

At last Thursdays meeting Frank put out a challenge to have a LED blink at about once per second, have the arduino play a continuous 1Khz sound, and have a button that will turn on another LED.

The random factor would be when you hit the Button, and it should be an instant on.

I took the challenge up, and made a program that works almost like Frank wanted - it's not quite what he asked for, but it is close and with a little more tweeking and playing around with can get very close to what he asked.

<http://youtu.be/Pv0MFDkctBI>

So, it blinks the LED at about 2 seconds per, plays "twinkle twinkle little star" pretty close to being right, and will almost instantly turn on the LED if the button is on - It takes it a second or so to realize the button has been released. And one of the strange things that I wasn't aware of was that while the interrupt is triggered it does stop counting millis value. This will end up causing some issues the longer the sketch runs.

The sketch can be found here:

<https://codebender.cc/sketch:61474>

The sketch works more or less how we talked about at the meeting.

```
int speakerPin = 11;
int ledPin = 8;
int length = 15; // the number of notes
char notes[] = "ccggaagffeeddc "; // a space represents a rest
int beats[] = { 1, 1, 1, 1, 1, 1, 2, 1, 1, 1, 1, 1, 1, 2, 4 };
int tempo = 300;
int ledState = LOW;
long previousMillis = 0;
long interval = 900;
int inruptpin = 9;
volatile int state = LOW;
```

The above of course sets up variables, the long interval variable is the one set for timing of the blinking led.

Below sets up plays the tune, it's just a PWM and a slight delay between notes, the tune is played from two different routines: "playTone" and "playNote"

```
void playTone(int tone, int duration) {
  for (long i = 0; i < duration * 1000L; i += tone * 2) {
    digitalWrite(speakerPin, HIGH);
```

```

        delayMicroseconds(tone);
        digitalWrite(speakerPin, LOW);
        delayMicroseconds(tone);
    }
}

void playNote(char note, int duration) {
    char names[] = { 'c', 'd', 'e', 'f', 'g', 'a', 'b', 'C' };
    int tones[] = { 1915, 1700, 1519, 1432, 1275, 1136, 1014, 956 };

    // play the tone corresponding to the note name
    for (int i = 0; i < 8; i++) {
        if (names[i] == note) {
            playTone(tones[i], duration);
        }
    }
}

```

This is to blink the LED and is called when the interrupt is triggered. The interrupt jumps from the programs current location anytime the trigger happens. The best method is to keep this subroutine very small - it can be larger than what I have, but it needs to be able to execute quickly.

```

void blink() {
    state = !state;
    digitalWrite(intruptpin, state);
}

```

More setup stuff - what is important here though is setting up the Interrupt - the UNO has 2 that can be used, 0 is on Pin 2 and 1 is on Pin 3 - <http://arduino.cc/en/Reference/attachInterrupt>

```

void setup() {
    pinMode(speakerPin, OUTPUT);
    pinMode(ledPin, OUTPUT);
    pinMode(intruptpin, OUTPUT);
    attachInterrupt(0, blink, HIGH);
}

```

This is where the magic happens.

First we set a variable called currentMillis to equal millis();

then a for loop will start to play the notes, also inside the for loop, we check if currentMillis is greater the interval, and if so we setup previousMillis to equal currentMillis, check the led state, and change the LED

```
void loop() {
  unsigned long currentMillis = millis();
  for (int i = 0; i < length; i++) {
    if (notes[i] == ' ') {
      delay(beats[i] * tempo); // rest
    } else {
      playNote(notes[i], beats[i] * tempo);
    }

    // pause between notes
    if (currentMillis - previousMillis > interval) { previousMillis = currentMillis;
      if (ledState == LOW) {ledState = HIGH;} else {ledState = LOW;}
      digitalWrite (ledPin, ledState);
    }
  }
}
```

So, the sketch is small, it run quickly, and it works pretty well.

CHEERLIGHTS!

