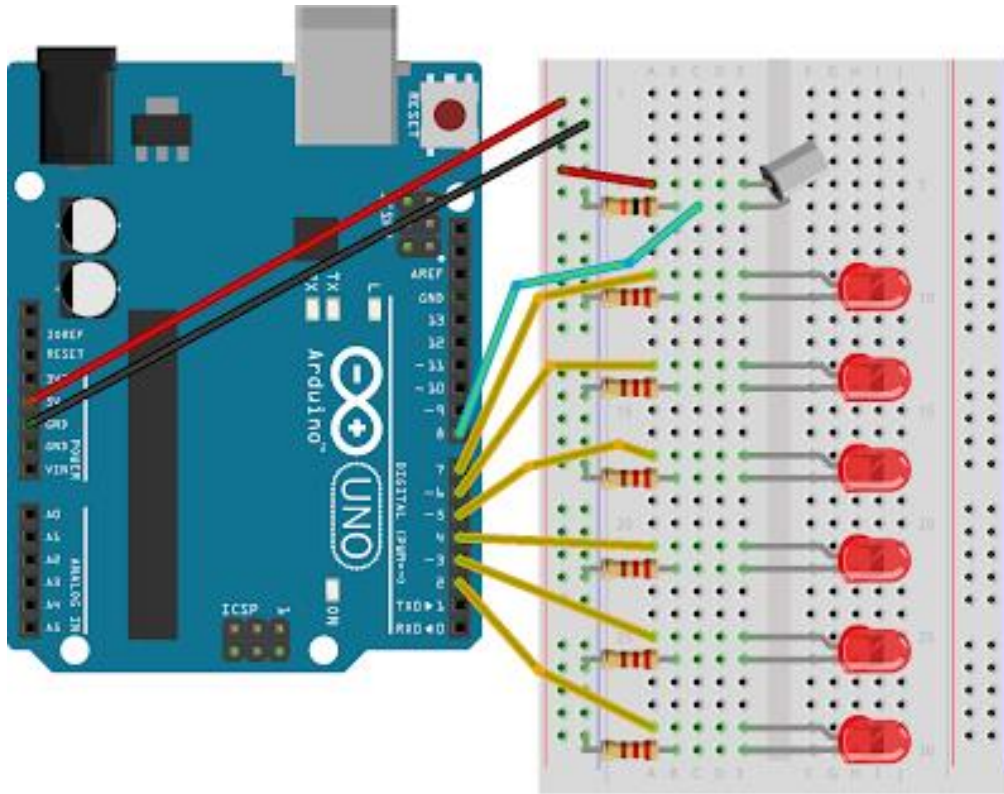


## Pràctica-1: Relotge

RELOTGE DIGITAL QUE ENCÉN UN LED CADA DEU MINUTS MITJANÇANT EL TEMPORITZADOR INCORPORAT D'ARDUINO



**Sensor d'inclinació** - Tipus d'interruptor que s'obre o tanca depenent de la seva orientació. Normalment es tracta de cilindres buits amb una bola de metall a l'interior que farà una connexió a través de dos cables quan s'inclina en la posició correcta. Es pot substituir per un pulsador qualsevol.



Programació:

```
const int switchPin = 8;

unsigned long previousTime = 0;

int switchState = 0;
int prevSwitchState = 0;

int led = 2;

long interval = 10000;

void setup(){
  for(int x = 2; x < 8; x++){
    pinMode(x, OUTPUT);
  }
  pinMode(switchPin, INPUT);
}

void loop(){
  unsigned long currentTime = millis();

  if(currentTime - previousTime > interval){
    previousTime = currentTime;
    digitalWrite(led, HIGH);
    led++;

    if(led == 7){
    }
  }
  switchState = digitalRead(switchPin);

  if(switchState != prevSwitchState){
    for(int x = 2; x < 8; x++){
      digitalWrite(x, LOW);
    }

    led = 2;
    previousTime = currentTime;
  }
  prevSwitchState = switchState;
}
```

### Explicació:

- Per començar, crea una constant anomenada **switchPin**. Aquest serà el nom del pin on es troba l'interruptor d'inclinació.
- Crea una variable de tipus **unsigned long** (enter llarg sense signe). Aquesta, mantindrà el temps que un LED va canviar per última vegada.
- Crea una variable per a l'estat de l'interruptor, i un altre per mantenir l'estat previ de l'interruptor. Més tard es compararan des d'un bucle.
- L'última variable serà l'interval per encendre cada LED, de tipus **long** (tipus de dades llargues). En 10 segons (el temps entre cada LED encès) 10,000 milisegons passaran.
- En el **setup()**, cal declarar els pins del 2 al 7 com sortides. Un bucle **for()** declara els sis com a sortida amb només 3 línies de codi. També cal declarar **switchPin** (pin de l'interruptor) com a entrada.
- En el **loop()**, obtenim la quantitat de temps que l'Arduino ha estat funcionant amb **millis()** i l'emmagatzemem en una variable local: **currentTime** (hora actual).
- Amb l'ús d'una sentència **if()**, podràs comprovar si ha transcorregut el temps suficient per encendre un LED. Restem l'hora actual amb el temps previ i comparem per veure si és més gran que la variable de l'interval.
- Una vegada hem establert **previousTime** (temps previ), encenem un LED, i incrementem la variable del led. La pròxima vegada que passi l'interval, el següent LED s'encendrà.
- Afegeix un altre **if()** en el programa per comprovar si el LED del pin 7 està encès.
- Llegim el valor de l'interruptor i l'emmagatzemem en la variable **switchState**.
- Amb un **if()** comprovaràs si l'interruptor està en una posició diferent de la que era anteriorment. L'avaluació **!=** comprova si **switchState** no és igual a **prevSwitchState**. Si són diferents, apaga els LEDs, retorna la seva variable al primer pin i restableix el temporitzador (**previousTime = currentTime**).
- Al final de bucle, guarda l'estat de l'interruptor a **prevSwitchState**, per poder comparar-lo amb el valor que s'obté a **switchState** en el següent bucle.

### 1. Introducció/Objectius

### 2. Components/Materials

### 3. Anàlisi-funcionament:

### 4. Anàlisi-Codi:

### 5. Canvis-realitzats:

### 6. Experimentacions:

### 7. Simulació-Tinkercad

### 8. Fotos/Videos

### 9. Aplicacions:

### 10. Problemes/Conclusions:

