**D.J.SANGHVI COLLEGE OF ENGINEERING**

INDUSTRIAL SUMMER TRAINING PROGRAMME

Project: Accelerometer to detect level of the surface using 12bit ADC and 8 leds.

By: Vatsal Makani

**Executive Summary**

This project intends to build A Level Surface Device using an Accelerometer.

**Step 1: Project specification**

The project is to build a accelerometer using Psoc3 using 8 leds and 12bit Analog to digital converter. The aacelerometer is used to detect the level of the surface.

In this project 8 leds are used. If the surface is perfectly horizontal the centre two leds will glow.

If the accelerometer is tilted in one direction either of the one led glows. The led that will glow will depend upon the tilt of the board.

Idea of the project is similar to the spirit level in which as the tube is turned in one direction the bubble in the tube moves in that direction.

In the project PSOC3 is being used which consists of delta sigma ADC.

**Step 2: Component Selection**

1. 1 access accelerometer. Usually there are 3 access of the accelerometer x,y,z access.We make the use of the x.

2.12 bit ADC. 12bit means it has a resolution of 12 bits.

3. Microcontroller.

4. Control register (7 bit).

**Step 3: Pin configuration**

1. Design pins should be equal to the physical pins. There should be 1 to 1 mapping. In the project number of the design pins is equal to 8.

2. Accelerometer consists of three input pins which is given to it by microcontroller. These pins are enable and select the mode of the accelerometer. These pins are

a. Enable: 1=enable, 0=disable

b. mode: 0=low power mode , 1= regular power mode.

c. st/mot (self test/motion): self testing mode=1, motion mode =0.

All the above pins are digital pins.

3. There is one analog pin to the imput of the ADC.

4. There are 7 output pins from the control register.

5. There are 8 leds. Pin no 3 of the control register is given to 4th and the 5th led.

6. Pin configuration

St : p15[0]

En : p15[2]

Mode : p15[1]

Led1 : p2[0]

Led2 : p2[1]

Led3 : p2[2]

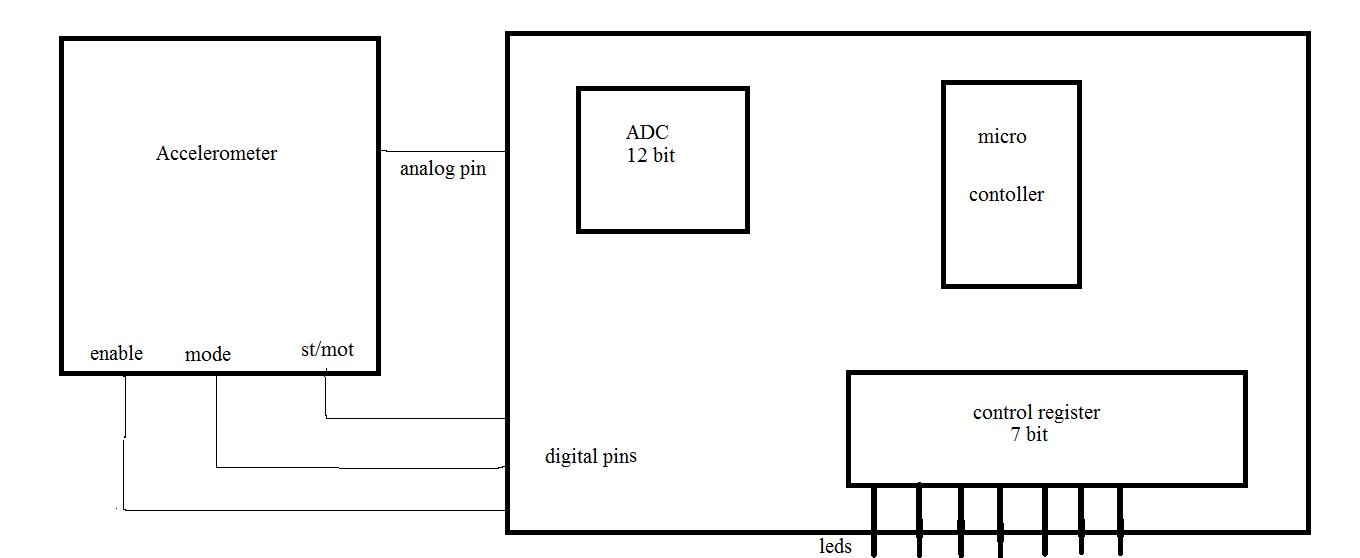
Led4 : p2[3]

