib.all;
4
5   entity INCTWO is
6      port (DIN : in std_logic_vector(15 downto 0);
7            DOUT : out std_logic_vector(15 downto 0));
8   end;
9
10  architecture Logic of INCTWO is
11  begin
12     ADD_COMP : ADD16 port map(DIN,"0000000000000010",DOUT);
13  end Logic;
```

Figure 1: INCTWO.vhdl

```
1  library ieee;
2  use ieee.std_logic_1164.all;
3  use ieee.numeric_std.all;
4
5  entity INS_MEM is
6     port (DIN : in std_logic_vector(15 downto 0);
7           DOUT : out std_logic_vector(15 downto 0)
8           );
9  end;
10
11 architecture Logic of INS_MEM is
12    type input is array (32 downto 0) of std_logic_vector(15 downto 0);
13    signal inputstruct : input;
14 begin
15    inputstruct(0) <= x"b200";
16    inputstruct(1) <= x"b401";
17    inputstruct(2) <= x"029a";
18    inputstruct(3) <= x"f602";
19    inputstruct(4) <= x"2001";
20
21    DOUT <= inputstruct(to_integer(unsigned(DIN)));
22 end Logic;
```

Figure 2: INS_MEM.vhdl

```
1  library ieee;
2  use ieee.std_logic_1164.all;
3  use work.lib.all;
4
5  entity SGNEXT6x16 is
6     port (DIN : in std_logic_vector(5 downto 0);
7           DOUT : out std_logic_vector(15 downto 0));
8  end;
9
10 architecture Logic of SGNEXT6x16 is
11
12 begin
13
14 DOUT<=DIN(5)&DIN(5)&DIN(5)&DIN(5)&DIN(5)&DIN(5)&DIN(5)&DIN(5)&DIN(5)&DIN(5)&DIN;
15 end Logic;
```

Figure 3: SGNEXT6x16.vhdl

```vhdl
library ieee;
use ieee.std_logic_1164.all;
use IEEE.Numeric_Std.all;
use work.lib.all;

entity DAT_MEM is
   port (DIN, ADD : in std_logic_vector(15 downto 0);
         WE, RE, CLK : in std_logic;
         DOUT : out std_logic_vector(15 downto 0)
         );
end;

architecture Logic of DAT_MEM is

   type memT is array (15 downto 0) of std_logic_vector(15 downto 0);
   signal mem : memT;
   signal read_address : std_logic_vector(15 downto 0);

begin
   process(CLK)
   begin
      if rising_edge(CLK) then
         if WE = '1' then
            mem(to_integer(unsigned(ADD))) <= DIN;
         end if;
         read_address <= ADD;
      end if;
   end process;

   process(RE)
   begin
      if RE = '1' then
         DOUT <= mem(to_integer(unsigned(read_address)));
      else
         DOUT <= "ZZZZZZZZZZZZZZZZ";
      end if;
   end process;
end Logic;
```

Figure 4: DAT_MEM.vhdl

```vhdl
library ieee;
use ieee.std_logic_1164.all;
use work.lib.all;

entity SHLONE is
   port (DIN : in std_logic_vector(15 downto 0);
         DOUT : out std_logic_vector(15 downto 0));
end;

architecture Logic of SHLONE is

begin

DOUT<=DIN(14 downto 0)&'0';

end Logic;
```

Figure 5: SHLONE.vhdl

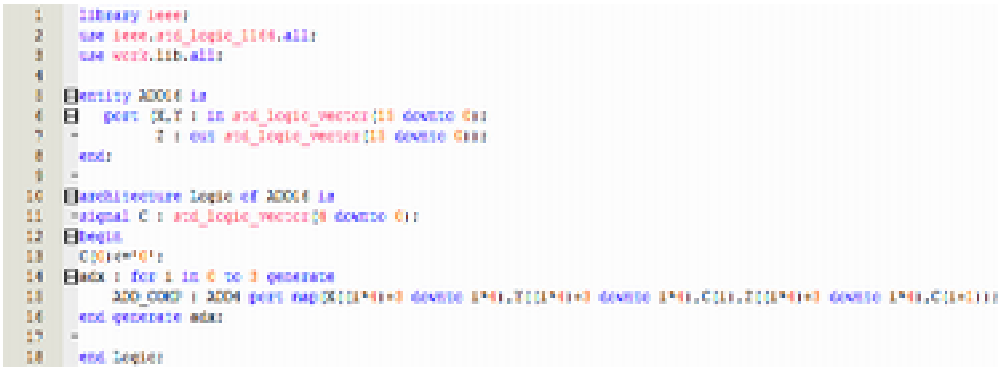

Figure 6: ADD16.vhdl

