ME433 05/05/16

LCD

* Intitializationcode writtenfor you
* Colours have been defined (can add more if you want
* 8 functions to make it workd
  + spi\_io
  + lcd\_init
    - software reset and delay
    - set pixel format (5 6 5 version)
    - gamma corrections
    - turn on screen
  + lcd\_draw pixel
    - tells screen address of pixel to change and what colour you want
    - calls lcd\_data16 sends 2 chars to send colour
  + lcd\_clearscreen
* need to write algorithm to show assci characters to the screen
  + each character takes up 8x5 pixels
  + require a hex20 offset ‘0’ 🡪 ASCII[0x30 -0x20]
  + draw character:
    - for (j = 0 to 5) // which column
      * for(I = 0 to 7) // which bit in that column
      * shift column bits
      * set 1 if pixel black, set 0 if pixel white
  + draw string:
    - while(string[q] != 0) // continue until reaching null
      * draw character (string(q))
      * q=q+1

USB CDC

* higher throughout of data, but no guarantee of when it will talk
* microchip 🡪 harmony 🡪 apps 🡪 USB 🡪 serial emulator
  + do not move out of this folder until complete
  + make a copy in case you mess it up
* in xide open cdc serial emulator project
  + firmware
* right click on project, properties
  + change to our pic
* compile to find bugs
  + comment out AN and CN initializations we will add our own later
  + bps switch state change the return to read channel A
  + comment out #pragma 🡪 replace with our own
* ls /dev/tty\* in command line lists communication ports (should see a usbmodem)
* screen /dev/tty.usbmodem to open screen to talk to the pic
* open app.c in xide
  + APP\_tasks
    - UART receive
      * If receive from UART write to screen
      * We will add sprint to send data to the computer
    - Cdc read
      * If receive certain character do something (start sending characters to computer)
      * Call this function in the UART receive