MODELLING OF AN INTEGRATED DIRECTIONAL UNIT

Team Members

SNEHIL SURESH WAKCHAURE

AJITH ERRABELLY

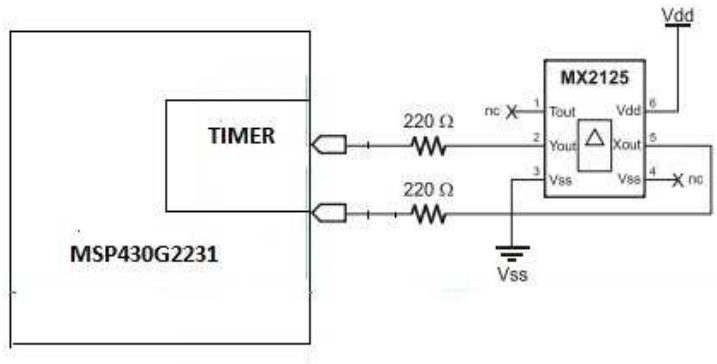
SRINIDHI SREEPAD

PRATHEEK RAJATH

Accelerometer MEMSIC2125

- Measures Tilt & Vibration
- Simple PWM pulse output of g-force for each axis
- Power Requirements: 3.3 to 5 VD
- Analog output of temperature (TOut pin)
- Operating temperature over 0 to 70 ℃
- The PWM output can be captured using the Timer Capture mode of MSP430
- Challenges: Calculation of Inclination using PWM data

MEMSIC2125 Hardware Interface (Using Timer)

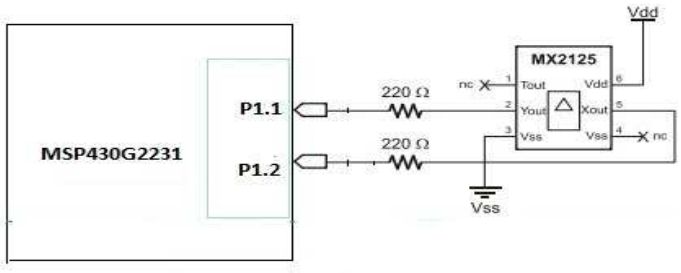


MEMSIC2125 Software Interface: (Using Timer)

Configure the Timer to operate to capture rising edge, SMCLK, CCIE=1
Within the Timer ISR:
Check if Rising edge
    Capture Start time
    Switch to falling edge capture settings
If falling edge, then
   Capture Stop time
   Switch to rising edge

Progress: Rising edge and falling edges detected, start and stop times captured, ISR coded
Waveforms tested on CRO, Values checked on IAR Testbench software
Challenges: Debugging is required to get correct value in TACCR0

MEMSIC2125 Hardware Interface: (Using Port1 Interrupts)



MEMSIC 2125 Software Interface: (Using Port1 Interrupts)

Configure 2 port pins to get PWM input from the Accelerometer
Detect rising edge using port interrupt
Within the Timer ISR:
Check if rising edge
    Capture Start time from TAR
    Switch to falling edge capture settings
If falling edge, then
   Capture Stop time from TAR
   Switch to rising edge
   Reset TAR

Progress: Rising edge and falling edges detected

Challenges: Testing & debugging yet to be done

GRAPHICAL USER INTERFACE

Serial Port Communication:

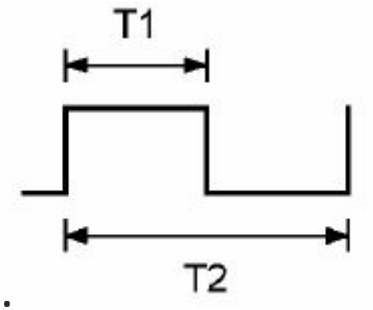
•Code for serial port communication and send transfer data from MSP430 to Matlab

```
% to open a serial port
s = serial('COM1' );
set(s,'BaudRate' ,57600);
fopen(s);

% to write to the serial port
fprintf(s,'writing to the serial port');

% to read from the serial port (num readings of the type 'char')
Buffer = fread(s,num, 'char');
```

Each serial port on the PC is labeled COM1, COM2, etc. You need to figure out which one you're attached to.