```
1   library ieee;
2   use ieee.std_logic_1164.all;
3   use work.lib.all;
4
5   entity MUX3x16 is
6   port(S : in std_logic_vector(1 downto 0);
7        X,Y,Z : in std_logic_vector(15 downto 0);
8        O : out std_logic_vector(15 downto 0));
9   end;
10
11  architecture Logic of MUX3x16 is
12  begin
13   with S select O <= X when "00",
14                      Y when "01",
15                      Z  when "10",
16                      "XXXXXXXXXXXXXXXX" when "11",
17                      "XXXXXXXXXXXXXXXX" when others;
18
19  end Logic;
```

Figure 7: MUX3x16.vhdl

Table 1: INCTWO Timing Delays

Progagation Delay						
	Input Port	Output Port	RR	RF	FR	FF
1	DIN[5]	DOUT[15]	13.126	13.126	13.126	13.126
2	DIN[5]	DOUT[14]	13.110	13.110	13.110	13.110
3	DIN[2]	DOUT[15]	12.823	12.823	12.823	12.823
4	DIN[2]	DOUT[14]	12.807	12.807	12.807	12.807
5	DIN[4]	DOUT[15]	12.764	12.764	12.764	12.764
6	DIN[4]	DOUT[14]	12.748	12.748	12.748	12.748
7	DIN[5]	DOUT[12]	12.708	12.708	12.708	12.708
8	DIN[5]	DOUT[13]	12.705	12.705	12.705	12.705
9	DIN[7]	DOUT[15]	12.700	12.700	12.700	12.700
10	DIN[7]	DOUT[14]	12.684	12.684	12.684	12.684
11	DIN[3]	DOUT[15]	12.653	12.653	12.653	12.653
12	DIN[3]	DOUT[14]	12.637	12.637	12.637	12.637
13	DIN[6]	DOUT[15]	12.598	12.598	12.598	12.598
14	DIN[6]	DOUT[14]	12.582	12.582	12.582	12.582
15	DIN[10]	DOUT[15]	12.548	12.548	12.548	12.548
16	DIN[10]	DOUT[14]	12.532	12.532	12.532	12.532
17	DIN[5]	DOUT[11]	12.438	12.438	12.438	12.438
18	DIN[2]	DOUT[12]	12.405	12.405	12.405	12.405
19	DIN[2]	DOUT[13]	12.402	12.402	12.402	12.402
20	DIN[4]	DOUT[12]	12.346	12.346	12.346	12.346
