



Σχεδίαση Ψηφιακών Συστημάτων

ΕΞΑΜΗΝΙΑΙΑ ΑΣΚΗΣΗ ΘΕΩΡΙΑΣ

ΜΕΡΟΣ 03

Τμήμα Μηχανικών Πληροφορικής
& Υπολογιστών

(Καθηγητής: Ιωάννης Βογιατζής)

Μάρκος Παντελιδάκης - 18390228 - ice18390228@uniwa.gr

1. Control Unit

```

1  LIBRARY ieee;
2  USE ieee.std_logic_1164.all;
3
4  ENTITY Control IS PORT (
5      OP_5to0: IN STD_LOGIC_VECTOR(5 DOWNTO 0);
6      RegDst, Branch: OUT STD_LOGIC;
7      MemRead, MemtoReg: OUT STD_LOGIC;
8      MemWrite, ALUSrc, RegWrite: OUT STD_LOGIC;
9      ALU_op: OUT STD_LOGIC_VECTOR(1 DOWNTO 0));
10 END Control;
11
12 architecture behavioural of Control is
13 begin
14     with OP_5to0 select
15     RegDst <=
16         '1' when "000000",
17         '0' when others;
18
19     with OP_5to0 select
20     Branch <=
21         '1' when "000100",
22         '0' when others;
23
24     with OP_5to0 select
25     MemRead <=
26         '1' when "100011",
27         '0' when others;
28
29     with OP_5to0 select
30     MemtoReg <=
31         '1' when "100011",
32         '0' when others;
33
34     with OP_5to0 select
35     ALU_op <=
36         "10" when "000000",
37         "01" when "000100",
38         "00" when others;
39
40     with OP_5to0 select
41     MemWrite <=
42         '1' when "101011",
43         '0' when others;
44
45     with OP_5to0 select
46     ALUSrc <=
47         '1' when "100011",
48         '1' when "101011",
49         '0' when others;
50
51     with OP_5to0 select
52     RegWrite <=
53         '1' when "000000",
54         '1' when "100011",
55         '0' when others;
56 end behavioural;

```

Εικ3.1.1: Κώδικας αρχείου VHDL κυκλώματος μονάδας ελέγχου MIPS.

```

1  LIBRARY ieee;
2  USE ieee.std_logic_1164.all;
3
4  entity Control_tb is
5  end Control_tb;
6
7  architecture test_b of Control_tb is
8  component Control port(
9      OP_5to0: IN STD_LOGIC_VECTOR(5 DOWNTO 0);
10     RegDst, RegWrite, ALUSrc, Branch : OUT STD_LOGIC;
11     MemRead, MemWrite, MemtoReg : OUT STD_LOGIC;
12     ALU_op: OUT STD_LOGIC_VECTOR(1 DOWNTO 0));
13  end component;
14
15  signal OP_5to0_1 : STD_LOGIC_VECTOR(5 DOWNTO 0);
16  signal RegDst1, RegWrite1, ALUSrc1, Branch1 : STD_LOGIC;
17  signal MemRead1, MemWrite1, MemtoReg1 : STD_LOGIC;
18  signal ALU_op1 : STD_LOGIC_VECTOR(1 DOWNTO 0);
19
20  BEGIN
21  M1 : Control port map(OP_5to0 => OP_5to0_1, RegDst => RegDst1, RegWrite => RegWrite1,
22  ALUSrc => ALUSrc1, Branch => Branch1, MemRead => MemRead1,
23  MemWrite => MemWrite1, MemtoReg => MemtoReg1, ALU_op => ALU_op1);
24  process
25  begin
26      OP_5to0_1 <= "000000"; wait for 50 ps;
27      OP_5to0_1 <= "100011"; wait for 50 ps;
28      OP_5to0_1 <= "101011"; wait for 50 ps;
29      OP_5to0_1 <= "000100"; wait for 50 ps;
30  end process;
31  end test_b;

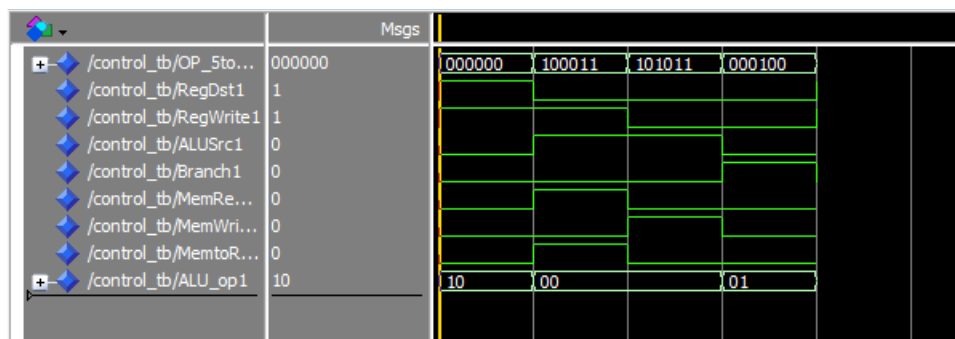
```

Εικ3.1.2: Κώδικας αρχείου VHDL testbench κυκλώματος μονάδας ελέγχου MIPS.

