

Introduction to Arduino

Week 2

Questions on last week?

Programming the Arduino

- We will focus on C/C++

Now Some more programming statements

For Loop

- A for loop is used when we want to execute a set of statements for a range of values
- Example
- int i =0;
- int i_squared = 0;
- for (i=0; i<=20; i++)
- {
- i_squared =i*i;
- }

If else statement

- Used to change what statements are run based on conditions

- if ($x > 0$)
 {
 do something
 }
else
 {
 do something else
 }

While loop

- int var =0;
- while(var < 20)
 - {
 - // do something repetitive 20 times
 - Serial.print("var is: ");
 - Serial.print(var);
 - Serial.println();
 - var++;
 - }

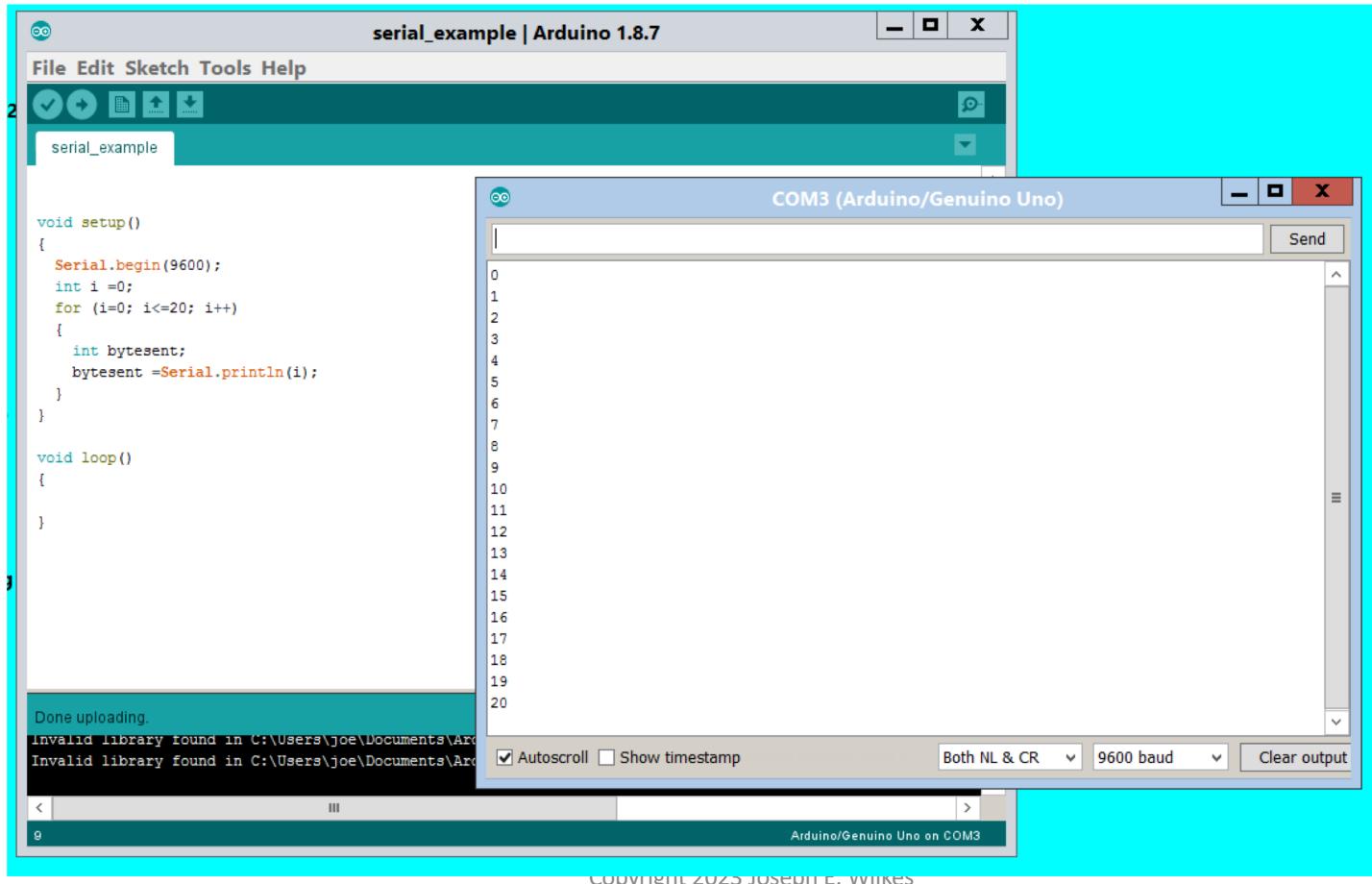
Serial command

- When we want to write something to the attached laptop we use the serial command
- First we have to set the speed of the serial port on the Arduino
- The serial terminal part of the Arduino development environment needs to be set to the same speed
- Typical speeds to use are 9600, 19200, 38400
- In the setup section we need the command
 - `serial.begin(9600)`

Serial Statements

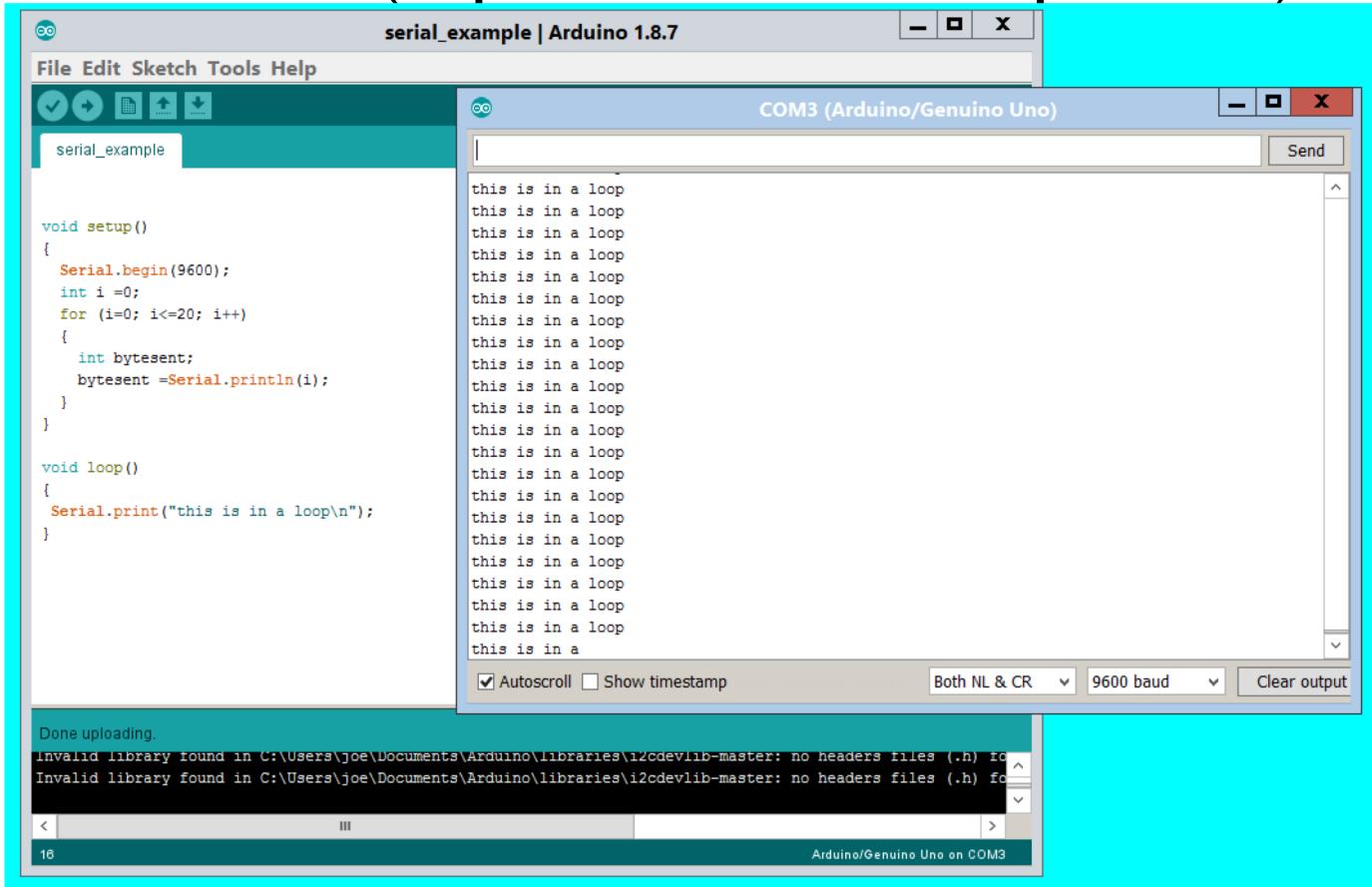
- Syntax
 - `Serial.print(value);`
 - `Serial.print(value, format);`
- Example
 - `serial.print(motor_speed);`
- For more information see:
 - <https://www.arduino.cc/reference/en/language/functions/communication/serial/>

