Mechatronics

Lecture

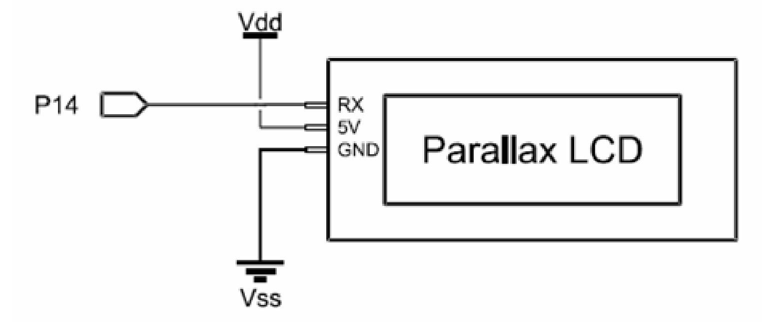
Smart Sensors

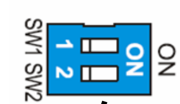
Liquid Crystal Display (LCD)

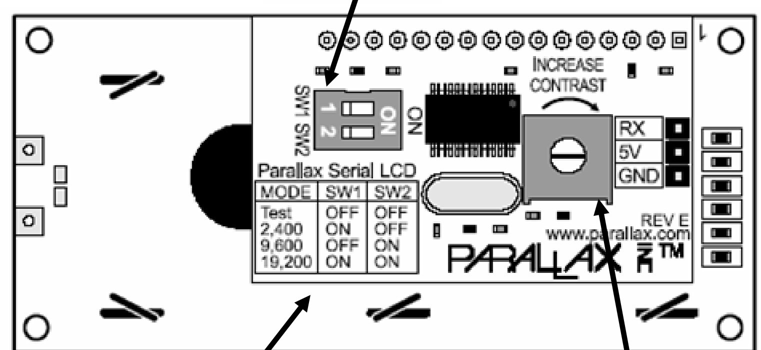
- Display measurement, status information, etc.
- Field-testing without being tethered to a PC/Laptop
- Parallax 2×16 serial LCD (non-backlit)
- 3-pin connection (V_{ss}, V_{dd}, and V_{sig})
- BS2 commands the LCD serially, using SEROUT

Interfacing LCD to BS2

- Connect BS2's V_{ss} , V_{dd}, and one I/O pin (say P14) to LCD's GND, 5V, and RX pins, respectively
- To test LCD module, on its backside, set switches SW1 and SW2 off
- Turn on power to BS2, LCD should display "Parallax, Inc." on top line and "www.parallax.com" on bottom line
- If display appears dim, adjust the contrast potentiometer
- Turn off power to BS2 and set SW2 ON to allow LCD to receive serial communication from BS2 at 9600 baud rate

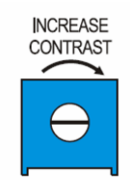


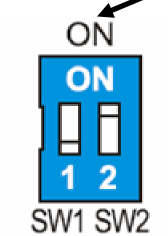




Parallax Serial LCD

MODE	SW1	SW2
Test	OFF	OFF
2,400	ON	OFF
9,6000	OFF	ON
19,200	ON	ON

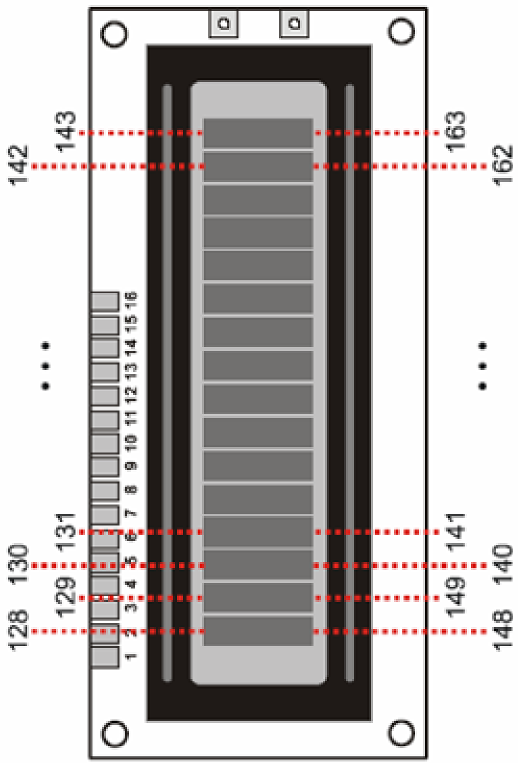




LCD: PBASIC Sample Code I

```
'{$STAMP BS2}
'{$PBASIC 2.5}
SEROUT 14, 84, [22, 12] 'Initialize LCD
PAUSE 5
SEROUT 14, 84, ["Hello World!", 13, "The LCD Works"]
```

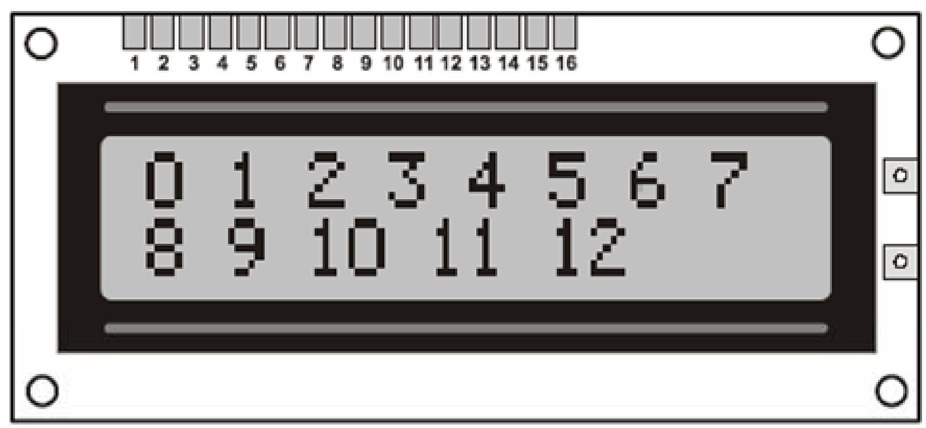
- SEROUT *Pin, BaudMode,* [*DataItem1, DataItem2, ...*]
- BaudMode argument for 9600 bits per second (bps), 8 data bits, no parity, true signal: 84
- DataItems: text to be displayed, control codes, formatters like DEC, BIN, HEX, etc.
- LCD must receive control code 22 from BS2 to turn on
- Control code examples—8: cursor left, 9: cursor right, 12: clear display (follow with PAUSE 5 to allow display to clear), 13: carriage return, 21: LCD off,
- 128 to 143 Position cursor on Line 0, character 0 to 15
- 148 to 163 Position cursor on Line 1, character 0 to 15
- SEROUT 14, 84, [128, "Hello", 148, "World!"]



