

Termómetro IR #28040

1.- INTRODUCCION

Este módulo de Parallax consiste en un termómetro que permite medir la temperatura de un objeto y está basado en el sensor de infra rojos (IR) MLX90614ESF-AAA-TO-39 de la firma MELEXIS. Este sensor está diseñado para medir sin contacto la temperatura de los objetos colocados dentro de su campo de acción. El sensor con el que está dotado el módulo 28040 tiene un campo de visión de 90° e integra un detector sensible a los infra rojos y un integrado para su control. Cada sensor se comunica mediante el protocolo SMBus con un coprocesador SX20AC/SS-G que Parallax a programado para simplificar su uso y facilitar la comunicación con el controlador principal del usuario (PIC, Basic Stamp, Arduino, Atmel, etc...).

Las figuras 1 y 2 muestran el aspecto del módulo 28040 así como el cono a campo de visión del sensor que incorpora.



Figura 1. El termómetro IR 28040

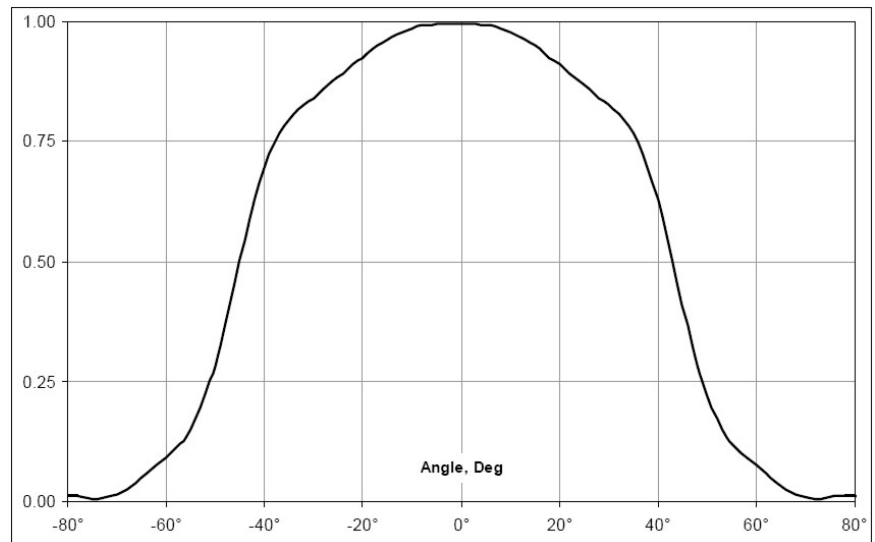


Figura 2. El campo de visión

2.- CARACTERISTICAS

Entre las características más relevantes de este termómetro infra rojo se resumen a continuación:

- Continuo flujo de los datos de salida y control independiente de alarma
- Datos de 16 bits que permite un rango de temperaturas de -70 a +380°C
- Detección automática de baudios para la comunicación con el controlador principal (host) de la aplicación del usuario: 2400, 4800, 9600, 19200 y 38400 baudios.
- El módulo 28040 se presenta en un circuito impreso con 5 patillas en formato SIL con paso 2.54 que facilita su empleo en todo tipo de boards y placas de prototipos.
- Se puede crear una red de múltiples termómetros empleando tan sólo una única señal de E/S para la comunicación con el host.
- El módulo se puede emplear de forma autónoma, sin conexión con host alguno, para el control de una señal de alarma por exceso de temperatura.
- Funciones de configuración que permiten un bajo consumo.

3.- ESPECIFICACION Y APLICACIONES

Sus especificaciones mecánicas y eléctricas se resumen a continuación:

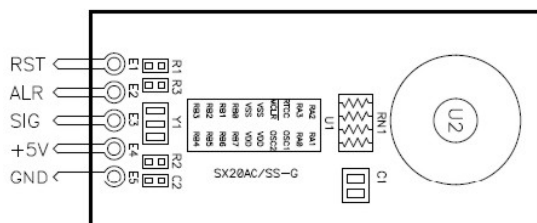
- Alimentación única de +5Vcc
- Consumo típico de 20mA. En el modo sleep de bajo consumo el consumo es de 5mA
- Conexiones en formato SIL con 5 patillas con paso 2.54 mm
- Temperatura de trabajo de 0 a 70°C.
- Dimensiones: 6.5 x 13.8 x 40.4 mm
- Peso: 7 gr.