Serial setup in Arduino Development Environment



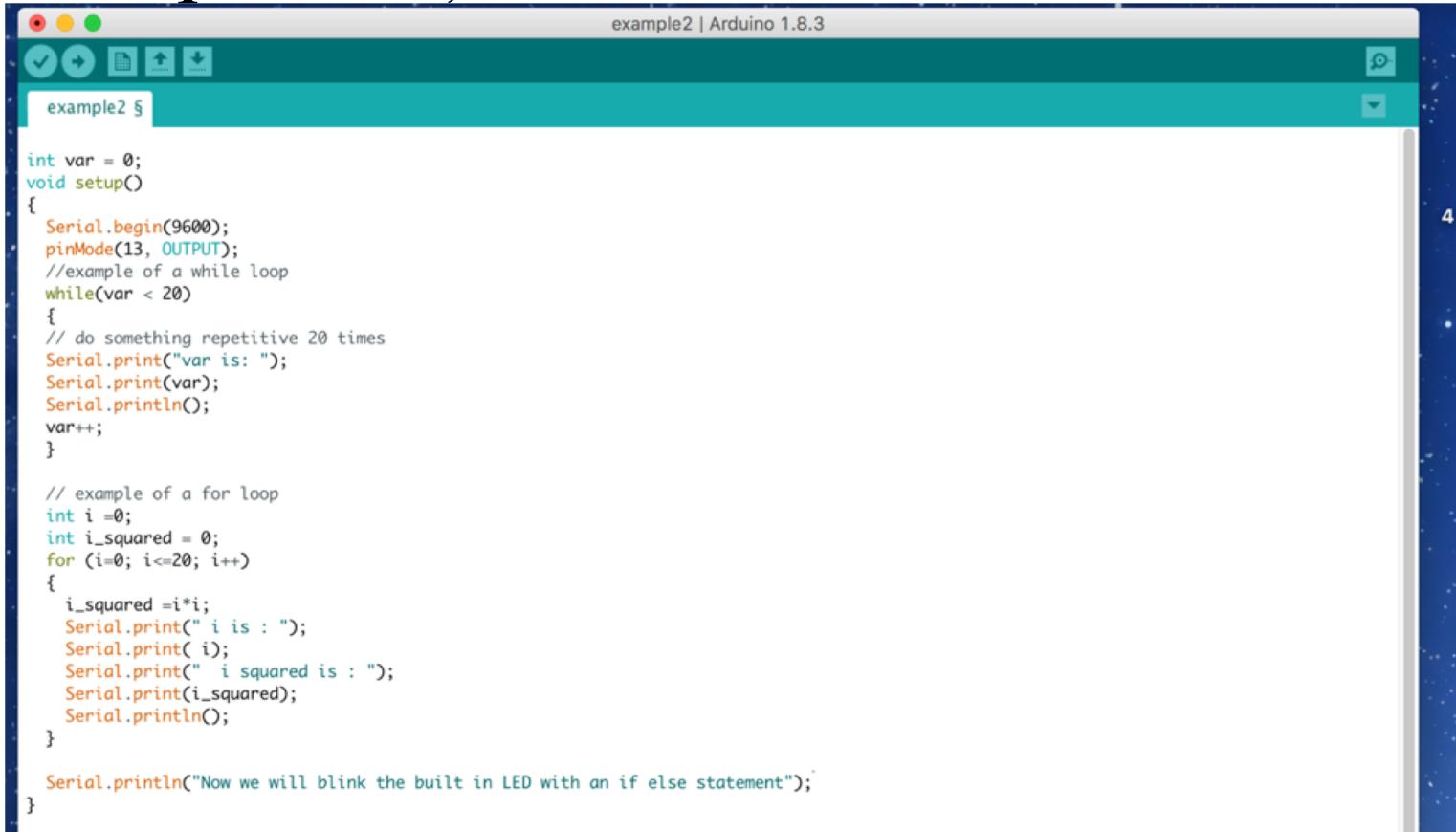
Copyright 2023 JOSEPH E. VVIIKES

Serial setup in Arduino Development Environment (Open serial_example.ino)



Copyright 2023 Joseph E. Wilkes

Examples of use of some of the statements (open example2.ino)



The screenshot shows the Arduino IDE interface with the title bar "example2 | Arduino 1.8.3". The code editor contains the following Arduino sketch:

```
int var = 0;
void setup()
{
    Serial.begin(9600);
    pinMode(13, OUTPUT);
    //example of a while loop
    while(var < 20)
    {
        // do something repetitive 20 times
        Serial.print("var is: ");
        Serial.print(var);
        Serial.println();
        var++;
    }

    // example of a for loop
    int i =0;
    int i_squared = 0;
    for (i=0; i<=20; i++)
    {
        i_squared =i*i;
        Serial.print(" i is : ");
        Serial.print(i);
        Serial.print(" i squared is : ");
        Serial.print(i_squared);
        Serial.println();
    }

    Serial.println("Now we will blink the built in LED with an if else statement");
}
```

Examples of use of some of the statements (open example2.ino)

```
void loop()
{
    //example of an if  else loop
    var = 5;
    if (var >5)
    {
        digitalWrite(13, HIGH);
    }
    else
    {
        digitalWrite(13, LOW);
    }
    delay(2000);
    var =6;
    if (var >5)
    {
        digitalWrite(13, HIGH);
    }
    else
    {
        digitalWrite(13, LOW);
    }
    delay(2000);
}

Done Saving.

Sketch uses 2512 bytes (7%) of program storage space. Maximum is 32256 bytes.
Global variables use 286 bytes (13%) of dynamic memory, leaving 1762 bytes for local variables. Maximum is 2048 bytes.
```

Output of examples

```
var is: 0
var is: 1
var is: 2
var is: 3
var is: 4
var is: 5
var is: 6
var is: 7
var is: 8
var is: 9
var is: 10
var is: 11
var is: 12
var is: 13
var is: 14
var is: 15
var is: 16
var is: 17
var is: 18
var is: 19
i is : 0 i squared is : 0
i is : 1 i squared is : 1
i is : 2 i squared is : 4
i is : 3 i squared is : 9
i is : 4 i squared is : 16
i is : 5 i squared is : 25
i is : 6 i squared is : 36
i is : 7 i squared is : 49
i is : 8 i squared is : 64
i is : 9 i squared is : 81
i is : 10 i squared is : 100
i is : 11 i squared is : 121
i is : 12 i squared is : 144
i is : 13 i squared is : 169
i is : 14 i squared is : 196
i is : 15 i squared is : 225
i is : 16 i squared is : 256
i is : 17 i squared is : 289
i is : 18 i squared is : 324
i is : 19 i squared is : 361
i is : 20 i squared is : 400
Now we will blink the built in LED with an if else statement
```

