

Práctica 1 – Interrupciones y Deep Sleep


Julián Cifuentes Vásquez, julian.cifuentesv@comunidad.iush.edu.co

Medellín, 11 de abril de 2024

Entregable de la práctica # 1.

Implementación un firmware que permite censar la temperatura de un sensor lm35 y muestra por serial la temperatura censada. La medición de la temperatura la debe hacer cuando se genere una interrupción en bajo para el pin 2. El programa debe permanecer el resto del tiempo en un modo de bajo consumo.

- [VIDEO](#)
- [CÓDIGO](#)



```
1  #include <avr/sleep.h>
2
3  #define sensorPin A0
4  #define ledPin 10
5  #define buttonPin 2
6  #define state 13
7
8  void sensorFunc();
9  void sleepFunc();
10 void interruptFunc();
```

```
1 void setup(){
2   Serial.begin(9600);
3   pinMode(ledPin, OUTPUT);
4   pinMode(buttonPin, INPUT);
5   pinMode(state, OUTPUT);
6
7   digitalWrite(state, HIGH);
8
9   attachInterrupt(digitalPinToInterrupt(buttonPin), interruptFunc, LOW);
10 }
11
12 void loop() {
13   sleepFunc();
14 }
```

```
1 void sensorFunc(){
2   int temp = analogRead(sensorPin);
3
4   float voltage = temp * 5.0 / 1023.0;
5   float tempC = voltage * 100.0;
6
7   Serial.print("Temperature: ");
8   Serial.print(tempC);
9   Serial.println(" °C");
10
11 }
```

```
1 void sleepFunc(){
2   Serial.println("Entering deep sleep mode...");
3
4   delay(750);
5   digitalWrite(state, LOW);
6
7   set_sleep_mode(SLEEP_MODE_PWR_DOWN);
8   sleep_enable();
9   sleep_mode();
10
11   digitalWrite(state, HIGH);
12
13 }
```

```
1 void interruptFunc(){
2   sleep_disable();
3
4   digitalWrite(ledPin, HIGH);
5   sensorFunc();
6   digitalWrite(ledPin, LOW);
7 }
```