21	DIN[4]	DOUT[13]	12.343	12.343	12.343	12.343
22	DIN[7]	DOUT[12]	12.282	12.282	12.282	12.282
23	DIN[7]	DOUT[13]	12.279	12.279	12.279	12.279

Total logic elements	19 / 33,216 (< 1 %)
Total combinational functions	19 / 33,216 (< 1 %)
Dedicated logic registers	0 / 33,216 (0 %)
Total registers	0
Total pins	32 / 475 (7 %)
Total virtual pins	0
Total memory bits	0 / 483,840 (0 %)
Embedded Multiplier 9-bit elements	0 / 70 (0 %)
Total PLLs	0 / 4 (0 %)

Figure 8: INCTWO Chip Resources

Table 2: INS_MEM Timing Delays

	Input Port	Output Port	RR	RF	FR	FF
1	DIN[2]	DOUT[0]	10.399	10.399	10.399	10.399
2	DIN[1]	DOUT[3]	10.077			10.077
3	DIN[1]	DOUT[4]	10.077			10.077
4	DIN[1]	DOUT[7]	10.077			10.077
5	DIN[0]	DOUT[0]	9.953	9.953	9.953	9.953
6	DIN[1]	DOUT[14]	9.833			9.833
7	DIN[0]	DOUT[13]	9.645	9.645	9.645	9.645
8	DIN[1]	DOUT[9]	9.585			9.585
9	DIN[0]	DOUT[12]	9.558			9.558
10	DIN[0]	DOUT[15]	9.528			9.528
11	DIN[0]	DOUT[14]	9.381			9.381
12	DIN[1]	DOUT[1]	9.180			9.180
13	DIN[0]	DOUT[10]	9.130			9.130
14	DIN[0]	DOUT[3]		9.608	9.608	
15	DIN[0]	DOUT[4]		9.608	9.608	
16	DIN[0]	DOUT[7]		9.608	9.608	
17	DIN[0]	DOUT[9]		9.411	9.411	
18	DIN[1]	DOUT[0]		10.124	10.124	
19	DIN[1]	DOUT[12]		9.731	9.731	
20	DIN[1]	DOUT[13]		9.818	9.818	
21	DIN[1]	DOUT[15]		9.701	9.701	
22	DIN[2]	DOUT[1]		9.411	9.411	

Total logic elements	19 / 33,216 (< 1 %)
Total combinational functions	19 / 33,216 (< 1 %)
Dedicated logic registers	0 / 33,216 (0 %)
Total registers	0
Total pins	32 / 475 (7 %)
Total virtual pins	0
Total memory bits	0 / 483,840 (0 %)
Embedded Multiplier 9-bit elements	0 / 70 (0 %)
Total PLLs	0 / 4 (0 %)

Figure 9: INS_MEM Chip Resources