Autoscroll No line ending 9600 baud Clear output

Copyright 2023 JOSEPH E. VILLEKES

Functions

- Often we find that we need to use the same set of code in multiple places
- Rather than duplicate the code we make it a function
 - Code duplication leads to errors when code is changed but not in all places
- A function can be simple or complicated
- Variables inside a function have no relationship to ones outside of the function
 - But do not reuse names

Example of a Function

- int a = 1;
 - int b = 2;
 - int c = add(a,b);
 - add(int a1, int a2)
 - {
 - int a1_plus_a2 = a1+a2;
 - return(a1_plus_a2);
 - }
- We add "int" before each variable to tell the compiler that each of the variables are integers (1, 2, 3, etc)

Global vs non-global variables

- When we use variables inside of:
 - `setup()`
 - `loop()`
 - `function()`
- They remain local to the function
 - This is called a local variable
- If we want a variable to be known to everything, we put it before the `setup()` code
 - This is called a global variable

Libraries

- Often there are a set of operations that everyone wants to do
- When they become popular enough, someone writes a good set of code
- The new set of code becomes a library
- The use of a library means the software development can be done more rapidly
- Once a library exists, anyone can use it
- It must be installed in a particular place on the laptop
- See <https://www.arduino.cc/en/Reference/Libraries> for common libraries
- A number of libraries come installed with the IDE, but you can also download or create your own.

Some of the Standard Libraries

- EEPROM
 - reading and writing to "permanent" storage
- Ethernet
 - for connecting to the internet using the Arduino Ethernet Shield, Arduino Ethernet Shield 2 and Arduino Leonardo ETH
- LiquidCrystal
 - for controlling liquid crystal displays (LCDs)
- SD
 - for reading and writing SD cards
- Servo
 - for controlling servo motors
- SPI
 - for communicating with devices using the Serial Peripheral Interface (SPI) Bus

Some of the Standard Libraries

- SoftwareSerial
 - for serial communication on any digital pins.
- Stepper
 - for controlling stepper motors
- TFT
 - for drawing text , images, and shapes on the Arduino TFT screen
- WiFi
 - for connecting to the internet using the Arduino WiFi shield
- Wire
 - Two Wire Interface (TWI/I2C) for sending and receiving data over a net of devices or sensors.

Some other useful libraries

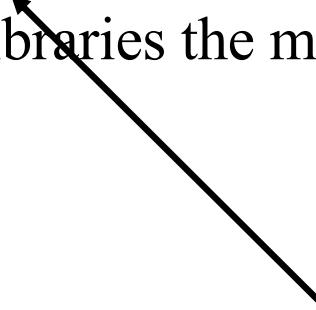
- AudioFrequencyMeter
 - Sample an audio signal and get its frequency back
- AudioZero
- RTC
- ArduinoSound
 - Play audio files from a SD card
 - Real Time Clock to schedule event
 - A simple way to play and analyze audio data

Some other useful libraries

- X10
 - Sending X10 signals over AC power lines
- XBee
 - for communicating with XBee modules in API mode
- FFT
 - frequency analysis of audio or other analog signals
- Tone
 - generate audio frequency square waves in the background on any microcontroller pin