Algunas de los posibles ejemplos de aplicación:

- Detección y medida en superficies, sin contacto con las mismas.
- Alarma para condiciones de exceso de temperatura
- Medición de temperaturas en sistemas de aire acondicionado en residencias, comercios y edificios industriales.
- Control industrial de temperaturas en piezas o elementos móviles.
- Detección de presencia humana o animal.
- Detección de movimiento.
- Control de temperaturas en múltiples zonas. Se pueden realizar redes de termómetros que se pueden leer con una única línea de datos.
- Relés térmicos o termostatos
- Medida de temperatura del cuerpo.

4.- PATILLAJE

La siguiente figura 3 junto con la tabla que le acompaña, muestran la distribución y descripción del patillaje del módulo termómetro IR 28040.



PIN	NOMBRE	DESCRIPCION
1	RST	Reinicio del coprocesador del módulo. Se activa por transición "1" – "0" – "1"
2	ALR	Salida de alarma por exceso de temperatura. Activa por "1"
3	SIG	Línea de E/S de datos serie para la comunicación con el controlador host del usuario
4	+5Vcc	Tensión de alimentación
5	GND	Tierra de alimentación

Figura 3. Patillaje del termómetro IR 28040

Se recomienda revisar el datasheet del sensor MLX90614 que incluye información técnica detallada del sensor propiamente dicho. También se puede visitar la página del fabricante www.melexis.com donde se pueden encontrar abundantes notas de aplicación.

5. CÓMO TRABAJA EL MÓDULO 28040

El módulo termómetro IR 28040 consta principalmente de un sensor IR de temperatura sin contacto MLX90614 y un coprocesador esclavo SX20AC/SS-G programado de fábrica por Parallax. La comunicación entre el sensor y el esclavo se realiza según el protocolo SMBus (una versión del I2C), pero esto es transparente para el usuario final. Lo verdaderamente importante es que la comunicación entre el módulo (esclavo) y el host del usuario se realiza en serie, empleando una única línea de E/S del microcontrolador del usuario o host (SIG). Para ello Parallax ha dispuesto de una serie de comandos que permiten y facilitan el uso de todas las propiedades del sensor.

Por su parte el sensor MLX90614 contiene una EEPROM interna donde se pueden ajustar una serie de parámetros de configuración como son:

- Dirección de esclavo del módulo/sensor en cuestión. Esto permite crear una red con múltiples módulos o termómetros IR 28040 que emplean la misma línea de E/S para la transferencia de datos serie entre cualquiera de ellos y el host de la aplicación del usuario.
- Valor de la alarma donde se establece la temperatura máxima que, en caso de sobrepasarse, activará la salida ALR del termómetro.
- Intervalos de tiempo empleados durante la transmisión de datos entre el termómetro IR 28040 y el host del usuario.

Entre los datos que devuelve el módulo cada vez que es leído por el host, se incluye un byte para el cálculo PEC (Packet Error Code). El valor PEC no es un valor lineal que muestre la integridad de los datos transferidos, pero mediante un procedimiento de CRC-8 puede servir para verificar si los datos transferidos son o no válidos. Si se establece una comunicación correcta entre el módulo 28040 y el host del usuario, no es necesario el empleo del byte PEC a no ser que el módulo sea retirado del circuito justo durante una transferencia de datos.

Tras la conexión de alimentación al módulo 28040, las temperaturas leídas se almacenan en una RAM interna, en la dirección 0x07. Existe un retardo mínimo de 1 mS entre la lectura de una medida y la siguiente. En el modo normal de funcionamiento este tiempo se puede ajustar entre 1 y 2550 mS. En el modo sleep de bajo consumo el tiempo se puede ajustar en intervalos de 1 a 255 segundos.

La temperatura para el disparo de la alarma puede ajustarse entre -70°C y +380°C. Por defecto queda ajustada de fábrica a 100°C.

6. PROTOCOLO DE COMUNICACIÓN

Parallax ha desarrollado, en el coprocesador que integra el módulo 28040, un intérprete con una serie de comandos que permiten el control total del sensor MLX90614 y que son enviados por el host mediante una sencilla comunicación serie. Estos comandos y/o sus parámetros se quedan almacenados en la RAM y/o EEPROM del sensor.

Todos los comandos que se envían al módulo van precedidos del byte 0x00. Esto mantiene a la línea serie a nivel "0" durante un breve intervalo de tiempo que sirve para alertar al módulo de que le llegará un nuevo comando.

A continuación se envía el comando. Todos los comandos comienzan por la cadena "**!TEM**". El carácter "!" se emplea para la sincronización y detección de baudios en la comunicación entre el módulo 28040 y el host. La cadena "**TEM**" identifica al propio módulo termómetro IR 28040. La cadena "**!TEM**" va seguida de otro carácter que identifica al comando propiamente dicho según la tabla que se muestra seguidamente. En general el formato de los comandos es el siguiente (las comas ',' separan los campos, pero no hay que enviarlas):

0x00, "!TEM?", Esclavo, Dir, byte L, byte H

0x00	Es el byte inicial con el que debe empezar todo comando que se vaya a enviar
!TEM?	Es la cadena inicial de todo comando. El signo '?' representa el comando en particular
Esclavo	Es la dirección que todo módulo termómetro IR 28040 tiene asignada. Por defecto es 0x5A y se puede modificar para crear redes de módulos.
Dir	Es la dirección interna a la que se accede para la ejecución de un determinado comando
Byte L	Representa el byte de menos peso del parámetro de un determinado comando
Byte H	Representa el byte de más peso del parámetro de un determinado comando

COMANDO	DESCRIPCION	ESCRIBE EN	RESPUESTA
"!TEMR"	Lee datos (temperatura) del sensor	---	Devuelve tres bytes: temperatura byte bajo, temperatura byte alto y byte PEC
"!TEMW"	Escribe datos de configuración	EEPROM: 0x20, 0x21 o 0x2E	Ninguna
"!TEMC"	Pone al sensor en modo de lectura continua	RAM: 0x07	Ninguna
"!TEMc"	Pone al sensor en modo de lectura y transmisión continua	RAM: 0x07	Devuelve la cadena "!TEM" seguida del byte de dirección de esclavo, byte bajo de la temperatura, el byte alto y el byte PEC
"!ITEMZ"	Pone al sensor en modo de lectura continua y bajo consumo	RAM: 0x07	Ninguna
"!ITEMz"	Pone al sensor en modo de lectura y transmisión continua y bajo consumo	RAM: 0x07	Devuelve la cadena "!TEM" seguida del byte de dirección de esclavo, byte bajo de la temperatura, el byte alto y el byte PEC
"!ITEMI"	Ajusta el intervalo de tiempo entre lectura y lectura continuas de la temperatura	RAM: 0x07	Ninguna
"!ITEMA"	Ajusta el valor de la temperatura de la alarma	RAM: 0x07	Ninguna
"!ITEMS"	Detiene las lecturas continuas de temperatura	----	Ninguna
"!IID"	Devuelve la versión o ID del firmware interno	----	Devuelve una cadena ASCII de tres caracteres "x.y"

6.1 Comando Read "R": Lectura de temperatura del sensor

Formato: $0x00, !TEMR, Esclavo, 0x07$

Devuelve tres bytes con la temperatura del sensor según la última lectura o medida realizada. Primero devuelve el byte de menos peso, luego el de más peso y por último el byte PEC. Los dos primeros bytes representan la temperatura medida y expresan unidades de 0.02 Kelvin

6.2 Comando Write "W": Escritura de datos en el sensor

Formato: $0x00, ¡TEMW, Esclavo, Dir, Byte L, Byte H$

Este comando se empleará para escribir parámetros de configuración y ajuste en la memoria EEPROM del sensor según el valor de *Dir* y que se resume en la siguiente tabla:

Dir	Objetivo	Byte L	Byte H	Defecto
0x20	Ajustar la temperatura de disparo de la alarma	Valor de 16 bits que representa unidades de 0.02 K de la temperatura (0°C = 273 K)		100°C (18650 unidades de 0.02K)
0x21	Ajustar los intervalos de tiempo entre lectura y lectura	Byte L * 10mS en el modo normal	Byte H * 1.15 seg. En el modo sleep	50 mS en normal 1.15 s. en sleep
0x2E	Ajusta la nueva dirección de esclavo del módulo y la configuración del modo de trabajo	Dirección esclavo entre 0x01 y 0x7F	Configuración: "C", "c", "Z", "z", "S"	Esclavo=0x5A Modo ="C"

Ejemplos: *0x00, !TEMW, 0x5A, 0x20, 0x28, 0x3C*
Ajusta la temperatura de alarma a 35°C (p.e. $(35+273)/0.02 = 15400 = 0x3C28$)

0x00, !TEMW, 0x5A, 0x2E, 0x33, "Z"
La nueva dirección de esclavo pasa a ser 0x33 y el modo de trabajo "Z" (bajo consumo)

Este comando implica la escritura de datos en la EEPROM del sensor, por lo que se debe esperar un mínimo de 50mS antes de enviar cualquier otro comando.

Igualmente este comando es de carácter general que permite realizar cualquier ajuste y configuración. En lo sucesivo se presentarán comandos de carácter específico que se derivan de éste y que pueden hacer lo mismo.

6.3 Comandos "C" y "c": Lectura continua del sensor

Formato: *0x00, !TEMc, Esclavo, 0x07*
0x00, !TEMW, Esclavo, 0x07

Al ejecutar el comando "C" se consigue que el sistema esté de forma constante leyendo la temperatura que ofrece el sensor. El intervalo de tiempo entre una lectura y la siguiente se determina mediante el comando "W" o el comando "I" y por defecto es de 50mS. El comando "c" es idéntico, pero además de leer la temperatura de forma constante y con un determinado intervalo, cada temperatura recién leída se transmite directamente al host.

Formato equivalente: *0x00, !TEMW, Esclavo, 0x2E, Esclavo, "C"*
0x00, !TEMW, Esclavo, 0x2E, Esclavo, "c"

Cuando envía el comando !TEMc (o su equivalente), el host se debe preparar para recibir datos de forma constante. Por cada vez que se realiza una lectura de la temperatura del sensor, se transmite una trama al host. Cada trama empieza con la cadena ASCII "TEM" seguido de un byte con la dirección del módulo que envía la medida, los bytes bajo y alto con el valor de la temperatura y, por último, el byte PEC.

Recordemos que cada lectura de temperatura y su consiguiente transmisión al host, se realiza a intervalos de tiempo que se puede ajustar mediante el comando "W" y/o el comando "I". En caso de que haya varios módulos 28040 en red, debemos asegurarnos de que los intervalos de tiempo entre una medida y la siguiente, sea diferente para cada módulo evitando así que dos o más coincidan en la transmisión de la temperatura.

6.4 Comandos "Z" y "z": Modo de trabajo

Formato: *0x00, !TEMZ, Esclavo, 0x07*
0x00, !TEMz, Esclavo, 0x07

El comando "Z" establece el modo de trabajo en bajo consumo con lectura continua de la temperatura del sensor. El comando "z" establece el modo de trabajo en bajo consumo con lectura continua de la temperatura del sensor y transmisión directa de la misma.

Por defecto entre lectura y lectura se emplea un intervalo de tiempo de 1.15 seg. Y se puede ajustar mediante los comandos "I" y/o "W". En caso de que haya varios módulos 28040 en red, debemos asegurarnos de que los intervalos de tiempo entre una medida y la siguiente, sea diferente para cada módulo evitando así que dos o más coincidan en la transmisión de la temperatura. El consumo pasa de los 20mA del modo normal, a los 5 mA.

Formato equivalente: *0x00, !TEMW, Esclavo, 0x2E, Esclavo, "Z"*
0x00, !TEMW, Esclavo, 0x2E, Esclavo, "z"

Para finalizar este modo de trabajo en bajo consumo se debe aplicar señal de RESET por la patilla correspondiente.

Al igual que con los comandos “C” o “c”, por cada vez que se realiza una lectura de la temperatura del sensor, se transmite una trama al host. Cada trama empieza con la cadena ASCII “TEM” seguido de un byte con la dirección del módulo que envía la medida, los bytes bajo y alto con el valor de la temperatura y, por último, el byte PEC.

6.5 Comando “I”: Ajusta intervalo de tiempo

Formato: *0x00, !ITEMI, Esclavo, Byte L, Byte H*

Este comando permite ajustar el tiempo de transmisión de los datos entre una medida y la siguiente que se empleará en los comandos “c” y “z”. En el modo normal de trabajo el intervalo se ajusta entre 1 y 255 unidades de 10mS (Byte L). En el modo de bajo consumo el intervalo se ajusta entre 1 y 255 unidades de 1.15 seg (Byte H).

Formato equivalente: *0x00, !ITEMW, Esclavo, 0x21, Byte L, Byte H*

Ejemplo: *0x00, !ITEMI, 0x5A, 0x0A, 0x04 → 0x00, !ITEMW, 0x5A, 0x21, 0x0A, 0x04*

El ejemplo, en los dos formatos, ajusta el intervalo en 100mS para el modo normal (10mS x 0x0A) y en 4.6 segundos para el modo de bajo consumo (1.15s x 0x04).

6.6 Comando “A”: Ajusta la temperatura de alarma

Formato: *0x00, !ITEMA, Esclavo, Byte L, Byte H*

Con este comando ajustamos la temperatura de alarma que, en caso de ser superada, activará la patilla ALR de alarma. Byte L y Byte H expresan el valor de la temperatura en unidades de 0.02 K

Formato equivalente: *0x00, !ITEMW, Esclavo, 0x20, Byte L, Byte H*

Ejemplo: *0x00, !ITEMA, 0x5A, 0xDA, 0x48 → 0x00, !ITEMW, 0x5A, 0x20, 0xDA, 0x48*

El ejemplo, en los dos formatos, ajusta la temperatura de alarma en 100°C: $(100 + 273) / 0.02 = 18650 = 0x48DA$

6.7 Comando “S”: Detener la lectura continua

Formato: *0x00, !ITEMS, Esclavo*

Este comando detiene la lectura continua de la temperatura.

Formato equivalente: *0x00, !ITEMW, Esclavo, 0x2E, Esclavo, “S”*

Ejemplo: *0x00, !ITEMS, 0x5A → 0x00, !ITEMW, 0x5A, 0x2E, 0x5A, “S”*

El ejemplo, en los dos formatos, ejecuta el comando “S” sin alterar la dirección de esclavo

6.8 Comando “!ID”: Devuelve la versión del firmware

Formato: 0x00, !ID, Esclavo

Este comando devuelve una cadena ASCII de tres caracteres “x.y” que expresan el número de la versión del firmware interno del módulo termómetro IR 28040.

7. EJEMPLOS

Junto con el módulo del termómetro IR 28040 se adjunta un CDROM que contiene librerías y ejemplos sencillos que muestran el funcionamiento de este dispositivo. Se proporcionan los programas fuente escritos en ensamblado y en C para el PIC16F886. Se trata de ejemplos didácticos cuya única pretensión es proporcionar los elementos básicos para manejar este sensor de cara su empleo en las múltiples aplicaciones en que puede ser utilizado. Se incluyen también una serie de librerías que facilitan el diseño de los programas que usan el módulo 28040.

Librería: *lcd4bitsPIC16*

Contiene rutinas para el control de una pantalla LCD mediante interface de 4 bits y que se usará para visualizar los resultados de los distintos ejemplos.

NOMBRE	PARAM. DE ENTRADA	PARAM. DE SALIDA	DESCRIPCION
UP_LCD	Ninguno	Ninguno	Configura las líneas de E/S para adaptarlas a la pantalla LCD
LCD_INI	Ninguno	Ninguno	Rutina para la inicialización de la pantalla LCD según especificaciones del fabricante
LCD_DATO	W=Contiene el carácter a visualizar	Ninguno	Envía a la pantalla el dato a visualizar en la posición actual del cursor
LCD_REG	W=Contiene el comando a ejecutar por parte de la pantalla LCD	Ninguno	Envía a la pantalla el comando que debe ejecutar

Librería: *RS232LOWPIC16*

El interface entre el host y el módulo de termómetro IR 28040 se realiza mediante una única línea y según el protocolo serie estándar. Esta librería contiene funciones que implementan, por software, la transmisión y recepción de datos hacia/desde el módulo por software.

NOMBRE	PARAM. DE ENTRADA	PARAM. DE SALIDA	DESCRIPCION
RxS	Ninguno	Rxdreg	Espera a recibir un carácter vía serie que quedará almacenado en Rxdreg
TxS	Txdreg	Ninguno	Transmite vía serie el carácter previamente almacenado en Txdreg

Librería: IR_28040_PIC16

Contiene funciones específicas para el control del módulo termómetro IR 28040 de Parallax

NOMBRE	PARAM. DE ENTRADA	PARAM. DE SALIDA	DESCRIPCION
IR_Reset	Ninguno	Ninguno	Señal de Reset del termómetro IR 28040 según especificaciones de Parallax: nivel bajo durante 5 mS y luego alta impedancia durante al menos 50 mS
IR_Read	Ninguno	IR_Dato_1=Byte L IR_Dato_2=Byte H IR_Dato_3=PEC	Lee el valor de la temperatura actual del termómetro IR 28040. El resultado de 16 bits se almacena en IR_Dato_1, IR_Dato_2 (byte L, byte H) y representa unidades de 0.02 Kelvin. En IR_Dato_3 se almacena e PEC.
IR_Alarma	IR_Dato_1=Byte L IR_Dato_2=Byte H	Ninguno	Ajusta el valor de la temperatura de alarma. Cuando esta se supere la línea ALR del termómetro IR 28040 se activará. Las variables IR_Dato_1 e IR_Dato_2 se suponen cargadas con el valor bajo y alto de la temperatura de alarma respectivamente. Esta temperatura se expresa en unidades de 0.02 K
IR_Write	IR_Dir_Com=Dirección interna del comando IR_Dato_1=parte baja del parámetro IR_Dato_2=parte alta del parámetro	Ninguno	Graba parámetros de configuración en la EEPROM interna del termómetro IR 28040 IR_Dir_Com debe contener la dirección interna del comando de la EEPROM. En IR_Dato_1 la parte baja del parámetro y en IR_Dato_2 la parte alta.
IR_ID	Cadena ASCII "x.y" en: IR_Dato_1 IR_Dato_2 IR_Dato_3		Obtiene la versión ID del firmware interno del termómetro IR 28040 que se deposita en las variables IR_Dato_1, IR_Dato_2 e IR_Dato_3 en formato ASCII x.y