Table 3: SGNEXT6x16 Timing Delays

	Input Port	Output Port	RR	RF	FR	FF
1	DIN[5]	DOUT[8]	9.135			9.135
2	DIN[5]	DOUT[7]	8.925			8.925
3	DIN[5]	DOUT[6]	8.901			8.901
4	DIN[5]	DOUT[10]	8.901			8.901
5	DIN[3]	DOUT[3]	8.898			8.898
6	DIN[4]	DOUT[4]	8.897			8.897
7	DIN[5]	DOUT[12]	8.877			8.877
8	DIN[5]	DOUT[14]	8.877			8.877
9	DIN[5]	DOUT[11]	8.857			8.857
10	DIN[5]	DOUT[5]	8.847			8.847
11	DIN[5]	DOUT[15]	8.847			8.847
12	DIN[5]	DOUT[9]	8.845			8.845
13	DIN[5]	DOUT[13]	8.835			8.835
14	DIN[2]	DOUT[2]	8.779			8.779
15	DIN[0]	DOUT[0]	4.848			4.848
16	DIN[1]	DOUT[1]	4.802			4.802

Total logic elements	0 / 33,216 (0 %)
Total combinational functions	0 / 33,216 (0 %)
Dedicated logic registers	0 / 33,216 (0 %)
Total registers	0
Total pins	22 / 475 (5 %)
Total virtual pins	0
Total memory bits	0 / 483,840 (0 %)
Embedded Multiplier 9-bit elements	0 / 70 (0 %)
Total PLLs	0 / 4 (0 %)

Figure 10: SGNEXT6x16 Chip Resources

Table 4: DAT_MEM Timing Delays

	Input Port	Output Port	RR	RF	FR	FF
1	RE	DOUT[11]	9.988	9.988	9.988	9.988
2	RE	DOUT[1]	9.978	9.978	9.978	9.978
3	RE	DOUT[3]	9.814	9.814	9.814	9.814
4	RE	DOUT[13]	9.814	9.814	9.814	9.814
5	RE	DOUT[6]	9.754	9.754	9.754	9.754
6	RE	DOUT[7]	9.672	9.672	9.672	9.672
7	RE	DOUT[14]	9.493	9.493	9.493	9.493
8	RE	DOUT[5]	9.245	9.245	9.245	9.245
9	RE	DOUT[12]	9.245	9.245	9.245	9.245
10	RE	DOUT[9]	9.204	9.204	9.204	9.204
11	RE	DOUT[4]	9.176	9.176	9.176	9.176
12	RE	DOUT[8]	9.176	9.176	9.176	9.176
13	RE	DOUT[0]	9.165	9.165	9.165	9.165
14	RE	DOUT[2]	9.097	9.097	9.097	9.097
15	RE	DOUT[15]	8.952	8.952	8.952	8.952
16	RE	DOUT[10]	8.851	8.851	8.851	8.851

