Programming the Microarchitecture Design

Ghassan Arnouk

SYSC 3006A
Summer 2020
Lab 4 Report
Group 1

Instructor: Michel Sayde
TA: Khalid Almahrog
Submitted: 2020/06/05

1 First Part

1.1 Control FSM Output Table

1.1.1 Complete the provided Control FSM Output Table for Part 1 for the Fetch, Decode, and Execution States for opcodes 0x01 (ADD) through 0x07 (NOT).

Table 1: Fetch, Decode, and Execution States for opcodes 0x01 (ADD) through 0x07 (NOT)

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
State Hex encoding	Unused (0)	IRCE	PCOE	C1OE	AADD	MARCE	MAROE	MDRCE	MDROE	MDRget	MDRput	IBRead	IBWrite	AOP	ANOP	DR	SXR	SYR	RegSEL	RegLD	T1CE	T1OE	T2CE	T2OE	Q7+	Q6+	Q5+	Q4+	Q3+	Q2+	Q1+	Q0+	Hex Encoding
F0 0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	1	2402 3801
F1 1	0	0	0	1	1	0	1	1	0	0	0	1	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	1	0	1B10 0602
F2 2	0	1	0	0	0	0	0	0	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	40C2 0003
Decode 3	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	1	0	0	0	0	0	1	1	1	2002 2107
E0 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	1	1	0	0	0	0	0	0	0	0	1	0	1	0002 B805
E1 5	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1	1	0	1	1	0	0	0	0	0	0	1	1	0	0004 7606
E2 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0003 2100
Dead 7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0000 0007

2 Second Part

2.1 Control FSM Output Table

2.1.1 Complete the provided Control FSM Output Table for Part 2 for NOP Instruction Execution 3 States. This table will extend the Control FSM Output Table for Part 1 (same FSM Output ROM).

Table 2: NOP Instruction Execution States

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
State Hex encoding	Unused (0)	IRCE	PCOE	C1OE	AADD	MARCE	MAROE	MDRCE	MDROE	MDRget	MDRput	IBRead	IBWrite	AOP	ANOP	DR	SXR	SYR	RegSEL	RegLD	T1CE	T1OE	T2CE	T2OE	Q7+	Q6+	Q5+	Q4+	Q3+	Q2+	Q1+	Q0+	Hex Encoding
F0 0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	1	2402 3801
F1 1	0	0	0	1	1	0	1	1	0	0	0	1	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	1	0	1B10 0602
F2 2	0	1	0	0	0	0	0	0	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	40C2 0003
Decode 3	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	1	0	0	0	0	0	1	1	1	2002 2107
E3 8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0002 0009
E4 9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0002 000A
E5 10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0002 0000
Dead 7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0000 0007

3 Third Part

3.1 Control FSM Output Table

3.1.1 Complete the provided Control FSM Output Table for Part 3 for NEG Instruction Execution States. This table will extend the Control FSM Output Table for Part 1 and 2 (same FSM Output ROM).

Table 3: NEG Instruction Execution States

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
State Hex encoding	Unused (0)	IRCE	PCOE	C1OE	AADD	MARCE	MAROE	MDRCE	MDROE	MDRget	MDRput	IBRead	IBWrite	AOP	ANOP	DR	SXR	SYR	RegSEL	RegLD	T1CE	T1OE	T2CE	T2OE	Q7+	Q6+	Q5+	Q4+	Q3+	Q2+	Q1+	Q0+	Hex Encoding
F0 0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	1	2402 3801
F1 1	0	0	0	1	1	0	1	1	0	0	0	1	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	1	0	1B10 0602
F2 2	0	1	0	0	0	0	0	0	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	40C2 0003
Decode 3	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	1	0	0	0	0	0	1	1	1	2002 2107
E6 11	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	1	1	0	0	1	0	0	0	0	0	1	1	0	0	0005 320C
E7 12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	1	0	0	0	0	1	1	0	1	0002 090D
E8 13	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	1	1	1	0	1800 060E
E9 14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0003 2100
Dead 7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0000 0007

3.2 Describe how NEG instruction is executed at each execution state.

NEG can be performed by initially doing a NOT operation in the ALU and then adding a 1 to it. The 1 is the constant received from C1OE. The add operation is similar to incrementing the PC in previous labs. The Opcode for the NEG operation is 0x17 while it is 0x00 for the NOP operation.

3.3 Decode ROM Table

3.3.1 Complete the provided FSM Decode ROM Table to show any entries that must be programmed (for all parts).

Table 4: FSM Decode ROM Table

Instruction	Address (Hex)	Contents (Hex)
NOP	00	08
ADD	01	04
SUB	02	04
MOV	03	05
AND	04	04
OR	05	04
XOR	06	04
NOT	07	05
NEG	17	0B

4 Fourth Part

4.1 Instruction Table

4.1.1 Complete the provided Main Memory Table to contain the encodings of the Test Program instructions as indicated. Then program the words of this table into the Main Memory. Be sure to include the Main Memory contents exactly as given in the table.

Table 5: Main Memory Table

Address (Hex)	Instruction (Hex)	Encoding (Hex)
0	R2 ← [R15]	0320 F000
1	R11 ← NOT [R2]	07B0 2000
2	R10 ← [R15]	03A0 F000
3	R15 ← [R10] - R[11]	02FA B000
4	EEBB FFFF	illegal instruction
5	NOP	0000 0000
6	R11 ← -[R11]	17B0 0000

4.2 Test Results

4.2.1 Cycle the System Clock through the execution of your Test program and show your logs here.

Table 6: Simulation Log of the Circuit

Instruction Register	RAM(1060,270)[15]	RAM(1060,270)[11]	RAM(1060,270)[2]	RAM(1060,270)[10]
00000000	00000000	00000000	00000000	00000000
0320f000	00000000	00000000	00000000	00000000
0320f000	00000001	00000000	00000000	00000000
0320f000	00000001	00000000	00000001	00000000
07b02000	00000001	00000000	00000001	00000000
07b02000	00000002	00000000	00000001	00000000
07b02000	00000002	fffffffe	00000001	00000000
03a0f000	00000002	fffffffe	00000001	00000000
03a0f000	00000003	fffffffe	00000001	00000000
03a0f000	00000003	fffffffe	00000001	00000003
02fab000	00000003	fffffffe	00000001	00000003
02fab000	00000004	fffffffe	00000001	00000003
02fab000	00000005	fffffffe	00000001	00000003
00000000	00000005	fffffffe	00000001	00000003
00000000	00000006	fffffffe	00000001	00000003
17b00000	00000006	fffffffe	00000001	00000003
17b00000	00000007	fffffffe	00000001	00000003
17b00000	00000007	00000002	00000001	00000003