**Designing with Arduino**

Claremont Splash 2017

Jane Wu, Aom Pongpiriyakarn

The goal of this class is to introduce to you a few components commonly found in DIY electronics, and walk you through your very own Arduino keypad project!

Materials: Arduino Uno, HXT900 9GR Micro Servo, 4x4 Keypad, LEDs

**Hello World**

void setup()

{

Serial.begin(9600); //If you want to use the serial port

Serial.println(“Hello World!”);

}

void loop()

{}

**Servo Control**

#include <Servo.h>

// Pins

#define SERVOPIN 6

Servo servo;

void setup()

{

Serial.begin(9600);

pinMode(SERVOPIN, OUTPUT);

servo.attach(SERVOPIN);

servo.write(0);

delay(2000); // 2000 milliseconds

}

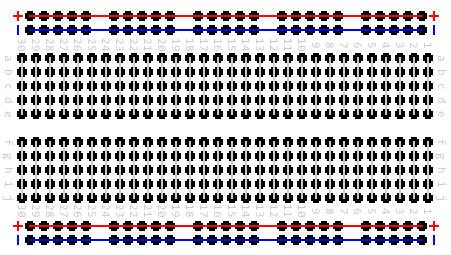
void loop()

{}

**Challenge:** Move your servo to a random position every 2 seconds.

**Breadboards**

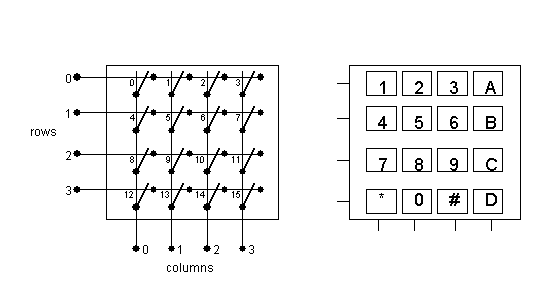
Breadboards are a staple of any DIY electronics project! While they may look complicated, breadboards are just a collection of connected holes.



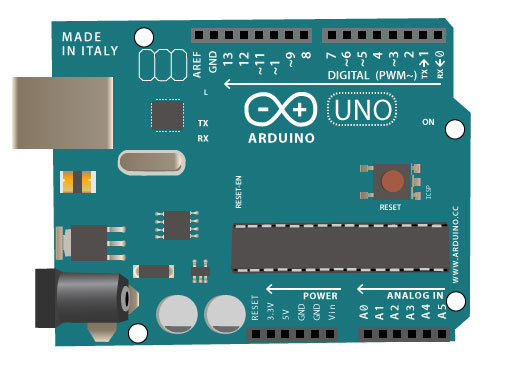
Rows are connected, but NOT across the dotted line

Columns are connected, but NOT holes in the same row

**Keypads – How Do They Work?**

****

Ever wondered how your microwave keypad knows which keys you’re pressing? The process is more than meets the eye. Rather than using separate wires for each individual button, standard keypads use a row and column system (shown above). The Arduino knows when a key is pressed because a key press causes a specific row and column to become connected.



**Connecting Your Keypad!**

To connect your keypad to the Arduino, use wires to connect each of the row and column pins to the DIGITAL pins on the Arduino Uno board.

**Keypad Lock Project**

**Setting Up**

#include <Keypad.h>

// Keypad pins, Left (\*) to Right (D)

// R1 R2 C0 R3 C1 C2 C3 R0

#define R0 2

#define R1 12

#define R2 11

#define R3 9

#define C0 10

#define C1 5

#define C2 4

#define C3 3

// Initialize keypad

const byte rows = 4;

const byte cols = 4;

char keys[rows][cols] = {

{'1','2','3', 'A'},

{'4','5','6', 'B'},

{'7','8','9', 'C'},

{'\*','0','#', 'D'}

};

byte rowPins[rows] = {R0,R1,R2,R3}; // define rows

byte colPins[cols] = {C0,C1,C2,C3}; // define columns

Keypad keypad =

Keypad(makeKeymap(keys), rowPins, colPins, rows, cols);

void setup() {

Serial.begin(9600);

}

void loop() {

// Get the current key being pressed

char key = keypad.getKey();

// If no key is being pressed

if(key != NO\_KEY) {

Serial.println(key);

}

}

**Tasks**

1. Pick a **secret** passcode! It can consist of any valid keypad characters.
2. Store your passcode as an array of characters.
3. Write a function that will check whether the correct passcode has been entered.
4. When the passcode is correctly entered, print “Success!” in the Serial port.

**Pseudocode**

void checkPassCode() {

// Replace with YOUR secret code

char secretCode[4] = [‘A’, ‘B’, ‘C’, ‘D’]

char key = the current key being pressed

int index = index of character we’re looking for

int isLocked = 1 // The door is currently locked

if (We already unlocked the door OR no key is pressed) {

return;

}

if (We are pressing the correct key) {

Add 1 to index

if (We’re at the last character) {

Print “Success!”

}

}

otherwise {

Reset index to 0

}

}

**Add Some Servo Action!**

To finish the lock, add the following code INSIDE the “loop()” function:

// If the code is correct

if(isLocked == 0) {

// INSERT YOUR SERVO CODE HERE

isLocked = 1;

}