

**Lab #1 - Introduction to HC11 Development**

Wayhar Ngeth, Habilou Sanwidi

January 16, 2020  
School of Electrical & Computer Engineering

1. **Introduction**

The goal for the first lab is to gain familiarity with the HC11 by executing some basic assembly programs and building and compiling those programs. There is also a goal of recognizing memory allocations within our code. This experiment aims to introduce to students some interesting features about what the HC11 can do.

1. **Procedures & Discussion**

We started off by setting up a communication protocol between the board and the PC. The parameters used are:

* Baud Rate **9600**
* Parity for error detection **None**
* Number of bits per character **8**
* # stop bits **1**
* Flow control **Xon/Xoff**

We then processed to uploading the compiler corresponding to the 6811-chip family, restarted the AsmIDE and reset the board using the SW2 key.

We then uploaded some test programs and observed how the HC11 reacted. Some of the reactions included a light show, messages displaying on the 7-segment display and musical tunes. Through testing sample code, we learned how to modify memory locations and their values. These test experiments also allowed us to visually see some functions of the board.

However, we did experience a minor setback towards the beginning of the experiment. When uploading the test programs, we learned after a few tries to always press the SW2 (Reset) button so that new programs can be uploaded. It took us a few minutes to understand conceptually what loading, opening, and building actually do inside the board.

1. **Conclusion**

Throughout this lab, we learned how to setup a communication between the HC11 board and a PC. We also got the opportunity to run some sample test and familiarized ourselves with HC11’s architecture, which would, without a doubt, be useful for future labs.

1. **Lab Participation**

Throughout this lab, my partner and I both got the opportunity to try every step of the process for ourselves and equally participated in the experiment. Habilou wrote down the explicit answers during the lab and completed the procedures and conclusion portion of the report. Wayhar completed the rest and ensured a quality report before submitting to eLC.

1. **Explicit Answers**

**4.1**

**The program allows the switches on the EVB to turn on and off the onboard LEDs.**

|  |  |
| --- | --- |
| Program | Observation |
| Ex2 | Light show |
| Ex3 | Displays HELP |
| Ex6 | Plays a tune |

**5.1   
 a. RM changes the register modifier and its content.**

**b. RM A changes the register modifier to a specified register**

|  |  |  |
| --- | --- | --- |
| Address | Byte | Instructions |
| D000 | 8e | lds |
| D001 | 8F | lds |
| D002 | ff | lds |
| D003 | 86 | ldaa |
| D004 | 00 | ldaa |
| D005 | B7 | staa |
| D006 | 10 | staa |
| D007 | 07 | staa |
| D008 | B6 | ldaa |
| D009 | 10 | ldaa |
| D00A | 03 | ldaa |
| D00B | B7 | staa |
| D00C | 7e | staa |
| D00D |  | staa |
| D00E |  | jmp |
| D00F |  | jmp |

**5.2**

**b. Changes the content of the memory at the specified ROM address.**

**c. By making use of Backspace & Spacebar**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Address | D000 | D001 | D002 | D003 | D004 | D005 |
| Byte | 8e | 8F | ff | 86 | 00 | B7 |
| Instructions | lds | stack | stack | ldaa | ldaa | staa |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Address | D006 | D007 | D008 | D009 | D010 | D00b | D00e |
| Byte | 10 | 07 | B6 | 10 | 03 | B7 | 7e |
| Instructions | ddrc+REGBLK | ddrc+REGBLK | ldaa | portc+REGBLK | portc+REGBLK | staa | jmp |