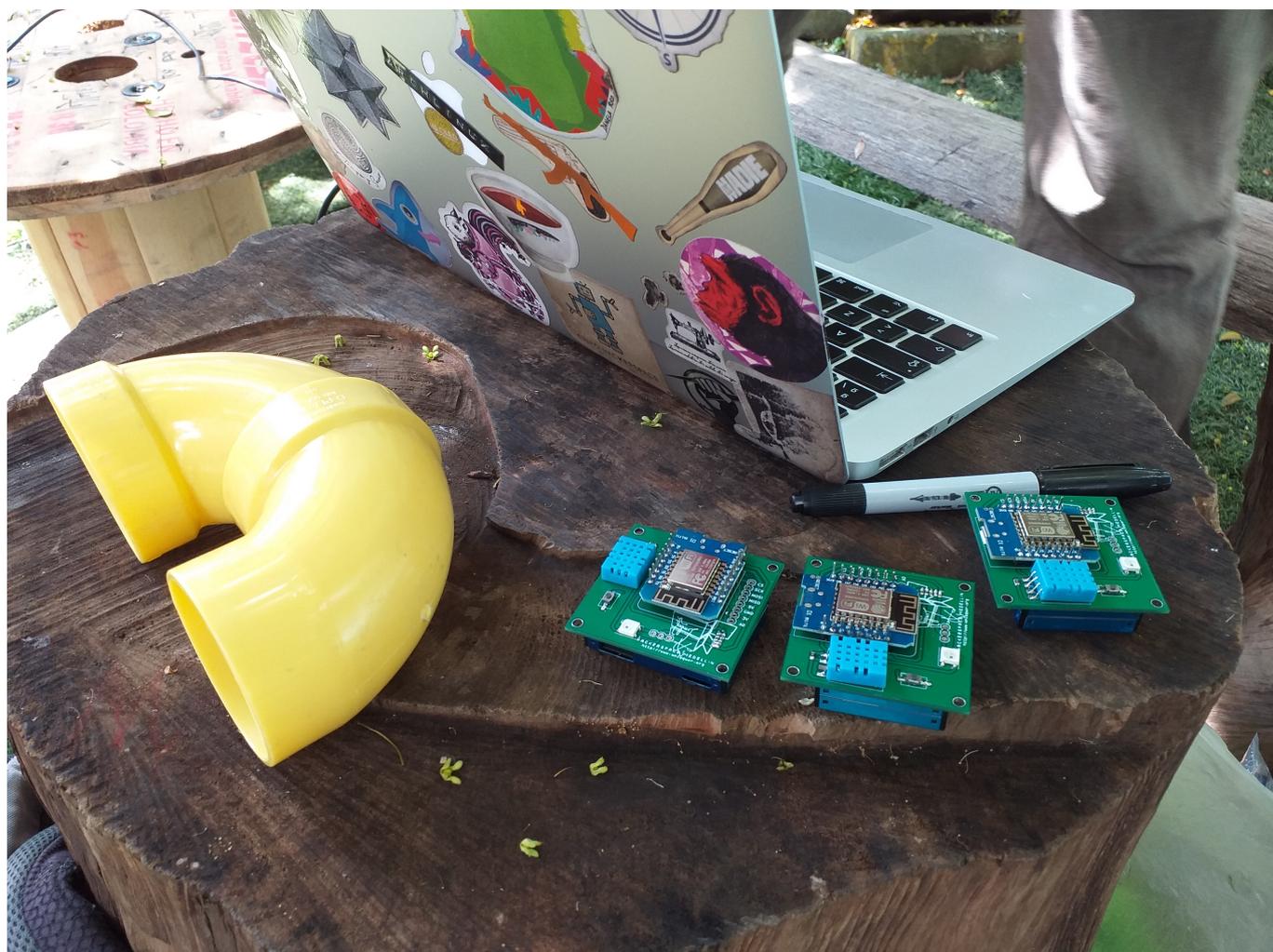# AQA - Taller Vestuario

Prendas de vestuario + AQA

Bitácora

2019-10-17

- Instalación de sensores para proceso de calibración

- [Visualización de los datos de los sensores instalados](#)

Referentes

- https://twitter.com/arturo182/status/1162762166107353089?s=20

Ideas y materiales que se pueden usar para el taller

- [instructable](#)
- [adrafruit](#)
- [Enlace externo](#)
- [Enlace externo](#)

prendas

[Falda led](#)