LCD: PBASIC Sample Code II

```
' {$STAMP BS2}
' {$PBASIC 2.5}
counter VAR Byte             'FOR...NEXT loop index
SEROUT 14, 84, [22, 12]   'Initialize LCD
PAUSE 5                       '5 ms delay for clearing display
FOR counter = 0 TO 12      'Count to 12; increment at 1/2 s
SEROUT 14, 84, [DEC counter, " "]
PAUSE 500
NEXT
END
```

- Display numbers 0 to 12 on LCD
- Each number is followed by a space
- When top line of LCD is filled up by 16 characters
  - text sent by BS2 wraps to the bottom line
  - if the bottom line is filled up by 16 characters then the text wraps again, to top line

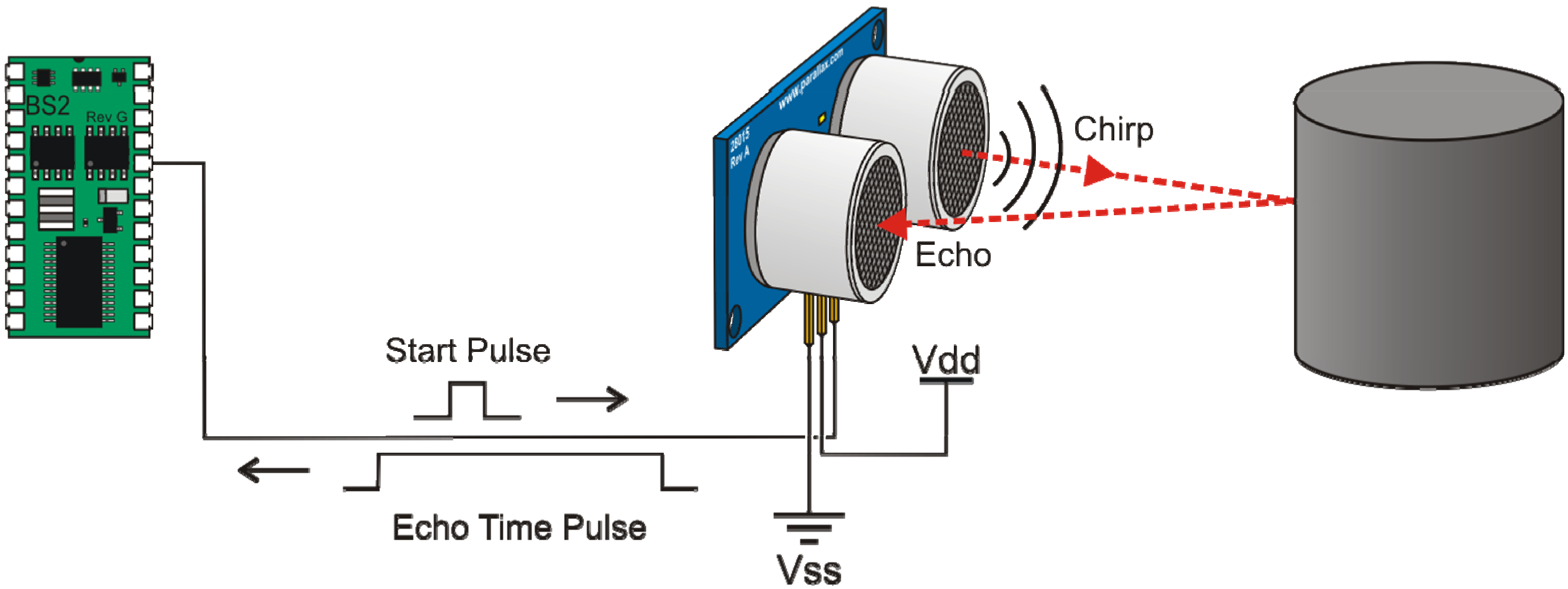


LCD: PBASIC Sample Code III

```
' Display elapsed time with BS2 and Parallax Serial LCD.
' {$STAMP BS2}
' {$PBASIC 2.5}
hours VAR Byte 'hours
minutes VAR Byte 'minutes
seconds VAR Byte 'seconds
SEROUT 14, 84, [22, 12] 'Initialize LCD
PAUSE 5 '5 ms to clear display
SEROUT 14, 84, ["Time Elapsed...", 13] 'Text & carriage return
SEROUT 14, 84, [" h m s"] 'Text on second line
DO 'Main Routine
'Calculate hours, minutes, seconds
IF seconds = 60 THEN seconds = 0: minutes = minutes + 1
IF minutes = 60 THEN minutes = 0: hours = hours + 1
IF hours = 24 THEN hours = 0
'Display digits on LCD on Line 1. The values 148, 153, 158
'place the cursor at character 0, 5, and 10 for the time values.
SEROUT 14, 84, [148, DEC2 hours,
                153, DEC2 minutes,
                158, DEC2 seconds ]
PAUSE 991 'Pause + program overhead ~ 1 second
seconds = seconds + 1 'Increment second counter
LOOP 'Repeat Main Routine
```

Ultrasonic Sensor—PING)))

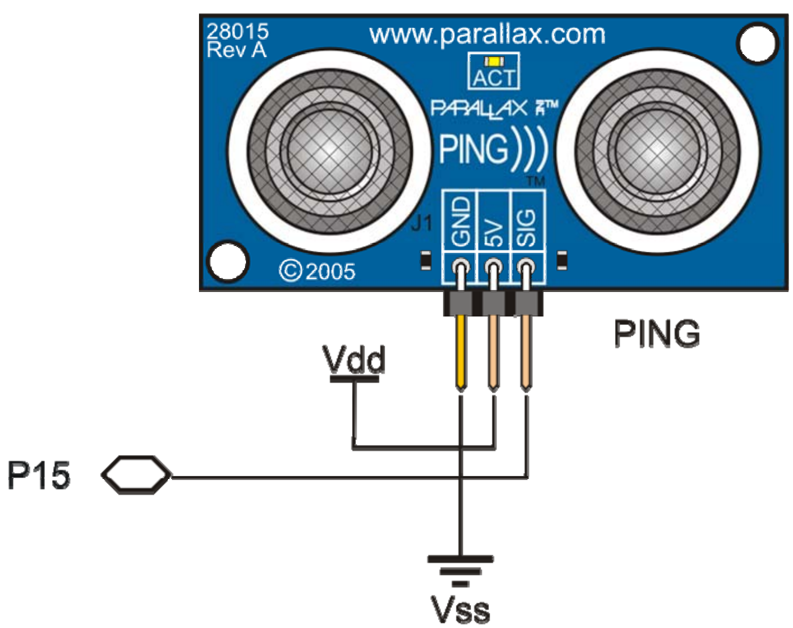
- Time-of-flight distance measurement
- Sensor emits a 40KHz tone and measures time till it receives the echo signal

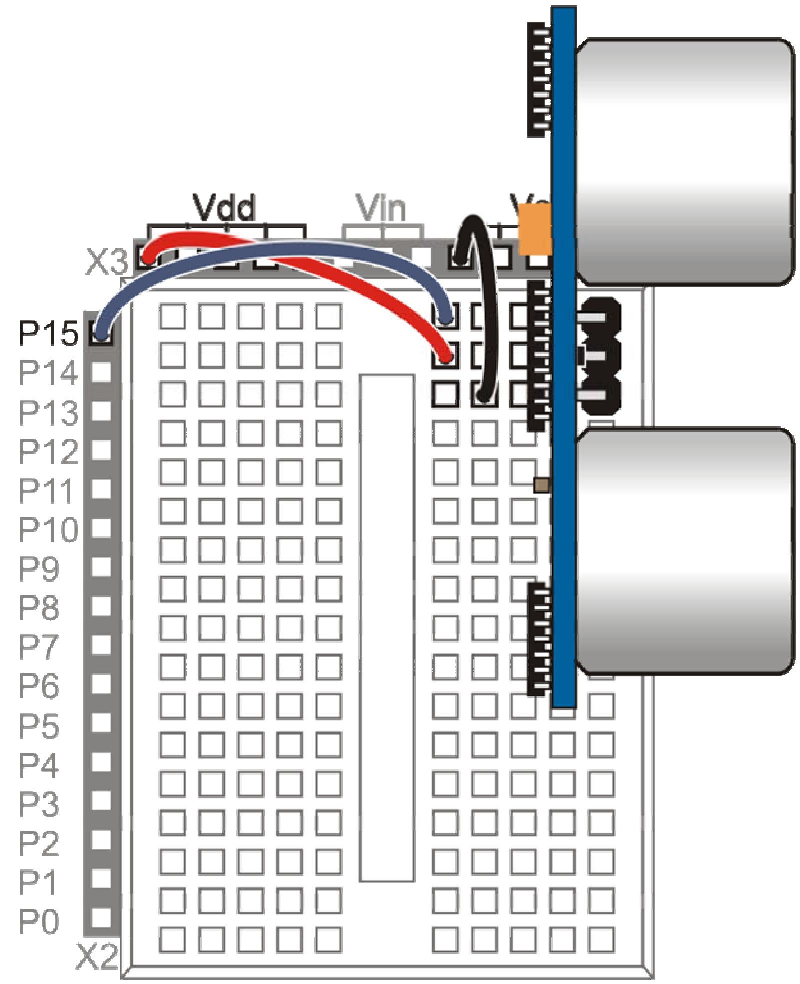


  - Round-trip time-of-flight yields distance measurement: $D=0.5\times C\times T$, D=distance (m), C=speed of sound in air @ 72 ºF (344.8 m/s), T=round trip time (s)
- Range: 3.3 meters

