// two leds stripes on pins 3 and 4 of Seeeduino Lotus

#include <FastLED.h>

#define NUM\_LEDS 30 // number of leds n stripe

#define DATA\_PIN\_1 3 // pin for strip 1

#define DATA\_PIN\_2 2 // pin for stripe 2

#define CLOCK\_PIN 13

// Define the array of leds for each stripe

CRGB leds\_1[NUM\_LEDS];

CRGB leds\_2[NUM\_LEDS];

int H[NUM\_LEDS]; // array for Hue of each LED

int H\_[NUM\_LEDS]; // array for desfasamento cores

int V[NUM\_LEDS]; // array for Value of each LED

int S[NUM\_LEDS]; // array for Saturation of each LED (unused?)

// one distance sensor HCSR04 on pin 7

#include "Ultrasonic.h"

Ultrasonic ultrasonic(7);

float counter;

void setup() {

// sanity check delay - allows reprogramming if accidently blowing power w/leds

delay(2000);

// set up LEDS

// FastLED.addLeds<NEOPIXEL, DATA\_PIN\_1>(leds\_1, NUM\_LEDS); // GRB ordering

// FastLED.addLeds<NEOPIXEL, DATA\_PIN\_2>(leds\_2, NUM\_LEDS);

FastLED.addLeds<WS2811, DATA\_PIN\_1>(leds\_1, NUM\_LEDS); // GRB ordering

FastLED.addLeds<WS2811, DATA\_PIN\_2>(leds\_2, NUM\_LEDS);

// Serial for ultrasonic sensor

Serial.begin(9600);

// value to desfasamento cores

int DESFAS = 16;

for (int k = 0; k < NUM\_LEDS; k++) {

H[k]=(30-(k+1))\*8;

Serial.println(H[k]);

}

for (int k = 0; k < NUM\_LEDS; k++) {

if ( k < (NUM\_LEDS-DESFAS) ){ H\_[k] = H[k+DESFAS]; }

else if ( k >= (NUM\_LEDS-DESFAS) ) {H\_[k] = H[k-(NUM\_LEDS-DESFAS)]; }

Serial.println(H[k]);

}

}

void loop() {

// counter // to engine animations

counter = counter + 1;

if (counter > 2,147,000,000) {counter = 0;}

long Dmax = 65; // length of the interaction distance

long Dmin = 5; // too close to sensor does not work well

// get distance

long D;

D = ultrasonic.MeasureInCentimeters();

// make V luminosity pulsing

//float L = 175 + 75\*abs(sin(((counter)\*6.283)/(D)));

int L = 255;

// we use the D values from 5 to 65!

if (D>Dmax+1 || D<Dmin) {

Serial.println(0);

// set all stripe to black

for( int j = 0; j < NUM\_LEDS; j++) {

V[j] = 0;

}

} else if (D<=Dmax) {

Serial.println(D-5);

for( int j = 0; j < (int)(D-5)/2; j++) {

V[j] = 0;

}

for( int j = (int)(D-5)/2; j < NUM\_LEDS; j++) {

V[j] = L;

}

}

for (int k = 0; k < NUM\_LEDS; k++) {

leds\_1[k] = CHSV(H\_[k],255,V[k]);

leds\_2[k] = CHSV(H\_[k],255,V[k]);

}

FastLED.show();

delay(100);

}