**Introduction**

In the first lab we were introduced to the microcontroller development board and its associated software. We have to connect the hardware, input data, run the program and collect the output.

**Preliminary Work**

Before programming I had to hook up the board. I connected the hardware to my dad’s older computer which had a serial port available because I did not have a USB-Serial adapter. Then I installed AxIDE from the CD and set the jumper to the correct position. Now I was ready.

**Lab Results**

For the lab I inserted the given content one by one using the MM command. After the content was entered, I double-checked using the MD command and made corrections as needed. Then I tried to run the program with the G 0100 command but the application just hung. I waited for quite some time, but nothing happened so I reset the board and continued. Thankfully the data wasn’t lost. I proceeded to trace the program one instruction at a time and copied the results to the tables below.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Table 1. Program Listing**   |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Address**  **(Hex)** | **Content** | **Opcode & Value** | **Effective Address or data Source** | **Addressing Mode** | | 0010 | 80 | --------- | --------- | --------- | | 0011-00FF | Skip | --------- | --------- | --------- | | 0100 | c6 | LDAB #$00 | 0101 | IMM | | 0101 | 00 | --------- | --------- | --------- | | 0102 | 17 | TBA | 0102 | INH | | 0103 | 4C | --------- | --------- | --------- | | 0104 | 83 | SUBD #$0064 | 0105-0106 | IMM | | 0105 | 00 | --------- | --------- | --------- | | 0106 | 64 | --------- | --------- | --------- | | 0107 | C6 | LDAB #$8A | 0108 | IMM | | 0108 | 8A | --------- | --------- | --------- | | 0109 | C9 | ADCB #$80 | 010A | IMM | | 010A | 80 | --------- | --------- | --------- | | 010B | D2 | SBCB $10 | 010C | DIR | | 010C | 10 | --------- | --------- | --------- | | 010D | D7 | STAB $11 | 010E | DIR | | 010E | 11 | --------- | --------- | --------- | | 010F | 86 | LDAA #$3A | 0110 | IMM | | 0110 | 3A | --------- | --------- | --------- | | 0111 | 8B | ADDA #$40 | 0112 | IMM | | 0112 | 40 | --------- | --------- | --------- | | 0113 | 18 | LDY #$0011 | 0114-0116 | IMM | | 0114 | CE | --------- | --------- | --------- | | 0115 | 00 | --------- | --------- | --------- | | 0116 | 11 | --------- | --------- | --------- | | 0117 | 18 | PSHY | 0117 | INH | | 0118 | 3C | XGDX | 0118 | INH | | 0119 | 8F | PSHX | 0119 | INH | | 011A | 3C | XGDX | 011A | INH | | 011B | 8F | PULA | 011B | INH | | 011C | 32 | PULB | 011C | INH | | 011D | 33 | --------- | --------- | --------- | | 011E | 86 | LDAA #$08 | 011F | IMM | | 011F | 08 | --------- | --------- | --------- | | 0120 | C6 | LDAB #$11 | 0121 | IMM | | 0121 | 11 | --------- | --------- | --------- | | 0122 | 3D | MUL | 0122 | INH | | 0123 | DD | --------- | --------- | --------- | | 0124 | 12 | --------- | --------- | --------- | | 0125 | 3F | STD $12 | 0125 | INH | | 0126 |  |  |  |  | | 0127 |  |  |  |  | | 0128 |  | STX #3FFF |  |  |   **Observations 1**  I was a bit confused with the results because the opcodes didn’t match the addresses. For example, the opcode LDY #$0011 had P=0115 when it should be in 0113. I had to move up many of the immediate mode opcodes in the table above.  Another problem was the end of the code didn’t seem right. STX was found in 0128, 0129, 0130 … . I didn’t check past 130 and did not find a SWI opcode.  By using the instruction set from the reference manual I was able to fill the addressing mode column and notice some patterns.  **Table 2.  Recording Single Step Results**   |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | Step | P | Y | X | A | B | C | S | N | Z | V | C | | 1 | 0102 | FFFF | 0011 | 50 | 00 | 14 | 002D | D | D | 0 | - | | 2 | 0103 | FFFF | 0011 | 00 | 00 | 14 | 002D | D | D | 0 | - | | 3 | 0107 | FFFF | 0011 | 00 | 9C | 10 | 002D | D | D | D | D | | 4 | 0109 | FFFF | 0011 | 00 | 8A | 18 | 002D | D | D | D | D | | 5 | 010B | FFFF | 0011 | 00 | 0A | 13 | 002D | D | D | D | D | | 6 | 010D | FFFF | 0011 | 00 | 89 | 1B | 002D | D | D | 0 | - | | 7 | 010F | FFFF | 0011 | 00 | 89 | 19 | 002D | D | D | 0 | - | | 8 | 0111 | FFFF | 0011 | 3A | 89 | 11 | 002D | D | D | D | D | | 9 | 0113 | FFFF | 0011 | 7A | 89 | 10 | 002D | D | D | D | D | | 10 | 0117 | FFFF | 0011 | 7A | 89 | 10 | 002D | D | D | D | D | | 11 | 0119 | FFFF | 0011 | 7A | 89 | 10 | 002D | D | D | D | D | | 12 | 011A | 7A89 | 0011 | FF | FF | 10 | 002B | D | D | D | D | | 13 | 011B | 7A89 | 0011 | FF | FF | 10 | 0029 | D | D | D | D | | 14 | 011C | FFFF | 0011 | 7A | 89 | 10 | 0029 | D | D | D | D | | 15 | 011D | FFFF | 0011 | 7A | 89 | 10 | 002A | D | D | D | D | | 16 | 011E | FFFF | 0011 | 7A | 89 | 10 | 002B | D | D | D | D | | 17 | 0120 | FFFF | 0011 | 08 | 89 | 10 | 002B | D | D | D | D | | 18 | 0122 | FFFF | 0011 | 08 | 11 | 10 | 002B | D | D | D | D | | 19 | 0123 | FFFF | 0011 | 00 | 88 | 11 | 002B | - | - | - | D | | 20 | 0125 | FFFF | 0011 | 00 | 88 | 11 | 002B | D | D | 0 | - | |

**Observations 2**

For this table, I moved the columns around according to AxIDE trace results. S seemed like the stack pointer and C looked like the code register value. I could not figure out how to get the carry value or any of the code register bits. I got the NZVC values from the manual’s instruction set [D=delta], but I need the initial value to find out whether it is either 0 or 1.

**Conclusion**

The first lab was needed to test for hardware/software problems before we get any further. I now feel comfortable with connecting the board and getting to the console. I was also able to witness the operations of a microcontroller by reading it’s memory.