Interfacing PING))) to BS2

- Connect BS2's V_{ss} , V_{dd}, and one I/O pin (say P15) to PING)))'s GND, 5V, and SIG pins, respectively





PING))): PBASIC Sample Code I

```
' {$STAMP BS2}
' {$PBASIC 2.5}
rawtime VAR Word
DO
PULSOUT 15, 5
PULSIN 15, 1, rawtime
DEBUG HOME, “rawtime = ", DEC5 rawtime
PAUSE 100
LOOP
```

- PULSOUT 15, 5: sends a 10μs pulse to P15
- PULSIN 15, 1, time: monitors for the return echo and stores it in the variable time (unit 2μs)

PING))): PBASIC Sample Code II

$$D_{cm} = \left(\frac{1}{2}\right) \times \overbrace{(100 \times 344.8)}^{\text{cm/s}} \times \overbrace{\left(T_{raw} \times 2 \times 10^{-6}\right)}^{\text{seconds}} = T_{raw} \times 0.03448$$

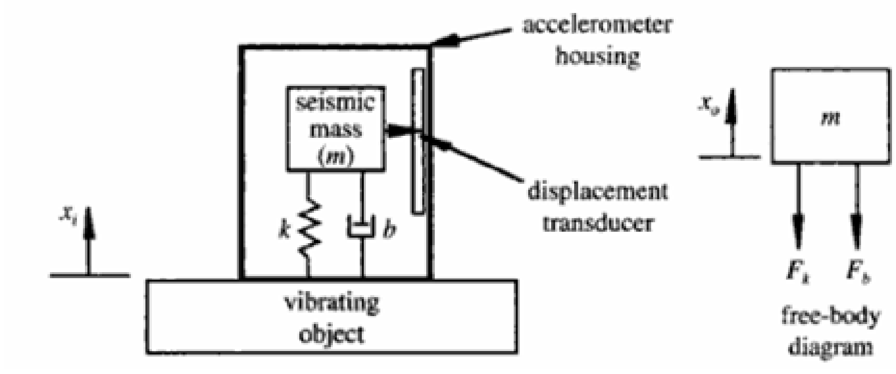
- Let cmConst=0.03448×65536=2260
- Now compute D_{cm} by using T_{raw}**2260

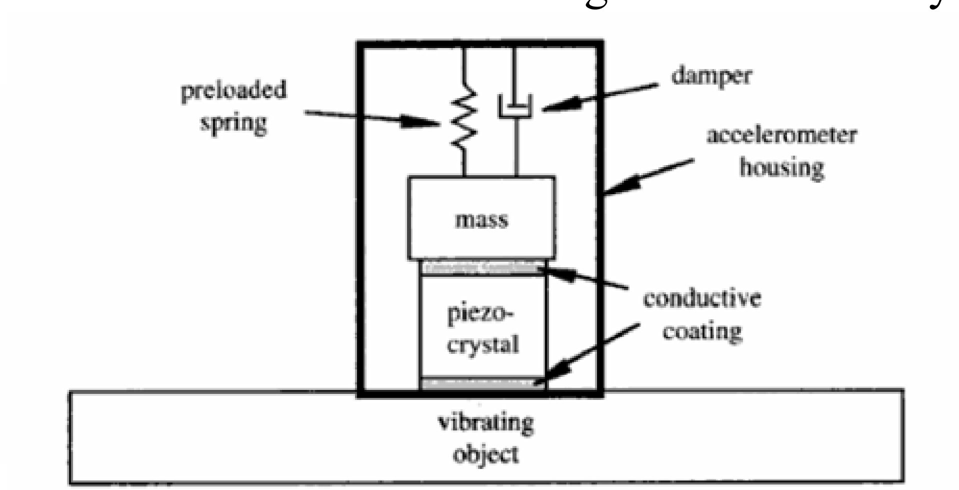
```
' {$STAMP BS2}
' {$PBASIC 2.5}
rawtime VAR Word
cmDist VAR Word
cmConst CON 2260
DO
PULSOUT 15, 5
PULSIN 15, 1, rawtime
cmDist=rawtime**cmConst
DEBUG HOME, “cmDist = ", DEC cmDist
PAUSE 100
LOOP
```

- For D_{inch} let inchConst=(0.03448/2.54)×65536=890
- Now compute D_{inch} by using T_{raw}**890

Accelerometer

- Electromechanical device to measure acceleration forces
  - Static forces like gravity pulling at an object lying at a table
  - Dynamic forces caused by motion or vibration
- How they work
  - Seismic mass accelerometer: a seismic mass is connected to the object undergoing acceleration through a spring and a damper;
  - Piezoelectric accelerometers: a microscopic crystal structure is mounted on a mass undergoing acceleration; the piezo crystal is stressed by acceleration forces thus producing a voltage
  - Capacitive accelerometer: consists of two microstructures (micromachined features) forming a capacitor; acceleration forces move one of the structure causing a capacitance changes.
  - Piezoresistive accelerometer: consists of a beam or micromachined feature whose resistance changes with acceleration
  - Thermal accelerometer: tracks location of a heated mass during acceleration by temperature sensing





Accelerometer Applications

- Automotive: monitor vehicle tilt, roll, skid, impact, vibration, etc., to deploy safety devices (stability control, anti-lock breaking system, airbags, etc.) and to ensure comfortable ride (active suspension)
- Aerospace: inertial navigation, smart munitions, unmanned vehicles
- Sports/Gaming: monitor athlete performance and injury, joystick, tilt
- Personal electronics: cell phones, digital devices
- Security: motion and vibration detection
- Industrial: machinery health monitoring
- Robotics: self-balancing

Helmet: Impact Detection

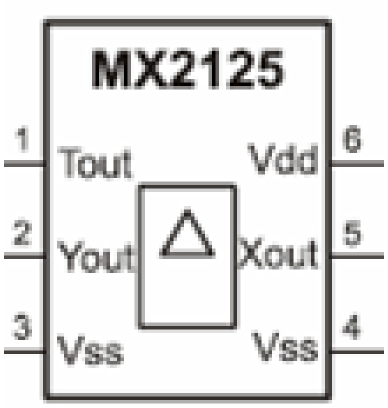
2 axis joystick

WII Nunchuk: 3 axis accelerometer

Segway

Memsic 2125 2-axis Accelerometer

- Measure acceleration, tilt angle, rotation angle
  - G-force measurements for X and Y axis reported in pulse-duration
- Temperature measurement: analog output (T_{out})
- Low current operation: < 4 mA @ 5VDC
- Measures 0 to ±2 g on either axis
- Resolution: <1 mg
- Operating temperature: 0 °C to 70 °C

