**Trabajo Práctico N4**

**Integrantes: Herrera Ailen , Morata Pablo, Moya Juan Ignacio , Muñoz Zoé.**

**Actividad 1.1**

MOV B,0x0300 ; Con B recorreremos los píxeles de la pantalla

MOV C,0x0333 ; Con A recorreremos cada color

; 7 6 5 4 3 2 1 0

; R R R G G G B B

lazo:

MOVB [B], CL

INC B ;Incrementamos la dirección al siguiente píxel

CMP B,0x0400

JZ fin

IN 0x0006

CMP A, 0x0032

JZ colorAmarillo

IN 0x0006

CMP A, 0x0031

JZ colorAzul

JMP lazo

fin:

HLT

colorAzul:

MOV C, 0x0333

JMP lazo

colorAmarillo:

MOV C, 252

JMP lazo

**Actividad 1.2**

Prints a 16x16 sprite into the visual display

JMP boot

vslDisplay EQU 0x300

sprite:

DB "\xFF\xFF\xFF\xFF\xFF\xC4\xC4\xC4"

DB "\xC4\xC4\xFF\xFF\xFF\xFF\xFF\xFF"

DB "\xFF\xFF\xFF\xFF\xC4\xC4\xC4\xC4"

DB "\xC4\xC4\xC4\xC4\xC4\xFF\xFF\xFF"

DB "\xFF\xFF\xFF\xFF\x8C\x8C\x8C\xF4"

DB "\xF4\x8C\xF4\xFF\xFF\xFF\xFF\xFF"

DB "\xFF\xFF\xFF\x8C\xF4\x8C\xF4\xF4"

DB "\xF4\x8C\xF4\xF4\xF4\xFF\xFF\xFF"

DB "\xFF\xFF\xFF\x8C\xF4\x8C\x8C\xF4"

DB "\xF4\xF4\x8C\xF4\xF4\xF4\xFF\xFF"

DB "\xFF\xFF\xFF\x8C\x8C\xF4\xF4\xF4"

DB "\xF4\x8C\x8C\x8C\x8C\xFF\xFF\xFF"

DB "\xFF\xFF\xFF\xFF\xFF\xF4\xF4\xF4"

DB "\xF4\xF4\xF4\xF4\xFF\xFF\xFF\xFF"

DB "\xFF\xFF\xFF\xFF\x8C\x8C\xC4\x8C"

DB "\x8C\x8C\xFF\xFF\xFF\xFF\xFF\xFF"

DB "\xFF\xFF\xFF\x8C\x8C\x8C\xC4\x8C"

DB "\x8C\xC4\x8C\x8C\x8C\xFF\xFF\xFF"

DB "\xFF\xFF\x8C\x8C\x8C\x8C\xC4\xC4"

DB "\xC4\xC4\x8C\x8C\x8C\x8C\xFF\xFF"

DB "\xFF\xFF\xF4\xF4\x8C\xC4\xF4\xC4"

DB "\xC4\xF4\xC4\x8C\xF4\xF4\xFF\xFF"

DB "\xFF\xFF\xF4\xF4\xF4\xC4\xC4\xC4"

DB "\xC4\xC4\xC4\xF4\xF4\xF4\xFF\xFF"

DB "\xFF\xFF\xF4\xF4\xC4\xC4\xC4\xC4"

DB "\xC4\xC4\xC4\xC4\xF4\xF4\xFF\xFF"

DB "\xFF\xFF\xFF\xFF\xC4\xC4\xC4\xFF"

DB "\xFF\xC4\xC4\xC4\xFF\xFF\xFF\xFF"

DB "\xFF\xFF\xFF\x8C\x8C\x8C\xFF\xFF"

DB "\xFF\xFF\x8C\x8C\x8C\xFF\xFF\xFF"

DB "\xFF\xFF\x8C\x8C\x8C\x8C\xFF\xFF"

DB "\xFF\xFF\x8C\x8C\x8C\x8C\xFF\xFF"

boot:

MOV C, sprite ; C points to the sprite

MOV D, vslDisplay ; D points to the fb

.loop:

MOVB AL, [C]; Print all the pixels

INC C

INC D

CMP A, 0xC4

JZ cambiarVerde

MOVB [D], AL

CMP D, 0x400

JNZ .loop

HLT

cambiarVerde:

MOVB AL, 0x15

MOVB [D], AL

JMP .loop

**Actividad 1.3**

MOV B, 0x2FE

MOV C, 5

MOVB [B], 224

loop:

CMP B, 0x30F

JE excepciones

CMP B, 0x3F0

JE excepciones

CMP B, 0x400

JNC FINAL

MOVB [B], 224

excepciones:

INC B

DEC C

JNZ loop

MOV C, 5

SUB B, 0x4

ADD B, 0x10

CMP B, 0x400

JC loop

FINAL:

HLT

**Actividad 2.1**

int led = 6;

void setup(){

Serial.begin(9600);

pinMode(2, INPUT);

pinMode(3, INPUT);

for(int i = 0; i<8; i = i+1){

pinMode(i, OUTPUT);

}

attachInterrupt(digitalPinToInterrupt(2), interrupcion\_uno, RISING);

attachInterrupt(digitalPinToInterrupt(3), interrupcion\_dos, RISING);

}

void loop(){

digitalWrite(led, HIGH);

delay(800);

digitalWrite(led,LOW);

led = led + 1;

if(led > 13){

led = 6;

}

}

void interrupcion\_uno(){

for (int i = 0; i < 40; i = i + 1) {

digitalWrite(7, HIGH);

delay(50);

digitalWrite(7, LOW);

delay(50);

}

digitalWrite(7,HIGH);

delay(1000);

digitalWrite(7,LOW);

}

void interrupcion\_dos(){

for (int i = 0; i < 40; i = i + 1) {

digitalWrite(12, HIGH);

delay(50);

digitalWrite(12, LOW);

delay(50);

}

digitalWrite(12,HIGH);

delay(1000);

digitalWrite(12,LOW);

}

1\_No, Arduino no permite interrupciones anidadas

2\_La interrupción de mayor prioridad es la número 2

3\_La función delay funciona de manera incorrecta debido a que utiliza interrupciones al ser ejecutada

4 \_si permitiera interrupciones anidadas, la función delay se ejecutaría sin problema alguno

5 \_no, se ejecutará la interrupción número 3 y luego una vez la interrupción dos ya que entra en cola a partir de un bit de pendiente, por lo tanto solo se ejecuta una vez

**Actividad 2.2**

const int Trigger = 5;

int Echo = 4;

bool activacion = false;

void setup() {

Serial.begin(9600);

pinMode(13, OUTPUT);

pinMode(2, INPUT);

pinMode(3, INPUT);

pinMode(Trigger, OUTPUT);

pinMode(Echo, INPUT);

digitalWrite(Trigger, LOW);

for(int i = 4; i<12; i = i+1){

pinMode(i, OUTPUT);

}

attachInterrupt(digitalPinToInterrupt(2), activacion, RISING);

attachInterrupt(digitalPinToInterrupt(3), desactivacion, RISING);

}

void loop()

{

long t;

long d;

digitalWrite(Trigger, HIGH);

delayMicroseconds(10);

digitalWrite(Trigger, LOW);

t = pulseIn(Echo, HIGH);

d = t/58.35;

Serial.print("Distancia: ");

Serial.print(d);

Serial.print("cm");

Serial.println();

if(d < 100 and activado == true){

Serial.println("ALERTA DE INTRUSO");

for(int i = 4; i<12; i = i+1){

digitalWrite(i, HIGH);

}

delay(100);

for(int i = 4; i<12; i = i+1){

digitalWrite(i, LOW);

}

}

delay(800);

}

void activacion(){

activado = true

Serial.print("Alarma activada");

}

void desactivacion(){

activado = false

Serial.print("Alarma desactivada");

}