For those that don't know about cheerlights -

CheerLights is an [ioBridge Labs](#) project that allows people's lights all across the world to synchronize, stay linked based on social networking trends. It's a way to connect physical things with social networking experiences and spread cheer at the same time.

Cheerlights is used alot around the holidays - I first knew about this 3 or 4 years ago, and it was around Christmas Time - the idea is neat, you post on twitter with a hash tag with "Cheerlights" and a color.

The colors that can be used: "red", "green", "blue", "cyan", "white", "warmwhite", "purple", "magenta", "yellow", "orange", "pink" and "oldlace" (a type of warmwhite) of course not all devices can display all colors. or all shades of color. Warm White and White look pretty much the same with my Neopixels.

There a lot of different devices and things that are all using cheerlights. But this is the 1st I've seen someone take a ethernet shield, and neopixles -

I based my sketch from this:

<https://github.com/bkonosky/Cheerlights-Arduino-Ethernet-ShiftBrite>

<http://www.cheerlights.com/post/14571330320/arduino-ethernet-shiftbrite-led-cheerlights>

and this:

<http://n0hio.wordpress.com/2013/12/06/cheerlights-project/>

Bkonosky is using something called a Shiftbrite LED, I've not heard of these, and it looks like they have been retired from sales, but here is the page I found.

<https://www.sparkfun.com/products/retired/10075>

N0hio - (Ham Radio guy) - Is using Neopixles, but he is controlling the Arduino over serial with a Raspberry PI and python script.

For those that don't know what Neopixels are, NeoPixel is an Adafruit product brand. and are digitally controlled RGB Leds - the Uno can control about 30 or so before you need to use a larger power supply, thou that does depend on how bright you make them, and if they are all turned on at the same time.

<http://www.adafruit.com/category/168>

Neopixels use 3 wires (4 if you have more then one pixel) They have a 5v VCC, Ground, Data IN and Data Out, if you only have one Pixel you only need 3 wires, nice!

The LEDs have a WS2811 or WS2812 chip built into them - and if you search eBay for WS2812 you'll find a lot of them cheap from China - You will also find some of the Adafruit products, they are reasonably priced for the most part.

I paid just under 9 dollars for my strip of 30 LEDs.

My code can be found here:

<https://codebender.cc/sketch:61632>

```
#include <SPI.h>
```

```
#include <Ethernet.h>
```

```
#include <Adafruit_NeoPixel.h>
```

```
// Local Network Settings
```

```
byte mac[] = { 0xD4, 0x28, 0xB2, 0xFF, 0x4C, 0x84 }; // Must be unique on local network
```

```
#define thingSpeakInterval 16000 // Time interval in milliseconds to get data from ThingSpeak  
(number of seconds * 1000 = interval)
```

```
// Variable Setup
```

```
long lastConnectionTime = 0;
```

```
String lastCommandString = "black";
boolean lastConnected = false;
int failedCounter = 0;

Adafruit_NeoPixel strip = Adafruit_NeoPixel(10, 6, NEO_GRB + NEO_KHZ800);

// Initialize Arduino Ethernet Client
EthernetClient client;

void setup() {
  delay(100);
  // Setup Serial
  Serial.begin(9600);
  delay(100);
  Serial.flush();
  delay(100);

  strip.begin();
  strip.show();

  // Start Ethernet on Arduino
  startEthernet();
}

void loop() {
  // Process CheerLights Commands
  if(client.available() > 0)
  {
    delay(100);
    Serial.println(client.available());
    String response;
    char charIn;
    do {
      charIn = client.read(); // read a char from the buffer
      response += charIn; // append that char to the string response
    } while (client.available() > 0);
    Serial.println(response.length());
    Serial.println(response);

    if (response.indexOf("white") > 0 || response.indexOf("warmwhite") > 0 ||
        response.indexOf("oldlace") > 0)
    {
      lastCommandString = "white";
      colorWipe(strip.Color(255,255,255),50);
    }
  }
}
```

```
}  
else if (response.indexOf("black") > 0 || response.indexOf("off") > 0)  
{  
lastCommandString = "black";  
colorWipe(strip.Color(0,0,0),50);  
}  
else if (response.indexOf("red") > 0)  
{  
lastCommandString = "red";  
colorWipe(strip.Color(255,0,0),50);  
}  
else if (response.indexOf("green") > 0)  
{  
lastCommandString = "green";  
colorWipe(strip.Color(0, 255, 0), 50);  
}  
else if (response.indexOf("blue") > 0)  
{  
lastCommandString = "blue";  
colorWipe(strip.Color(0,0,255),50);  
}  
else if (response.indexOf("cyan") > 0)  
{  
lastCommandString = "cyan";  
colorWipe(strip.Color(0,255,255),50);  
}  
else if (response.indexOf("magenta") > 0)  
{  
lastCommandString = "magenta";  
colorWipe(strip.Color(255,0,255),50);  
}  
else if (response.indexOf("yellow") > 0)  
{  
lastCommandString = "yellow";  
colorWipe(strip.Color(255,255,0),50);  
}  
else if (response.indexOf("purple") > 0)  
{  
lastCommandString = "purple";  
colorWipe(strip.Color(102,51,204),50);  
}  
else if (response.indexOf("orange") > 0)  
{  
lastCommandString = "orange";
```

```

colorWipe(strip.Color(255,153,0),50);
}
else if (response.indexOf("pink") > 0)
{
    lastCommandString = "pink";
    colorWipe(strip.Color(255,53,153),50);
}
/*else if (response.indexOf("warmwhite") > 0)
{
    lastCommandString = "warmwhite";
    fadeToColor(lastCommand,WARMWHITE,stepdelay);
    for (int i = 0; i < 3; i++) {
        lastCommand[i] = WARMWHITE[i];
    }
}
else if (response.indexOf("black") > 0)
{
    lastCommandString = "black";
    fadeToColor(lastCommand,BLACK,stepdelay);
    for (int i = 0; i < 3; i++) {
        lastCommand[i] = BLACK[i];
    }
}*/
else
{
    lastCommandString = "(no match)";
}
// Echo command
delay(200);
Serial.print("CheerLight Command Received: ");
Serial.println(lastCommandString);
delay(200);
}
// Disconnect from ThingSpeak
if (!client.connected() && lastConnected)
{
    Serial.println("...disconnected");
    client.stop();
}
// Subscribe to ThingSpeak Channel and Field
if(!client.connected() && (millis() - lastConnectionTime > thingSpeakInterval))
{
    subscribeToThingSpeak();
}

```

```

// Check if Arduino Ethernet needs to be restarted
if (failedCounter > 3 ) {startEthernet();}
lastConnected = client.connected();
delay(100);
} // End loop
void subscribeToThingSpeak()
{
if (client.connect("api.thingspeak.com", 80))
{
Serial.println("Connecting to ThingSpeak...");
failedCounter = 0;
Serial.println("Sending Request");
client.println("GET /channels/1417/field/1/last.txt HTTP/1.0");
client.println();
lastConnectionTime = millis();
}
else
{
failedCounter++;
Serial.println("Connection to ThingSpeak Failed (" +String(failedCounter, DEC)+")");
Serial.println();
lastConnectionTime = millis();
}
}
void startEthernet()
{
client.stop();
Serial.println("Connecting Arduino to network...");
Serial.println();
delay(1000);
// Connect to network and obtain an IP address using DHCP
if (Ethernet.begin(mac) == 0)
{
Serial.println("DHCP Failed, reset Arduino to try again");
Serial.println();
}
else
{
Serial.println("Arduino connected to network using DHCP");
Serial.println();
}
delay(1000);
}

```



```
void colorWipe(uint32_t c, uint8_t wait) {  
  for(uint16_t i=0; i<strip.numPixels(); i++) {  
    strip.setPixelColor(i, c);  
    strip.show();  
    delay(wait);  
  }  
  
}
```

I'll have a video of the Neopixels Cheerlights in action posted shortly.