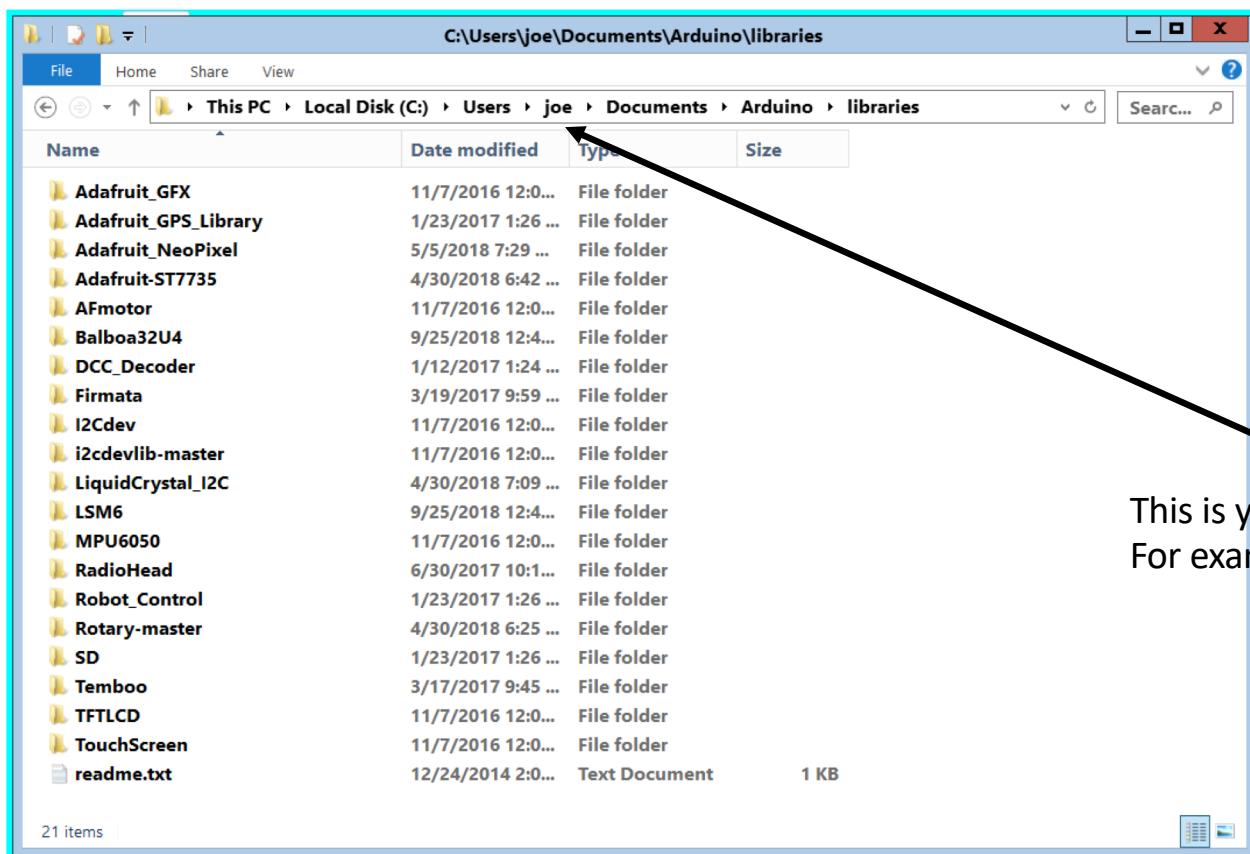
Installing Libraries

- Each library is installed in a different way
- Read the information about the library for directions to install it
- For Windows, libraries are installed in:
 - C:\Users/user_name\Documents\Arduino\libraries
- When we need libraries the method to install them will be shown



This is your login name
For example on Joe's computer it is joe

Example on Joe's Windows 11 laptop



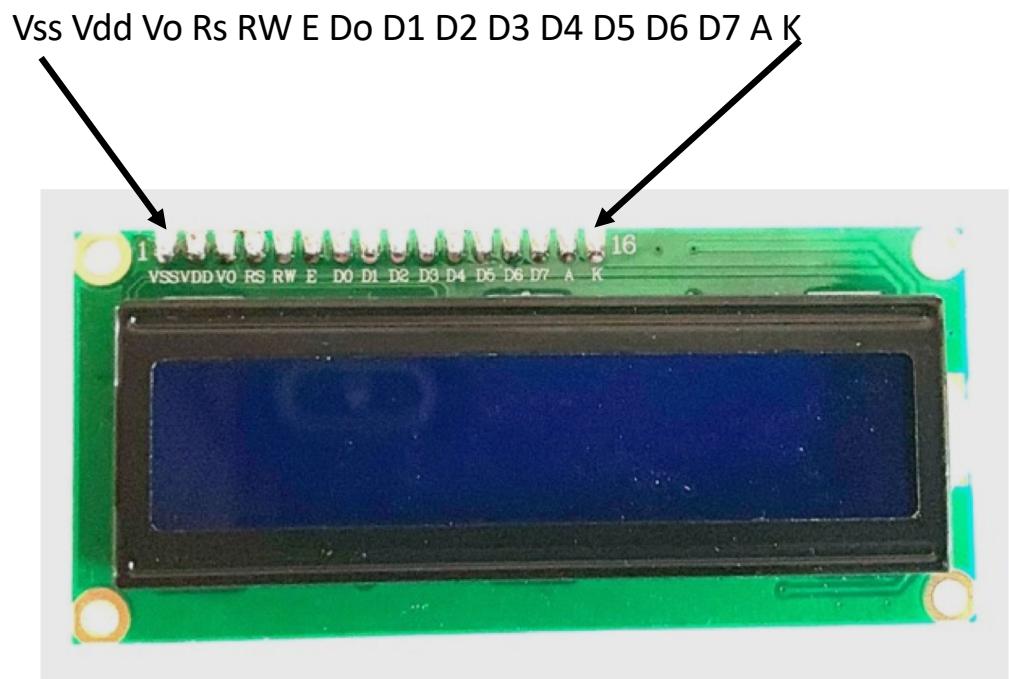
Another Example: LCD Display

- LCD displays are popular
- We will show how to use a simple LCD display and
- Use the LCD library