Total logic elements	0 / 33,216 (0 %)
Total combinational functions	0 / 33,216 (0 %)
Dedicated logic registers	0 / 33,216 (0 %)
Total registers	0
Total pins	51 / 475 (11 %)
Total virtual pins	0
Total memory bits	256 / 483,840 (< 1 %)
Embedded Multiplier 9-bit elements	0 / 70 (0 %)
Total PLLs	0 / 4 (0 %)

Figure 11: DAT_MEM Chip Resources

Table 5: SHLONE Timing Delays

	Input Port	Output Port	RR	RF	FR	FF
1	DIN[13]	DOUT[14]	8.918			8.918
2	DIN[2]	DOUT[3]	8.903			8.903
3	DIN[10]	DOUT[11]	8.868			8.868
4	DIN[14]	DOUT[15]	8.811			8.811
5	DIN[6]	DOUT[7]	8.801			8.801
6	DIN[1]	DOUT[2]	8.794			8.794
7	DIN[9]	DOUT[10]	8.794			8.794
8	DIN[5]	DOUT[6]	8.786			8.786
9	DIN[8]	DOUT[9]	8.551			8.551
10	DIN[11]	DOUT[12]	8.511			8.511
11	DIN[7]	DOUT[8]	8.159			8.159
12	DIN[3]	DOUT[4]	8.006			8.006
13	DIN[12]	DOUT[13]	7.976			7.976
14	DIN[4]	DOUT[5]	7.944			7.944
15	DIN[0]	DOUT[1]	4.802			4.802

Total logic elements	0 / 33,216 (0 %)
Total combinational functions	0 / 33,216 (0 %)
Dedicated logic registers	0 / 33,216 (0 %)
Total registers	0
Total pins	32 / 475 (7 %)
Total virtual pins	0
Total memory bits	0 / 483,840 (0 %)
Embedded Multiplier 9-bit elements	0 / 70 (0 %)
Total PLLs	0 / 4 (0 %)

Figure 12: SHLONE Chip Resources

Table 6: ADD16 Timing Delays

	Input Port	Output Port	RR	RF	FR	FF
1	X[1]	Z[15]	25.065	25.065	25.065	25.065
2	X[1]	Z[14]	24.660	24.660	24.660	24.660
3	Y[1]	Z[15]	24.516	24.516	24.516	24.516
4	Y[1]	Z[14]	24.111	24.111	24.111	24.111
5	Y[2]	Z[15]	24.071	24.071	24.071	24.071
6	X[2]	Z[15]	23.869	23.869	23.869	23.869
7	Y[3]	Z[15]	23.858	23.858	23.858	23.858
8	X[1]	Z[13]	23.761	23.761	23.761	23.761
