



PROJECTE FINAL

AMOR DE MADRE

Enric Marquès No



PERQUÈ HE DECIDIT AQUEST PROJECTE?

- ❖ A la nit, quan encenem la llum del menjador per anar al lavabo, molestem a la nostra mare perquè la seva habitació està connectada al menjador per una finestra.
- ❖ La següent diapositiva és un petit planol que ajuda a que us feu una idea del perquè he escollit això i on posar les dos leds i els dos sensors

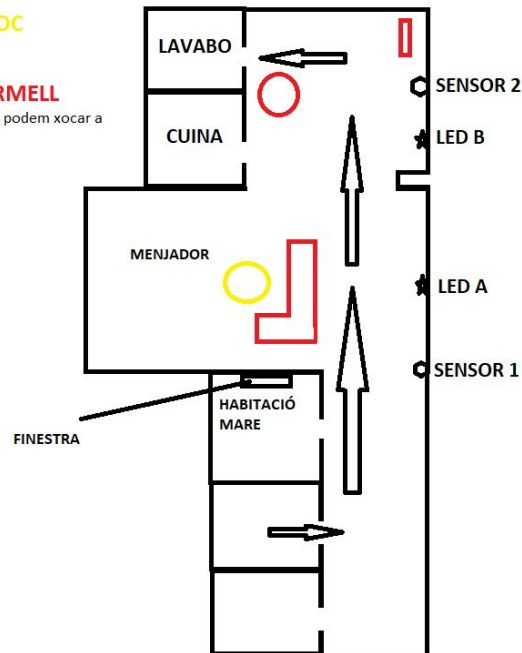
PERQUÈ HE DECIDIT AQUEST PROJECTE?

COLOR GROC

Llum

COLOR VERMELL

Objectes la qual, podem xocar a la nit



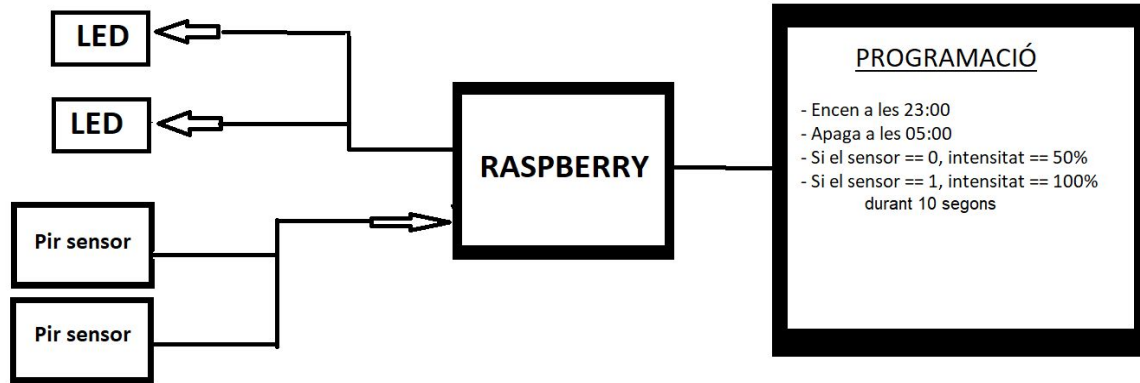
DESCRIPCIÓ

- ❖ El *LED A* i el *LED B* estaran encesos de les 23:00 pm fins a les 05:00 am amb una intensitat del 50%.
- ❖ Com que la Raspberry Pi no té hora, la treuré desde la connexió a internet.
- ❖ Utilitzaré una NodeMCU ESP8266 a un Led i un sensor perquè hi ha una distància de 3 metres entre el Led A i el Led B, i el sensor 1 i sensor 2.
- ❖ Quan algú passi per el sensor 1 o sensor 2, encendrà el *LED A* i el *LED B*. La intensitat canviarà al 100% durant 10 segons.

OBJECTES QUE UTILITZARÉ

- ❖ 2 LEDS
- ❖ 2 SENSORS
- ❖ RASPBERRY PI
- ❖ 2 PROTOBOARD
- ❖ 1 NODEMCU ESP8266

DIAGRAMA DE BLOCS



CODI PART 1

- ❖ **ahora:** Variable que ha de buscar l'hora en aquell moment
- ❖ **hora_encen:** Quan s'ha d'encendre
- ❖ **hora_apaga:** Quan s'ha d'apagar
- ❖ Els pins **12** i **14** corresponen als **2** Leds
- ❖ Els pins **2** i **5** corresponen als sensors
- ❖ El Led **12** l'he denominat com a **pwm1**
- ❖ El Led **14** l'he denominat com a **pwm2**
- ❖ El Led **12** i **14** començen amb **0** d'intensitat
- ❖ **Def hora_actual:** Busca la hora que és en aquell moment

```
import RPi.GPIO as GPIO
import time

ahora = ("00:00:00")
hora_encen = ("23:00:00")
hora_apaga = ("05:00:00")

GPIO.setmode(GPIO.BCM)
GPIO.setwarnings(False)
GPIO.setup([12,14],GPIO.OUT)
GPIO.setup([2,5],GPIO.IN)

pwm_led1=GPIO.PWM(12,500)
pwm_led2=GPIO.PWM(14,500)
pwm_led1.start(0)
pwm_led2.start(0)

def hora_actual():
    ahora = time.strptime("%H:%M:%S")
```

