# Lab 4 Write Up

## Schematic

Please find the schematic for the lab 4 project as follows. Schematics shows LCD\_ENABLE connected from SPLD to LCD. Figure also shows connections for EEPROM and LCD.



Figure : Board Schematic for Lab 4

## Signoff Sheet

Front Sheet :

Text, letter

Description automatically generated

Figure : Front side of sign off sheet

Back Sheet :

Text, letter

Description automatically generated

Figure : Back side of sign-off sheet

## Board Top

A picture containing text, electronics, circuit

Description automatically generated

Figure : Board Top Layout

## Board Bottom

A picture containing map

Description automatically generated

Figure : Board Top Layout

## Lab 4 Part 1

### I2C EEPROM

Implemented an EEPROM I2C device driver with the ability to bit-bang write and read a byte at any EEPROM I2C address using function calls from C. Please see the UI used for the same as follows:

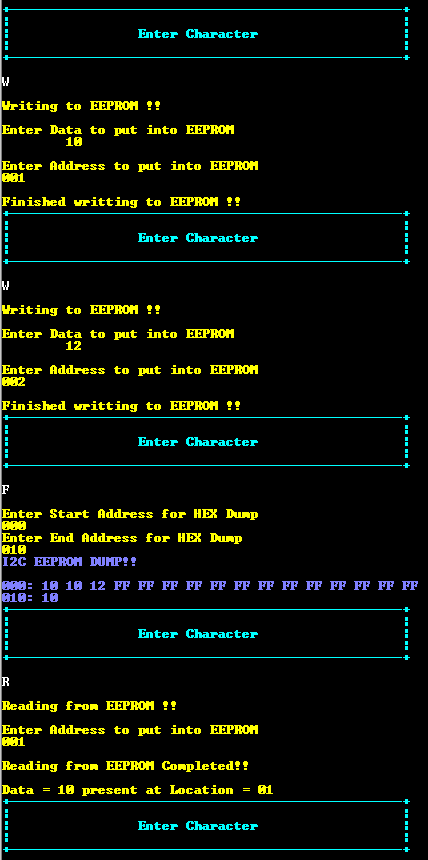
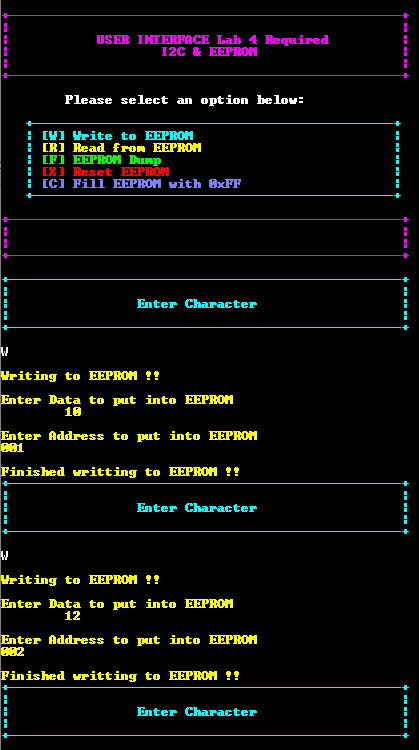


Figure : UART Console for EEPROM Part 1

Figure below shows the Logic Analyzer for EEPROM write when data 10 is loaded in memory address 10 H. Figure shows the write cycle operation as expected. We have a slave address follows by ACK. Data byte is followed by ACK bit.

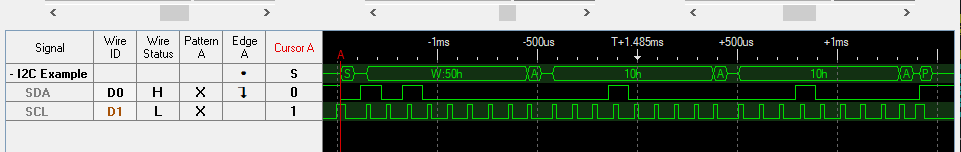


Figure : EEPROM Write

Figure below shows the cycle for EEPROM read which is behaving as expected when trying to read value stored at address 010H. The highlighted part shows the binary representation of a data read from address which is nothing but a 10. Read operations contains slave address followed by data byte.

