Project 1: ICOM 4215 – Computer Architecture and Organization
Spring 2011

The following file is provided for testing the first project:

Location of Instruction (hex)	Instruction	Opcode	Operand	IR (bin)	Effect
00 -01	LDI 25	01 110	000 0001 1001	0111000000011001	The accumulator gets a 00011001. Zero, Carry, Overflow and Negative flags are set to 0 (false). Note: the number in accumulator is 25.
02-03	STA r1	01 011	001 0000 0000	0101100100000000	Register 1 gest a 00011001. Note: the number on register 1 is a 25.
04-05	LDI -12	01 110	000 11110100	0111000011110100	The accumulator gets a 11110100. Negative flag changes to a 1. Note: the number in the accumulator is a -12 following 2's complement notation.
06-07	STA r2	01 011	010 0000 0000	0101101000000000	Register 2 gest a 11110100. Note: The number in register 2 is -12, following 2's complement notation.
08-09	LDI 2	01 110	000 0000 0010	0111000000000010	The accumulator gets a 00000010. Negative flag changes to 0. Note: The number in the accumulator is a 2.
0A-0B	STA r3	01 011	011 0000 0000	0101101100000000	Register 3 gets a 00000010. Note: The number in register 3 is a 2.
0C-0D	LDI 42	01 110	000 0010 1010	0111000000101010	The accumulator gets a 0010 1010. Note: The number in the accumulator is a 2A in hex (42 in decimal)
0E-0F	STA r7	01 011	111 0000 0000	0101111100000000	Register 7 gest a 2A in hex (42 in decimal). This will be used to jump to address 2A.
10-11	STA [80]	01 101	000 1000 0000	0110100010000000	Memory location 128 (80 in hex) gets a 00000010. Note: address 128 gets a 2.
12-13	ADDC r1	00 011	001 0000 0000	0001100100000000	ALU adds 00000010 and 00011001 getting a 00011011. This is loaded into the Accumulator. All flags set to 0. Note: The ALU adds 2 plus 25 and results 27. This result is saved in the accumulator.
14-15	STA [81]	01 101	000 1000 0001	0110100010000001	Store 00011011 into memory location 129 (81 in hex). Note: the contents of the alu, that is, a 27, is saved in address 129.

16-17	ADDC r2	00 011	010 0000 0000	0001101000000000	ALU adds 00011011 and 11110100 getting a 00001111. Carry flag set to 1. Overflow, Neg, Zero set to 0. Note: Add 27 plus -12 using 2's complement notation. Results in 15 with no overflow, but carry.
18-19	STA [82]	01 101	000 1000 0010	0110100010000010	Store 00001111 in memory location 130 (82 in hex). Note: 15 is saved in address 130.
1A-1B	MUL r3	00 100	011 0000 0000	0010001100000000	Multiply 1111 to 0010. Obtain a 0011110 in the accumulator. Carry, Overflow, Neg and Zero flasg are 0.
1C-1D	DIV r3	00 101	011 0000 0000	0010101100000000	Divide 1110 by 0010. Result in accumulator is 00000111. All flags zero.
1E-1F	LDA [FA]	01 100	000 1111 1010	0110000011111010	Load Accumulator with contents of keyboard (address 250 which is FA in hex). The contents is unknown until user presses the key. The accumulator will contain the ascii code of the letter or character pressed.
20-21	STA r1	01 011	001 0000 0000	0101100100000000	Register 1 gest a ???. Note: the number on register 1 is unknown.
22-23	NEG	00 110	000 0000 0000	0011000000000000	Twos complement of the contents of the accumulator. Carry and overflow flags are 0 but we do not know contents of Zero and Neg flags since they depend on what ascii character was selected.
24-25	ADDC r1	00 011	001 0000 0000	0001100100000000	Add a number and its complement. Accumulator will have a zero. Zero flag and carry flags are set. All other flags are zero.
26-27	BRZ	10 000	000 0000 0000	1000000000000000	Jump if zero to instruction located in address 42 (2A in hex).
28-29	STA r2	01 011	010 0000 0000	0101101000000000	This instruction should not be executed. If executed, register 2 gets a 0.
2A-2B	LDI 78	01 110	000 01001110	0111000001001110	Load Accumulator with 01001110. All flags to zero.
2C-2D	STA [FC]	01 101	000 1111 1100	0110100011111100	Display a N in first display location, as well as in location 252 in memory. Note; the ascii code for a capital N is 78.
2E-2F	STOP	11 111	000 0000 0000	1111100000000000	Cease operation. No changes to anything.

IR (bin)	Hex
0111000000011001	7019
0101100100000000	5900
0111000011110100	70F4
0101101000000000	5A00
0111000000000010	7002
0101101100000000	5B00
0111000000101010	702A
0101111100000000	5F00
0110100010000000	6880
0001100100000000	1900
0110100010000001	6881
0001101000000000	1A00
0110100010000010	6882
0010001100000000	2300
0010101100000000	2B00
0110000011111010	60FA
0101100100000000	5900
0011000000000000	3000
0001100100000000	1900
1000000000000000	8000
0101101000000000	5A00
0111000001001110	704E
0110100011111100	68FC
1111100000000000	F800

Code

7019
5900
70F4
5A00
7002
5B00
702A
5F00
6880
1900
6881
1A00
6882
2300
2B00
60FA
5900
3000
1900
8000
5A00
704E
68FC
F800