

# Hitachi<sup>®</sup> H48C 3-Axis Accelerometer Module (#28026)

## General Description

The Hitachi H48C 3-Axis Accelerometer is an integrated module that can sense gravitational (g) force of  $\pm 3g$  on three axes (X, Y, and Z). The module contains an onboard regulator to provide 3.3-volt power to the H48C, analog signal conditioning, and an MCP3204 (four channel, 12-bit) analog-to-digital converter to read the H48C voltage outputs. All components are mounted on a breadboard-friendly, 0.7 by 0.8 inch module. Acquiring measurements from the module is simplified through a synchronous serial interface. With the BASIC Stamp<sup>®</sup> 2 series, for example, this is easily handled with the SHIFTOUT and SHIFTIN commands.

## Features

- Measure  $\pm 3g$  on any axis
- Uses MEMS (Micro Electro-Mechanical System) technology, with compensation for calibration-free operation
- Onboard regulator and high-resolution ADC for simple connection to microcontroller host - compatible with BASIC Stamp 2 series SHIFTOUT and SHIFTIN commands
- Free-fall output indicates simultaneous 0g on all axes
- Small, breadboard-friendly package: 0.7" x 0.8" (17.8 mm x 20.3 mm)
- Wide operational range:  $-25^{\circ}$  to  $75^{\circ}$  C

## Application Ideas

- Tilt measurement in robotics applications
- Multi-axis vibration measurement in transit and shipping systems
- Multi-axis movement/lack-of-movement for alarm systems

## Packing List

Verify that your H48C Accelerometer kit is complete in accordance with the list below:

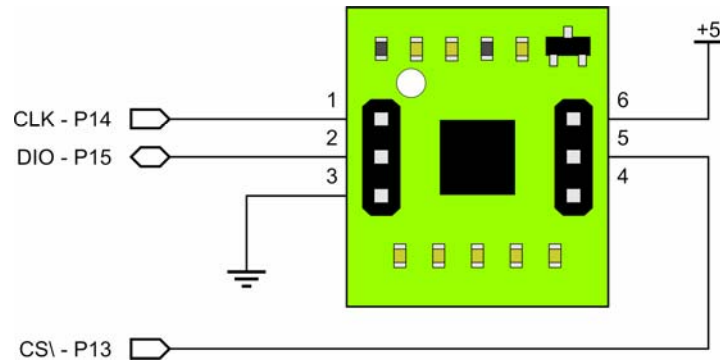
- Hitachi H48C 3-Axis Accelerometer module
- Documentation

Note: Demonstration software files may be downloaded from [www.parallax.com](http://www.parallax.com).  
(See [http://www.parallax.com/detail.asp?product\\_id=28026](http://www.parallax.com/detail.asp?product_id=28026))

## Essential Connections

Connecting the H48C module to the BASIC Stamp 2 controller is a straightforward operation, requiring just three I/O pins (the CLK and DIO pins may be shared in systems requiring the use of more than one H48C module). See Figure 1 for connection details.

Figure 1. H48C Connections



## How It Works

Through MEMS (Micro Electro-Mechanical System) technology and built-in compensation, the H48C accelerometer provides simultaneous outputs through analog conditioning circuitry to an MCP3204 ADC. To "read" g-force of a given axis we actually read the voltage output from that axis and calculate g-force using this formula:

$$G = ((axis - vRef) / 4095) \times (3.3 / 0.3663)$$

In the formula, *axis* and *vRef* are expressed in counts from the ADC, 4095 is the maximum output count from a 12-bit ADC channel, 3.3 is the H48C supply voltage, and 0.3663 is the H48C output voltage for 1g (when operating at 3.3v). In practice this can be simplified to:

$$G = (axis - vRef) \times 0.0022$$

Using the BASIC Stamp 2 module as a host controller, we should multiply the 0.0022 by 100 (to 0.22) to express the result in units of 0.01g. Using the \*\* operator, we are able to multiply by 0.22 and convert the raw readings to g-force with this bit of code:

```
IF (axCount >= rvCount) THEN
  gForce = (axCount - rvCount) ** GfCnv      ' positive g-force
ELSE
  gForce = -((rvCount - axCount) ** GfCnv)  ' negative g-force
ENDIF
```

