

# EJERCICIO 3

---

## *IDEA INICIAL (y que realizaré más adelante)*

---

La idea de este programa es disponer de un ATtiny que se comunique con un Arduino por bluetooth (USART) o por radiofrecuencia para enviar datos de temperatura cada 10 minutos. Sin embargo aún no dispongo del módulo HC-05 bluetooth ni del emisor/receptor de RF (están pedidos pero aún pueden tardar una o dos semanas más, cuando disponga de éstos lo implementaré de esta forma). El ATtiny debería poder estar semanas o meses sin tener que cambiarle las baterías o que éste se cargue mediante una placa solar, por lo que el he decidido investigar varias librerías sobre como dormir al Arduino o a los ATtiny haciendo que estos gasten la mínima energía.

## *SOLUCIÓN REALIZADA*

---

Por ahora voy a implementar el programa en el Arduino de forma que sea éste el que lea y calcule la temperatura, la guarde en la memoria interna y después se ponga a dormir durante 8 segundos (el máximo tiempo del WDT). El programa guarda en la EEPROM la temperatura cada diez minutos aproximadamente. El Arduino se encuentra totalmente congelado (solo funciona el reloj para el wdt, por lo que gasta microAmperios) durante 8 segundos y después se despierta, comprueba si es la 75ª vez que se entre en el loop() (8seg por 75 veces = 10 minutos). Si ha entrado por 75ª vez calcula la temperatura y la guarda en la eeprom, sino se vuelve a apagar.

He usado la librería "LowPower.h" disponible en GitHub para congelar el micro.

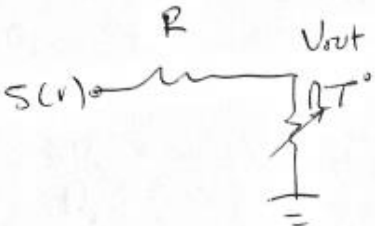
La librería "CalculoTemp.h" propia para calcular la temperatura a partir de la impedancia del sensor.

La librería <EEPROM.h> para guardar los datos en la eeprom. Cuando se llega a la última posición se sobrescribe a partir de la primera posición.

Con el fin de comprobar si el Arduino se duerme y se despierta correctamente cada 8 segundos, he hecho que cada vez que se despierte cambie de encendido a apagado o viceversa el led del pin 13. Lo más complicado de realizar este código ha sido entender cómo funciona el WDT, el funcionamiento de la librería LowPower y el cálculo de la temperatura.

He leído que lo ideal para el uso de mi sensor sería usar un puente de wheatstone para evitar que la corriente que pasa por éste no altere la medida de la temperatura, sin embargo he determinado que no es necesario ya que en la corriente que va a pasar por el sensor durará menos de un microsegundo cada diez minutos (he tenido cuidado en usar una función de la librería LowPower que apague el módulo ADC).

Calculo de la temperatura:



$$V_{out} = \frac{R_T}{R_T + R} \cdot S$$

$$V_{out} \cdot R_T + V_{out} \cdot R = R_T \cdot S$$

$$R_T (V_{out} - S) = -V_{out} \cdot R$$

$$R_T = \frac{S + V_{out} \cdot R}{S - V_{out}}; \quad 1023 \cdot \frac{V_{out}}{S} = V_{medido}$$

$$V_{out} = \frac{V_{medido}}{2048}$$

$$R_T = \frac{\frac{V_{medido}}{2048} \cdot R}{S - \frac{V_{medido}}{2048}} = \frac{\frac{V_{medido}}{2048} \cdot 10(k\Omega)}{S - \frac{V_{medido}}{2048}} = \frac{V_{medido} \cdot 10(k\Omega)}{1023 - V_{medido}}$$