Iluminación

[Como conectar cinta de leds](#)

código incial para hacer una animación en la matrix de leds

```cpp
#include<FastLED.h>
#define LED_PIN     D3
#define LED_TYPE    WS2812B
#define COLOR_ORDER GRB
#define f false
#define t true


const uint8_t kMatrixWidth  = 8;
const uint8_t kMatrixHeight = 8;
#define NUM_LEDS (kMatrixWidth * kMatrixHeight)

int BRIGHTNESS = 60;   // this is half brightness
CRGB leds[kMatrixWidth * kMatrixHeight];

#define amarillo CRGB::Yellow
#define black CRGB::Black
#define rojo CRGB::Red

int loop_cnt = 0;
uint16_t speed = 20;
static uint16_t x;
static uint16_t y;
static uint16_t z;
uint16_t scale = 31;
uint8_t noise[kMatrixWidth][kMatrixHeight];

// Fill the x/y array of 8-bit noise values using the inoise8 function.
/*
void fillnoise8() {
  for(int i = 0; i < kMatrixWidth; i++) {
    int ioffset = scale * i;
    for(int j = 0; j < kMatrixHeight; j++) {
      int joffset = scale * j;
      noise[i][j] = inoise8(x + ioffset,y + joffset,z);
    }
  }
  z += speed;
}
*/
void setup() {
  LEDS.addLeds<LED_TYPE,LED_PIN,COLOR_ORDER>(leds,NUM_LEDS);
  FastLED.setBrightness(BRIGHTNESS);
```

```cpp
  // Initialize our coordinates to some random values
  x = random16();
  y = random16();
  z = random16();
}

#define ESCENAS 8

CRGB matrix[ESCENAS][8][8] = {
  {
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
CRGB::Green, CRGB::Green,CRGB::Green},
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
CRGB::Green, CRGB::Green,CRGB::Green},
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
CRGB::Green, CRGB::Green,CRGB::Green},
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
CRGB::Green, CRGB::Green,CRGB::Green},
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
CRGB::Green, CRGB::Green,CRGB::Green},
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
CRGB::Green, CRGB::Green,CRGB::Green},
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
CRGB::Green, CRGB::Green,CRGB::Green},
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
CRGB::Green, CRGB::Green,CRGB::Green},
  },
  {
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
CRGB::Green, CRGB::Green,CRGB::Green},
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
CRGB::Green, CRGB::Green,CRGB::Green},
    {CRGB::Green, CRGB::Black, CRGB::Black,CRGB::Black, CRGB::Black,
CRGB::Black, CRGB::Black,CRGB::Green},
    {CRGB::Green, CRGB::Black, CRGB::Black,CRGB::Black, CRGB::Black,
CRGB::Black, CRGB::Black,CRGB::Green},
    {CRGB::Green, CRGB::Black, CRGB::Black,CRGB::Black, CRGB::Black,
CRGB::Black, CRGB::Black,CRGB::Green},
    {CRGB::Green, CRGB::Black, CRGB::Black,CRGB::Black, CRGB::Black,
CRGB::Black, CRGB::Black,CRGB::Green},
    {CRGB::Green, CRGB::Black, CRGB::Black,CRGB::Black, CRGB::Black,
CRGB::Black, CRGB::Black,CRGB::Green},
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
CRGB::Green, CRGB::Green,CRGB::Green},
  },
  {
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
CRGB::Green, CRGB::Green,CRGB::Green},
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
CRGB::Green, CRGB::Green,CRGB::Green},
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
```

```cpp
CRGB::Green, CRGB::Green,CRGB::Green},
    {CRGB::Green, CRGB::Black, CRGB::Black,CRGB::Black, CRGB::Black,
CRGB::Black, CRGB::Black,CRGB::Green},
    {CRGB::Green, CRGB::Black, CRGB::Black,CRGB::Black, CRGB::Black,
CRGB::Black, CRGB::Black,CRGB::Green},
    {CRGB::Green, CRGB::Black, CRGB::Black,CRGB::Black, CRGB::Black,
CRGB::Black, CRGB::Black,CRGB::Green},
    {CRGB::Green, CRGB::Black, CRGB::Black,CRGB::Black, CRGB::Black,
CRGB::Black, CRGB::Black,CRGB::Green},
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
CRGB::Green, CRGB::Green,CRGB::Green},
  },
  {
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
CRGB::Green, CRGB::Green,CRGB::Green},
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
CRGB::Green, CRGB::Green,CRGB::Green},
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
CRGB::Green, CRGB::Green,CRGB::Green},
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
CRGB::Green, CRGB::Green,CRGB::Green},