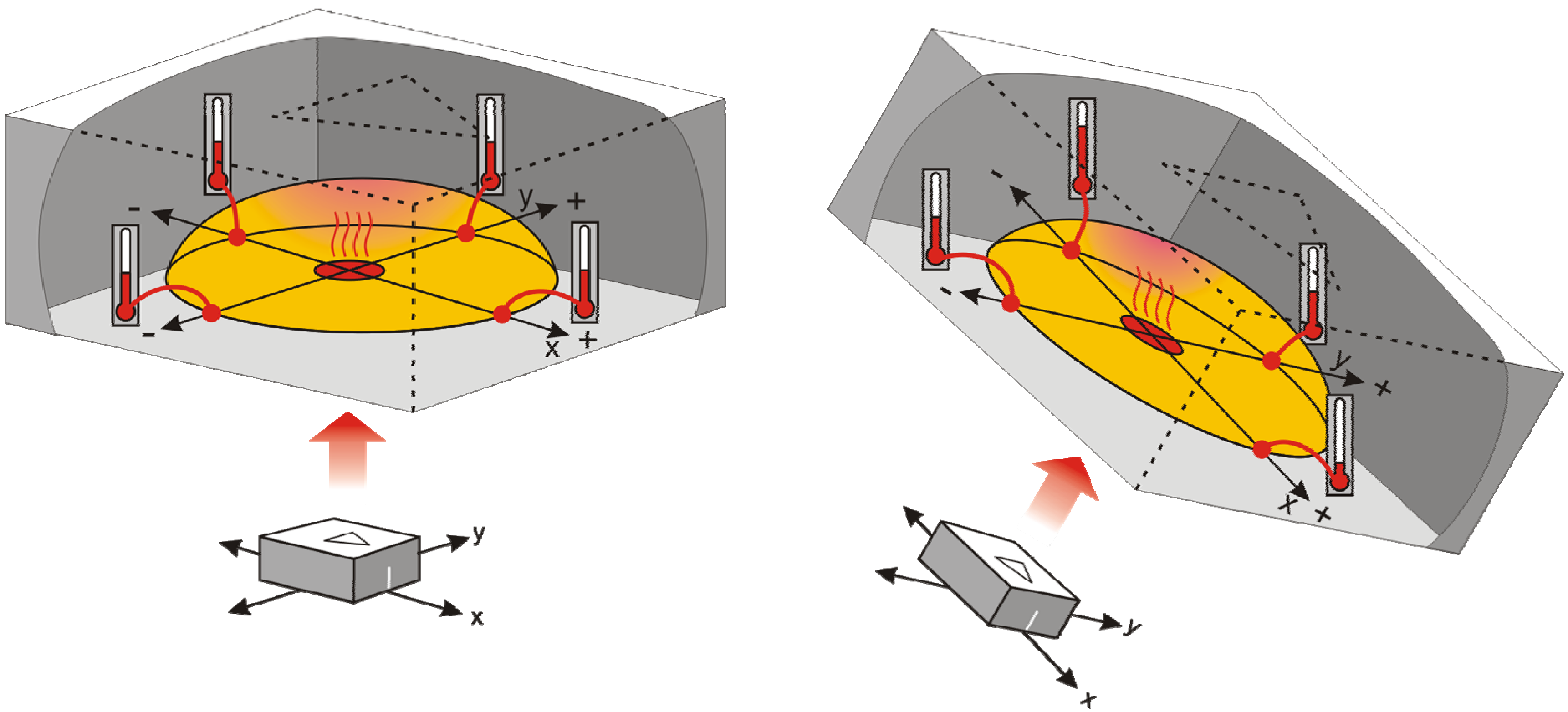


Accelerometer Module

MX2125 Chip

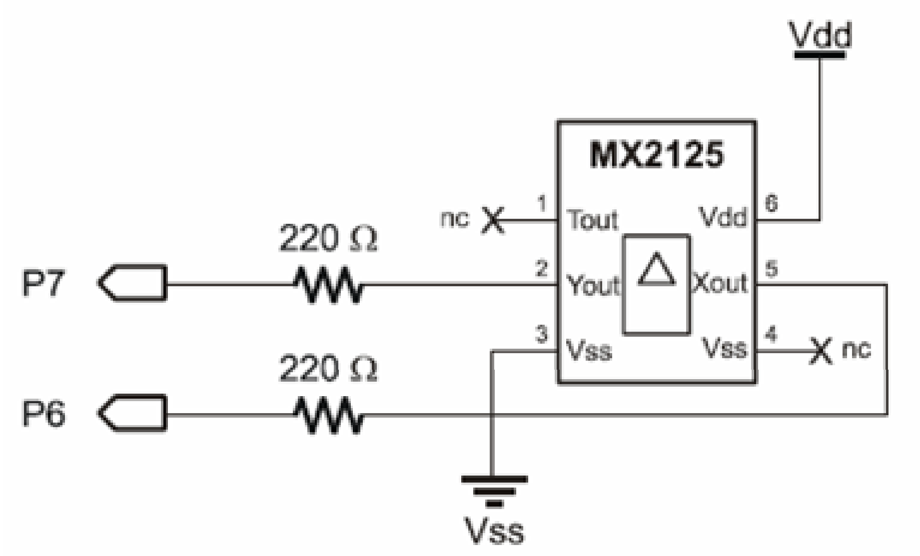
MX2125 Accelerometer: How it Works

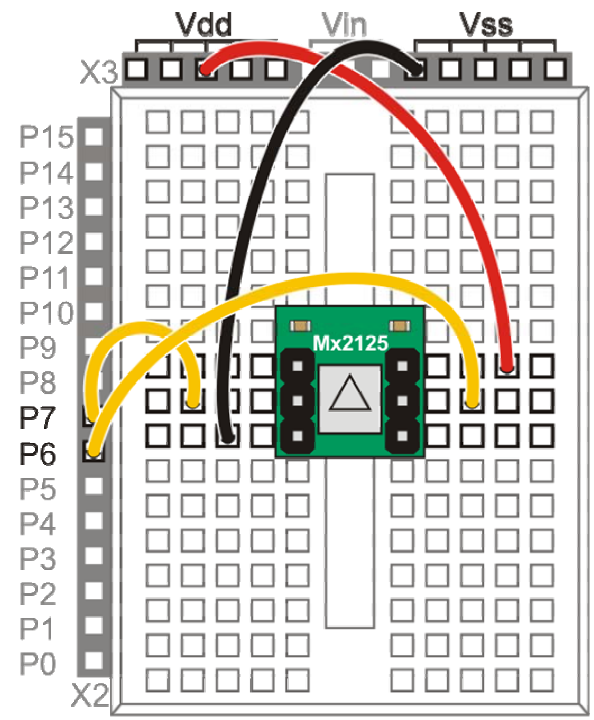
- A MEMS device consisting of
  - a chamber of gas with a heating element in the center
  - four temperature sensors around its edge
- Hold accelerometer level→hot gas pocket rises to the top-center of the accelerometer's chamber→all sensors measure same temperature
- Tilt the accelerometer→hot gas pocket collects closer to one or two temperature sensors→sensors closer to gas pocket measure higher temperature
- MX2125 electronics compares temperature measurements and outputs pulses (pulse duration encodes sensor o/p)



Interfacing Accelerometer to BS2

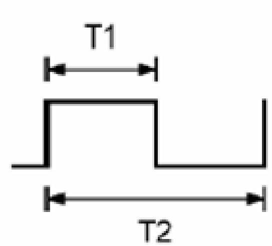
- Connect BS2's V_{ss} , V_{dd}, and two I/O pin (say P6 and P7) to MX2125's pins 3, 6, 5, and 2, respectively





- X_{out} and Y_{out} pulse outputs are set to 50% duty cycle at 0g; the duty cycle changes in proportion to acceleration
- G Force can be computed from the duty cycle as shown below
- T_{out} provides analog output 1.25 volts @25.0°C, output change: 5 mV/°C