CODI PART 2

```
def encen(pin):
    pwm_led1.ChangeDutyCycle(50)
    pwm_led2.ChangeDutyCycle(50)
    if GPIO.input(2) or GPIO.input(5):
        pwm_led1.ChangeDutyCycle(100)
        pwm_led2.ChangeDutyCycle(100)
        time.sleep(15)

def apaga(pin):
    GPIO.output(pin,GPIO.LOW)

def main():
    while (1):
        ahora = time.strftime("%H:%M:%S")
        if ("23:00:00" <= ahora <= "23:59:59") or ("00:00:00" <= ahora <= "05:00:00"):
            encen(12)
            encen(14)
        else:
            apaga(12)
            apaga(14)

main()
```


EXPLICACIÓ CODI PART 2.1

Si ahora (hora actual) en aquell moment està

entre les **23:00:00** i les **23:59:59** o entre les

00:00:00 i les **05:00:00**, els pins **12** i **14**

s'encenen amb un **50%** d'intensitat.

Si la hora en aquell moment no es entre les **23:00** i les **05:00** s'apaga (intensitat **0%**)

He utilitzat un **between** en per fer que s'encenés des de les **23:00:00** fins les **05:00:00**

```
def main():  
    while (1):  
        ahora = time.strftime("%H:%M:%S")  
        if ("23:00:00" <= ahora <= "23:59:59") or ("00:00:00" <= ahora <= "05:00:00"):  
            encen(12)  
            encen(14)  
        else:  
            apaga(12)  
            apaga(14)  
  
main()
```

EXPLICACIÓ CODI PART 2.2

Si són entre les **23:00** i les **05:00** s'encendran amb un

50% d'intensitat tan com el pin **12** i el **14**.

Si algun dels dos sensors detecten a algú, la intensitat passarà

al **100%** amb un time sleep de **15** segons

(estaran al **100%** d'intensitat durant **15** segons).

Si no és el cas, s'apagarà.

```
def encen(pin):  
    pwm_led1.ChangeDutyCycle(50)  
    pwm_led2.ChangeDutyCycle(50)  
    if GPIO.input(2) or GPIO.input(5):  
        pwm_led1.ChangeDutyCycle(100)  
        pwm_led2.ChangeDutyCycle(100)  
        time.sleep(15)  
  
def apaga(pin):  
    GPIO.output(pin, GPIO.LOW)
```

PROBLEMES QUE HE TINGUT

- ❖ Un petit problema que he tingut ha sigut que quan havia de posar que encenés a les 23:00:00 i s'apagués a les 05:00:00, no funcionava. La solució que he trobat a sigut dividir-ho en dos. Que s'encenés de 23:00:00 a 23:59:59 i de 00:00:00 a 05:00:00.
- ❖ El segon problema que he trobat ha sigut la distància del sensor 2 i el led B de la protoboard, ja que era una distància de 3 metres. La solució que he trobat ha sigut fer una segona protoboard amb una esp8266. Per tant quedaria: una protoboard amb una raspberry, un sensor i un led i, una altre protoboard amb un esp8266, un sensor i un led.
- ❖ Quan anava a fer el mqtt em vaig adonar que la esp8266 no tenia ip i desde Thonny (la aplicació que vaig usar per el micropython en el esp8266) li faig posar una.

IP A LA ESP8266

```
import machine
import network
wlan = network.WLAN(network.STA_IF)
wlan.connect('IOT2022', 'arduinouno')
wlan.ifconfig(('10.199.160.213', '255.255.255.0', '10.199.160.254', '8.8.8.8'))
```

1- for /L %i in (1,1,255) do @ping -n 1 -w 1 192.168.1.%i

2- arp -a

3-

Find MAC Address Vendors. Now.

Enter a MAC Address

a8-48-fa-c0-63-10

Espressif Inc.

ESP8266 PART1

```
import machine
import time

ahora = time.localtime()

p12 = machine.Pin(12, machine.Pin.OUT)
p14 = machine.Pin(14, machine.Pin.OUT)
p2 = machine.Pin(2, machine.Pin.IN)
p5 = machine.Pin(5, machine.Pin.IN)

led1 = machine.PWM(machine.Pin(12))
led2 = machine.PWM(machine.Pin(14))
led1.freq(90)
led2.freq(90)
led1.duty(511)
led2.duty(511)

def encen():
    #recollir valor p2 i p5
    valorP2=p2.value()
    valorP5=p5.value()
    if (valorP2 == 1) or (valorP5 == 1):
        print("PIR")
        led1.duty(1023)
        led2.duty(1023)
        time.sleep(7)
        led1.duty(211)
        led2.duty(211)

def apaga(pin):
    pin.off()
```

ESP8266 PART2

```
def main():  
    while (1):  
        ahora = time.localtime()  
        hora=ahora[3]  
        if (23 <= hora <= 24) or (0 <= hora <= 5):  
            encen()  
            print("Ences")  
        else:  
            apaga(p12)  
            apaga(p14)  
            print("Apagat")  
  
main()
```