6.1 Ejemplo 1: Obtener la versión ID del firmware interno

Objetivos

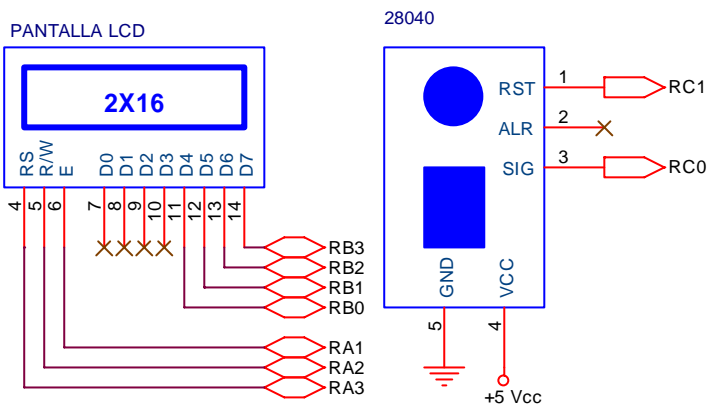


Figura 4. Esquema de conexiones del ejemplo 1

El ejemplo extrae y visualiza sobre la pantalla LCD la versión actual del firmware interno del módulo termómetro IR 28040

Esquema

Se muestra en la figura 4. La pantalla LCD se conecta con RB0:RB3 en un interface de 4 bits. Con RA1:RA3 se generan las señales de control.

Por su parte la transferencia de datos entre el controlador y el termómetro IR 28040 se realiza a través de la línea RC0. Por RC1 se genera la señal RESET de inicio del módulo.

Comentarios

Se trata de un ejemplo muy sencillo que hace uso de la función IR_ID para obtener la versión ID y visualizarla sobre la pantalla LCD. En la fotografía de la figura 5 se muestra el resultado de la ejecución de éste ejemplo.

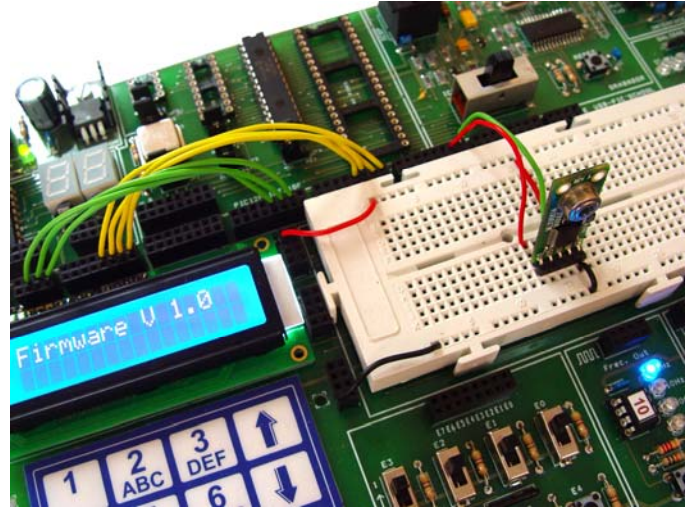


Figura 5. Ejecución del ejemplo 1

6.2 Ejemplo 2: Medida y visualización de la temperatura.

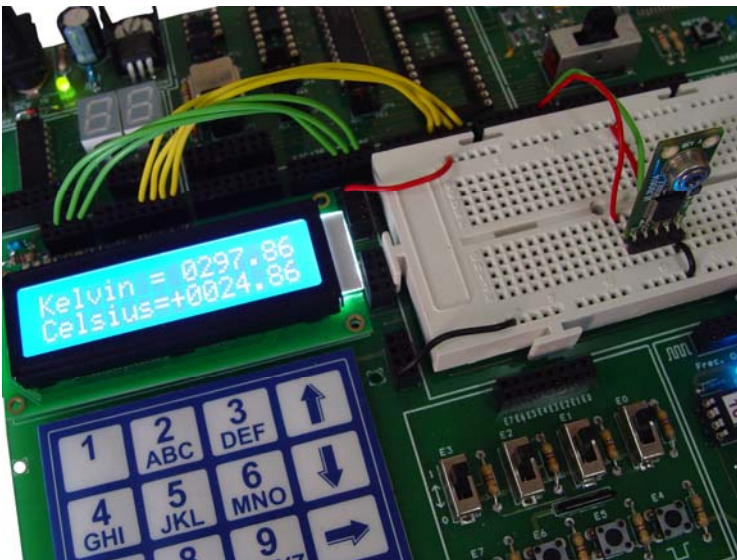
Objetivos

El ejemplo consiste en medir la temperatura y visualizarla sobre la pantalla LCD.

Esquema

Se emplea el mismo esquema de conexiones que en el ejemplo anterior

Comentarios



En la pantalla LCD se visualiza la temperatura tanto en Kelvin como en °C (celsius) El resultado que ofrece el termómetro expresa 0.02 unidades de kelvin. Este se convierte a grados kelvin y a grados centígrados antes de visualizarse tal y como se muestra en la fotografía de la figura 6.

Recordemos que la medida se realiza sin contacto con el sensor. Basta con acercarlo al mismo

Figura 6. Visualizando la temperatura

6.3 Ejemplo 3: Ajuste de la temperatura de la alarma.

Objetivos

Este ejemplo muestra una de las formas de ajustar la temperatura a la que debe saltar la alarma en caso de superarse. Emplea la función IR_Alarma de la librería.

Esquema

Se muestra en la figura 7. Con respecto al esquema de los dos ejemplos anteriores, lo único que se añade es el led S0 conectado con la salida ALR del módulo.

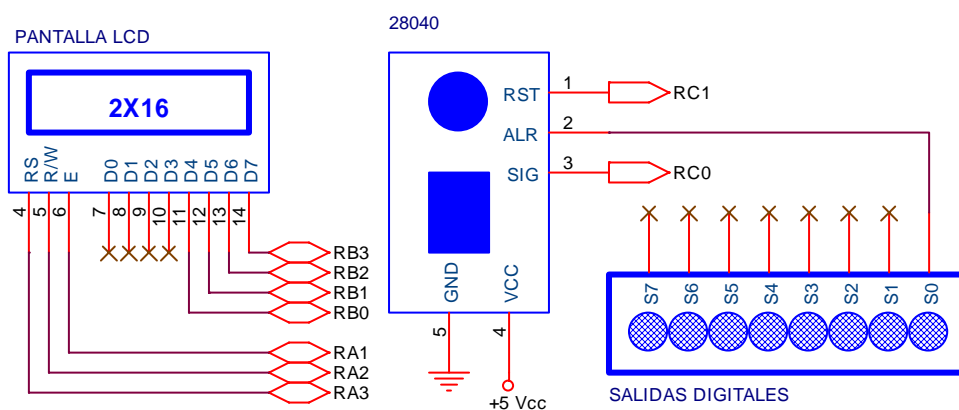


Figura 7. Esquema de montaje para el ejemplo 3

Comentarios

El usuario establece la temperatura de alarma deseada y se le envía al termómetro. La salida ALR de éste se conecta con uno de los leds de salida del laboratorio. Este se activará cada vez que la temperatura medida supere a la de la alarma ajustada. En la pantalla LCD se visualiza la temperatura tanto en Kelvin como en °C (celsius).

Se puede comprobar que una vez se ha grabado la temperatura deseada, el módulo 28040 es totalmente independiente respecto a nuestro controlador Host. Efectivamente, manteniendo únicamente su alimentación, el módulo se comporta como un termostato que dispara una señal cada vez que se supera la temperatura ajustada.

6.4 Ejemplo 4: Ajuste de la temperatura de la alarma.

Objetivos

Nos muestra otra forma de realizar el ajuste de la temperatura de la alarma. Emplea la función IR_Write de la librería. Esta función es de propósito general y se puede emplear para realizar éste y otros ajustes y configuraciones

Esquema

Se muestra en la figura 8. En este caso la salida de alarma del módulo 28040 se conecta con la entrada RC2 del PIC. La salida RC3 de éste mismo se conecta con el led S7 que se activará cada vez que el micro detecte que la temperatura medida es superior a la ajustada como alarma.

