

Lab 5

- We are shooting to have the majority of the class finishing tonight
- I made a mistake in last weeks notes
 - `#define BASE_EEPROM_ADDRESS 0x0F00`
 - Should say `#define BASE_EEPROM_ADDRESS 0x7F00`
 - Address range should be `0x7F00 – 0x7FFF`

Lab 5

- I have my microcontroller running at 8MHz without the PLL enabled
- SPI initialization
 - Set port direction
 - Set CS to 1
 - SSP1CON1
 - SSPEN to enable
 - Idle state for clock is low
 - Master mode, Fosc/64
 - SSP1STAT
 - Sample at end of data output time
 - Transmission on transition from active to idle

Lab 5

- I2C initialization
 - Set port direction (input)
 - Set ports to digital
 - SSP1CON1
 - SSPEN to enable
 - I2C master mode, $\text{clock} = F_{\text{osc}} / (4 * (\text{SSPxADD} + 1))$
 - SSP1STAT
 - Sample at end of data output time
 - SSP1CON3
 - Enable interrupt on detection of start or restart condition
 - Enable interrupt on detection of stop condition
 - Set SSPADD

Lab 5

- SPI write
 - Similar to reading
 - You need to break into two command sequences
 - Enable writing
 - Set CS to low
 - Enable writing via the write enable latch
 - Set CS to low
 - Write
 - Set CS to low
 - Send write command
 - Send MSB of address
 - Send LSB of address
 - Send the value to write
 - Set CS to high
 - Brief pause (5 ms)

Lab 5

- I2C writing
 - Simpler than reading
 - Send start condition (same as reading)
 - Send write command (same as reading)
 - Send MSB of address (same as reading)
 - Send LSB of address (same as reading)
 - Send value to write (different from reading – same as sending a byte of the address)
 - Send stop condition (same as reading)
 - Brief pause (5 ms)

Lab 5

- Questions
- Where are we
 - Buttons and state change
 - LCD output
 - Initialization of EEPROM
 - EEPROM read stub
 - EEPROM write stub