

Acelerómetro de 2 ejes #28017

1.- INTRODUCCION

El módulo 28017 consiste en un acelerómetro de 2 ejes basado en el sensor MX2125 de Memsic. Es capaz de medir ángulos de giro, colisiones, aceleración, rotación y vibraciones en un rango de hasta +/- 3 g sobre los dos ejes X e Y. El sensor es un circuito integrado con formato SMD que está fijado sobre una pequeña plaquita de circuito impreso diseñada por Parallax, que facilita el acceso a todas las señales del mismo así como su inserción en cualquier tarjeta de aplicación, prototipos o módulos board.

2.- CARACTERISTICAS

A continuación se resumen las más relevantes:

- Medida de hasta +/- 3 g en ambos ejes X e Y
- Cada eje proporciona una señal de salida modulada en anchura (PWM) que representa la posición de los mismos.
- Se presenta en un pequeño circuito impreso con 6 patillas en formato DIP y con paso 2.54mm como se muestra en la figura 1.
- Salida analógica de temperatura por la patilla TOut
- Totalmente compensado en el rango de temperatura de 0 a 70°
- Abundante información suministrada por el fabricante en www.memsic.com
- Se adjunta un CDROM con librerías y ejemplos desarrollados por Ingeniería de Microsistemas Programados escritos en lenguaje C y en ensamblador.

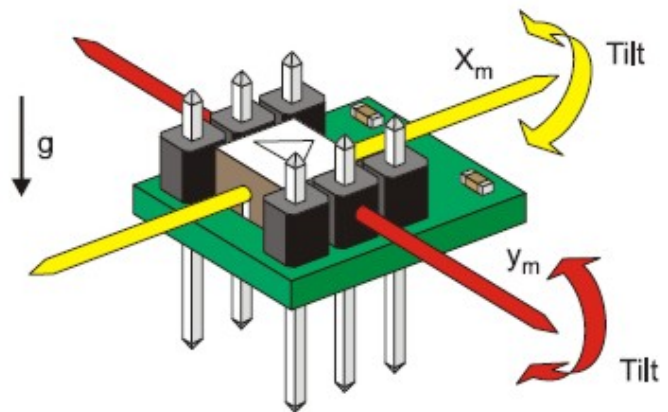


Figura 1. El módulo 28017

3.- ESPECIFICACION Y APLICACIONES

Las eléctricas y mecánicas más importantes del acelerómetro 28017 se citan a continuación:

- Tensión de alimentación de 3.3 hasta 5V con un consumo inferior a 5mA
- Señal de salida PWM de 100Hz (periodo de 10000 µS) y con un ciclo útil proporcional a la aceleración. Compatible con TTL y CMOS.
- Dimensiones de 10.7 x 10.7 x 11.8 mm
- Rango de temperatura de 0 a 70°C

Entre sus múltiples aplicaciones cabe citar:

- Detección de aceleración en los ejes X e Y para sistemas móviles (p.e. robots)
- Detección de movimientos para mandos de juegos (p.e. similar a los mandos de la Wii)
- Sistemas de alarmas por movimiento.
- Detección de impactos.
- Monitorización de movimientos humanos
- Control de piloto automático en sistemas o modelos de RC

4. TEORIA DE FUNCIONAMIENTO

El sensor MX2125 que incorpora el módulo 28017 de Parallax consta de un receptáculo cuadrado con un elemento calorífico que calienta una burbuja de gas, y cuatro sensores de temperatura o termopilas a cada lado del receptáculo como se muestra en la figura 2.

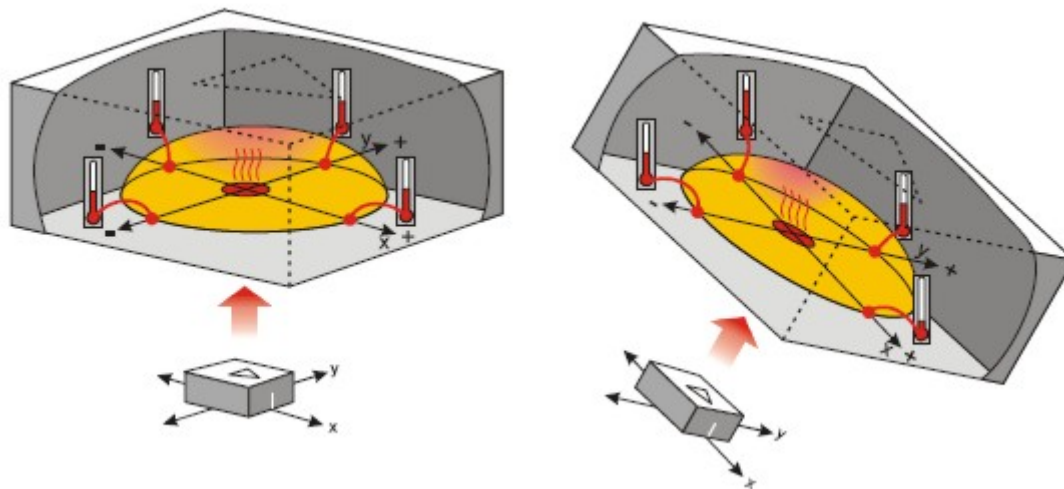


Figura 2. Interior del sensor MX2125

Cuando el sensor se encuentra totalmente nivelado, la burbuja de gas caliente se desplaza centrada hacia arriba en el interior del receptáculo. Los cuatro sensores de temperatura o termopilas registran un mismo valor térmico.

Cuando el sensor sufre algún tipo de giro en cualquiera de sus ejes, la burbuja de gas se desplaza en el interior de la carcasa provocando un aumento de temperatura en algunas de las termopilas y reducción de temperatura en otras. Comparando esas temperaturas se detecta tanto la aceleración dinámica como estática (gravedad y giro). La electrónica integrada en el sensor MX2125 convierte las medidas de temperatura en señales PWM fácilmente manejables por cualquier controlador (PIC, Atmel, Basica Stamp, Arduino, etc..)

5.- PATILLAJE

El encapsulado y distribución de las patillas del módulo acelerómetro 28017 se muestra en la figura 3 y la descripción de las mismas en la tabla adjunta.

Pin Nº	Nombre	Descripción
1	Tout	Salida analógica de temperatura
2	Yout	Salida PWM del eje Y
3	GND	Tierra de alimentación
4	GND	Tierra de alimentación
5	Xout	Salida PWM del eje X
6	Vdd	Alimentación de +3.3V hasta +5V

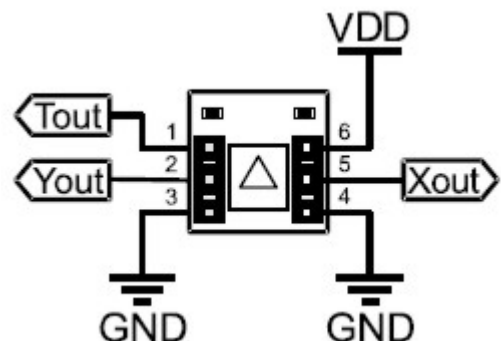


Figura 3. Distribución de patillas del 28017

6. PROTOCOLO DE COMUNICACIÓN

Cada eje proporciona una señal PWM, como la mostrada en la figura 4, con una frecuencia de 100Hz (periodo de 10000 µS) cuya anchura es proporcional a la aceleración. Cualquier tipo de controlador puede medir la duración de la anchura del pulso y obtener así los valores de aceleración.

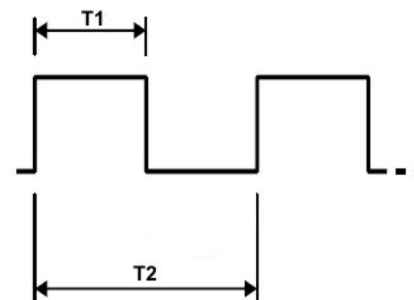


Figura 4. Señal de salida para cada eje

Con una alimentación de +5V, una aceleración de 0g se corresponde con el 50% del ciclo útil aunque puede fluctuar entre 48.7% y 51.3%. Esto significa que si la señal PWM de salida tiene un periodo de 10000 µS (100 Hz), una anchura del ciclo útil de 5000 µS (50%) se corresponde con una aceleración de 0 g. La aceleración se calcula con la siguiente fórmula proporcionada por el fabricante:

$$A(g) = ((T1 / T2) - 0.5) / 12.5 \%$$

De cara a emplear un microcontrolador, la siguiente ecuación es equivalente y facilita los cálculos que hay que realizar:

$$A(g) = (((T1 / 10) - 500) * 8) / 1000$$

Conocida la aceleración de cualquiera de los ejes, también se puede calcular el ángulo de giro del mismo mediante la función arco seno (asin) de g:

$$\text{ángulo} = \text{asin}(g)$$

Algunos compiladores como el compilador C de la firma CCS que hemos empleado en los ejemplos que proponemos, el resultado de la función asin() se ofrece en radianes. Para pasarlos a grados sexagesimales se realiza la siguiente operación:

$$1 \text{ Radián} = 360/2\pi = 57.295779^\circ$$

7. EJEMPLOS

Junto con el módulo acelerómetro 28017 se adjunta un CDROM que contiene librerías y ejemplos sencillos realizados por Ingeniería de Microsistemas Programados S.L., que muestran el funcionamiento de este dispositivo. Se proporcionan los programas fuente escritos en ensamblador 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 a 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:

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: MSE_Mat_PIC16.inc

Esta librería contiene una serie de funciones que resuelven las operaciones matemáticas más elementales y que son empleadas por algunos de los ejemplos escritos en ensamblador. Los ejemplos escritos en C no necesita de esta librería ya que la mayor parte de las funciones matemáticas están integradas en el propio compilador.

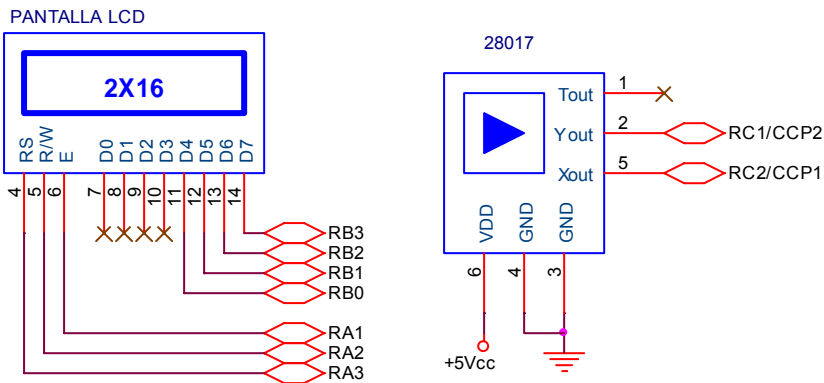
NOMBRE	PARAM. DE ENTRADA	PARAM. DE SALIDA	DESCRIPCION
Sum_BCD	Mat_Dato_AL+Mat_Dato_BL	Mat_Dato_BL Mat_Dato_AH	Suma dos números BCD (00-99): Mat_Dato_AL + Mat_Dato_BL. El resultado se almacena en Mat_Dato_BL. En Mat_Dato_AH queda, si lo hubiera, el overflow (resultado mayor de 99)
Sub_BCD	Mat_Dato_AL- Mat_Dato_BL		Resta dos números BCD (00-99): Mat_Dato_AL - Mat_Dato_BL. El resultado se almacena en Mat_Dato_AL. En Mat_Dato_BH queda, si lo hubiera, el overflow (A<B).
Sum16	Mat_Dato_BH:Mat_Dato_BL+ Mat_Dato_AH:Mat_Dato_AL	Mat_Dato_BH Mat_Dato_BL C	Suma dos números de 16 bits. Mat_Dato_BH:Mat_Dato_BL+ Mat_Dato_AH:Mat_Dato_AL. El resultado se deposita en Mat_Dato_BH:Mat_Dato_BL. Si hay llevada en el bit 16º, C=1
Sub16	Mat_Dato_AH:Mat_Dato_AL- Mat_Dato_BH:Mat_Dato_BL.	Mat_Dato_BH Mat_Dato_BL C	Resta dos números de 16 bits. Mat_Dato_AH:Mat_Dato_AL- Mat_Dato_BH:Mat_Dato_BL. El resultado se almacena en Mat_Dato_BH:Mat_Dato_BL. Si hay llevada en el bit 16º, C=0
Comp16	Mat_Dato_AH:Mat_Dato_AL(A) Mat_Dato_BH:Mat_Dato_BL(B)	C, Z	Compara dos nº de 16 bits contenidos en Mat_Dato_AH:Mat_Dato_AL(A) con Mat_Dato_BH:Mat_Dato_BL(B). Si A>B --> STATUS<C>=1 y <Z>=0; Si A<B --> STATUS<C>=0 y <Z>=0; Si A=B --> STATUS<C>=1 y <Z>=1
Mul8x8	Mat_Dato_AL * Mat_Dato_BL.	Mat_RES2:Mat_RES3	Multiplica dos nºs de 8 bits contenidos en Mat_Dato_AL y Mat_Dato_BL. El resultado de 16 bits se almacena en Mat_RES2 y Mat_RES3 (LSB).
Mul16x16	Mat_Dato_BH:Mat_Dato_BL* Mat_Dato_AH:Mat_Dato_AL	Mat_RES0:Mat_RES3	Multiplica dos números de 16 bits: Mat_Dato_BH:Mat_Dato_BL* Mat_Dato_AH:Mat_Dato_AL. El resultado de 32 bits se almacena en Mat_RES0:Mat_RES3 (LSB)
Div16x16	Mat_Dato_AH:Mat_Dato_AL/ Mat_Dato_BH:Mat_Dato_BL	Mat_Dato_AH:Mat_Dato_AL Mat_RES3:Mat_RES2	Esta rutina divide dos números de 16 bits. El dividendo se almacena en Mat_Dato_AH:Mat_Dato_AL y el divisor en Mat_Dato_BH:Mat_Dato_BL. El cociente se almacena en Mat_Dato_AH:Mat_Dato_AL y el resto en Mat_RES3:Mat_RES2.
Bits8_BCD	W	Mat_RES0	Convierte un número binario de 8 bits en el registro W, en 2 dígitos BCD (de 00 a 99). El resultado se almacena en Mat_RES0. P.e. W=0x2A, Mat_RES0=42 (0x2A = 42).
Bits16_BCD	Mat_Dato_AH:Mat_Dato_AL	Mat_RES0 Mat_RES1 Mat_RES2 (LSB)	Esta rutina convierte un número binario de 16 bits situado en Mat_Dato_AH y Mat_Dato_AL y, lo convierte en 5 dígitos BCD que se depositan en las variables Mat_RES0, Mat_RES1 y Mat_RES2, siendo esta última la de menos peso.
BCD_Bits16	Mat_RES0:Mat_RES2 (LSB)	Mat_Dato_AH:Mat_Dato_AL	Convierte un número de 5 dígitos en BCD, en un número de 16 bits. En Mat_RES0:Mat_RES2 (LSB) se encuentra en nº de 5 dígitos. El resultado se almacena en Mat_Dato_AH:Mat_Dato_AL.
BCD_ASCII	Mat_RES0:Mat_RES2 (LSB) Mat_Point	Mat_RES0(MSB):Mat_RES6(LSB)	Convierte un número BCD de hasta 6 dígitos almacenados en Mat_RES0:Mat_RES2 (LSB) En una cadena ASCII que se almacena a partir de Mat_RES0(MSB) hasta Mat_RES6(LSB). La variable Mat_Point indica el lugar donde colocar un punto decimal (0 =ninguno). P.e. Mat_Point=1, el resultado sería .xxxxxx; Mat_Point=3, el resultado sería xx.xxxx

6.1 Ejemplo 1: Medir la anchura del pulso de los ejes X e Y

Objetivos

Se trata de un ejemplo básico consistente en leer el valor de la anchura de los pulsos que ofrecen las salidas X e Y del sensor y visualizarlas sobre el LCD.

Esquema



La figura 5 muestra el esquema empleado para el desarrollo de éste y otros ejemplos posteriores. La pantalla LCD se gestiona mediante un interface de 4 bits de datos con las líneas RB0:RB3 del controlador y mediante las señales de control E, R/W y RS que se conectan con RA1:RA3 respectivamente. El módulo acelerómetro dispone de las salidas Xout e Yout por las ofrece sendas señales PWM. Se conectan con las entradas RC2 y RC1 que acceden a los módulos CCP1 y CCP2 del PIC.

Figura 5. Esquema de montaje del ejemplo 1

Comentarios

La señal PWM que genera el sensor expresa en μs la posición de los dos ejes X e Y. Cuando el grado de inclinación en ambos es de 0° , las señales de salida PWM para ambos ejes es del 50% del ciclo. Según el fabricante dicho ciclo es de $10000\mu\text{s}$ (100Hz), por lo que las señales tendrán un ciclo útil (duty) de unos $5000\mu\text{s}$ en reposo (el sensor sobre un plano horizontal). Cualquier movimiento del sensor sobre el eje X o Y, hará que la anchura de salida correspondiente aumente o disminuya en función del sentido del movimiento.

La rutina **Read28017** se encarga de leer, secuencialmente, la anchura de los pulsos que ofrecen las salidas de los eje X e Y. Emplea para ello los módulos CCP1 y CCP2 del PIC. Estos capturan el valor del Timer 1 cada vez que se detecta un flanco descendente (fin del ciclo útil) por la patilla CCP1 y CCP1. El Timer 1 se pone a 0 cuando se detecta un flanco ascendente (inicio del ciclo útil) y evoluciona cada $1\mu\text{s}$. Así, los módulos CCP1 y CCP2 contienen un valor que expresan, en μs , la duración de las señales PWM de los ejes X e Y respectivamente.

El valor capturado para ambos ejes se divide entre 10 para, de alguna manera, filtrar ruidos y variaciones provocadas por variaciones de temperatura. En la pantalla LCD se visualiza, como se muestra en la fotografía de la figura 6, el valor de ambos ejes. Se recuerda que cuando el sensor está en absoluto reposo, el ciclo útil tiene una anchura del 50% del periodo de la señal PWM. Esto significa que la anchura de los pulsos debe ser de unos $5000\mu\text{s}$ que, divididos entre 10 nos dará el valor de 500 aproximadamente.

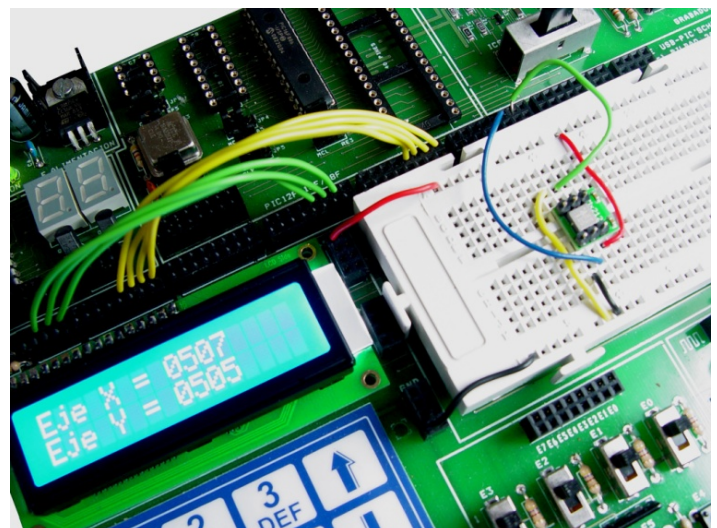


Figura 6. Ejecución del ejemplo 1

6.2 Ejemplo 2: Medida de la posición relativa de los dos ejes.

Objetivos

Medir y visualizar sobre la pantalla LCD la duración del ciclo útil de las señales de los ejes X e Y, relativas a una posición de calibración o referencia.

Esquema

El mismo que en el ejemplo 1

Comentarios



El programa comienza tomando una muestra del valor del eje X y del Y cuando el sensor se supone en reposo. Realiza una calibración inicial con valor 0 en reposo. En función de los giros o movimientos realizados en cada eje, la pantalla LCD visualiza el signo y el valor de esos giros respecto a la posición 0 de calibración. Así pues disponemos de información sobre los movimientos relativos al centro que realiza el sensor.

Los valores y el signo que aparecen en la pantalla LCD indican claramente, como se muestra en la figura 7, el tipo de movimiento realizado y pueden usarse en todo tipo de aplicaciones que requieran ese tipo de datos (posición, desplazamiento, etc..)

Figura 7. Ejecución del ejemplo 3

6.3 Ejemplo 3: Detección de movimiento.

Objetivos

Realizar un ejemplo de aplicación consistente en una alarma que se activará cuando se detecte un determinado movimiento del sensor en cualquiera de sus ejes.

Esquema

Es el mismo que el empleado en los ejemplos anteriores al que se añade el mostrado en la figura 8. Simplemente la salida RA0 se conecta con el led S7 del laboratorio USB-PIC'School y que servirá para señalar el estado de alarma.

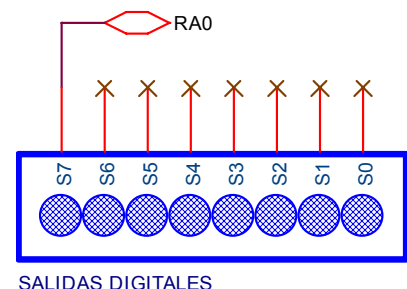


Figura 8. Esquema necesario para el ejemplo 3

Comentarios

El programa comienza tomando una muestra del valor del eje X y del Y cuando el sensor se supone en reposo, nivelado. Realiza una calibración inicial con valor 0.

Quando se detecta un movimiento del sensor en cualquier sentido de cualquier eje, superior a un determinado máximo permitido (con relación al valor de calibración), se dispara una situación de alarma que se visualiza en el LCD y produce una intermitencia por la salida RA0 conectada a un led del laboratorio, tal y como se muestra en la fotografía de la figura 9.

Se trata de una posible aplicación real de alarma que se activa al desplazar un objeto o cuando sufre un movimiento brusco. El umbral de disparo se puede modificar variando las constantes "X_Max" e "Y_Max" del programa.

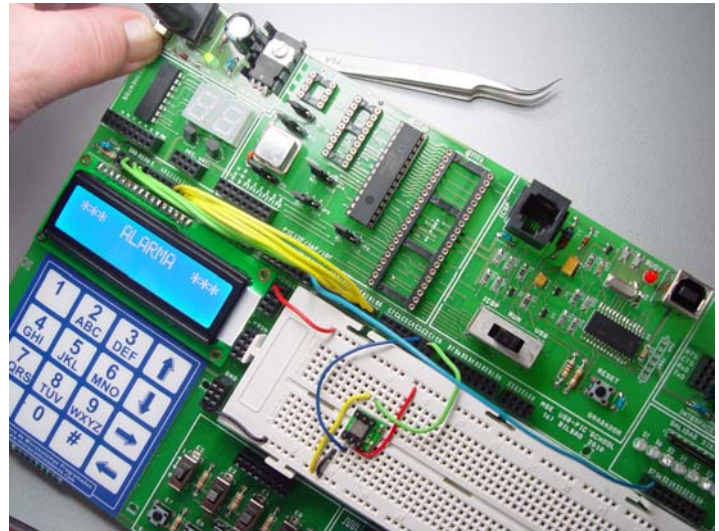


Figura 9. Ejecución del ejemplo 3

6.4 Ejemplo 4: Cálculo y visualización de la aceleración (g).

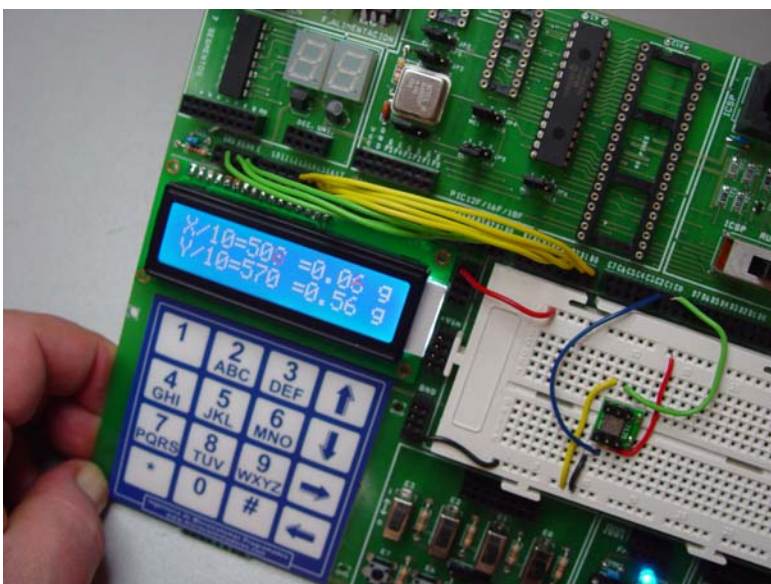
Objetivos

Una vez que se conoce el valor del ciclo útil de las señales que proporcionan cualquiera de los ejes y que representa el desplazamiento, podemos calcular y visualizar la aceleración ejercida en esos ejes.