Connections between LCD and Arduino

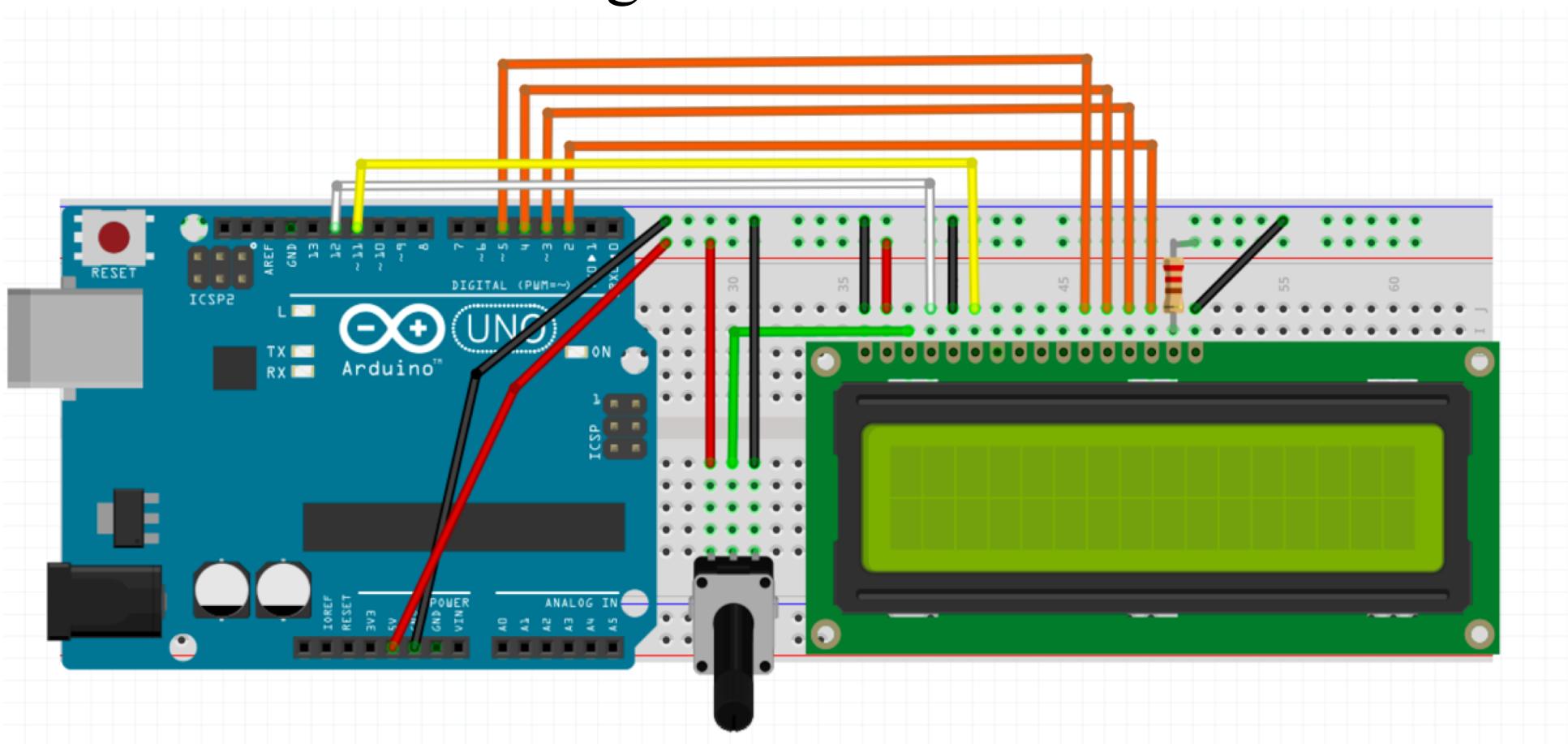
LCD1602 Arduino

VSS	->	GND
VDD	->	+5V
VO	->	10K Potentiometer
RS	->	D12
RW	->	GND
E	->	D11
D0	->	no connection
D1	->	no connection
D2	->	no connection
D3	->	no connection
D4	->	D5
D5	->	D4
D6	->	D3
D7	->	D2
A	->	330Ω
K	->	GND

