C++ Code Snippets

PART III: Serial Communications via LCD for Arduino IDE/Teensy 3.2

John R. Wright, Jr., PhD, CSTM, CLSSGB, F.ATMAE
ITEC 467, Mobile Robotics
“The Knack”

https://www.youtube.com/watch?v=g8vHhgh6oM0

PowerTip PC1602Q B
16 x 2 LCD Display

BPI-216 LCD Driver Board

Baud Rate
Up= 9600bps
Down= 2400bps

Contrast
= Darker

NOTE: It's not necessary to connect both +5 or both GND pins; they are duplicated for convenience (and to allow a reversible connector).

Backlight
Up= ON
Down= OFF

Pads for external BL switch

https://seetron.com/bpi216/bpi216hdw.html
7404 IC

http://tayloredge.com/reference/Packages/pinouts/index.html
Welcome to Teensy 3.2
32 Bit Arduino-Compatible Microcontroller

To begin using Teensy, please visit the website & click Getting Started.

www.pjrc.com/teensy
Sample Code for BPI-216

/*
This code was adapted from http://seetrontech.blogspot.com/2011/12/arduino-hello-world-for-bpi-216-serial.html
J. Wright, 2017
Note: To use this with an Arduino or Teensy, one must invert the TX Serial Output with an inverter gate

Inverter used: 7404 IC
Pin 14 = +5
Pin 7 = GND
Pin 1 = Input from Microcontroller TX
Pin 2 = Inverted Output to SER input of LCD Module (Seetron BPI-216 driving a PowerTip PC1602Q B)
*/
```cpp
#include <SoftwareSerial.h> // Library already in your IDE

const int rxPin = 255 // Not used for LCD – set to invalid pin as we need
// something for command below
const int txPin = 5 // Connect BPI/BPK's SER input to this pin

const char clearScreen[ ] = {
  254,1,254,128,0}; // https://www.seetron.com/bpi216/bpi216prog.html
const char message1[ ] = "Hello 467 World";
const char message2[ ] = "Code Hard";
const char message3[ ] = "Flowchart?";

SoftwareSerial mySerial = SoftwareSerial(rxPin, txPin);
```
void setup()
{
    pinMode(txPin, OUTPUT);  // Define pin mode for tx:
digitalWrite(txPin, LOW);  // Stop bit state for inverted serial
mySerial.begin(9600);    // Set the data rate
delay(1000);          // Wait for 1000ms to establish communications
mySerial.print(clearScreen);
mySerial.print(message1);
}

void loop()
{
    delay(1000);
    mySerial.print(clearScreen);
delay(1000);
    mySerial.print(message2);
delay(1000);
    mySerial.print(clearScreen);
delay(1000);
    mySerial.print(message3);
}
Video of Teensy to BPI-216 LCD

https://youtu.be/rBc9dd79qe4
Parallax 27977-ND
2x16 LCD

Sample Code for the Parallax 27977-ND LCD


This LCD does not require an inverter IC. This LCD does allow for tones/sound to be generated.

Sample code below developed/tested by Ethan Bressler (2019) & Edited by J. Wright (2019)*/

```c
#include <SoftwareSerial.h> //Library already in your IDE
const int rxPin= 7; // Not used, but needed for command below
const int txPin= 8; // 255 produces a “/” character on first pass so we use 7
const char message1[ ] = "Hello 467 World";
const char message2[ ] = "Code Really Hard";

SoftwareSerial mySerial= SoftwareSerial(rxPin, txPin);
```
void setup()
{
    pinMode(txPin, OUTPUT);  // define pin mode for tx:
digitalWrite(txPin, LOW);  // Stop bit state for inverted serial
    mySerial.begin(9600);  // Set the data rate
delay(1000);  // wait for 1000ms to establish com
}

void loop()
{
    mySerial.write(220);  //220 represents Note A (sound) -- Use Parallax PDF
delay(5000);
    mySerial.print(message1);  //Prints “Hello 467 World”
delay(1000);
    mySerial.write(12);  //Clears screen and set to position 0,0
delay(1000);
    mySerial.write(220);
    mySerial.print(message2);  //Prints “Code Really Hard”
delay(1000);
    mySerial.write(12);
delay(1000);
    mySerial.write(12);
    delay(1000);
}
Which One Do I have?

Seetron BPI-216

- **Baud Rate**
  - Up = 9600bps
  - Down = 2400bps

- **Contrast**
  - Down = Darker

- **Backlight**
  - Up = ON
  - Down = OFF

- **Pads for external BL switch**

- **Serial In**
  - GND
  - +4.8-5.3V

Parallax 27977

NOTE: It’s not necessary to connect both +5 or both GND pins; they are duplicated for convenience (and to allow a reversible connector).
Do I have to use an LCD?

YES! – Refer to the Contest Rules & for Check Point #2

5. ROBOT OPERATION

The robot must show the Contest Judge by using the provided LCD module/screen that it has found the candle before it attempts to put it out (i.e. “Candle Found” / “Scanning for Candle” or similar messages).

For example, the robot cannot just flood the arena structure with CO2 thereby putting the candle out by accident. Failure to use the LCD module properly will result in a disqualification for that run. It is the responsibility of the teams to ensure accurate use of the LCD to the Contest Judge.
Any Questions?

https://camdenkelly.com/common-job-interview-questions-and-answers/