**DEVICE DRIVERS**

**PROJECT REPORT**

**PROJECT TITLE-ACCELEROMETER ATMEGA8 USING USB**

**Submitted by**

**SHYAM KRISHNAN K V**

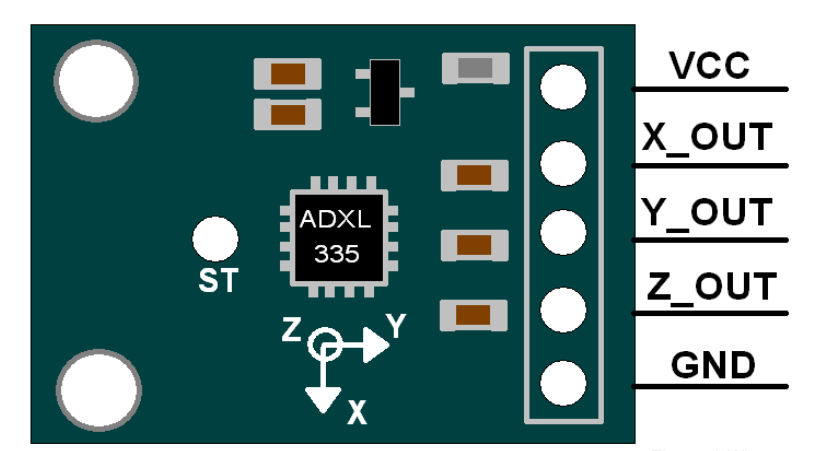
**A SIDDARTHA**

* **SUMMARY-**
* **COMPONENTS USED**

1. ADXL335 ACCELEROMETER
2. ATMEGA 8 MICROCONTROLLER
3. USBASP PROGRAMMER
4. CONNECTING WIRES AND BREADBOARDS
5. RESISTORS,CAPACITORS AND ZENER DIODES

**1.ADXL335-**An accelerometer(ADXL335) is an electromechanical device that will measure acceleration force. It shows acceleration, only due to cause of gravity i.e. g force. It measures acceleration in g unit. It can be used for tilt sensing applications (Example: In mobile phones, gaming applications etc).The ADXL335 measures acceleration along X, Y and Z axes. It gives analog voltage output proportional to the acceleration along the 3 axes. These voltages can be converted to digital signal using ADC and then processed by microcontroller to find out the tilt. Features of ADXL335 are as follows:-

1. 3V-6V DC Supply Voltage
2. Onboard LDO Voltage regulator
3. Can be interface with 3V3 or 5V Microcontroller.
4. All necessary Components are populated.
5. Ultra Low Power: 40uA in measurement mode, 0.1uA in standby@ 2.5V
6. Tap/Double Tap Detection
7. Free-Fall Detection
8. Analog output

****

**ACCELEROMETER ADXL335**

1.VCC:Power supply pin

2. X\_OUT:X axis analog output

3. X\_OUT:Y axis analog output

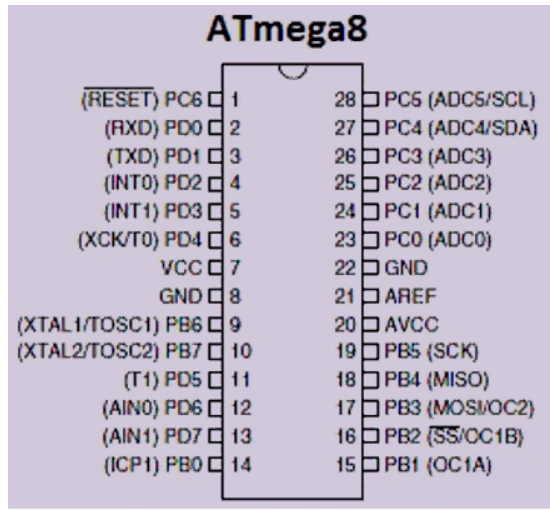
4. X\_OUT:Z axis analog output

**2.ATMEGA 8 Microcontroller-**

1.**ATmega8** is an 8-bit AVR microcontroller that is based on RISC architecture and is mainly used in the embedded system and industrial automation projects.