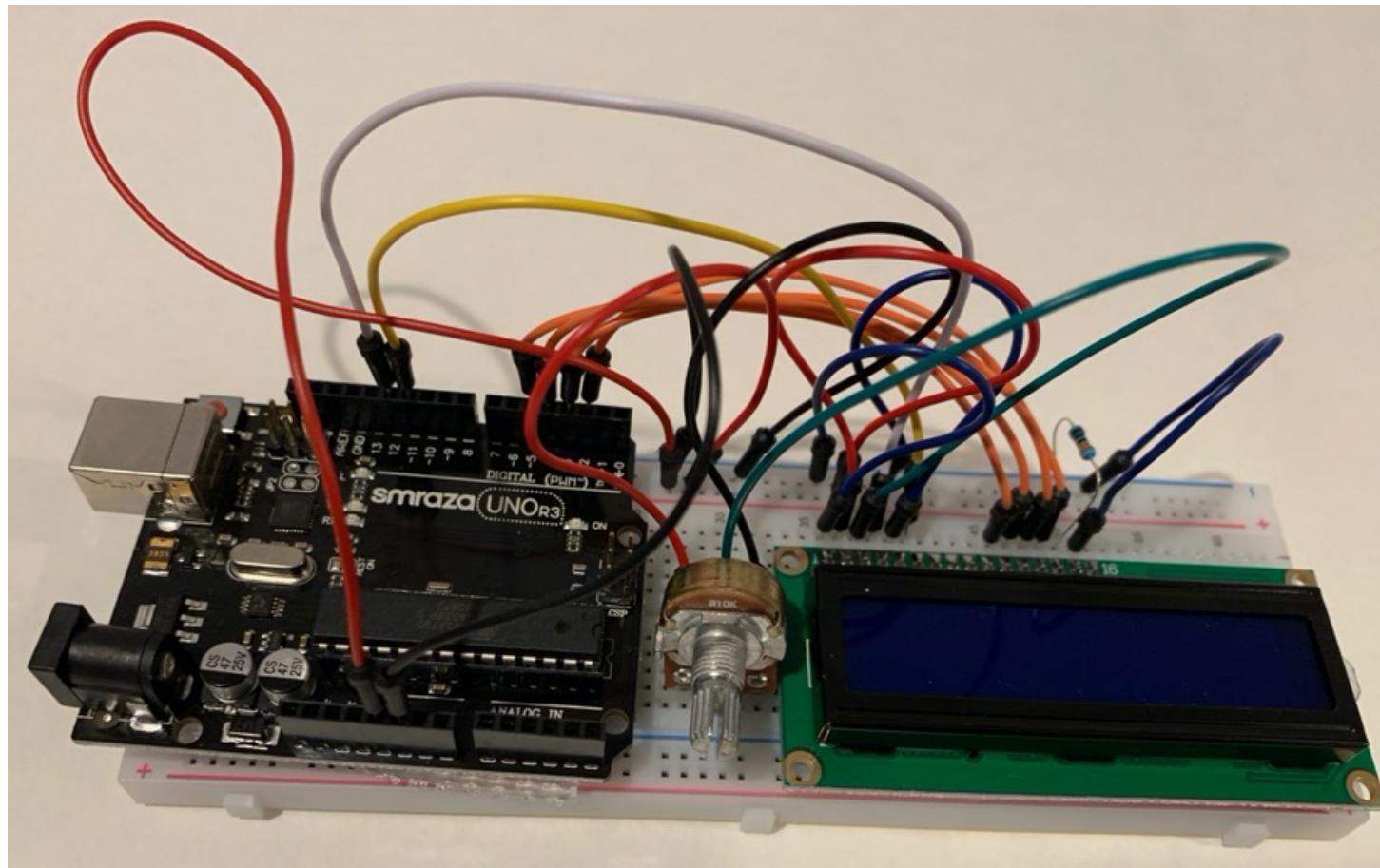


15 character by 2 line display

One View of Wiring



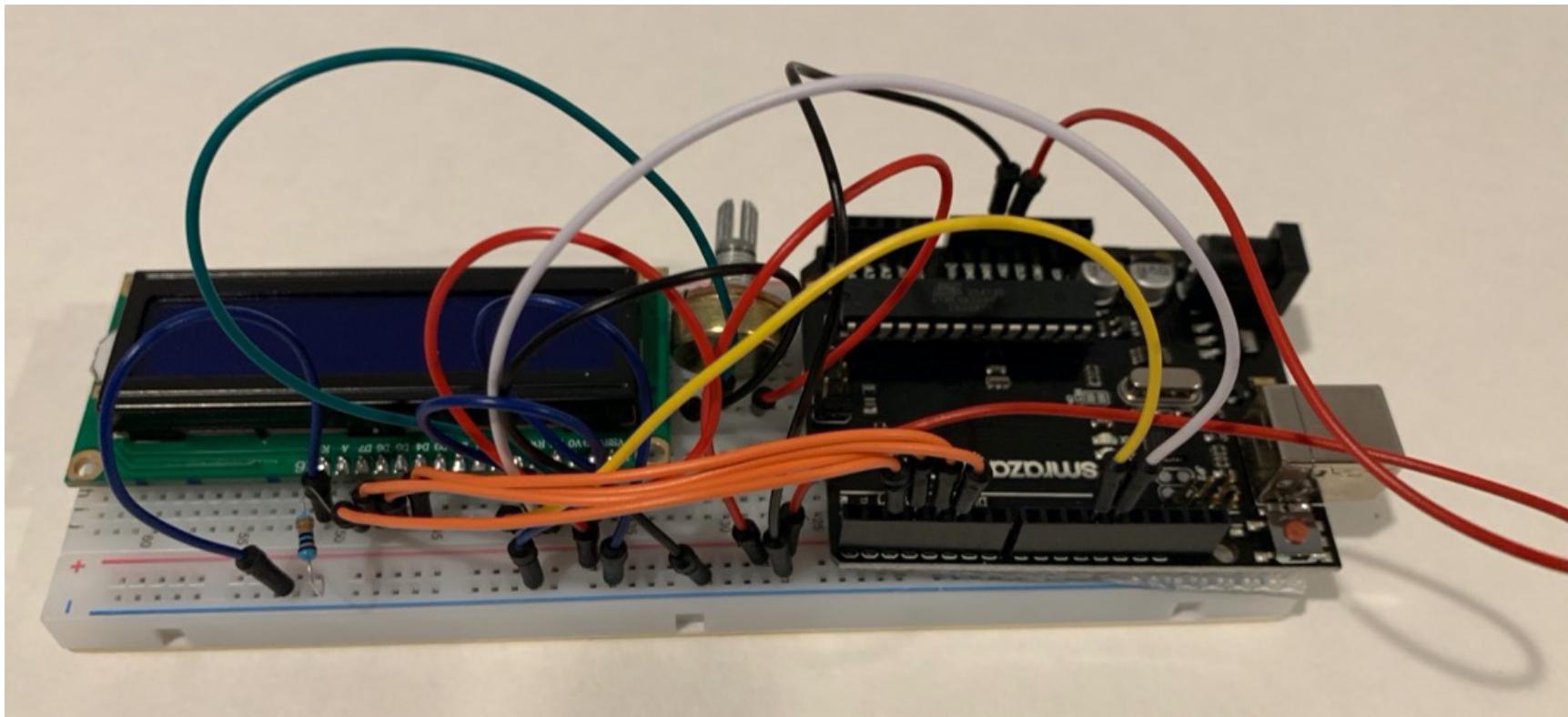
LCD display experimental setup



Copyright 2023 Joseph E. Wilkes

28

LCD display experimental setup (another view)

