

## Codigo:

```
#include <Arduino.h>
#define LED 16
unsigned long myTime;
#define interruptPin 0

void Task1(void * parameter);
void anotherTask(void * parameter);

void setup(){
  Serial.begin(115200);

  xTaskCreate(
    anotherTask, /* Task function. */
    "another Task", /* name of task. */
    10000, /* Stack size of task */
    NULL, /* parameter of the task */
    1, /* priority of the task */
    NULL); /* Task handle to keep track of created task */

  xTaskCreate(
    Task1, /* Task function. */
    "Task 1", /* name of task. */
    10000, /* Stack size of task */
    NULL, /* parameter of the task */
    1, /* priority of the task */
    NULL);
}

void loop(){
  Serial.println("this is ESP32 Task");
  delay(1000);
}

void Task1(void * parameter){
  pinMode(LED,OUTPUT);
  for(;;){
    Serial.println(myTime); // prints time since program started
    delay(500);
    digitalWrite(LED,HIGH);
    Serial.println("ON");
    delay(500);
    digitalWrite(LED,HIGH);
    Serial.println("OFF");
    digitalWrite(LED,LOW);
    delay(500);
  }
}

void anotherTask( void * parameter ){
  /* loop forever */
  for(;;){
    Serial.println("this is another Task");
    delay(1000);
  }
}
```

```

vTaskDelete( NULL );
}

}

```

## Funcionament:

En el cas d'aquesta pràctica volem que se'ns encengui i apagueu un led usant les tasques que hem vist ja a l'apartat 1, per això veiem com el setup del nostre codi serà el mateix però posarem una tasca més com veiem aquí:

```

void setup(){
  Serial.begin(115200);

  /* we create a new task here */
  xTaskCreate(
    anotherTask, /* Task function. */
    "another Task", /* name of task. */
    10000, /* Stack size of task */
    NULL, /* parameter of the task */
    1, /* priority of the task */
    NULL); /* Task handle to keep track of created task */

  xTaskCreate(
    Task1, /* Task function. */
    "Task 1", /* name of task. */
    10000, /* Stack size of task */
    NULL, /* parameter of the task */
    1, /* priority of the task */
    NULL);
}

```

Amb això fet ens falta definir el loop i declarar la funció i exactament que volem que aquesta faci, dit això veiem com el loop serà senzill ja que només volem que ens escrigui per pantalla un missatge.

```

void loop(){
  Serial.println("this is ESP32 Task");
  delay(1000);
}

```

Finalment, declararem les dues tasques que hem creat anteriorment, primer la Task1, la qual serà l'encarregada del LED:

```

void Task1(void * parameter){
  pinMode(LED,OUTPUT);
  for(;;){
    Serial.println(myTime); // prints time since program started
    delay(500);
    digitalWrite(LED,HIGH);
    Serial.println("ON");
    delay(500);
    digitalWrite(LED,HIGH);
    Serial.println("OFF");
    digitalWrite(LED,LOW);
    delay(500);
  }
}

```

```
}
```

Seguidament declarem la tasca "anothertask" la qual treu per pantalla el següent missatge:

```
void anotherTask( void * parameter ){  
    /* loop forever */  
    for(;;){  
        Serial.println("this is another Task");  
        delay(1000);  
    }  
    vTaskDelete( NULL );  
}
```