    {CRGB::Green, CRGB::Black, CRGB::Black,CRGB::Black, CRGB::Black,
CRGB::Black, CRGB::Black,CRGB::Green},
    {CRGB::Green, CRGB::Black, CRGB::Black,CRGB::Black, CRGB::Black,
CRGB::Black, CRGB::Black,CRGB::Green},
    {CRGB::Green, CRGB::Black, CRGB::Black,CRGB::Black, CRGB::Black,
CRGB::Black, CRGB::Black,CRGB::Green},
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
CRGB::Green, CRGB::Green,CRGB::Green},
  },
  {
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
CRGB::Green, CRGB::Green,CRGB::Green},
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
CRGB::Green, CRGB::Green,CRGB::Green},
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
CRGB::Green, CRGB::Green,CRGB::Green},
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
CRGB::Green, CRGB::Green,CRGB::Green},
    {CRGB::Green, CRGB::Black, CRGB::Black,CRGB::Black, CRGB::Black,
CRGB::Black, CRGB::Black,CRGB::Green},
    {CRGB::Green, CRGB::Black, CRGB::Black,CRGB::Black, CRGB::Black,
CRGB::Black, CRGB::Black,CRGB::Green},
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
CRGB::Green, CRGB::Green,CRGB::Green},
  },
  {
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
CRGB::Green, CRGB::Green,CRGB::Green},
```

```cpp
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
CRGB::Green, CRGB::Green,CRGB::Green},
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
CRGB::Green, CRGB::Green,CRGB::Green},
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
CRGB::Green, CRGB::Green,CRGB::Green},
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
CRGB::Green, CRGB::Green,CRGB::Green},
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
CRGB::Green, CRGB::Green,CRGB::Green},
    {CRGB::Green, CRGB::Black, CRGB::Black,CRGB::Black, CRGB::Black,
CRGB::Black, CRGB::Black,CRGB::Green},
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
CRGB::Green, CRGB::Green,CRGB::Green},
  },
  {
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
CRGB::Green, CRGB::Green,CRGB::Green},
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
CRGB::Green, CRGB::Green,CRGB::Green},
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
CRGB::Green, CRGB::Green,CRGB::Green},
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
CRGB::Green, CRGB::Green,CRGB::Green},
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
CRGB::Green, CRGB::Green,CRGB::Green},
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
CRGB::Green, CRGB::Green,CRGB::Green},
    {CRGB::Green, CRGB::Green, CRGB::Green,CRGB::Green, CRGB::Green,
CRGB::Green, CRGB::Green,CRGB::Green},
  },
  {
    {CRGB::Black, CRGB::Black, CRGB::Black,CRGB::Black, CRGB::Black,
CRGB::Black, CRGB::Black,CRGB::Black},
    {CRGB::Black, CRGB::Black, CRGB::Black,CRGB::Black, CRGB::Black,
CRGB::Black, CRGB::Black,CRGB::Black},
    {CRGB::Black, CRGB::Black, CRGB::Black,CRGB::Black, CRGB::Black,
CRGB::Black, CRGB::Black,CRGB::Black},
    {CRGB::Black, CRGB::Black, CRGB::Black,CRGB::Black, CRGB::Black,
CRGB::Black, CRGB::Black,CRGB::Black},
    {CRGB::Black, CRGB::Black, CRGB::Black,CRGB::Black, CRGB::Black,
CRGB::Black, CRGB::Black,CRGB::Black},
    {CRGB::Black, CRGB::Black, CRGB::Black,CRGB::Black, CRGB::Black,
CRGB::Black, CRGB::Black,CRGB::Black},
    {CRGB::Black, CRGB::Black, CRGB::Black,CRGB::Black, CRGB::Black,
CRGB::Black, CRGB::Black,CRGB::Black},
    {CRGB::Black, CRGB::Black, CRGB::Black,CRGB::Black, CRGB::Black,
CRGB::Black, CRGB::Black,CRGB::Black},
  },
```

```
};

void loop() {
  /*fillnoise8();*/

  for(int i = 0; i< kMatrixHeight; i++) {
    for(int j = 0; j< kMatrixWidth; j++) {
      leds[i*kMatrixWidth + j] = matrix[loop_cnt%ESCENAS][i][j];
    }
  }
  FastLED.show();
  delay(500);
  for(int i = 0; i< kMatrixHeight; i++) {
    for(int j = 0; j<kMatrixWidth; j++) {
      leds[i*kMatrixWidth + j] = CRGB::Black;
    }
  }
  delay(500);
  loop_cnt++;
}
```

Algunas fotografias de la 5 sesión del taller

From:
<https://wiki.unloquer.org/> -

Permanent link:
<https://wiki.unloquer.org/proyectos/vestuario_aqa?rev=1574508985>

Last update: **2019/11/23 11:36**