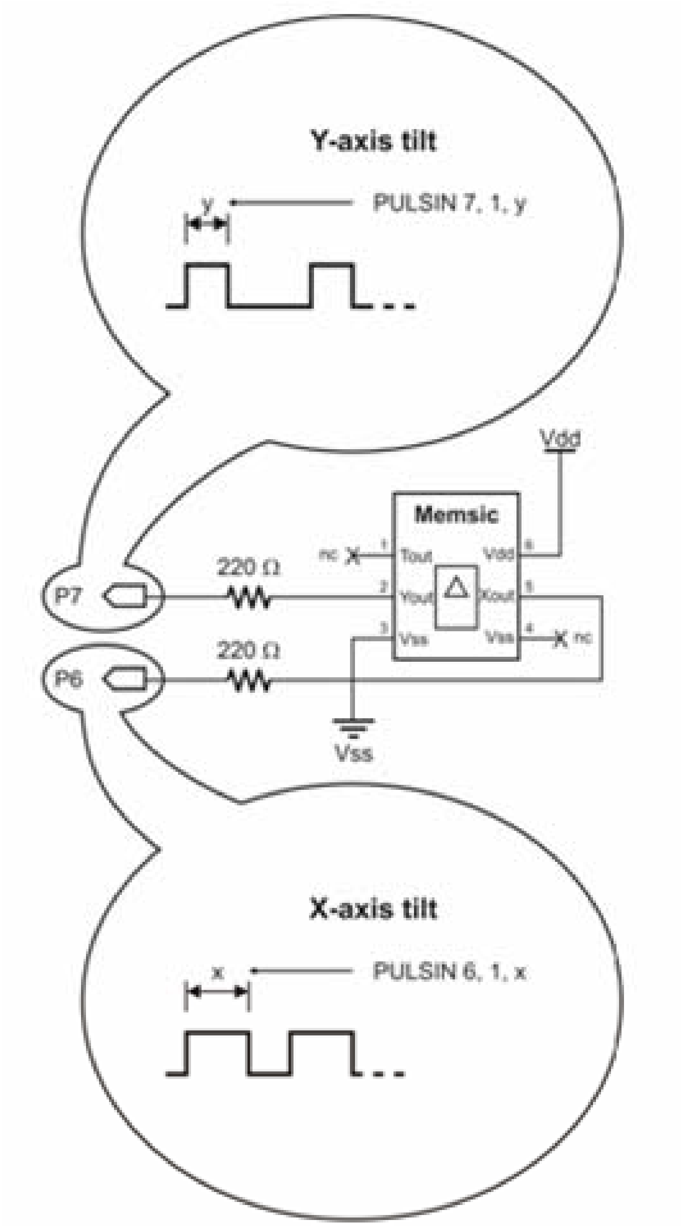
Memsic 2125 Pulse Output



$$A(g) = ((T1 / T2) – 0.5) / 12.5\%$$

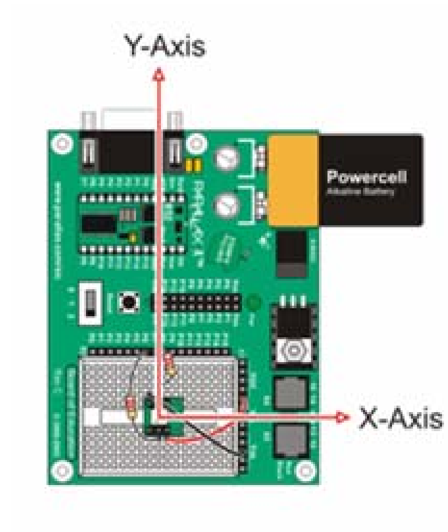
T2 duration is calibrated to 10 milliseconds at 25° C (room temperature)

Accelerometer Axis Pulse Measurements



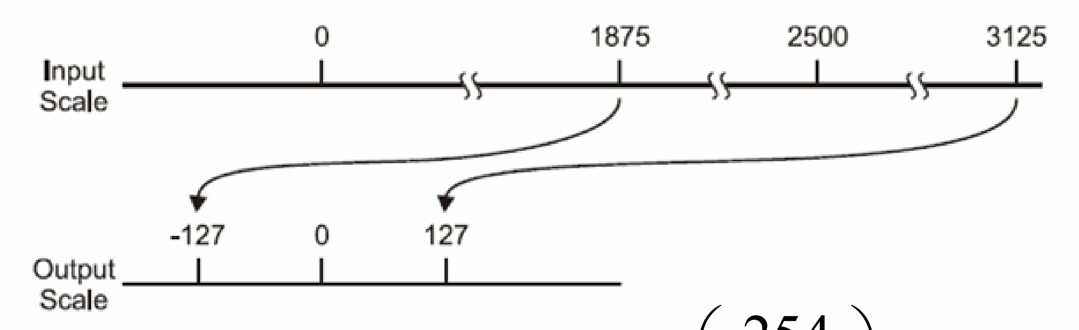

```
'{$STAMP BS2}
'{$PBASIC 2.5}
x VAR Word
y VAR Word
DEBUG CLS
DO
PULSIN 6, 1, x
PULSIN 7, 1, y
DEBUG HOME, DEC4 ? x, DEC4 ? y
PAUSE 100
LOOP
```

Pulsin o/p range: 1875 to 3125
When level: o/p=2500



Pulse Measurements: Offset and Scaling

- **Let X_{raw}= Pulsin output**
- **$X_{raw} \in$ {1875, 3125} and when level X_{raw}=2500**
- **We wish X_{out}: $X_{raw} \rightarrow X_{out} \in$ {-127, 127}, and X_{out}=0 when level**



$$X_{\text{out}} = \left(X_{\text{raw}} - 2500\right)\times\left(\frac{254}{1250}\right)$$

$$= X_{\text{raw}} \times\left(\frac{254}{1250}\right) - 508$$

- **Let Scale=INT((254/1250) ×65536)=13316**
- **Now compute X_{out} by using X_{raw}**13316-508**

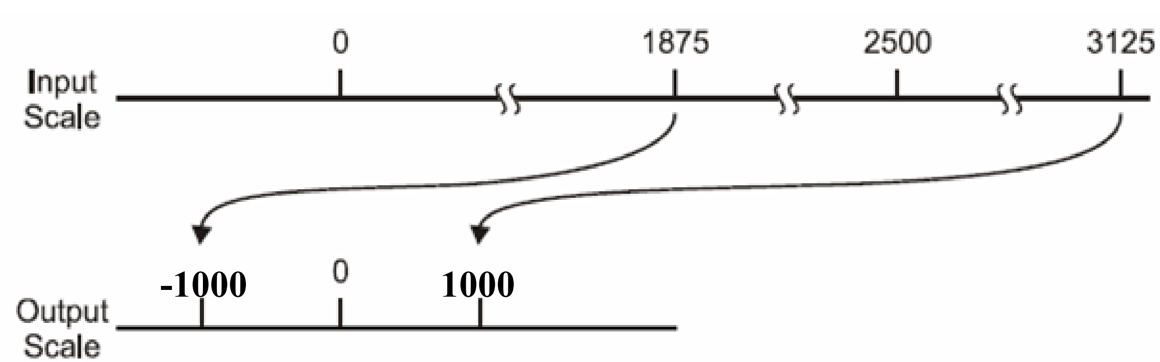
```
'{$STAMP BS2}
'{$PBASIC 2.5}
scalecon CON 13316
xraw VAR Word
yraw VAR Word
Xo VAR Word
Yo VAR Word
DEBUG CLS
DO
PULSIN 6, 1, xraw
PULSIN 7, 1, yraw
Xo=xraw**scalecon-508
Yo=yraw**scalecon-508
DEBUG HOME, SDEC Xo, SDEC Yo
PAUSE 100
LOOP
```

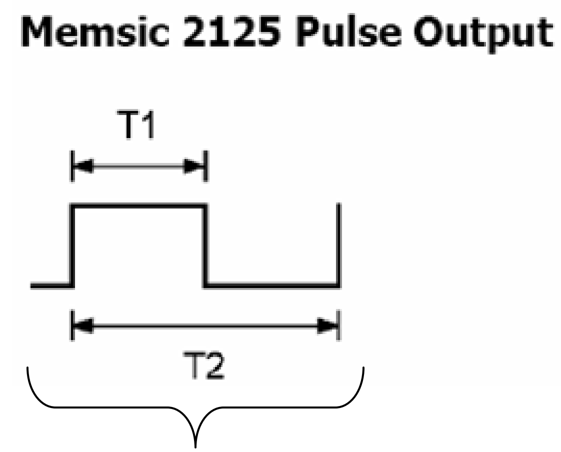
Clamp input range to {1875, 3125} using the following:

```
xout=(xraw Min 1875 Max 3125) **scalecon-508
yout=(yraw Min 1875 Max 3125) **scalecon-508
```

g-Force Measurements in mili-g—I

- **Let T_{raw}= Pulsin output (2μs units)**
- **$T_{raw} \in \{1875, 3125\}$ and when level T_{raw}=2500**
- **T_{raw}=1875→-g (-1000 milli-g) and T_{raw}=3125→g (1000 mili-g)**
- **So, we wish T_{out}: $T_{raw} \rightarrow T_{out} \in \{-1000, 1000\}$, and T_{out}=0 when level**