9	Y[2]	Z[14]	23.666	23.666	23.666	23.666
10	X[2]	Z[14]	23.464	23.464	23.464	23.464
11	Y[3]	Z[14]	23.453	23.453	23.453	23.453
12	X[1]	Z[12]	23.339	23.339	23.339	23.339
13	Y[1]	Z[13]	23.212	23.212	23.212	23.212
14	Y[4]	Z[15]	23.025	23.025	23.025	23.025
15	X[4]	Z[15]	22.978	22.978	22.978	22.978
16	X[3]	Z[15]	22.913	22.913	22.913	22.913
17	X[0]	Z[15]	22.829	22.829	22.829	22.829
18	Y[1]	Z[12]	22.790	22.790	22.790	22.790
19	Y[2]	Z[13]	22.767	22.767	22.767	22.767
20	X[1]	Z[10]	22.660	22.660	22.660	22.660
21	Y[4]	Z[14]	22.620	22.620	22.620	22.620
22	X[4]	Z[14]	22.573	22.573	22.573	22.573
23	X[2]	Z[13]	22.565	22.565	22.565	22.565
24	Y[3]	Z[13]	22.554	22.554	22.554	22.554
25	Y[0]	Z[15]	22.548	22.548	22.548	22.548
26	X[3]	Z[14]	22.508	22.508	22.508	22.508

Total logic elements	37 / 33,216 (< 1 %)
Total combinational functions	37 / 33,216 (< 1 %)
Dedicated logic registers	0 / 33,216 (0 %)
Total registers	0
Total pins	48 / 475 (10 %)
Total virtual pins	0
Total memory bits	0 / 483,840 (0 %)
Embedded Multiplier 9-bit elements	0 / 70 (0 %)
Total PLLs	0 / 4 (0 %)

Figure 13: ADD16 Chip Resources

Table 7: MUX3x16 Timing Delays

	Input Port	Output Port	RR	RF	FR	FF
1	S[0]	O[9]	11.890	11.890	11.890	11.890
2	X[8]	O[8]	11.681			11.681
3	X[3]	O[3]	11.671			11.671
4	S[0]	O[4]	11.656	11.656	11.656	11.656
5	S[0]	O[5]	11.577	11.577	11.577	11.577
6	S[0]	O[10]	11.491	11.491	11.491	11.491
7	S[0]	O[3]	11.375	11.375	11.375	11.375
8	S[0]	O[2]	11.353	11.353	11.353	11.353
9	S[0]	O[7]	11.343	11.343	11.343	11.343
10	Y[9]	O[9]	11.315			11.315
11	X[9]	O[9]	11.257			11.257
12	S[0]	O[8]	11.209	11.209	11.209	11.209
