

# APLICACIÓN DE UN SISTEMA DE MEDIDA INTELIGENTE

*Iñigo Aguirre<sup>1</sup>, M<sup>a</sup> Concepción Sáenz<sup>2</sup> e Iñigo Javier Oleagordia<sup>3</sup>*

*<sup>1</sup>Universidad del País Vasco. inigo.aguirre@ehu.es*

*<sup>2</sup>Universidad del País Vasco. iepsachc@lg.ehu.es*

*<sup>3</sup>Universidad del País Vasco. jtpolagi@lg.ehu.es*

## RESUMEN

En este artículo presentamos una aplicación de un sensor inteligente conectado al ordenador. Esta aplicación se utiliza, en la EUITI de Eibar (Universidad del País Vasco), como un ejemplo para los trabajos que deberán presentar los alumnos y las alumnas en Instrumentación Electrónica. Con ello pretendemos que las clases magistrales se enriquezcan con la exposición pública de los trabajos. También conseguimos aplicar en los trabajos conocimientos adquiridos en otras asignaturas.

## 1. INTRODUCCIÓN

Esta aplicación está compuesta por un sensor de temperatura de *Analog Devices* (AD7416) [1], un microcontrolador de *Microchip Technology Inc.* (16F84A) [2] y para la visualización en el ordenador el software de *Nacional Instruments* (LabVIEW) [3]. Para la representación del esquema se ha utilizado el software de *Labcenter Electronics* (ISIS Professional), para realizar el layout el software de *Labcenter Electronics* (ARES Professional) [4] y para programar el PIC16F84A utilizamos el software MPLAB.

Existe la posibilidad de que parte del alumnado no tenga conocimientos de algunas de las herramientas enumeradas. Esto no supone un problema debido a que se podrán utilizar otros recursos más conocidos por este sector del alumnado como, por ejemplo el PSPICE, el ORCAD etc. De hecho, el trabajo está abierto, dentro de un margen, a las propuestas del alumnado.

## 2. SISTEMA DE MEDIDA

En la Figura 1 podemos ver el diagrama de bloques del sistema de medida realizado. El sistema consta de tres partes:

- Sensor de temperatura AD7416 de Analog Devices.
- Microcontrolador PIC16F84A de Microchip.
- Ordenador.

El sensor de temperatura se conecta al microcontrolador a través del bus serie I<sup>2</sup>C, por otro lado el microcontrolador se conecta, a través del puerto paralelo. Además, veremos el valor de temperatura medido mediante el programa LabVIEW, con el que no sólo visualizamos la señal sino que también se realizará un histórico de alarmas. Seguidamente explicaremos los distintos elementos de los que consta el sistema de medida.

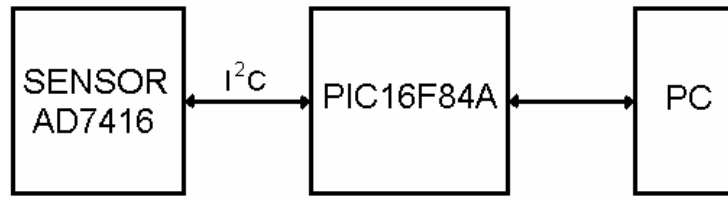


Figura 1: Sistema de medida y control típico.

### 3. SENSOR DE TEMPERATURA AD7416

#### 3.1. Características

El AD7416 es un sensor de temperatura que posee las siguientes características:

- Rango de temperatura (-40°C ÷ +125°C).
- Precisión de ± 1 °C.
- Convertidor Analógico – Digital de 10 Bit.
- Indicador de sobre temperatura.
- Bus serie I<sup>2</sup>C.
- Tres bits de direccionamiento.

#### 3.2. Función de los pines

En la Tabla 1 explicaremos la función de los 8 pin del integrado.

nº	Nombre	Función
1	SDA	I/O digital. Bus serie bidireccional para datos
2	SCL	Entrada digital. Clock del bus serie
3	OTI	Salida digital. Señal que se activa cuando el valor de la temperatura supera un valor de configuración
4	GND	Tensión de referencia
5	A2	Entrada digital. El MSB del direccionamiento del sensor
6	A1	Entrada digital. El bit intermedio del direccionamiento del sensor
7	A0	Entrada digital. El LSB del direccionamiento del sensor
8	VDD	Tensión de alimentación. 2,7V ÷ 5,5V

Tabla 1: Función de los pines del sensor.