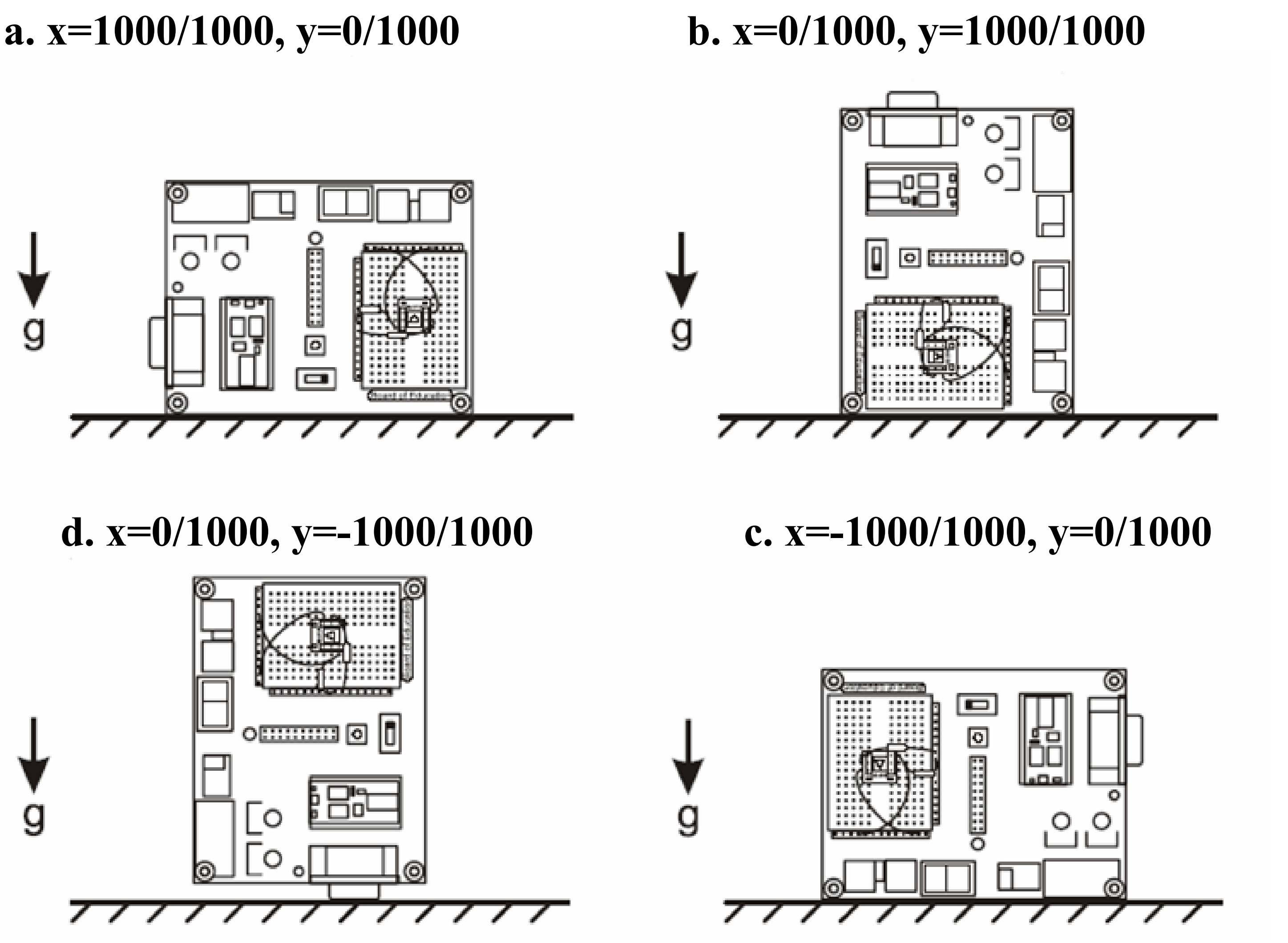


T_1: Pulsin output returns T_{raw}
T_2: 10milli-seconds @ 25ºC

- **Moreover, recall g force is given by**

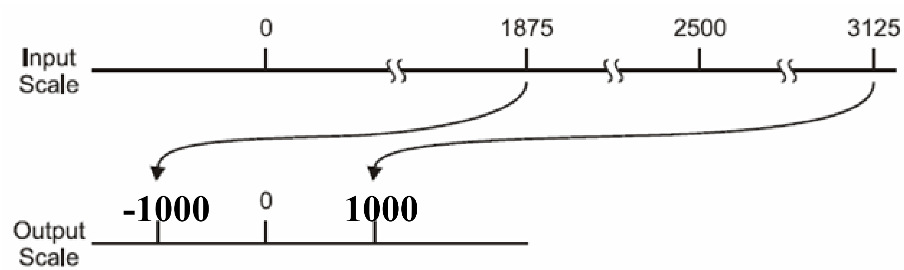
$$g_{\text{Force}} = \left(\frac{T_1}{T_2} - 0.5\right) \times \left(\frac{1}{12.5\%}\right) \quad (\text{units}: \text{g})$$

g-Force Measurements in mili-g—II



Sample Readings at Various Orientations (start at top left, rotate clockwise)

g-Force Measurements in mili-g—III



- T_1: Pulsin output returns T_{raw} in 2μs units
- T_2: 10mili-seconds @ 25ºC
- Thus,
  $T_1 = 2\times10^{-6}\times T_{raw}$ seconds $= 2\times10^{-3}\times T_{raw}$ mili-seconds

$$
\begin{aligned}
T_{\text{out}} &= \left(T_{\text{raw}} - 2500\right)\times\left(\frac{2000}{1250}\right)\\
&= \left(\frac{2\times T_{\text{raw}}}{10}\right)\times\left(\frac{1000}{125}\right) - 2500\times\left(\frac{2000}{1250}\right)\\
&= \left(\frac{2\times T_{\text{raw}}}{10}\right)\times 8 - 4000\\
&= \left(\left(\frac{2\times T_{\text{raw}}}{10}\right) - 500\right)\times 8
\end{aligned}
$$

$$
\begin{aligned}
g_{\text{Force}} &= \left(\frac{T_1}{T_2} - 0.5\right)\times\left(\frac{1}{12.5\%}\right),\ (\text{units}: \text{g})\\
&= \left(\frac{T_1}{T_2} - 0.5\right)\times\left(\frac{1}{12.5\%}\right)\times 10^3,\ (\text{units}: \text{milli-g})\\
&= \left(\frac{T_{\text{raw}}\times 2\times 10^{-3}}{10} - 0.5\right)\times\left(\frac{100}{12.5}\right)\times 10^3\\
&= \left(\frac{T_{\text{raw}}\times 2}{10} - 500\right)\times 8
\end{aligned}
$$

MX2125 Angle of Rotation in Vertical Plane—I

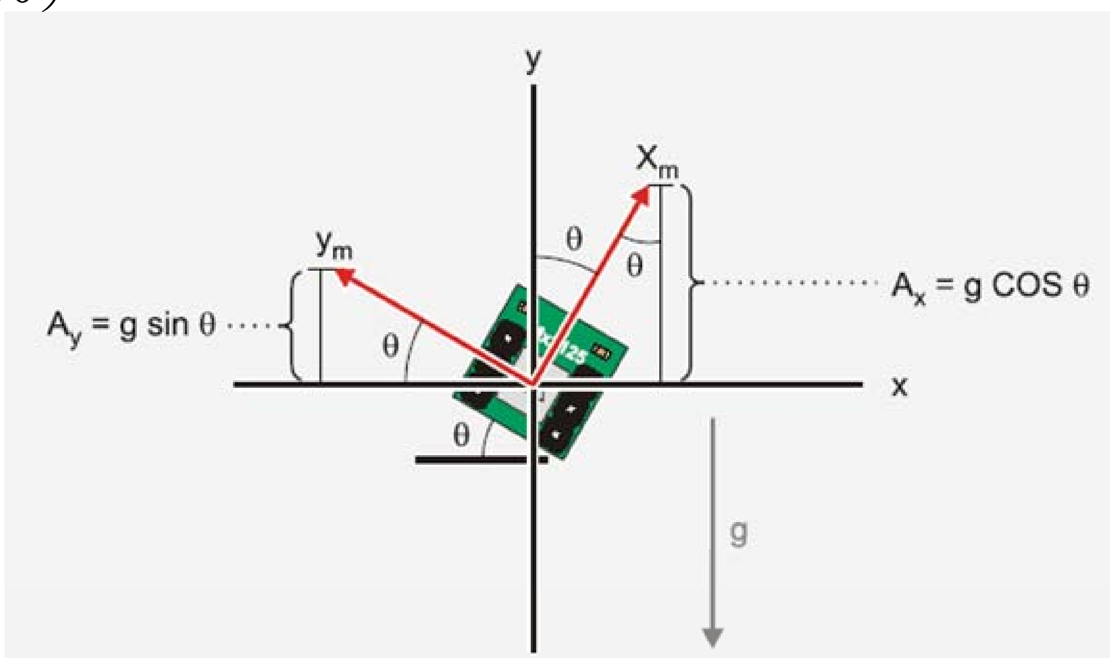
- MX2125's angle of rotation in the vertical plane:

$$\theta = \tan^{-1}\left(\frac{A_y}{A_x}\right),\ \text{BS2 returns } A_x, A_y \in \{1875, 3125\}$$

- To compute $\tan^{-1}(Y/X)$ use PBASIC ATAN command: X ATN Y; ATN requires X, $Y \in \{-127, 127\}$ which is accomplished using

$$X = (A_x - 2500)\times\left(\frac{254}{1250}\right)$$
$$= A_x \times\left(\frac{254}{1250}\right) - 508$$

- **Let INT((254/1250) ×65536)=13316**
- **Now compute X by using Ax**13316-508**

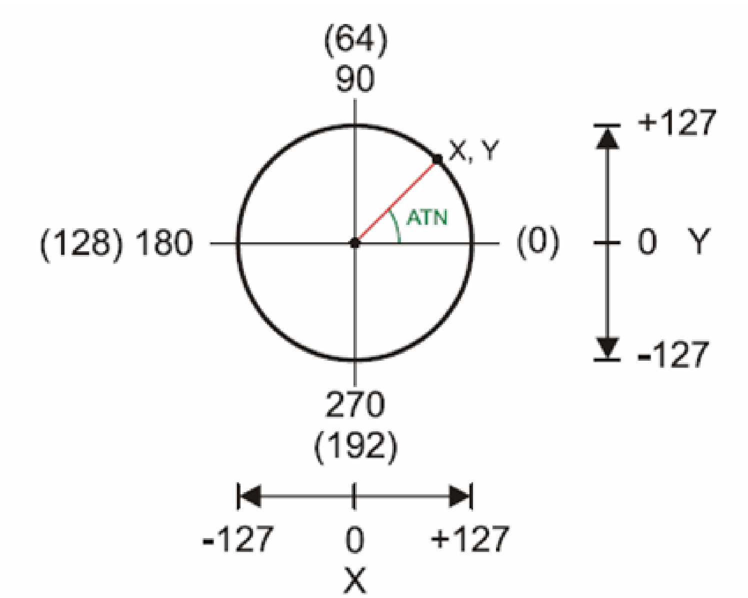


MX2125 Angle of Rotation in Vertical Plane—II

- ATN returns its output in binary radians (i.e., a circle is split up into 256 segments instead of 360 segments as in degrees)
- Convert ATN output from brad to degrees as follows:

$$\theta_{Deg} = \theta_{BRad} \times \left(\frac{360}{256}\right)$$

- **Let INT((360/256)×256)=360**
- **Now compute θ_{Deg} by using θ_{BRad}*/360**



Unit circle in degrees and binary radians

MX2125 Angle of Rotation in Vertical Plane: Sample Code

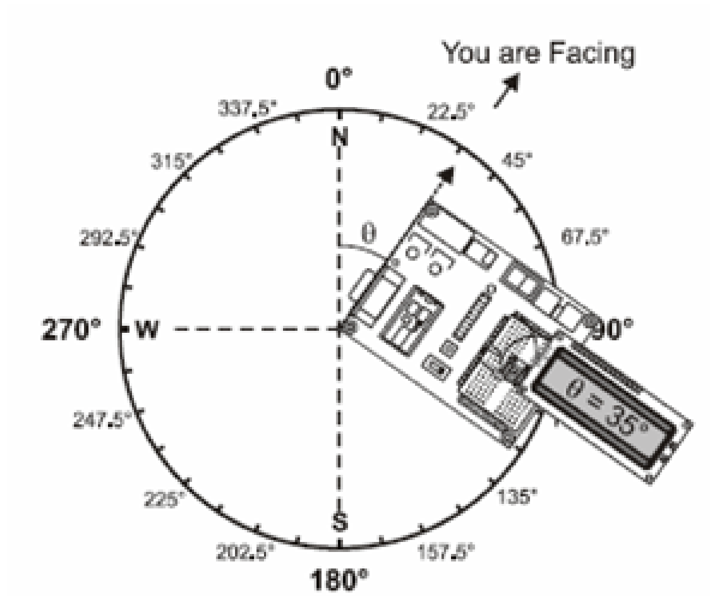
```
'{$STAMP BS2}
'{$PBASIC 2.5}
scale1 CON 13316
scale2 CON 360
Ax VAR Word
Ay VAR Word
angle VAR Word
DEBUG CLS
DO
PULSIN 6, 1, Ax
PULSIN 7, 1, Ay
Ax=(Ax MIN 1875 MAX 3125)**scale1-508
Ay=(Ay MIN 1875 MAX 3125)**scale1-508
angle=Ax ATN Ay
angle=angle*/scale2
DEBUG HOME, " Ax =", SDEC Ax, " Ay=", SDEC Ay, " angle=", SDEC3 angle, 176, "   "
PAUSE 300
LOOP
```

HM55B Compass Module

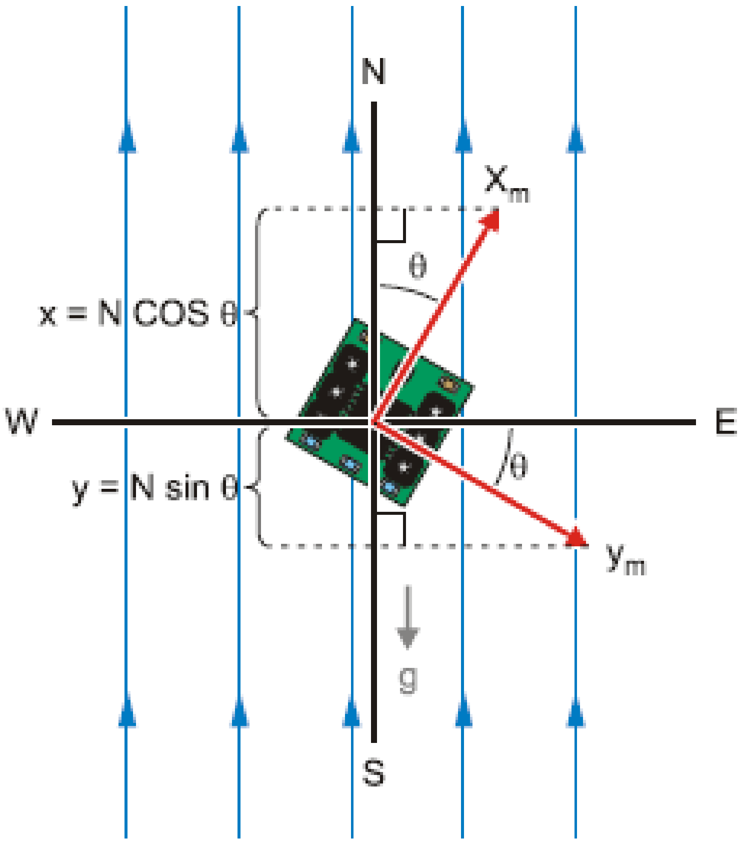
- Dual axis magnetic field sensor
- Sensitive to microtesla (μT) variations in magnetic field strength
- Operates on I=30-45 mA @ 5VDC
- Sensitivity: 1 to 1.6 **μT**
- Conversion time: 30 to 40 ms between start measurement and data-ready
- Built-in resistor protection for data pins
- Operating Temp.: 0 to 70 °C

