

Lab 5

- Devices we will be interfacing
 - LCD
 - EEPROM
 - Potentiometer
 - Pushbuttons
- New groups of two
 - If you are behind on any of the labs, use the PIC 18 board

Lab 5

- Potentiometer
 - Specify the address into the EEPROM
 - Use 8 MSBs from the potentiometer
 - Define an base address in your code
 - `#define BASE_EEPROM_ADDRESS 0x0F00`
 - Add the pot address to the base address and display the address value on line 1 of the LCD in hex and decimal
 - “ 0x7FFF 32,767 ” (15 of the available 16 characters)
 - Display the value stored at that address in the EEPROM on line 2 of the LCD in hex, binary, and decimal
 - “FF 11111111 255” (15 of available 16 characters)
 - As the potentiometer is moved, read the value at that address from the EEPROM and display

Lab 5

- Since we are reading (versus writing), there should not be issues related to reducing the life of the EEPROMs
- You can simply read the value from the EEPROM each time the top 8 bits of the potentiometer change

Lab 5

- When the rightmost button is pushed
 - The current address is saved
 - The potentiometer will now allow the selection of an 8 bit value
 - Line 1 of LCD maintains the selected address
 - Line 2 not displays the 8 bit value selected by the potentiometer (same format as previously defined)
 - If the rightmost button is pushed again, the mode is cancelled and the display returns to displaying address and content data on the LCD and selecting addresses with the pot
 - If the leftmost non reset button is pushed, the value selected by the potentiometer is written to the selected address of the EEPROM, and the mode returns to selecting addresses and displaying values

Lab 5

- You can use my LCD functions if you need (has anyone implemented the LCD code?)
- Start working on
 - The LCD output
 - Should be the same for either board
 - The pushbutton functionality
 - Should be same for either board, other than possibly the ports
 - Create a dummy function to read and write values to the EEPROM
 - Write – just return
 - Read – return the address -1

Lab 5

- Keep track of any code you write that is specific to the DEM 2 board
- The only real difference between the programs is accessing the EEPROM which is isolated in three functions
 - Initialization of EEPROM
 - Reading from EEPROM
 - Writing to EEPROM

Lab 5

- Things to consider
 - Using interrupts versus polling for
 - Pushbutton
 - EEPROM activities
 - Potentiometer changes
- I use polling for everything

Lab 5 Modes

- Modes transitions based on button pushes
 - Mode 0 (default mode)
 - Display address based on pot value + base address on top line
 - Display value from EEPROM at that address on bottom line
 - Read from EEPROM at address from pot + base address
 - Mode 1
 - Display address based on pot value on top line, stays constant once you enter mode 1
 - Display value of pot on bottom line

Lab 5 Transitions

- Mode 0
 - Transition to mode 1 on right button push
 - Keep base address + pot as the address value
- Mode 1
 - Transition to mode 0 on right button push
 - No write to EEPROM (cancel write)
 - Transition to mode 0 on left (non reset) button push
 - Write the pot value (top 8 bits) to the address defined when transitioning to mode 1

Lab 5 General Flow

- Initialize LCD & EEPROM
- Setup pot and LEDs
- Infinite loop
 - Get value from pot
 - Update mode based on buttons
 - Perform required action
 - Read from EEPROM
 - Write to EEPROM
 - Only write once per mode transition
 - Update LCD

Where Are We With Lab 4

- Who is done with part A & part B
- Who is done with part A
- Who has partial functionality

Lab 5

- Communicating with the EEPROM is the most complicate process we have hit
 - Setup the MSSP
 - Initialize contact with the EEPROM
 - Tell the EEPROM what you want to do (read or write)
 - Tell the EEPROM the address you want to use
 - 16 bit address is broken into two 8 bit address since SPI & I2C have 8 bit interfaces
 - Send or receive the data

Lab 5

- Of the two interfaces, I2C is slightly more complicated to use
 - Primarily due to more complicated hand shaking
 - Checking acknowledgement bits
 - Setting additional bits
- Read the EEPROM datasheet
- We will cover the read/write sequence for both again on Tuesday
- Next Tuesday last day for full credit for lab 2