### 3.3. Diagrama de bloques

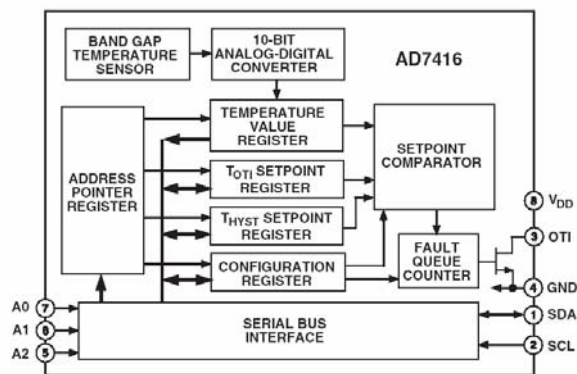


Figura 2: Diagrama de bloques del sensor AD7416.

La Figura 2 nos muestra el diagrama de bloques del sensor. Estos bloques son:

#### 3.3.1. Medida de la temperatura

El método utilizado para medir la temperatura se basa en el valor de la tensión base-emisor ( $V_{BE}$ ) del transistor, trabajando a corriente constante. Este método necesita una calibración individualizada de cada chip. La técnica que se utiliza es la medición de la variación de ( $V_{BE}$ ) cuando trabaja a dos valores de corriente. Esto lo podemos ver en (1).

$$\Delta V_{BE} = KT / q \cdot \ln(N) \quad (1)$$

Donde: K es la constante de Boltzmann, q es la carga del electrón, T (K) es la temperatura absoluta y N es la relación entre las corrientes.

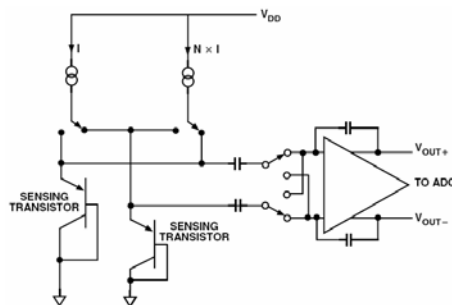


Figura 3: Técnica de medida de la temperatura.

#### 3.3.2. El ADC

El ADC recoge el valor ( $V_{BE}$ ) y aporta una señal digital de 10 bit en formato complemento a dos. La resolución del ADC es de 0.25 °C. Aunque, teóricamente, el rango es de 255 °C se utiliza en un margen entre -40 °C y +125 °C. Este valor se guarda en el Temperature Value Register. La conversión, para valores positivos de temperatura lo tenemos en (2) en la Tabla 2 vemos la relación, de algunos valores, entre la T(°C) y la salida digital.

$$T(^{\circ}C) = \text{CodigoADC} / 4 \quad (2)$$

Temperatura (°C)	Salida digital
-128°C	10 0000 0000
-50°C	11 0011 1000
-0.25°C	11 1111 1111
0°C	00 0000 0000
0.25°C	00 0000 0001
50°C	00 1100 1000
125°C	01 1111 0100
127°C	01 1111 1100

Tabla 2: Formato de los datos de temperatura.

### 3.3.3. Temperature Value Register

Temperature Value Register. Es el registro, de sólo lectura, donde se almacena el valor digital de la temperatura. Este registro es de 16 bit de los cuales se utilizan los 10 MSB para almacenar el dato.

### 3.3.4. TOTI Setpoint Register

TOTI Setpoint Register. Este registro de 16 bit, de lectura/escritura, almacena el valor de temperatura por encima de la cual se activa la señal OTI. Este registro utiliza los 9 MSB para realizar la comparación.

### 3.3.5. THYST Setpoint Register

THYST Setpoint Register. Este registro de 16 bit, de lectura/escritura, almacena el valor de temperatura para regular el valor de histéresis. Este registro utiliza los 9 MSB para realizar la comparación.

