

Basic Express Nota de aplicación

Utilizar un acelerómetro ADXL202 con BasicX

Introducción

El dispositivo analógico ADXL202 es un acelerómetro de 2 ejes bastante asequible que puede conectarse fácilmente a pequeños microcontroladores de 8 bits. Este dispositivo es capaz de leer aceleraciones estáticas y dinámicas con un rango de medidas que oscilan entre $\pm 19.6 \text{ m/s}^2$ ($\pm 2 \text{ g}$). La resolución depende del ancho de banda, y es de 49 mm/s^2 (5 mg) a 60 Hz.

El sensor genera un pulso con una señal con un ancho de banda modulada (PWM) para cada uno de los dos canales de salida. El componente de aceleración de cada eje se deriva fácilmente al medir los ciclos de funcionamiento (duty cycle) de cada señal PWM, que significa que no es necesario utilizar ningún convertidor de señal analógica a digital (ADC).

Interfaz de Hardware

Figura 1 (abajo) indica cómo se debe conectar un acelerómetro ADXL202EB, que es la placa de evaluación del ADXL202:

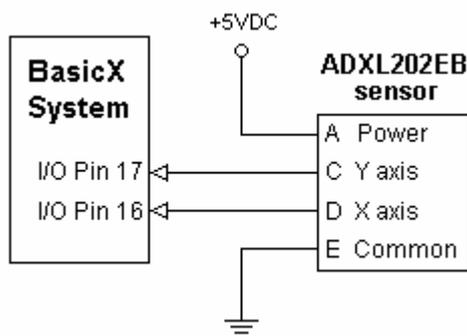


Figura a Conexión de un sensor al sistema BasicX

Tenga en cuenta que es posible que BasicX y el sensor compartan la misma fuente de alimentación de +5 VDC, aunque esta opción puede presentar algún ruido. Se debería utilizar por tanto una fuente de alimentación independiente si desee minimizar el ruido.

Interfaz de Software

Estos son los símbolos utilizados en esta sección:

- A_X = componente X del vector de aceleración, unidades no dimensionales de gravedades
- A_Y = componente Y del vector de aceleración, unidades no dimensionales de gravedades

- K_1 = Constante, no dimensional
- K_2 = Constante, no dimensional
- P = Periodo de señal PWM, segundos
- T_1 = Duración del pulso ascendente, segundos
- T_3 = Duración del pulso descendente, segundos

Figura 2 (abajo) ilustra la generación de una señal PWM por el canal 1:

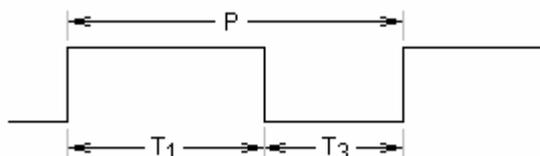


Figura 2 Señal PWM

El componente A_x del eje x está determinado de la siguiente manera (A_y es similar):

$$A_x = \frac{T_1 / P - K_1}{K_2}$$

Las constantes K_1 y K_2 varían en cierto modo entre sensores y deben estar determinado empíricamente. Los valores nominales son 0.5 para K_1 y 0.125 para K_2 .

El primer paso en la medición de la aceleración es determinar el periodo P . Podemos hacer llamadas sucesivas a "PulseIn" para medir los anchos de los pulsos ascendentes o descendentes en Figura 2. Se toman y se hacen los promedios de algunas muestras para obtener el periodo. En el siguiente ejemplo tomamos 20 muestras:

```

Const PinX As Byte = 16, PinY As Byte = 17
Dim SumT1 As Single, SumT3 As Single, Period As Single
Dim T1 As Single, T3 As Single, i As Integer
Dim AvgT1 As Single, AvgT3 As Single
Const NSamples As Integer = 20
SumT1 = 0.0
SumT3 = 0.0
For i = 1 to NSamples
    Call PulseIn(PinX, 1, T1) ' High-going pulse.
    Call PulseIn(PinX, 0, T3) ' Low-going pulse.
    SumT1 = SumT1 + T1
    SumT3 = SumT3 + T3
Next
AvgT1 = SumT1 / CSng(NSamples)
AvgT3 = SumT3 / CSng(NSamples)
Period = AvgT1 + AvgT3 ' Units are in seconds.
    
```

El periodo cambia relativamente lento, por lo que no es necesario realizar este proceso más de una vez por minuto aproximadamente. Así mismo, sólo es necesario medir un eje, dado que el periodo es el mismo para los dos ejes.

Debe tener en cuenta que “PulseIn” devuelve unidades de segundos de valores decimales (floating points), que facilita la escritura de expresiones. “PulseIn” está sobrecargado – la versión de valor entero (*integer*) puede utilizarse si se necesita un rendimiento superior.

Una vez que se conoce el periodo, es muy sencillo medir cada eje. El siguiente proceso da como resultado el componente de aceleración filtrado de cada eje:

```
Sub GetAccelerations( _
    ByVal Period As Single, _
    ByRef Ax As Single, _
    ByRef Ay As Single)
' Este proceso lee cada eje del acelerómetro ADXL202.
' Tiene como resultado ambos componentes de un vector de aceleración 2D.
' Las unidades se expresan en gravedades
Dim T1X As Single, T1Y As Single
Dim SumX As Single, SumY As Single
Dim i As Byte, T1 As Single
Const NSamples As Byte = 20
Const PinX As Byte = 16, PinY As Byte = 17
SumX = 0.0
SumY = 0.0
For i = 1 to NSamples
    Call PulseIn(PinX, 1, T1)
    SumX = SumX + T1
    Call PulseIn(PinY, 1, T1)
    SumY = SumY + T1
Next
' Se realizan los promedios de cada eje.
T1X = SumX / CSng(NSamples)
T1Y = SumY / CSng(NSamples)
' Se determina los componentes de aceleración.
Ax = ((T1X / Period) - 0.5) / 0.125
Ay = ((T1Y / Period) - 0.5) / 0.125
End Sub
```

Programas de ejemplo

Los dos programas de ejemplo se proporcionan como ficheros independientes. El programa “DisplayAcceleration” simplemente muestra los componentes de un vector de aceleración 2D expresados en unidades de cm/s^2 . El programa “DisplayAngle” muestra los ángulos de inclinación del acelerómetro.

© 1998-2001 by NetMedia, Inc. All rights reserved.

Basic Express, BasicX, BX-01, BX-24 and BX-35 are trademarks of NetMedia, Inc.

All other trademarks are the property of their respective owners.

2.00.A

Traducción Española: Alicia Bernal, revisión: Pablo Pompa www.superrobotica.com