Mechanical Compass

Hitachi HM55B



HM55B Compass Module

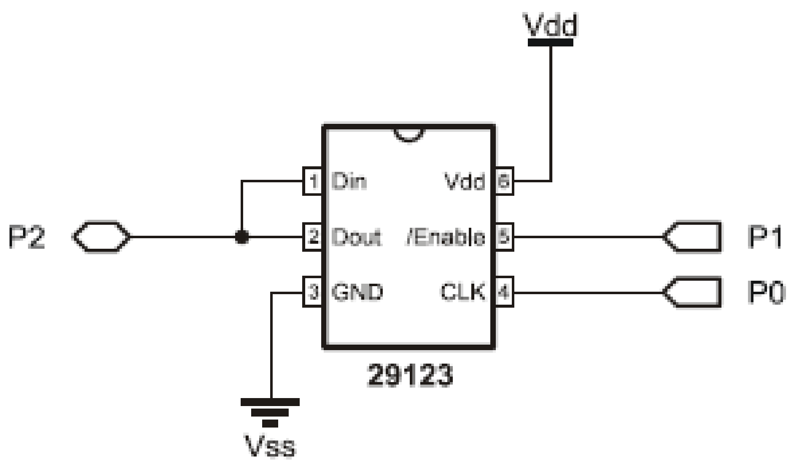


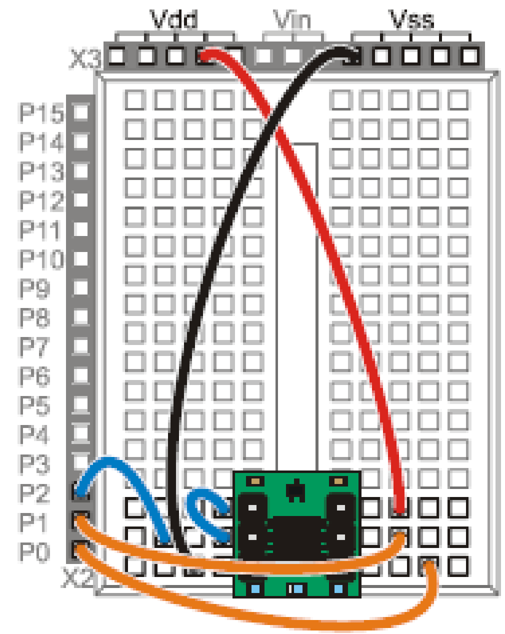
$$\tan\theta = \frac{-N\sin\theta}{N\cos\theta} = \frac{-y}{x}$$

$$\tan^{-1}(\tan\theta) = \tan^{-1}\left(\frac{-y}{x}\right)$$

$$\theta = \tan^{-1}\left(\frac{-y}{x}\right)$$

HM55B Compass Module with BS2

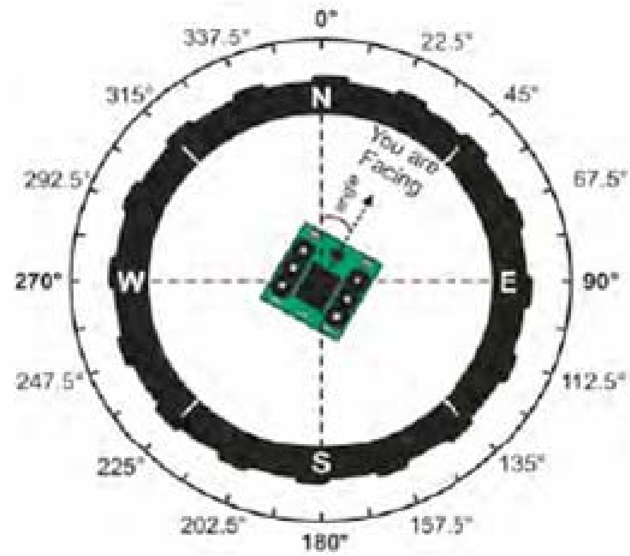


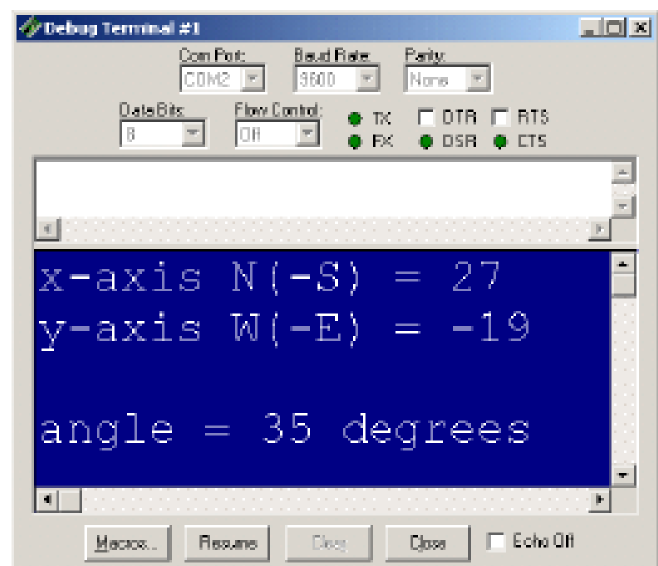


- Connect BS2's Vss , Vdd, and three I/O pin (say P0, P1 and P2) to HM55B's pins 3, 6, 4, 5, and 2, respectively.
- Din and Dout are shorted to use only one pin for sending and receiving data from HM55B.

HM55B Compass Module with BS2

```
Compass_Get_Axes:
HIGH En: LOW En
SHIFTOUT DinDout,clk,MSBFIRST,[Reset\4]

HIGH En: LOW En
SHIFTOUT DinDout,clk,MSBFIRST,[Measure\4]
status = 0

DO
   HIGH En: LOW En
   SHIFTOUT DinDout,clk,MSBFIRST,[Report\4]
   SHIFTIN DinDout,clk,MSBPOST,[Status\4]
LOOP UNTIL status = Ready

SHIFTIN DinDout,clk,MSBPOST,[x\11,y\11]
HIGH En

IF (y.BIT10 = 1) THEN y = y | NegMask
IF (x.BIT10 = 1) THEN x = x | NegMask

RETURN

‘ To get the agnle
angle = x ATN -y ' Convert x and y to brads
angle = angle */ 361
```





Calibration

WHY?

- Software Calibration
- compensate for the effects of magnetic fields
- corrects for the HM55B chip's axis sensitivity, offset and skew errors

HOW?

- Make a printout of the 16-segment compass shown.
- Align the printout to magnetic north with the aid of the magnetic compass.
- Affix the aligned printout to your work surface.
- Make sure to set the magnetic compass well away from the printout before continuing.
- Align the Compass Module to magnetic north by lining up the edge of your carrier board with the dashed line that passes through the 0° mark.