2.The Program memory is 8K Flash, enough to store a number of instructions while other two memories RAM and EEPROM contain 1K and 512 Bytes respectively.

3.Other features of this module include are a power-up timer, a watchdog timer, Brown out Detection, In-Circuit Serial Programming and five sleep modes.

****

**ATMEGA 8 MICROCONTROLLER**

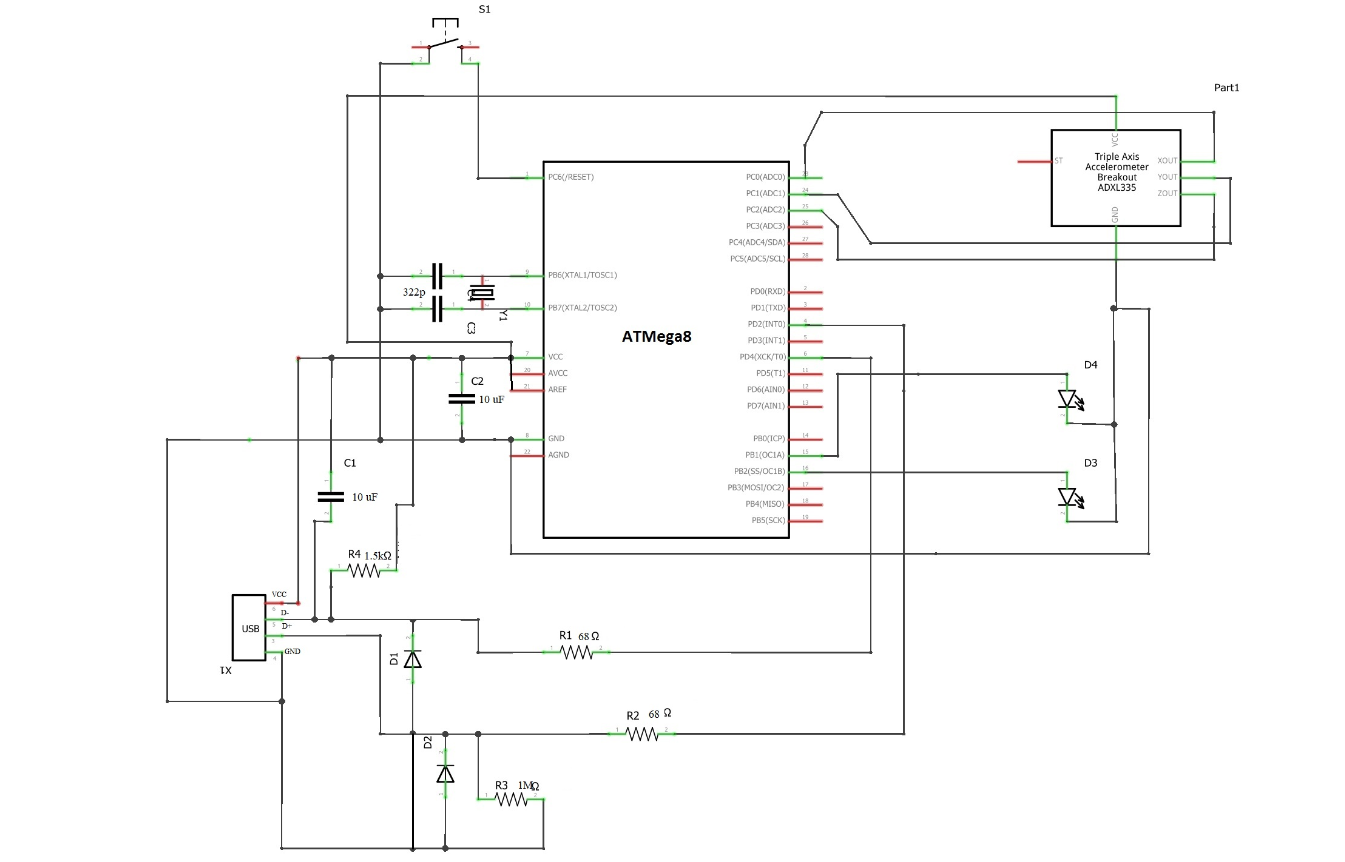
**Accelerometer** device can be implemented over low cost **USB hardware** and data can be reported back to kernel through Input Subsystem. These events reported to Input Core can be further used to provide user space application.