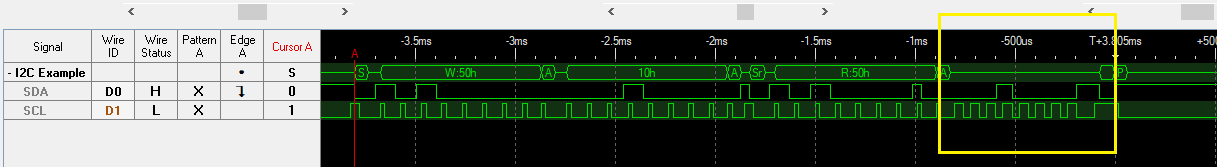


Figure : EEPROM Read

Figure below shows the sequence of bit pattern when function EEPROM rest is called. The reset sequence involves sending a start condition on the I2C bus, followed by the device address with the R/W bit set to write (0), and then sending the bit pattern 8 SDA high, followed by an I2C clock, and one stop bit. The 8 SDA high bits serve as a command to reset the device.

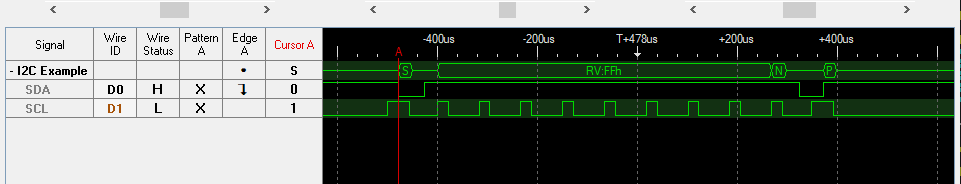


Figure : EEPROM Reset

Timing calculation is performed on LCD and it perfectly shows that the rise time and fall time calculation perfectly satisfies the limits mentioned in I2C specifications and EEPROM data sheet. Observed fall time is 55 ns and observed rise time is 76 ns, which is well under the requirements of 300 ns for each of them