13	X[5]	O[5]	11.045			11.045
14	Y[5]	O[5]	10.964			10.964
15	S[0]	O[13]	10.942	10.942	10.942	10.942
16	S[0]	O[15]	10.905	10.905	10.905	10.905
17	S[1]	O[9]	10.893	11.599	11.599	10.893
18	X[7]	O[7]	10.846			10.846
19	Y[14]	O[14]	10.768			10.768
20	S[0]	O[14]	10.760	10.760	10.760	10.760
21	Y[2]	O[2]	10.739			10.739
22	S[0]	O[12]	10.736	10.736	10.736	10.736
23	Y[13]	O[13]	10.726			10.726
24	Y[10]	O[10]	10.713			10.713
25	X[13]	O[13]	10.680			10.680
26	S[0]	O[11]	10.664	10.664	10.664	10.664

Total logic elements	32 / 33,216 (< 1 %)
Total combinational functions	32 / 33,216 (< 1 %)
Dedicated logic registers	0 / 33,216 (0 %)
Total registers	0
Total pins	66 / 475 (14 %)
Total virtual pins	0
Total memory bits	0 / 483,840 (0 %)
Embedded Multiplier 9-bit elements	0 / 70 (0 %)
Total PLLs	0 / 4 (0 %)

Figure 14: MUX3x16 Chip Resources

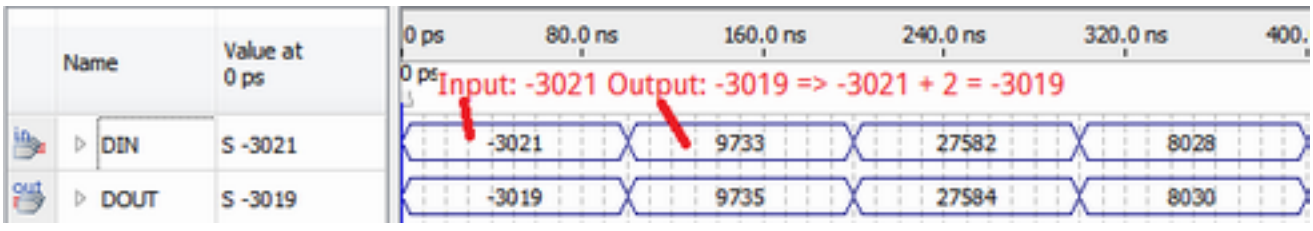

Figure 15: INCTWO Simulation

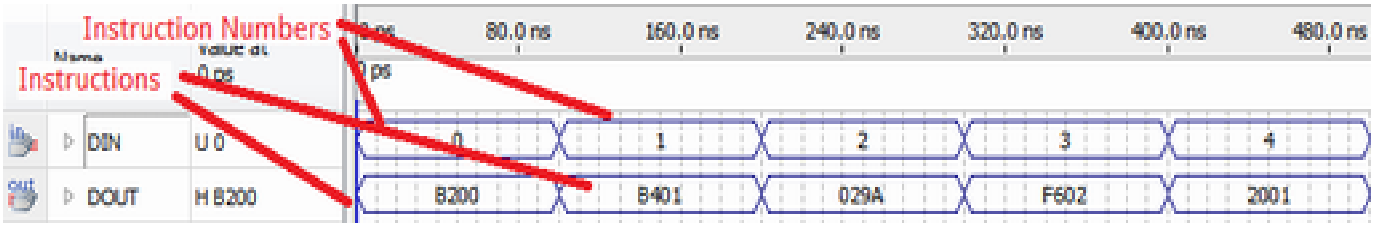

Figure 16: INS_MEM Simulation

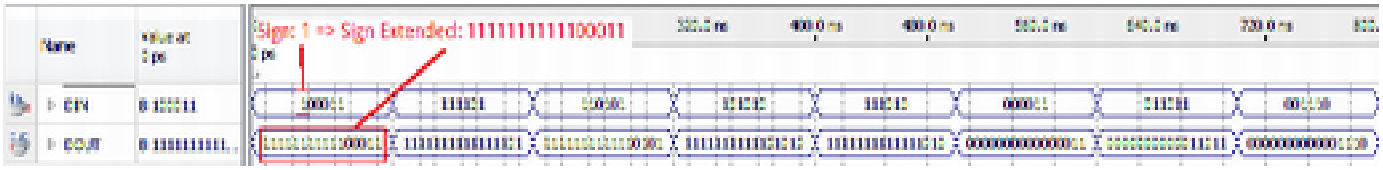

Figure 17: SGNEXT6x16 Simulation

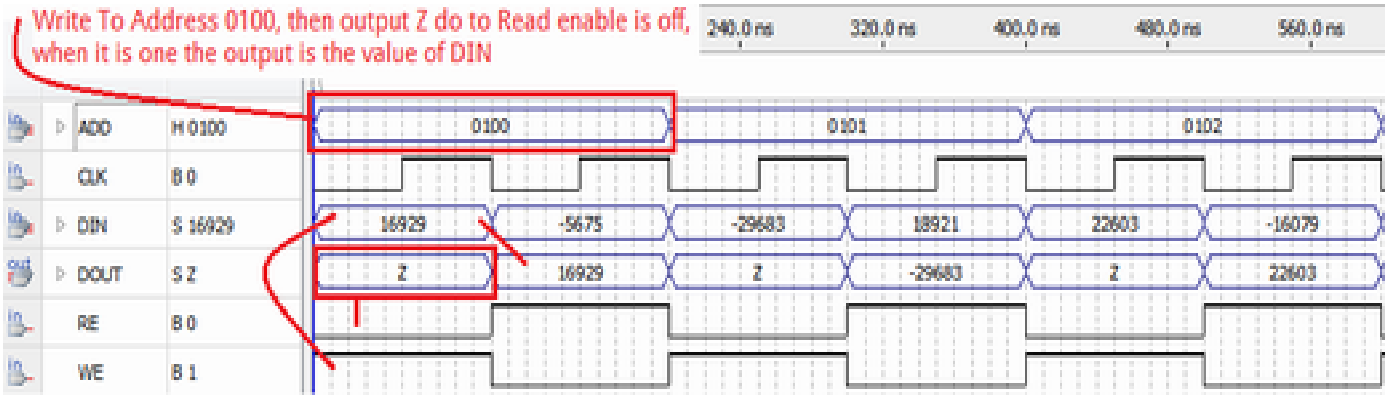

Figure 18: DAT_MEM Simulation

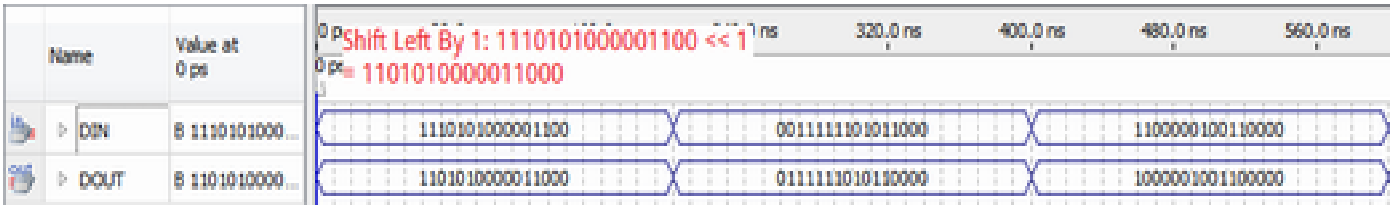

Figure 19: SHLONE Simulation

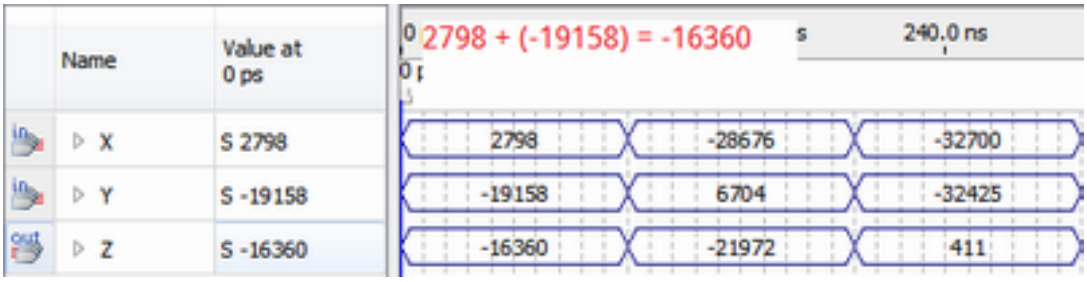

Figure 20: ADD16 Simulation

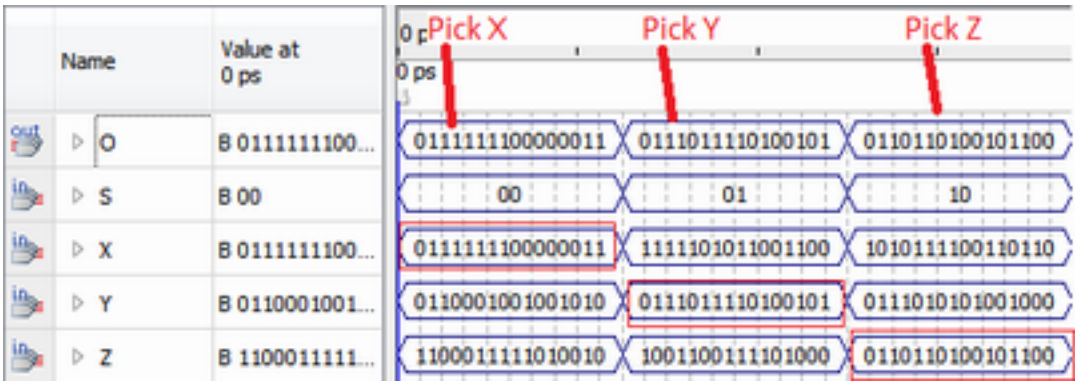

Figure 21: MUX3x16 Simulation

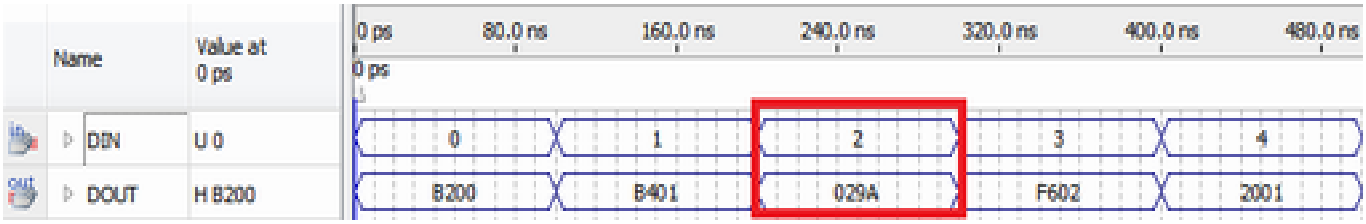

Figure 22: Simulation of the third instruction of the program

Table 8: Binary Code for the program

Hex	**Binary**
0xB200	1011001000000000
0xB401	1011010000000001
0x029A	0000001010011010
0xF602	1111011000000010
0x2001	0010000000000001