Esquema

El mismo que en el ejemplo 1

Comentarios



El programa toma muestras del valor del ciclo útil, en uS, tanto del eje X como del Y. Los visualiza y, a partir de ellos, calcula y visualiza la aceleración (g) según la ecuación del fabricante:

$$A(g) = ((T1/T2)-0.5)/12.5\%$$

T1 representa en uS el valor del ciclo útil de la señal PWM que proporciona el sensor y T2 representa el valor en uS del periodo de la señal (10000uS a 100Hz). Para facilitar los cálculos, esa ecuación se puede convertir en esta otra equivalente:

$$A(g) = ((T1/10)-500)*8, \text{ siendo el resultado en milésimas de g}$$

Figura 10. Ejecución del ejemplo 4

En la pantalla LCD se visualiza el valor de la señal PWM de cada eje dividido entre 10, y la aceleración (g) correspondiente. En el ejemplo no se contempla una calibración inicial y su ejecución se muestra en la figura 10.

6.5 Ejemplo 5: Cálculo y visualización del ángulo.

Objetivos

Conocida la anchura del pulso de las señales que proporcionan ambos ejes, se calcula la aceleración (g) como en el anterior ejemplo y de aquí se obtiene el ángulo de giro de cualquiera de los ejes. Dados los cálculos que hay que realizar, este ejemplo se ha resuelto en lenguaje C. En caso de no disponer del compilador apropiado, el usuario siempre puede usar el ejecutable correspondiente adjuntado en el CDROM

Esquema

El mismo que en el ejemplo 1

Comentarios

El programa toma muestras del valor del ciclo útil, en uS, tanto del eje X como del Y. Los visualiza y, a partir de ellos, calcula y visualiza la aceleración (g) según la ecuación del fabricante:

$$A(g) = ((T1/T2)-0.5)/12.5\%$$

T1 representa en uS el valor del ciclo útil de la señal PWM que proporciona el sensor y T2 representa el valor en uS del periodo de la señal (10000uS a 100Hz). Para facilitar los cálculos, esa ecuación se puede convertir en esta otra equivalente:

$$A(g) = ((T1/10)-500)*8, \text{ siendo el resultado en milésimas de g}$$

Conocido el valor de (g) podemos calcular el ángulo desplazado en cada eje según:

$$\text{Ángulo} = \arcseno(g) = \text{asin}(g)$$

El compilador C de la firma CCS que hemos empleado, ofrece un resultado en radianes de la función asin(). Para convertir esos radianes en grados sexagesimales basta con aplicar:

$$1 \text{ Radián} = 360/2\pi = 57.295779^\circ$$

En la pantalla LCD se visualiza el valor de la señal PWM de cada eje dividido entre 10 y respecto al punto de calibración inicial, la aceleración (g) y el ángulo correspondiente. En el ejemplo se contempla la calibración inicial que se toma como valor relativo de referencia, por lo que en el cálculo de la aceleración se elimina la constante 500. Se hace notar que con inclinaciones superiores a los 60° aprox. el sensor deja de comportarse de forma lineal según las especificaciones del fabricante, por lo que la medida del ángulo pierde precisión. Según la figura 11 hemos inclinado el laboratorio mediante el empleo de un cartabón, por lo que el ángulo de giro en el eje X es de unos 30°

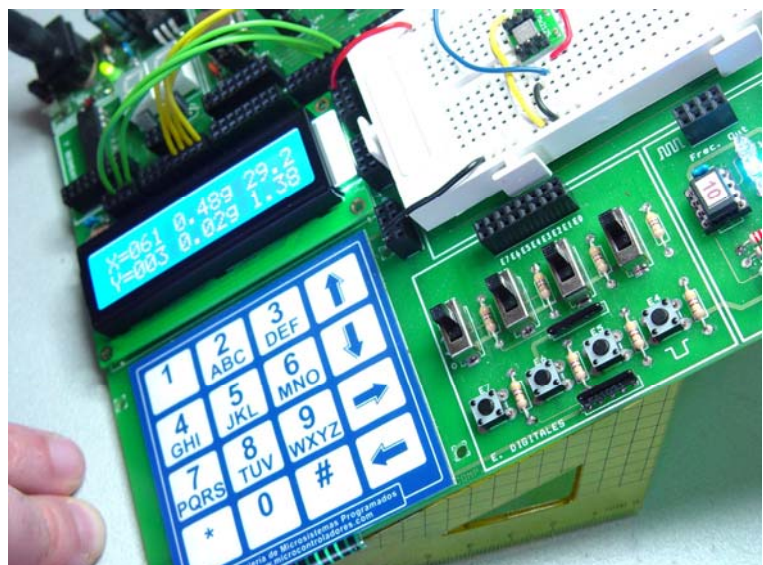


Figura 11. Ejecución del ejemplo 5