### 3.3.6. Configuration Register

Configuration Register. Es un registro de lectura/escritura de 8 bit que se utiliza para configurar los modos de operación del sensor. Por ejemplo, la salida OTI se puede configurar como comparador o como interruptor. Si funciona como comparador la señal se activa cuando la temperatura sobrepasa el valor de  $T_{OTI}$  y se desactiva cuando es inferior a  $T_{HYST}$ . Si funciona como interruptor se activa tanto si supera el valor  $T_{OTI}$  como si es inferior a  $T_{HYST}$  y se resetea con un acceso de lectura. Si la temperatura supera el valor  $T_{OTI}$  y una vez peseteado si vuelve a superar el valor no se vuelve a activar hasta que no sea inferior a  $T_{HYST}$ .

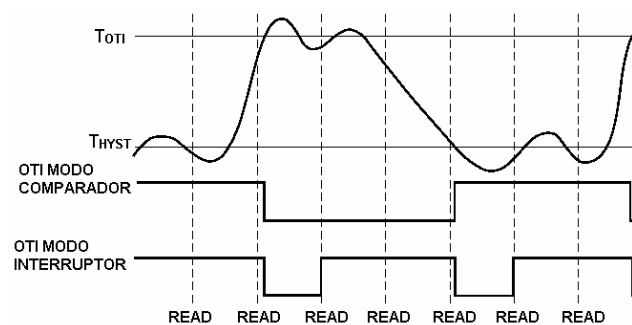


Figura 4: Modos de configuración de la señal OTI.

### 3.3.7. Fault Queue

Fault Queue. Se utiliza para evitar falsas activaciones de la señal OTI en ambientes con mucho ruido. Para ello se utilizan dos bit del registro de configuración con los que configuramos el número de eventos necesarios para que la señal se active.

### 3.3.8. Address Pointer Register

Address Pointer Register. Este es un registro de 8 bit que almacena el direccionamiento de uno de los registros vistos anteriormente. Para ello utiliza los dos LSB.

### 3.3.9. Serial Bus Interface

Serial Bus Interface. El sensor se comunica a través del bus serie I<sup>2</sup>C. El sensor se conecta a este bus como esclavo bajo el control de un master, que en nuestra aplicación es el PIC16F84A. Este bus realiza la comunicación a través de dos hilos. Los hilos utilizados se denominan SCL y SDA. Mientras que el hilo SDA es bidireccional, utilizable tanto por los master como por los slave. El SCL es una señal de clock que sólo la puede controlar el master.

## 4. CIRCUITO DE MEDIDA

En la Figura 5 podemos ver el circuito completo formado por el sensor, el microcontrolador y la comunicación con el PC. El componente más reseñable es el microcontrolador PIC16F84A, que por su bajo precio y su gran desarrollo en los últimos años resulta un componente atractivo para trabajar con él.

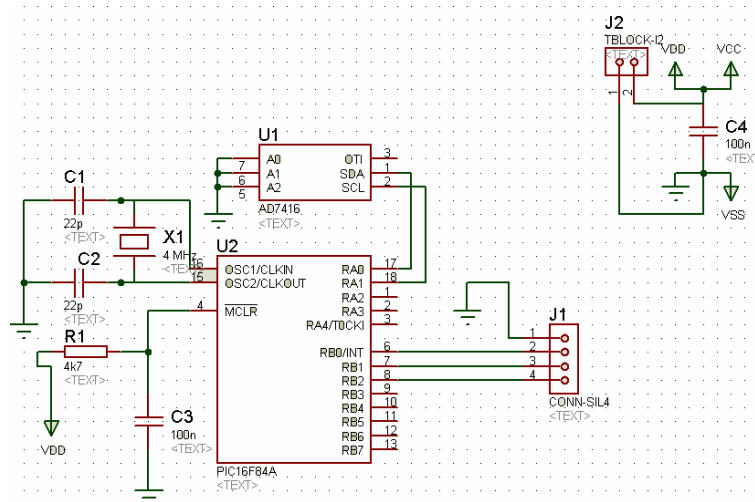


Figura 5: Esquema completo del sistema de medida.

En la Figura 5 podemos ver el circuito completo formado por el sensor, el microcontrolador y la comunicación con el PC a través del conector J1. El esquema se ha realizado con el *Labcenter Electronics* (ISIS Professional) [4], que es un programa fácil e intuitivo. Por otro lado con el programa *ARES Professional* de *Labcenter Electronics* [4] obtenemos el ruteado. En la Figura 6 Presentamos el ruteado con los componentes.

