Using the ADXL202 Accelerometer with BasicX

Introduction

The Analog Devices ADXL202 is a low cost 2 axis accelerometer that can be easily connected to small 8-bit microcontrollers. The device is capable of reading both static and dynamic accelerations with a measurement range of ±19.6 m/s^2 (±2 *g*). Resolution depends on bandwidth, and is 49 mm/s^2 (5 m*g*) at 60 Hz.

The sensor generates a pulse width modulated (PWM) signal on each of the 2 output channels. The acceleration component of each axis is easily derived by measuring the duty cycle of each PWM signal, which means an Analog to Digital Converted (ADC) is not required.

Hardware interface

Figure 1 (below) illustrates how to connect an ADXL202EB, which is the evaluation board version of the ADXL202:
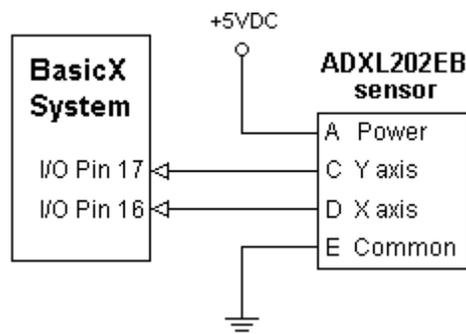
Figure 1
Connecting sensor to BasicX system

Note that it is possible to have the BasicX and sensor share the same +5 VDC power source, but this is not the lowest noise option. A separate power supply should be used if you want to minimize noise.

Software interface

Symbols used in this section:

A_X = X component of acceleration vector, nondimensional units of gravities
A_Y = Y component of acceleration vector, nondimensional units of gravities
K_1 = Constant, nondimensional
K_2 = Constant, nondimensional
P = Period of PWM signal, seconds
T_1 = Duration of high-going pulse, seconds
T_3 = Duration of low-going pulse, seconds

Figure 2 (below) illustrates the PWM signal generated for 1 channel:
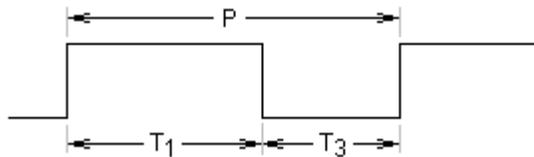
Figure 2
PWM signal

The X-axis acceleration component A_X is determined as follows (A_Y is similar):

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

The constants K_1 and K_2 vary somewhat between sensors and need to be determined empirically. Nominal values are 0.5 for K_1 and 0.125 for K_2 .

The first step in measuring acceleration is to determine the period P. We can make successive calls to PulseIn to measure the widths of the high-going and low-going pulses in Figure 2. Several samples can be taken and averaged to get the period. Here we take 20 samples:

```
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.
```

The period drifts relatively slowly, so this procedure doesn't need to be done more than once per minute or so. Also, only 1 axis needs to be measured, since the period is the same for both axes.

Note that PulseIn returns units of floating point seconds, which makes expressions easier to write. PulseIn is overloaded -- the integer version can be used if higher performance is required.

Once the period is known, it is simple to measure each axis. The following procedure returns the filtered acceleration component of each axis:

```
Sub GetAccelerations( _
    ByVal Period As Single, _
    ByRef Ax As Single, _
    ByRef Ay As Single)

' This procedure reads each axis of an ADXL202 accelerometer. Both
' components of a 2D acceleration vector are returned. Units are
' in gravities.

    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

    ' Take averages for each axis.
    T1X = SumX / CSng(NSamples)
    T1Y = SumY / CSng(NSamples)

    ' Determine the acceleration components.
    Ax = ((T1X / Period) - 0.5) / 0.125
    Ay = ((T1Y / Period) - 0.5) / 0.125

End Sub
```

Example Programs

Two example programs are provided as separate files. Program *DisplayAcceleration* simply displays each component of the 2D acceleration vector in units of cm/s^2. Program *DisplayAngle* displays the tilt angle of the accelerometer.

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