**Hardware Development Timeline**

* **29th August** - Given two PCB boards for Price Display and Cash Register. Began soldering as per Part 1 Schematics on both the boards simultaneously.
* **3rd September** - Continued soldering on both the boards
* **4th September** - Completed soldering on Board 1. Tested to see proper functionality by flashing Boot86.hex into the EEPROM of Board 1 and connected to PC to see output via Hyperterminal but the test failed and there was no output. Began debugging the board. Also, continued soldering on Board 2.
* **5th September** - Completed debugging on Board 1 and figured out the problem. It was due to improper soldering of SMD 100ohm resistor at position R13. Resoldered the resistor and checked for proper functionality with Boot86.hex and Timer.hex and Board 1 worked properly. Completed soldering on board 2 and waiting for testing.
* **10th September -** Completed testing and Board 2 worked properly. Collected circuit for Part 2 of Hardware.
* **14th September -** Started designing the schematics for the different components of the Part II hardware.
* **16th September -** Completed the schematics and discussed about possible extra hardware features for the project.
* **19th September -** Had hardware schematics evaluation. The following were the changes that were asked to be made by us:-
  + Interface LCD and KeyPad through 8255 and BCD and speaker through 80188. Use latches while interfacing on the 80188.
  + LCD Display System:-
    - Vo (for contrast) - Connect it through a 103 potentiometer (one leg to Vcc, other leg to ground and third leg to Vo)
    - LED+ and LED- (for backlight) connect a resistor to LED- and then to ground. Find the value of the resistor to be used. Check page 13 of LCD datasheet on how to do it.
  + Keypad System:-
    - Stick with the 4+3+3 design for the two keypads
    - Use resistors to pull up to 5V for all the 10 pins. The resistance values are around 1000ohm. Poll for which key is being pressed by sending and then detecting a low signal. This is done for better sensitivity.
  + BCD 7-segment display System:-
    - Use two latches (one instead of 7447 decoder and one instead of the 3-8 decoder).
    - Also use the given 8pnp and 6npn transistors while interfacing. The 8pnp are meant for choosing the pattern to be displayed while the 6npns are used while choosing which BCD display to use.
    - The arithmetic operations to be done on ASM side instead of on software side.
    - Transistor connection : Latch -> base -> collector and emitter (connect emitter to resistors same as port A)
    - Use common anode displays and use the 8 pnps for choosing the pattern for these
    - Remove the NOT gates
    - Put individual resistors in the BCD legs as well as in the base current part and also for the amplification part.
  + Speaker System:-
    - Use Opamps while interfacing
    - Gain should be still lesser
    - Remember to use Band Pass filter somewhere in the range of 800-1000Hz
* **26th September -** Wrote code for the basic functionality of the LCD system. I.e. making it display a particular character on the first line of the LCD many times. The following are the Problems --> Solutions that we experienced while testing.
  + There was no output on the LCD system once we sent the HEX file through Hyperterminal --> The Timer.asm skeleton which we used was wrong since we hadn’t declared many necessary constants. So we made those changes.
  + Pin 3 of 8255 Port B was always high irrespective of the input --> The 8255 chip was damaged so we had to change the chip
  + The output of the LCD was very dim and the text was difficult to read --> The power supply adapter which we used didn’t supply 5V so all the outputs were dim. So we had to solder two wires (5V and GND) to the power source and take the voltage supply from the power source through the wires.
  + The number of characters that were displayed on the screen was always lesser than the number of characters that were supposed to be displayed --> After displaying every character, we had to wait until the LCD was ready to display the next character. So we had to add code that would keep polling the Busy Flag on the LCD and perform the next operation based on the busy flag value.
* **27th September -** Next in line is to make the LCD display words which are passed through the ASM program. Also, to complete the Keypad circuit and see whether the basic functionalities are working.
* **1st October -** LCD is now working fully. Words were not being displayed because of an improper declaration of a variable in the data segment. User defined variables have to be declared at the end of the data segment only. Also, keypad is more or less working. Need to make changes to the code to make it work properly.
* **3rd October -** LCD done and dusted now. Added the LCD code into procedures and refactored the entire code. Keypad also works now. There was a problem with the polling part of the keypad which made it go into an infinite loop. This problem has now been solved. Next step is to make it work for two keypads, create a separate Keypad.asm file and also to display the character that is being pressed on Hyperterminal. BCD and speaker we started building the circuit.
* **4th October -** Circuit for the 7-Segment Display completed on breadboards. Next step is to write code and check if the display system works.
* **10th October -** The keypad code completely works. Also, we have integrated both the keypads to work together at the same time. The following was the problem we faced and the corresponding solution:-
  + Once we pressed a key, Hyperterminal kept showing multiple key presses since the loop which checks the key being pressed was too fast and it kept detecting the same key-press again and again - We added a flag that makes sure that only one instance of every key-press is detected.
* **11th October -** 7-Segment basic code now works. We are able to give a particular pattern and also control which BCD displays that pattern through our ASM code. Now we have to complete the rest of the ASM code and the next hardware component to complete is the speaker system.
* **17th October –** Tried to integrate keypad with BCD by calling Timer 2 interrupt at 3ms intervals. However this failed due to improper usage of stack in the ISR. To be corrected and then complete the code. Speaker still in progress.
* **23rd October –** Wire-wrapping for the BCD circuit is complete. Debugging for it is still left. Speaker finally showing signs of working. Output from EEPROM is now reading properly. We just have to play the sound through the speakers after proper filtering. Next plan is to complete keypad BCD integration and start off with serial communication.
* **24th October –** Speaker finally works! The 7-segment display also works fully on its own and it is also able to interact with a keypad input. Have to now make it more extensive. Speaker had a few stupid but critical problems which we solved. The DAC chip was not inserted properly. We tried getting input from the EEPROM without connecting datelines to it. And finally, we used a PRINT\_2HEX function to view the data sent to the DAC on the Hyperterminal screen. Big blunder since it modifies the data and hence we ended up getting shit on the speaker. Corrected all these and it now works fine. Maybe have to do a bit more filtering and make it capable of playing multiple sounds.
* **2nd November –** Speaker now works with multiple files. Started integrating keypad with speaker and BCD. Also, completed wire-wrapping for the daughter board but yet to test it out.
* **6th November –** Cash Register works now with Keypad+BCD+Speaker! User can now give input on keypad and it is reflected on BCD and speaker. Wire wrapping also checked perfectly.
* **7th November –** Basic serial communication with LCD board working perfectly. We are now able to give input for both lines of LCD through the serial port. Next up to implement is a protocol that the LCD board would follow to communicate through serial.
* **14th November –** Basic serial communication working for Cash Register. We are able to send a price and say it on the speaker. More modifications to be done to make it say all types of numerical prices.
* **18th November –** Basic calculator functions done. The speaker is now able to say any type of price. The program can now add any type of number and also subtract any type of number.
* **22nd November –** Hardware side of the project almost done! We are now able to test out multiple transactions and everything seems to be working fine for the basic features.
* **28th November –** We are done with hardware side of coding!
* **29th November –** Done with implementing rolling of text on the LCD. Now completely done with both the boards. Only testing left.