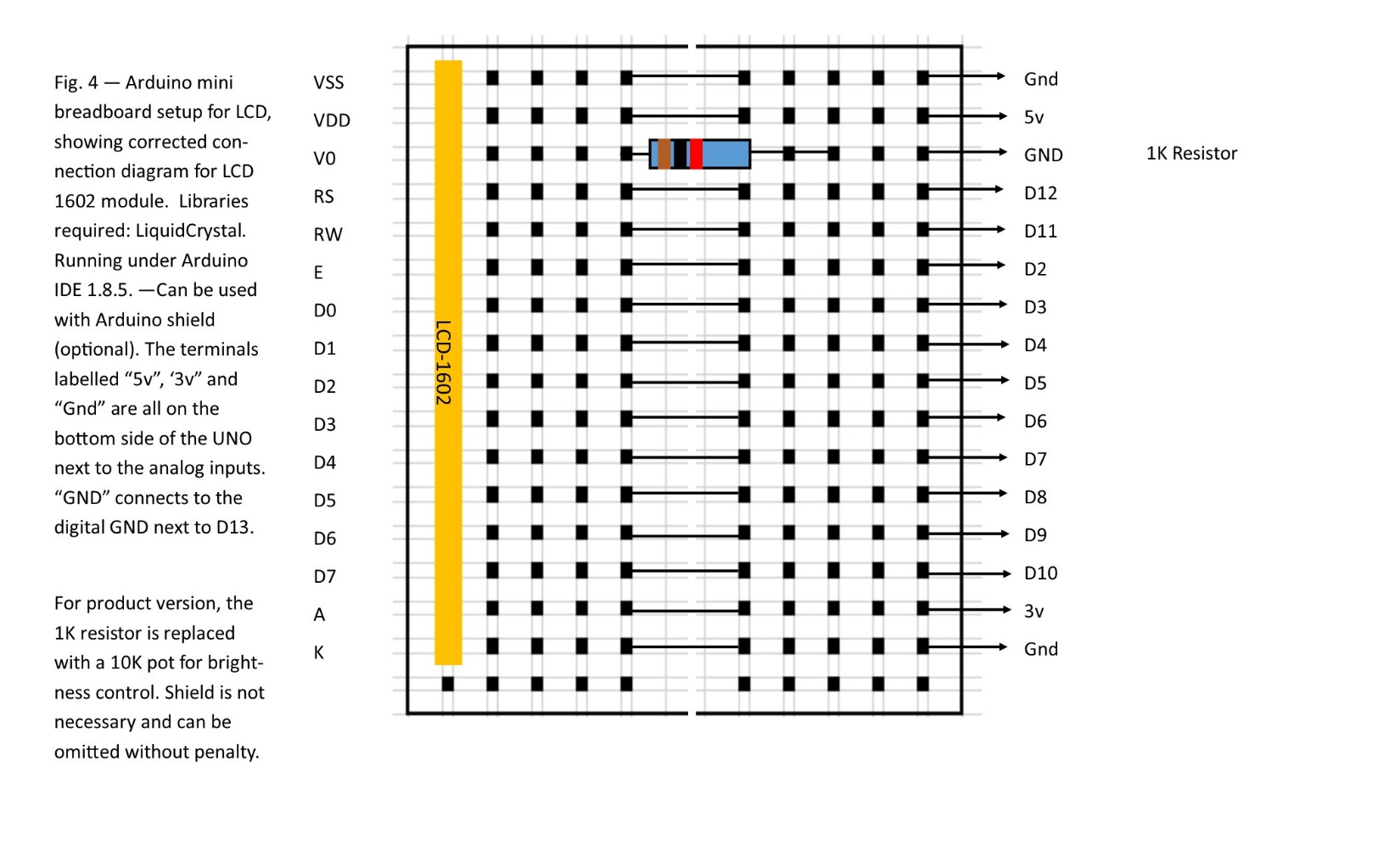
**Project 4 – Product Price Display with LCD\_1602**

Sketch displays a corporate name on line 0 of the LCD\_1602. Items and their prices, and a phone number are sequentially displayed on line 1.

This program also has a user-written function, lineWrite(int, String), which writes a line of text one character at a time using the LcdDataWrite() function on line 0 or line 1 of the display. This function expects either 0 or 1 for the first argument, the line number to write, and a 16-character text string for the second argument. The function does not error check for string length.



**Code for LCD\_Viv**

/\* Product Pricing Display v. 1.0.2 by Bill Jenkins 07-20-2018

using LCD\_1602 display

This will dsplay the company name on line 0

and a list of item prices, then, a phone number on line 1

\*/

// This is for the LCD\_1602 module

// Connections between LCD\_1602 and Arduino

// using 8-bit interface.

// 4-bit interface does not work for some reason.