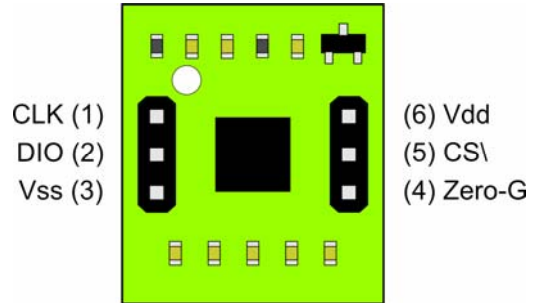
Note the **IF-THEN** structure which prevents a negative number from being divided – this is illegal in PBASIC 2.x and will not return the correct result. By restructuring the conversion equation for negative g-forces we can indeed arrive at the correct value. The output value, *gForce*, is a signed integer.

In application the analog signal conditioning circuitry affects the rate at which readings can be taken H48C module. The filter/buffer circuit is designed to minimize noise while maintaining the highest possible signal resolution into the ADC. By design, the filter circuit limits MC48C axis output rail-to-rail rise/fall time to about five milliseconds. Since MCP3204 has a significantly higher sample rate, the

sampling rate of the module is dictated by the filter circuitry and works out to about 200 samples per second.

## Pin Definitions and Ratings

- (1) CLK Synchronous clock input
- (2) DIO Bi-directional data to/from host
- (3) Vss Power supply ground (0v)
- (4) Zero-G "Free-fall" output; active-high
- (5) CS\ Chip select input; active-low
- (6) Vdd +5vdc



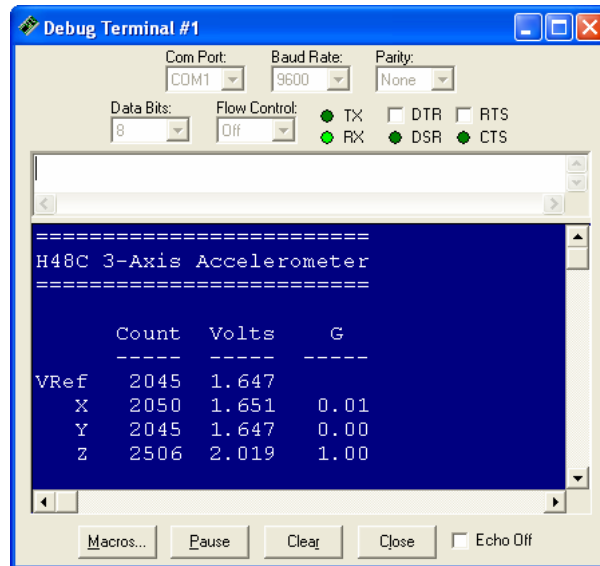
Symbol	Quantity	Minimum	Typical	Maximum	Units
$V_{DD}$	Operating voltage	4.5	5.0	5.5	V
$V_{SS}$	Ground reference connection		0		V
$I_{DD}$	Supply current		7	10	mA
$V_{IH}$	High Level Voltage Input <sup>†</sup>	0.7 $V_{DD}$			V
$V_{IL}$	Low Level Voltage Input <sup>†</sup>			0.3 $V_{DD}$	V
$V_{OH}$	High Level Voltage Output <sup>†</sup>	4.1			V
$V_{OL}$	Low Level Voltage Output <sup>†</sup>			0.4	V
	Sample Rate			200	sps
	ADC (MCP3204) Resolution <sup>†</sup>	12			bits
	Measurement Range <sup>§</sup>	-3		3	g
	Sensitivity <sup>§</sup>		366.3		mV/g
	Accuracy <sup>§</sup>	10			%
	Non-linearity <sup>§</sup>	-2		2	%
	Operating Temperature Range <sup>§</sup>	-25		75	°C
	High Level Zero-G Voltage Output <sup>§</sup>	3.2	3.3		V
	Zero-G Output Delay <sup>§</sup>			1	ms

<sup>†</sup> From Microchip MCP3204 Datasheet

<sup>§</sup> From Hitachi H48C Datasheet

## Demonstration Program

This demonstration uses the BASIC Stamp 2 series microcontroller to read the reference voltage and output channels from the H48C using the onboard MCP3204 analog-to-digital converter. For each channel the raw count, channel voltage, and g-force for the X, Y, and Z axes are displayed as shown below:



## BASIC Stamp 2 Source Code

```
' =====  
'  
' File..... H48C_3-Axis.BS2  
' Purpose.... Hitachi H48C 3-Axis Accelerometer Demonstration  
' Author..... Copyright (c) 2005-2006 Parallax, Inc.  
' E-mail..... support@parallax.com  
' Started....  
' Updated.... 02 FEB 2006  
'  
' {$STAMP BS2}  
' {$PBASIC 2.5}  
'  
' =====  
'  
' ----- [ I/O Definitions ] -----  
'  
Dio          PIN    15          ' data to/from module  
Clk          PIN    14          ' clock output  
CS           PIN    13          ' active-low chip select  
'  
' ----- [ Constants ] -----  
'  
XAxis       CON    0          ' adc channels  
YAxis       CON    1  
ZAxis       CON    2
```

```

VRef          CON      3

Cnt2Mv        CON      $CE4C          ' counts to millivolts
                                           ' 0.80586 with **

GfCnv         CON      $3852          ' g-force conversion
                                           ' 0.22 with **

' ----- [ Variables ] -----

axis          VAR      Nib            ' axis selection
rvCount       VAR      Word           ' ref voltage adc counts
axCount       VAR      Word           ' axis voltage adc counts
mVolts        VAR      Word           ' millivolts
gForce        VAR      Word           ' axis g-force

dValue        VAR      Word           ' display value
dPad          VAR      Nib            ' display pad

' ----- [ Initialization ] -----

Reset:
HIGH CS              ' deselect module
DEBUG CLS,           ' paint display
"=====", CR,
"H48C 3-Axis Accelerometer", CR,
"=====", CR,
CR,
"      Count  Volts    G  ", CR,
"      -----  -----  -----", CR,
"VRef", CR,
"  X", CR,
"  Y", CR,
"  Z", CR,

' ----- [ Program Code ] -----

Main:
FOR axis = XAxis TO ZAxis          ' loop through each axis
GOSUB Get_H48C                     ' read vRef & axis counts

dValue = rvCount                    ' display vRef count
DEBUG CRSRXY, 6, 6
GOSUB RJ_Print

dValue = axCount                    ' display axis count
DEBUG CRSRXY, 6, (7 + axis)
GOSUB RJ_Print

mVolts = rvCount ** Cnt2Mv          ' convert vref to mv
DEBUG CRSRXY, 13, 6,               ' display
    DEC (mVolts / 1000), ".",
    DEC3 mVolts

mVolts = axCount ** Cnt2Mv          ' convert axis to mv
DEBUG CRSRXY, 13, (7 + axis),

```

```

        DEC (mVolts / 1000), ".",
        DEC3 mVolts

' calculate g-force
' -- "gForce" is signed word

IF (axCount >= rvCount) THEN
    gForce = (axCount - rvCount) ** GfCnv      ' positive g-force
ELSE
    gForce = -((rvCount - axCount) ** GfCnv) ' negative g-force
ENDIF
DEBUG CRSRXY, 20, (7 + axis),                ' display g-force
    " " + (gForce.BIT15 * 13),
    DEC1 (ABS(gForce) / 100), ".",
    DEC2 ABS(gForce)
NEXT
PAUSE 200
GOTO Main

' -----[ Subroutines ]-----

' Reads VRef and selected H48C axis through an MCP3204 ADC
' -- pass axis (0 - 2) in "axis"
' -- returns reference voltage counts in "rvCount"
' -- returns axis voltage counts in "axCounts"

Get_H48C:
    LOW CS
    SHIFTOUT Dio, Clk, MSBFIRST, [%11\2, VRef\3] ' select vref register
    SHIFTTIN Dio, Clk, MSBPOST, [rvCount\13]     ' read ref voltage counts
    HIGH CS
    PAUSE 1
    LOW CS
    SHIFTOUT Dio, Clk, MSBFIRST, [%11\2, axis\3] ' select axis
    SHIFTTIN Dio, Clk, MSBPOST, [axCount\13]    ' read axis voltage counts
    HIGH CS
    RETURN

' -----

' Right-justify value in 5-digit field
' -- move cursor first, then call with value in "dValue"

RJ_Print:
    LOOKDOWN dValue, >=[10000, 1000, 100, 10, 0], dPad
    DEBUG REP " "\dPad, DEC dValue
    RETURN

```