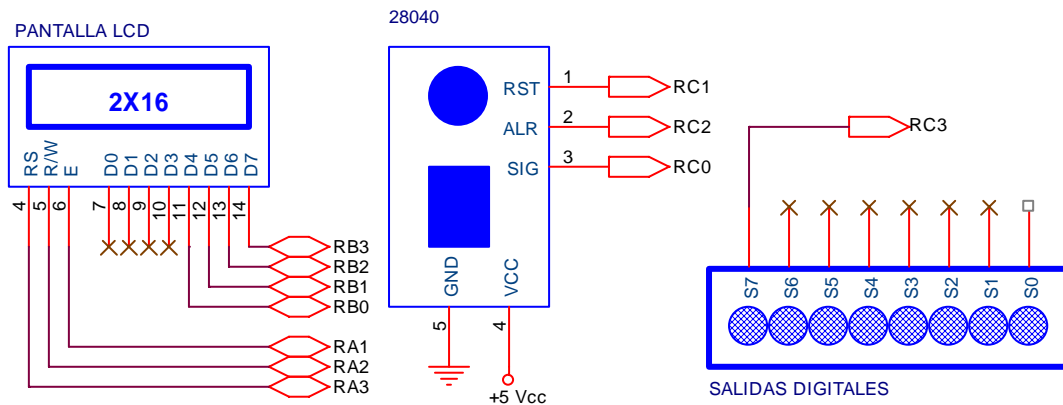


Figura 8. Esquema de montaje del ejemplo 4

Comentarios

El ejemplo muestra otra forma de ajuste de la temperatura de alarma mediante el empleo de la función IR_Write. Esta función se emplea también para escribir y ajustar otros parámetros tales como intervalos, dirección de esclavo y modo de trabajo por defecto.

La salida ALR del 28040 se conecta con la entrada RC2. Cada vez que se produzca un exceso de temperatura, se activa la salida RC3 conectada a un led y en la pantalla LCD aparecerá el mensaje oportuno. En la pantalla LCD se visualiza la temperatura en °C (celsius).

En esta ocasión interviene el controlador para detectar una situación de alarma y actuar en consecuencia. La fotografía de la figura 9 se muestra la ejecución del ejemplo. La temperatura de alarma se ajusta a 100°C. Cuando se le acerca un objeto (p.e. la punta de un soldador) y se supera esa temperatura. El programa los detecta, presenta un mensaje en el LCD y activa la salida S7 conectada con RC3.

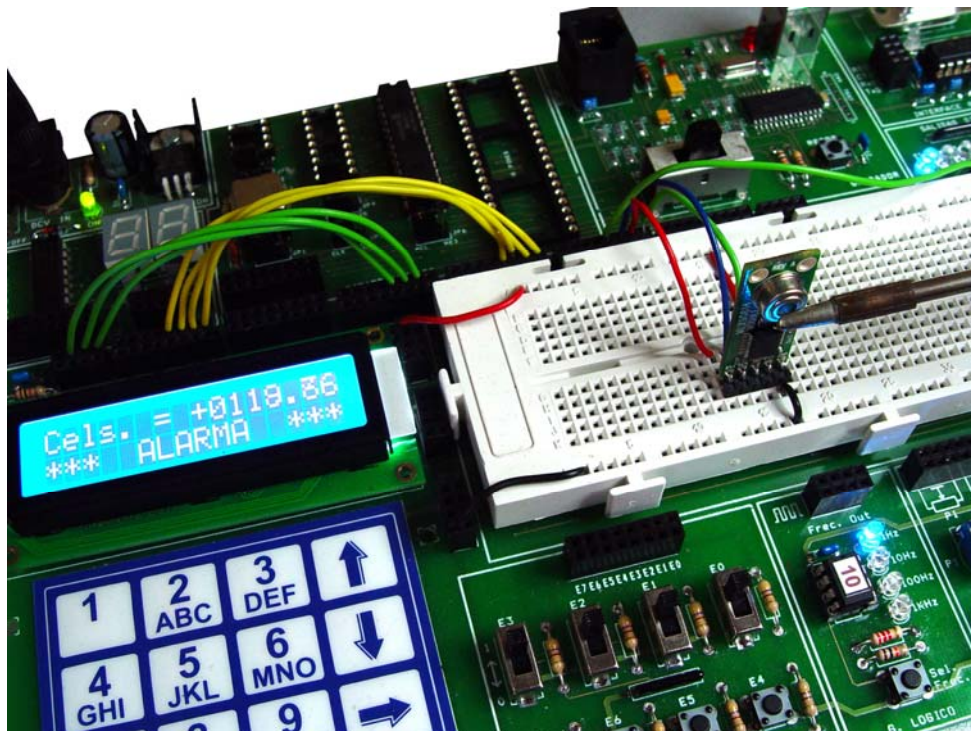


Figura 9. Ejecución del ejemplo 4