Πίν3.1.1: Πίνακας ζητούμενων τιμών για τη μονάδα ελέγχου.

OP_5to0	RegDst	RegWrite	ALUSrc	Branch	MemRead	MemWrite	MemtoReg	ALU_op
000000	1	1	0	0	0	0	0	10
100011	0	1	1	0	1	0	1	00
101011	0	0	1	0	0	1	0	00
000100	0	0	0	1	0	0	0	01

Οι τιμές του πίνακα 3.1 επαληθεύονται από τη κυματομορφή της Εικόνας 2.3.



Εικ3.1.3: Παράδειγμα κυματομορφής κυκλώματος μονάδας ελέγχου MIPS.

2. Sign Extension Unit

```

1  LIBRARY ieee;
2  USE ieee.std_logic_1164.all;
3
4  ENTITY SIGN_Extension is port (
5    Instr_15to0 : in  std_logic_vector (15 downto 0);
6    Sign_extended: out std_logic_vector (31 downto 0));
7  END SIGN_Extension;
8
9  architecture behavioural of SIGN_extension is
10 begin
11   Sign_extended <=
12     x"0000" & Instr_15to0 when Instr_15to0(15)='0' else
13     x"FFFF" & Instr_15to0 when Instr_15to0(15)='1';
14 end behavioural;

```

Εικ3.2.1 Κώδικας αρχείου VHDL κυκλώματος μονάδας επέκτασης προσήμου.

Στη συνέχεια βάζουμε τις ζητούμενες τιμές στο testbench αρχείο όπως βλέπουμε και στην εικόνα 3.2.2.

```

1  LIBRARY ieee;
2  USE ieee.std_logic_1164.all;
3
4  entity SIGN_Extension_tb is
5  end SIGN_Extension_tb;
6
7  architecture test_b of SIGN_Extension_tb is
8  component SIGN_Extension is port(
9    Instr_15to0 : in std_logic_vector (15 downto 0);
10   Sign_extended: out std_logic_vector (31 downto 0));
11  end component;
12
13  signal Instr_15to0_1 : std_logic_vector (15 downto 0);
14  signal Sign_extended1 : std_logic_vector (31 downto 0);
15  BEGIN
16  M1 : SIGN_Extension port map(Instr_15to0 => Instr_15to0_1, Sign_extended => Sign_extended1);
17  process
18  begin
19    Instr_15to0_1 <= x"0010"; wait for 50 ps;
20    Instr_15to0_1 <= x"1001"; wait for 50 ps;
21    Instr_15to0_1 <= x"80A0"; wait for 50 ps;
22  end process;
23  end test_b;

```

Εικ3.2.2: Κώδικας αρχείου VHDL testbench κυκλώματος μονάδας επέκτασης προσήμου.

	Msgs	
/sign_extension_tb/...	00000000000010000	00000000000010000 00010000000000001 1000000010100000
/sign_extension_tb/...	0000000000000000...	000000000000000000000000000010000 00000000000000000000000000001 11111111111111111111000000010100000

Εικ3.2.3: Παράδειγμα κυματομορφής κυκλώματος μονάδας επέκτασης προσήμου.

3. Κύκλωμα (αριστερής) ολίσθησης κατά 2

```

1  LIBRARY ieee;
2  USE ieee.std_logic_1164.all;
3  USE IEEE.STD_LOGIC_UNSIGNED.ALL;
4
5  entity shiftleft2 is port (
6      In1: in  std_logic_vector(31 downto 0);
7      d:  out std_logic_vector(31 downto 0));
8  end shiftleft2;
9
10 architecture behavioural of shiftleft2 is
11
12 BEGIN
13     d <= In1(29 downto 0) & "00";
14 end behavioural;

```




Εικ3.3.1 Κώδικας αρχείου VHDL κυκλώματος αριστερής ολίσθησης κατά 2.

```

1  LIBRARY ieee;
2  USE ieee.std_logic_1164.all;
3
4  entity shiftleft2_tb is
5  end shiftleft2_tb;
6
7  architecture test_b of shiftleft2_tb is
8  component shiftleft2 is port(
9      In1 : in  std_logic_vector(31 downto 0);
10     d : out std_logic_vector(31 downto 0));
11  end component;
12
13  signal In1l : std_logic_vector(31 downto 0);
14  signal d1 : std_logic_vector(31 downto 0);
15  BEGIN
16  M1 : shiftleft2 port map(In1 => In1l, d => d1);
17  process
18  begin
19      In1l <= x"0000AAAF"; wait for 50 ns;
20      In1l <= x"FFFFAAFF"; wait for 50 ns;
21  end process;
22  end test_b;
23

```

Εικ3.3.2: Κώδικας αρχείου VHDL testbench κυκλώματος αριστερής ολίσθησης κατά 2.

	Msgs	
		
 /shiftleft2_tb/In11	0000000000000000...	00000000000000001010101010101111
 /shiftleft2_tb/d1	0000000000000000...	0000000000000000101010101010111100

Εικ3.3.3: Παράδειγμα κυματομορφής κυκλώματος αριστερής ολίσθησης κατά 2.