Finalmente veremos el resultado obtenido en la Figura 7.

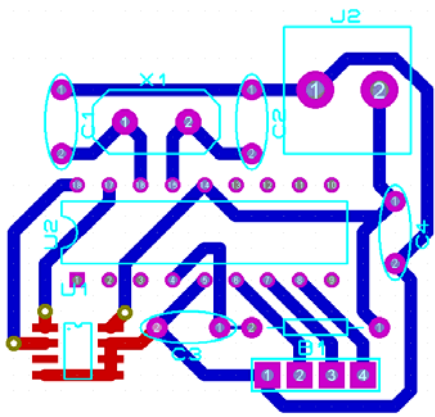


Figura 6: Ruteado del sistema de medida.

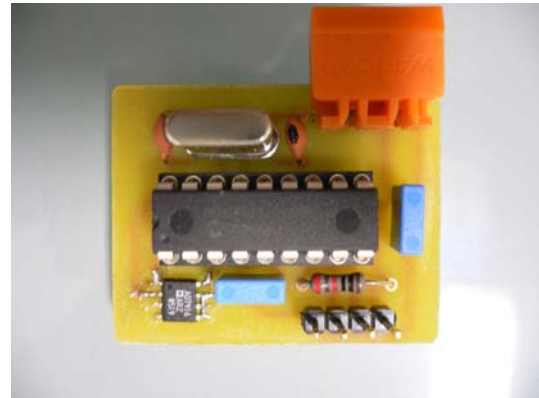


Figura 7: Circuito de medida con el AD7416.

## 5. PROGRAMA DEL MICROCONTROLADOR

El programa del microprocesador tiene dos cometidos: el primero leer la temperatura del sensor y finalmente enviar la lectura vía serie al ordenador [5].

En la “Aplicación note AN-561” que podemos localizar en [1] tenemos todo el programa. En este documento además de poseer el programa principal posee las siguientes subrutinas:

- Inicialización del Puerto: Esta posee la configuración de los puertos de entrada y salida, así como la configuración del Timer para generar interrupciones permitiendo que se lea periódicamente la temperatura.
- Inicio y final de la comunicación con el sensor: En todas las transferencias de información mediante el protocolo I2C se necesitan estas subrutinas, en las cuales en el inicio se ponen las señales SDA y SCL a nivel bajo y al finalizar en nivel alto.
- Envío del byte. Esta subrutina se llama dos veces, una para direccionar el registro y la segunda para capturar el dato de la temperatura.
- Lectura del byte. Esta subrutina lee el valor de la temperatura y la almacena en dos byte: HIGHBYTE y LOWBYTE
- Lectura de la temperatura. Esta subrutina engloba las anteriores.
- Transmisión. Se encarga de la transmisión a través del puerto paralelo

## 6. PROGRAMA EN LABVIEW

En el panel vamos a ver representado la evolución de la temperatura en el tiempo, para mejor visión se añade el valor numérico. Por otro lado, en el programa controlamos el puerto paralelo. La comunicación se realiza mediante tres hilos: SYNC, SCLK y SDATA.

La señal SYNC la activa el PIC para iniciar la transferencia de datos, posteriormente se genera la señal de reloj (SCLK), desde el LabVIEW [3], que en cada flanco de bajada recoge el bit de SDATA, obteniendo un array de 16 bits, de los cuales 10 bits son válidos. Este array de 10 bits se convierte un valor numérico, que será el valor de temperatura en (°C). En las próximas figuras podemos ver tanto el panel como las diferentes partes del diagrama.