Baud Rate or M*odulation Rate* is defined as the number of distinct symbol changes (signaling events) made to the transmission medium per second in a digitally modulated signal or a line code.

Determine the tilting angle from Duty cycle:



Formula for calculating g force.

A(g) = ((T1 / T2) -0.5) / 12.5%

The T1 duration (Memsic output) is captured by PULSIN in the variable xRaw.

xRaw = xRaw */ Scale

xGForce = ((xRaw / 100) - 500) * 8

At this point the standard equation provided by Memsic can be applied, adjusting the values to account for the pulse-width in microseconds. Fortunately, one divided by divided by 0.125 (12.5%) is eight,hence the final multiplication. The result is a signed value representing g-force in milli-g's (1/1000th g).

g = ((t1 / 100 ms) - 0.5) *8

Obtaining Tilting Angle

```
Read_Tilt:
GOSUB Read_G_Force
disp = ABS xmG / 10 MAX 99 ' x displacement
GOSUB Arcsine
xTilt = angle * (-2 * xmG.BIT15 + 1) ' fix sign
disp = ABS ymG / 10 MAX 99 ' y displacement
GOSUB Arcsine
yTilt = angle * (-2 * ymG.BIT15 + 1) ' fix sign
RETURN
' Trig routines courtesy Tracy Allen, PhD. (www.emesystems.com)
Arccosine:
disp = disp */ 983 / 3 ' normalize input to 127
angle = 63 - (disp / 2) ' approximate angle
DO ' find angle
IF (COS angle <= disp) THEN EXIT
angle = angle + 1
LOOP
angle = angle */ 360 ' convert brads to degrees
RETURN
Arcsine:
GOSUB Arccosine
angle = 90 - angle
RETURN
```

The first line "disp = disp */ 983 / 3", in the program listing you have, is a normalization step, to scale the values the Memsic puts out down to the range of +/- 127.

Note that the Stamp trig functions are only good to 8 bits around the whole circle. The circle is divided into 256 angular segments, called "brads" (binary radians). And the sine and cosine is mapped into the interval -127 to +127, to represent the scale from -1 to +1.