Led5 : p4[0]

Led6 : p4[1]

Led7 : p4[2] and Led8 : p4[3]

**Step 4: Component configuration**



led’s

**Step 5: Algorithm**

It follows the following steps:

**Step 1:** Initialize the accelerometer, ADC.

**Step 2:** Get the value from ADC.

**Step 3:** Compare with 0g offset.

0g offset=1.65v

Vdd=3.3v

**Step 4:** voltage range is from 0 to 3.3v but in order to save power upper bounded

voltage is reduced from 3.3 to 2.048v.In psoc the reference line voltage is

1.024v. Now because of the changes made the 1.65v will correspond to 3300.

**Step 5:** The 12 bit resolution is lowered down to 5 bit resolution. This is done as

follows: The 12bit value is shifted 7 times to the right. Now the upper 5 bits of

the msb is made 0. So the 12 resolution is lowered down to 5 bit resolution.

**Step 6:** Add 3 (Because we have 8leds).

**Step 7:** Define the set limit.

**Step 8:** Shift 1 by answer. Initially the register value is 1. And if the 2nd led

Needs to be glowed it is shifted by that value. Therefore that particular led will

glow while others are off since others receive zero.

Code:

#include<device.h>

#include <project.h> // include the header files containing api.

#include"bubble.h"

void main()

{

int16 value; //declare the datatype. It is a 16 bit integer value

en\_Write(1); //enable the accelerometer.

st\_Write(0); //to set accelerometer in motion mode

mode\_Write(1); //to use it in regular power mode

CYGlobalIntEnable; //enable global interrupt.

led\_reg\_Write(0); //control register is named as led\_reg and initially

it is kept zero

ADC\_Start(); //It is an api to start ADC

ADC\_StartConvert(); //Start the conversion

while(1) // infinite loop

{

//ADC\_IsEndConversion();

value=ADC\_GetResult16();

value=value-3300;

value=value>>7;

value+=3;

if(value<0) //comparision is done

{

value=0;

}

if(value>6)

{

value=6;

}

value=1<<value; //destination=source that has to be shifted<<amount by which it

has to be shifted

>>Right shift

<<Left shift

led\_reg\_Write(value); //write the value in the register.

}

}

**Explanation:**

As the accelerometer is rotated it gives a corresponding value to the ADC.

First the header files are included. Then the accelerometer is enabled. There are some hardwar device which has the capability to interrupt the microcontroller. Therefore the global interrupt is set. Then ADC is enabled and it converts and places in the variable called as value. There are two methods of creating a infinite loop. First is while loop as shown. Second one is using for loop. It is done as for(;;).

ADC can be used in two modes –single mode and contionous mode. The project makes the use of the continuous mode.

Then the value is shifted by seven in order to reduce resolution. Then a value 3 is added. This is done in order to glow the corresponding led. If the value is greater or smaller than a particular voltage it will glow the end leds only.hence the limit is set.

Now the value after this step corresponds to the corresponding led that has to be glown.

This value is then compared. After adding 3 still if the value is less than zero it is kept zero. This is done using if statement. This means that if the value is still less than zero after adding 3 the last led should be glown no matter what is the value.

The same comparision is done on the other side. If the value is greater than six after adding three the last led on the other side should glow.

In Last step the register contains one. Depending upon the value present in variable ‘value’ the one present in the register is shifted accordingly.

Finally this value is written in the register. And the led glows.

**Step 6. Test and debug:**

The project designed was tested and was successfully implemented.

**Conclusion:**

This project may be used by  [stonemasons](http://en.wikipedia.org/wiki/Stonemason), [bricklayers](http://en.wikipedia.org/wiki/Bricklayer), other building trades workers, [surveyors](http://en.wikipedia.org/wiki/Surveyor_(surveying)" \o "Surveyor (surveying)),[millwrights](http://en.wikipedia.org/wiki/Millwright) and other metalworkers, and in some [photographic](http://en.wikipedia.org/wiki/Photography) or [videographic](http://en.wikipedia.org/wiki/Videography" \o "Videography) work. Hence this project finds a wide application in day to day life.