// 1602 Arduino

// 1 VSS Gnd

// 2 VDD 5v

// 3 V0 TO 1K RESISTOR, OTHER LEAD TO GND

// 4 RS D12

// 5 RW D11

// 6 E D2

// 7 D0 D3

// 8 D1 D4

// 9 D2 D5

// 10 D3 D6

// 11 D4 D7

// 12 D5 D8

// 13 D6 D9

// 14 D7 D10

// 15 A 3v

// 16 K Gnd

// variables for LCD\_1602

int DI = 12;

int RW = 11;

int DB[] = {3, 4, 5, 6, 7, 8, 9, 10};// use array to select pin for bus

int Enable = 2;

// variable for delay after writing line

int duration = 4000; // delay time in milliseconds

void LcdCommandWrite(int value) { // define all pins for LCD\_1602

int i = 0;

for (i=DB[0]; i <= DI; i++) // assign value for bus

{

digitalWrite(i,value & 01); // for 1602 LCD, it uses D7-D0( not D0-D7) for signal identification; here, it’s used for signal inversion.

value >>= 1;

}

digitalWrite(Enable,LOW);

delayMicroseconds(1); // wait for 1ms

digitalWrite(Enable,HIGH);

delayMicroseconds(1); // wait for 1ms

digitalWrite(Enable,LOW);

delayMicroseconds(1); // wait for 1ms

}

void LcdDataWrite(int value) { // writes a single character to the LCD

// initialize all pins

int i = 0;

digitalWrite(DI, HIGH);

digitalWrite(RW, LOW);

for (i=DB[0]; i <= DB[7]; i++) {

digitalWrite(i,value & 01);

value >>= 1;

}

digitalWrite(Enable,LOW); // send HIGH pulse to Enable line

delayMicroseconds(1);

digitalWrite(Enable,HIGH);

delayMicroseconds(1);

digitalWrite(Enable,LOW);

delayMicroseconds(1); // wait for 1ms

}

// function written 07-20-2018 by Bill Jenkins

// writes an entire line of text to either line 0 or line 1

// one character at a time using function LcdDataWrite()

// If writing to line 0, it clears the screen before writing

// function expects a string of 16 characters,

// no error checking done on string length

void lineWrite (int linenum, String linetext){ // line 0 or 1, text to write

int pos=0; // char position inside linetext

if (linenum==0) {

LcdCommandWrite(0x01); // clear the screen, cursor position returns to 0

delay(20);

LcdCommandWrite(0x80); // set cursor position at first line, first position

}

else // don't clear screen if writing to line 1

LcdCommandWrite(0xc0); // set cursor position at second line, first position

delay(20);

while (linetext[pos]) {

LcdDataWrite(linetext[pos]); // write one character to the LCD

pos++; // increment position counter

}

}

void setup (void) {

int i = 0;

for (i=Enable; i <= DI; i++) {

pinMode(i,OUTPUT);

}

delay(100);

// initialize LCD after a brief pause for LCD control

LcdCommandWrite(0x38); // select as 8-bit interface, 2-line display, 5x7 character size

delay(50);

LcdCommandWrite(0x06); // set input mode

// auto-increment, no display of shifting

delay(20);

LcdCommandWrite(0x0E); // display setup

// turn on the monitor, cursor on, no flickering

delay(20);

LcdCommandWrite(0x01); // clear the screen, cursor position returns to 0

delay(100);

LcdCommandWrite(0x80); // display setup

// turn on the monitor, cursor on, no flickering

delay(20);

// show boot up screen

lineWrite(0, "Display v. 1.0.2");

lineWrite(1, "by Bill Jenkins ");

delay(3000);

}

void loop (void) {

lineWrite(0, "ExoticAromaProd."); // line 0 text does not change

lineWrite(1, "Sm. Wax $ 2.00"); // line 1 - price info

delay(duration);

lineWrite(1, "Med. Wax $ 7.00"); // line 1 - price info

delay(duration);

lineWrite(1, "Lg. Wax $ 9.00"); // line 1 - price info

delay(duration);

lineWrite(1, "Sm. Lamp $14-up"); // line 1 - price info

delay(duration);

lineWrite(1, "Lg. Lamp $30-up"); // line 1 - price info

delay(duration);

lineWrite(1, " 773-377-5504 "); // line 1 - phone number

delay(duration);

}

**Construction of Product**

Since all pins necessary for this project are on the digital header at the top and the analog connector at the bottom (for the power connections), no Shield is required for construction of dedicated model. Pin jumpers are to be hot glued in place to the analog and digital headers after all wiring is complete. Female sockets on the jumpers are to be hot glued to the header on the back of the LCD\_1602. A 10K pot will replace the 1K resistor for brightness control of the LCD and backlight. Finished product should be approximately 2.5” tall x 4” wide x 2.5” deep. Since the display is a small 16-character, 2-line LCD panel, it is well suited to things such as a price display, or anything displaying information to people at distances of not more than about 18”.

Front Panel: 4”W x 2.5” H

Hole for LCD panel 1”H x 2 13/16” W

Hole for 10K pot 0.265” dia. (17/64”) for Brightness control.

Rear Panel: Same size as Front

Hatch for batteries (Battery options: 6xAA or 2x18650)

Batteries should be replaced or recharged once pack reaches 7.1V.

Side Panels: 2.5” W x 2.5” H

Holes for USB and 2.1mm barrel jacks on L. side panel only

Top/Bottom Panels: 4” W X 2.5” D

No holes, rubber feet at bottom optional.