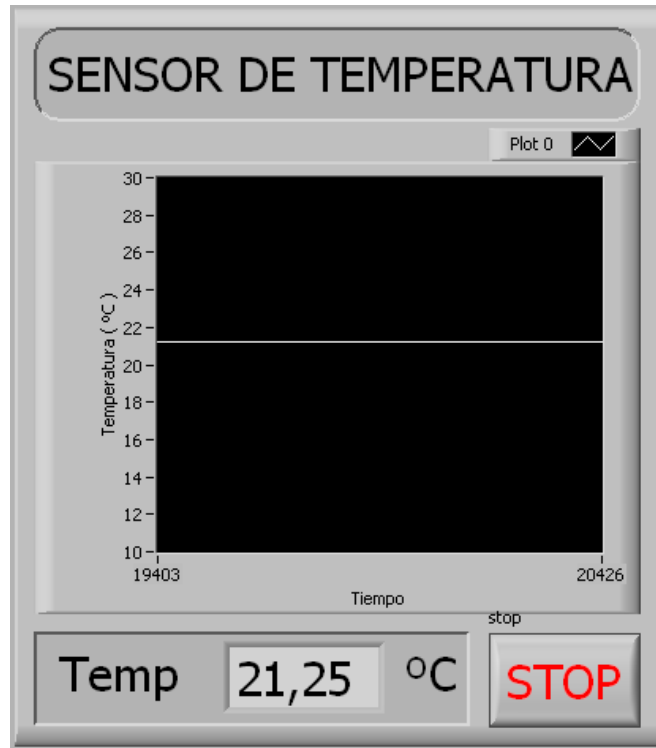


Figura 8: Interfaz de usuario.

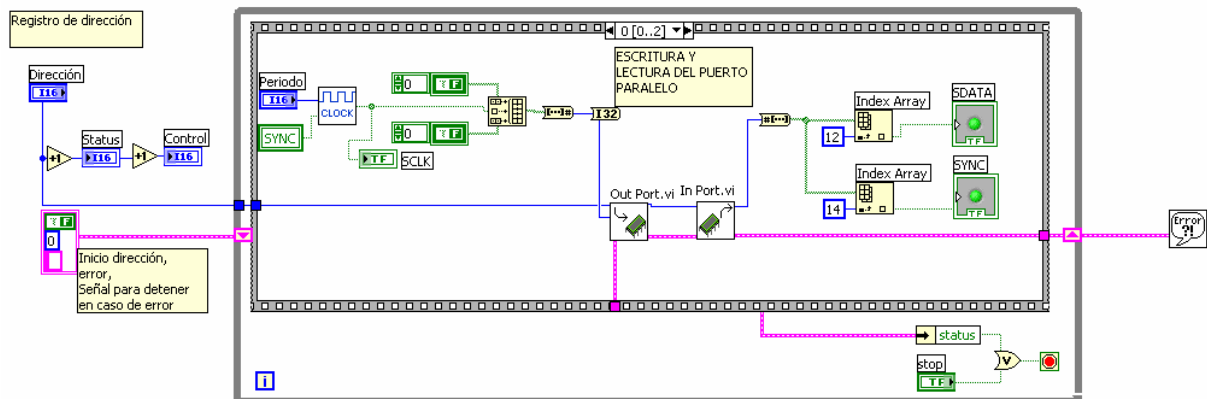


Figura 9: Programa de control del puerto paralelo.

Este diagrama recoge los valores de las señales SDATA y SYNC y aporta la señal de reloj de la señal de salida, del PC, SCLK. Posteriormente se va creando el array, hay que tener en cuenta que el primer valor adquirido corresponde con el MSB, como he dicho anteriormente seleccionamos los 10 MSB y convertimos a su valor numérico en °C.

## 7. CONCLUSIONES

Este tipo de trabajos además de dar la posibilidad de utilizar los conocimientos de otras disciplinas, pretende dar a conocer al alumnado una variedad de sensores comerciales y una posible aplicación. Este tipo de trabajos, aunque supone un esfuerzo por parte del alumnado,

son muy bien recibidos ya que todos los trabajos son expuestos en clase delante de los compañeros y todos ellos repercuten, en gran medida, sobre toda la clase.

## 8. BIBLIOGRAFÍA

[1] <http://www.analog.com> [Consulta: 15 febrero 2006]

[2] <http://www.microchip.com> [Consulta: 23 marzo 2006]

[3] LÁZARO, A. M. *LabView 7.1. Programación gráfica para el control de instrumentación*. Thomson, 2005, ISBN: 84-9732-391-2.

[4] <http://www.labcenter.co.uk> [Consulta: 7 mayo 2006]

[5] PALACIOS, E., REMIRO, F., LÓPEZ, L. J. *Microcontrolador PIC16F84A. Desarrollo de proyectos*. Rama, 2004, ISBN: 84-7897-600-0.