Experiment #18: Advanced Frequency Measurement

This experiment uses **PULSIN** to create a responsive frequency meter.

New PBASIC elements/commands to know:

- PULSIN

Building The Circuit

Use the same circuit as in Experiment #18.

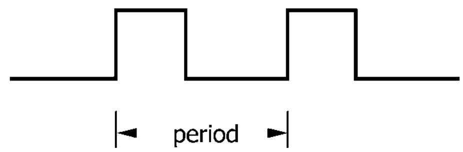
```
' ==============================================================================
'
'   File...... Ex18 - FreqIn2.BS2
'   Purpose... Frequency Input
'   Author.... Parallax
'   E-mail.... stamptech@parallaxinc.com
'   Started...
'   Updated... 01 MAY 2002
'
'   {$STAMP BS2p}
'
' ==============================================================================


' ------------------------------------------------------------------------------
' Program Description
' ------------------------------------------------------------------------------

' This program monitors and displays the frequency of a signal on Pin 0.


' ------------------------------------------------------------------------------
' I/O Definitions
' ------------------------------------------------------------------------------

FreqPin         CON     0                       ' frequency input pin


' ------------------------------------------------------------------------------
' Constants
```

```
' -------------------------------------------------------------------------------

Convert         CON     $0200                   ' input to uSeconds (BS2)
' Convert       CON     $00CC                   ' BS2sx
' Convert       CON     $00C0                   ' BS2p


' -------------------------------------------------------------------------------
' Variables
' -------------------------------------------------------------------------------

pHigh           VAR     Word                    ' high pulse width
pLow            VAR     Word                    ' low pulse width
period          VAR     Word                    ' cycle time (high + low)
freq            VAR     Word                    ' frequency


' -------------------------------------------------------------------------------
' Program Code
' -------------------------------------------------------------------------------

Main:
  PULSIN FreqPin, 0, pHigh                      ' get high portion of input
  PULSIN FreqPin, 1, pLow                       ' get low portion of input
  period = (pHigh + pLow) */ Convert            ' calculate cycle width in uSecs
  freq = 50000 / period * 20                    ' calculate frequency

  ' display on DEBUG screen

  DEBUG Home
  DEBUG "Period...... ", DEC period, " uS    ", CR
  DEBUG "Frequency... ", DEC freq, " Hz    "
  GOTO Main                                     ' do it again

  END
```

Behind The Scenes

In the last experiment, you learned that the frequency of a signal is defined as the number of cycles per second. You created a simple frequency meter by counting the number of pulses (cycles) in one second. This method works well, especially for low-frequency signals. There will be times, however, when project requirements will dictate a quicker response time for frequency measurement.

The frequency of a signal can be calculated from its period, or the time for one complete cycle.



By measuring the period of an incoming signal, its frequency can be calculated with the equation (where the period is expressed in seconds):

frequency = 1 / period

The BASIC Stamp's `PULSIN` function is designed to measure the width of an incoming pulse. By using `PULSIN` to measure the high and low portions of an incoming signal, its period can be calculated and the frequency can be determined. The result of `PULSIN` is expressed in units of two microseconds. Thus, the formula for calculating frequency becomes:

frequency = 500,000 / period

This creates a problem for BASIC Stamp math though, as it can only deal with 16-bit numbers (maximum value is 65,535). To fix the formula, we convert 500,000 to 50,000 x 10 and rewrite the formula like this

frequency = 50,000 / period * 10

Run the program and adjust the 10K pot. Notice that the `DEBUG` screen is updated without delay and that there is no "hunting" as when using `COUNT` to determine frequency.