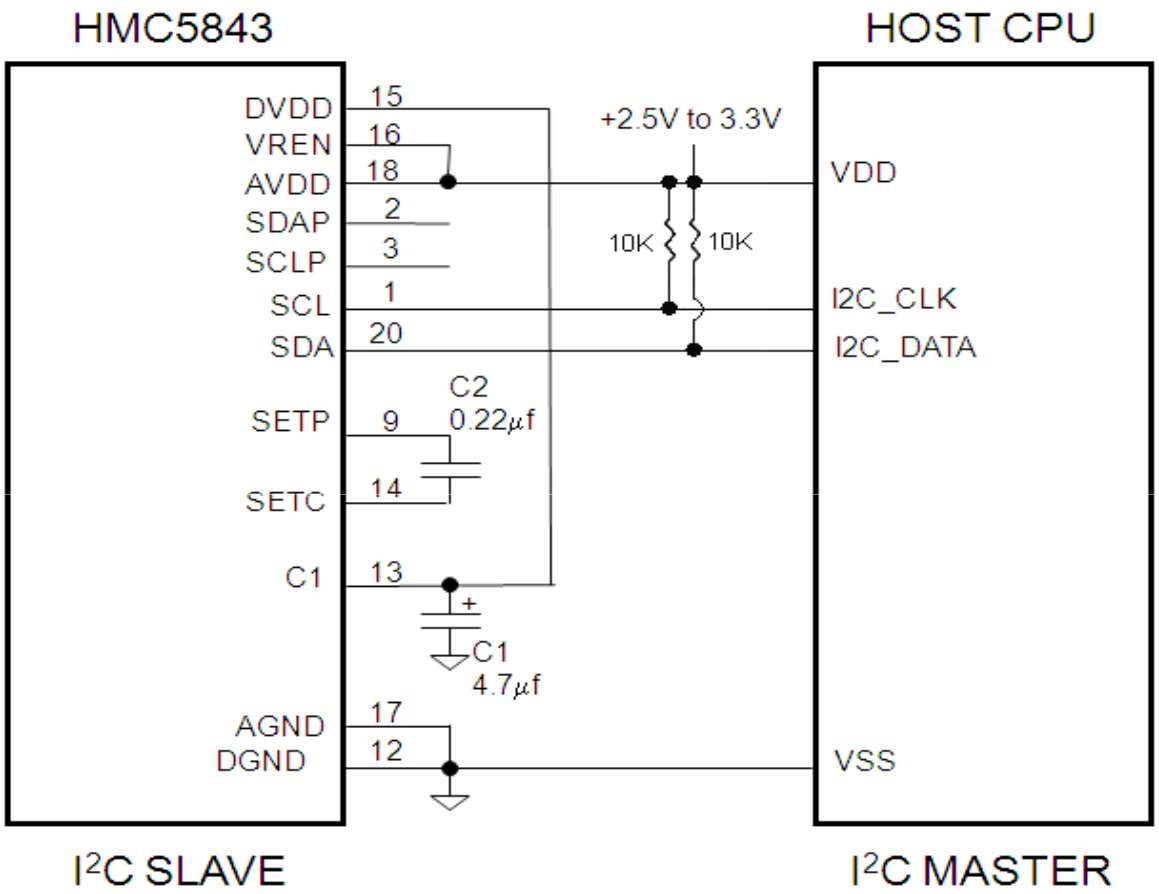
Having the Stamp's ATN function, you could use ATN (X/SQR(16129-(X*X))), where X is the value from the Memsic, without any iteration. {for trig, if the sine of an angle is X, then the tangent of that angle is X/SQR(1-X^2)}. That is just a thought. I haven't used the formula for anything. Since 1 is represented as 127, 1^2 has to be 127^2 =16129.

HMC5843 MAGNETOMETER

Type: 3-axial digital magnetometer

It includes 1043 series magneto-resistive sensors and an ASIC, which contains ADC and I2C serial bus interface.

Hardware interface



Using the Sparkfun breakout board
MSP430 - Port P1.6, P1.7
HMC5843 – Pin SCL and SDA

Software interface

Using I2C serial bus interface

- 2 wire I2C
- 7-bit serial address and 1 Acknowledgement bit.
- 8-bit read address and 8-bit write address.

HMC5843 is used as the slave

- Used in the Slave Transmitter mode
- Supports 100kHz to 400kHz data speed

Mode of operation

- Used in continuous measurement mode
- Output is taken at the 6 Data output registers, which gives hex values for each axis
- Angles are calculated using the given formulae in the datasheet.

Status:

- Working on interfacing with MSP430.
- Component ordered - Shipment yet to arrive

Challenges:

- Calculating the azimuth angle.
- Calibration of values obtained at the magnetometer.

Piezo vibration sensor

Principle Of Operation- Transverse Effect

-A small AC voltage is
created when film moves back and forth.
-converting output to DC using ADC
-When vibration is sensed the measurements are sent to the
PC using UART.
-If not vibration- takes measurements

Task: -Sense Vibration (in progress)
-Integrating parts of code as a whole system.

OVERALL STATUS:

1. Accelerometer Interface ready –Testing yet to be done
2. Magnetometer- Interfacing understood, Components re-ordered
3. Interface with the computer- Pilot codes ready
4. System Integration & Vibration sensor- Work in progress