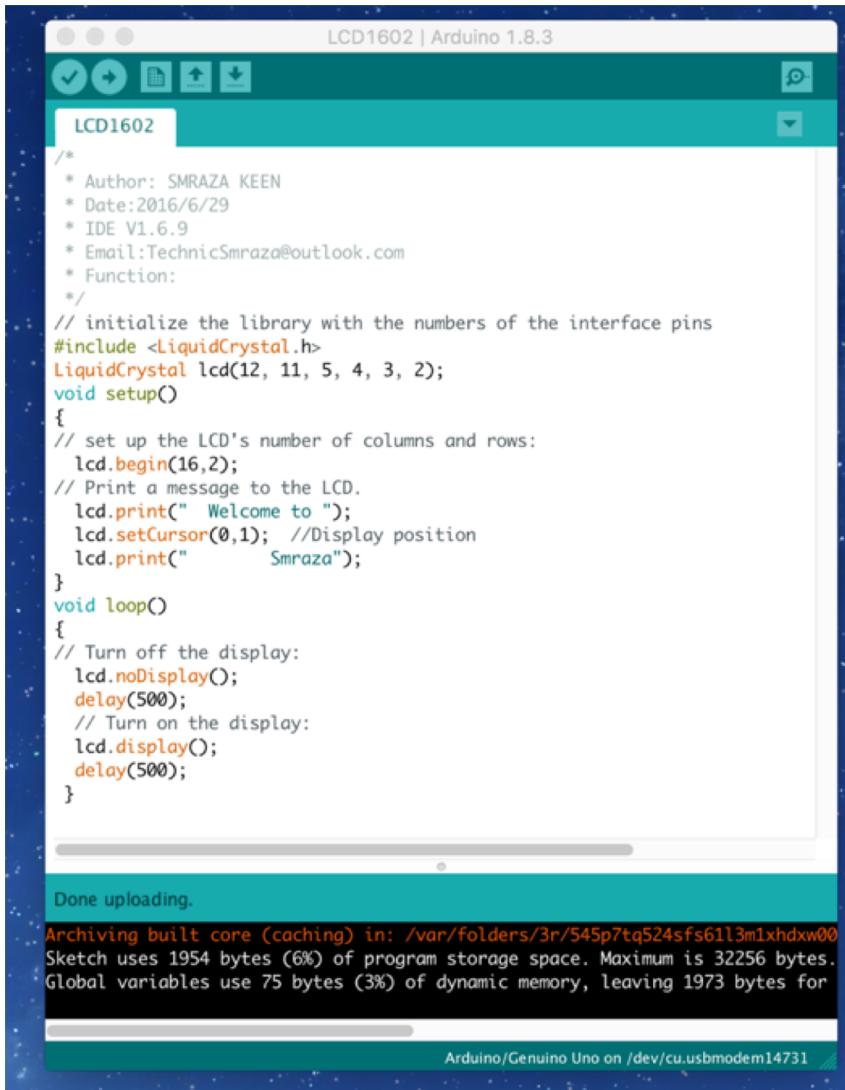


Liquid Crystal Library

- Go to <https://playground.arduino.cc/Main/LiquidCrystal>
- Download: LiquidCrystal_1.zip
- Install library

LCD Display

- We import some libraries
- We define some constants



The screenshot shows the Arduino IDE interface with a sketch titled "LCD1602". The code initializes the LiquidCrystal library, sets up the LCD with 16 columns and 2 rows, and prints a welcome message. It then turns off and on the display in a loop. The status bar at the bottom indicates the sketch is archiving to a folder and provides memory usage details.

```
/*
 * Author: SMRAZA KEEN
 * Date:2016/6/29
 * IDE V1.6.9
 * Email:TechnicSmraza@outlook.com
 * Function:
 */
// initialize the library with the numbers of the interface pins
#include <LiquidCrystal.h>
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
void setup()
{
    // set up the LCD's number of columns and rows:
    lcd.begin(16,2);
    // Print a message to the LCD.
    lcd.print(" Welcome to ");
    lcd.setCursor(0,1); //Display position
    lcd.print(" Smraza");
}
void loop()
{
    // Turn off the display:
    lcd.noDisplay();
    delay(500);
    // Turn on the display:
    lcd.display();
    delay(500);
}

Done uploading.
Archiving built core (caching) in: /var/folders/3r/545p7tq524sfs61l3m1xhdw00
Sketch uses 1954 bytes (6%) of program storage space. Maximum is 32256 bytes.
Global variables use 75 bytes (3%) of dynamic memory, leaving 1973 bytes for
```

Sample Output of LCD



Questions ?

Homework Week 2

- Setup your development environment, if not done previously
- Look at www.arduino.cc
- Explore github
- Look at changing what is displayed on the LCD display