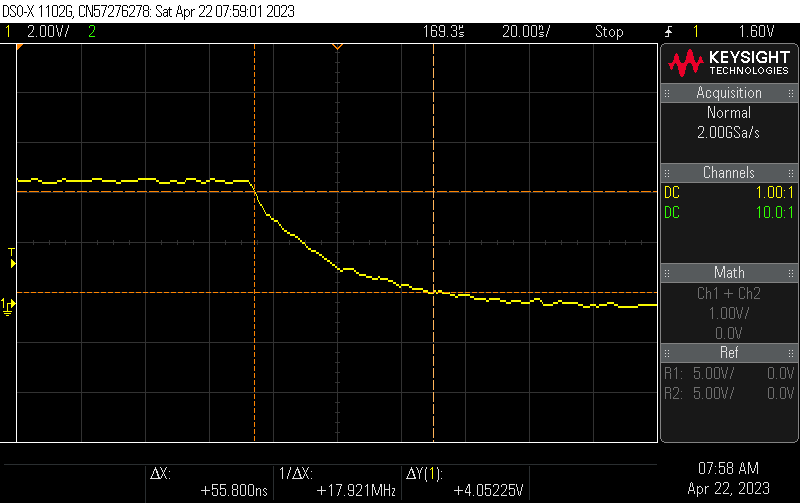


Figure : Fall Time below 300 ns

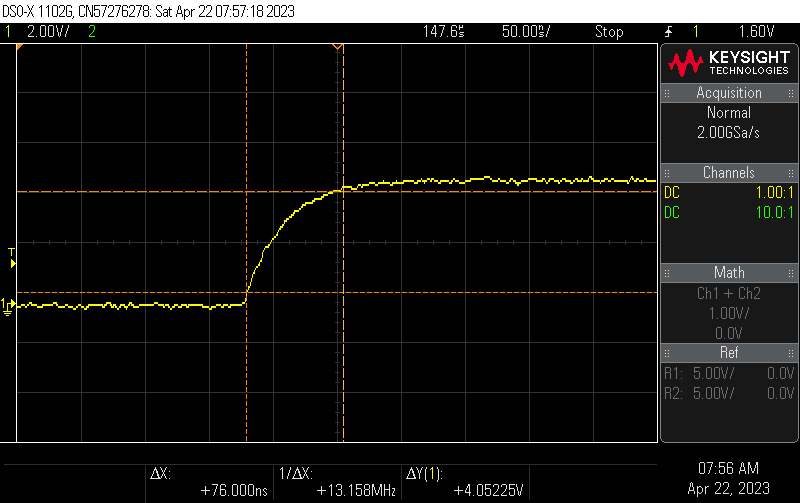
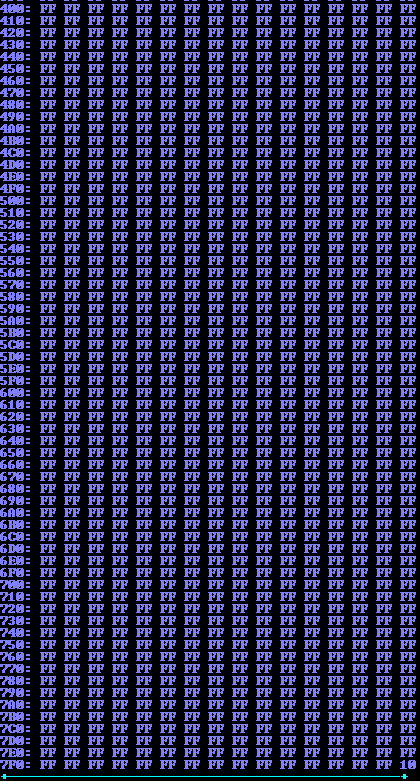
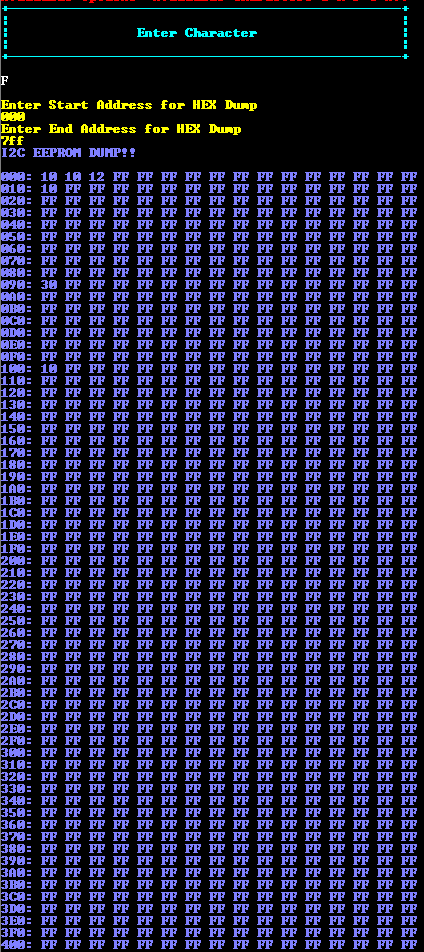


Figure : Rise time below 300 ns

Figure below verifies that HEXDUMP works for 000 to 7FF.



# Lab 2 & Lab 3 Part 3 Supplemental 1 :

The lab 2 deals with using LCD and implementing a full-scale timer with a real time clock implementing using Timer interrupt. Ideal output with author info and clock running is as follows .

A picture containing text, electronics, circuit

Description automatically generated

Figure : LCD with author info and real-time clock

The ideal UART interface looks like this as shown in below picture. Which takes inputs to print character, string, got to address and (x,y). Also it includes options ot printf a custom character based on user inputs as well two built in customer characters as seen in photograph below .

A picture containing graphical user interface

Description automatically generated

Figure : UART interface showing HEX dump of CGRAM AND DDRAM

Above interface also shows the LCD hex dump after printing string “Kiran Jojare Finished Lab 4” as seen and verified using HEXDUMP of CGRAM and DDRAM.   
  
Figure options to pause the elapsed time and reset the elapsed time has also been tested and verified as seen in sign off sheet.

Timing analysis of LCD’s using logic analyser on E, R/W and RS signal has been proven to be under specified range as seen in the image below. Setup time is observed to be 1.085 usec and hold time is observed to be 1.095 usec as seen in the screenshot below.

A computer screen capture

Description automatically generated with low confidence

Figure : Setup and Hold Time Timing Analysis

Custom characters has been implemented using all 8 characters codes and is visible as follows:  
  
A picture containing text, electronics, circuit

Description automatically generated

Figure : Customer Character CU Logo

A close-up of a computer

Description automatically generated with low confidence

Figure : Custom Character with Smileys Logo

The code also implements the logic where input arguments are taken from the user for each row value as well as customer character code and respective customer character is then printed on the desired X and Y co-ordinates.