**3.VUSB-**V-USB is a software-only implementation of a low-speed USB device for Atmel AVR microcontrollers, making it possible to build USB hardware with almost any AVR microcontroller, not requiring any additional chip .Advantages of using microcontrollers with USB hardware:-

1. Standard AVR controllers are usually easier to obtain.
2. Most of the controllers with USB support are only available in SMD, which is almost impossible to handle for hobbyists.
3. V-USB comes with a free shared Vendor- / Product-ID pair.
4. A good free ANSI-C compiler (GNU gcc) and a free development system for Windows (Win AVR) are available for AVR.
5. AVR controllers are faster than most of the controllers with integrated USB and cost less.
6. Stand-alone operation: Some of the USB controllers download their firmware from the host computer into RAM. They don’t work without connection to the host.
7. AVR controllers have on-chip EEPROM.

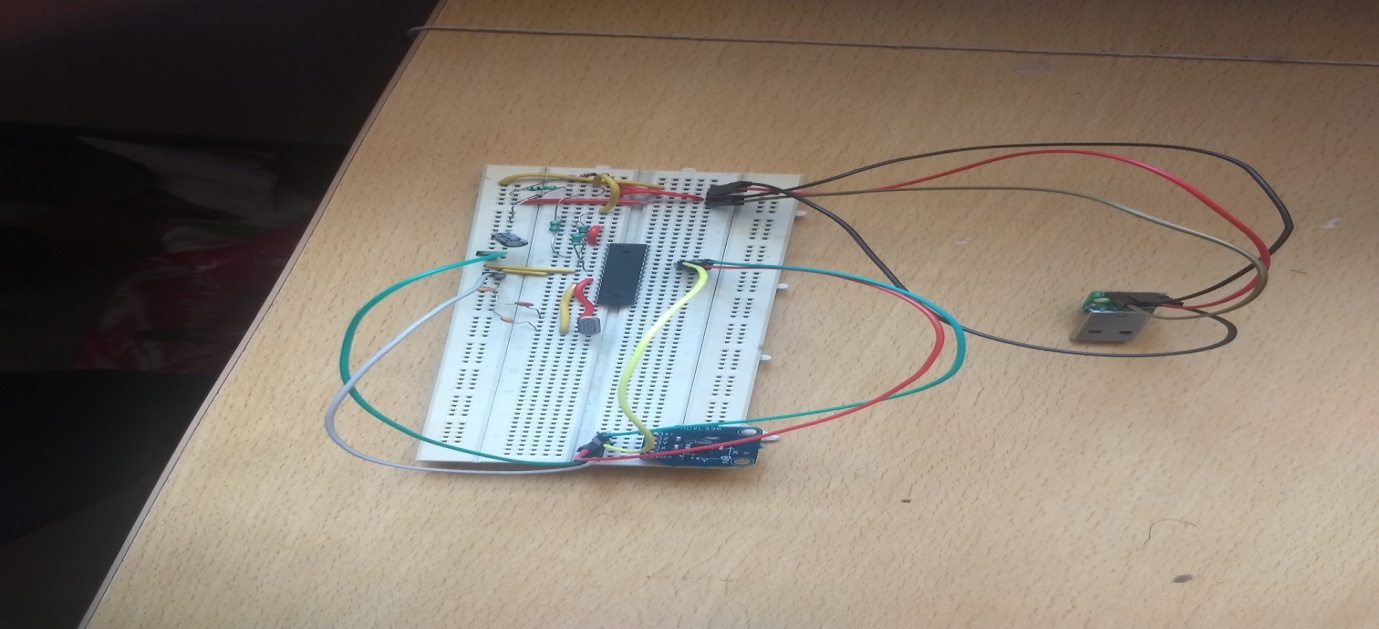
* **HARDWARE DESIGN**

**1.SCHEMATIC**



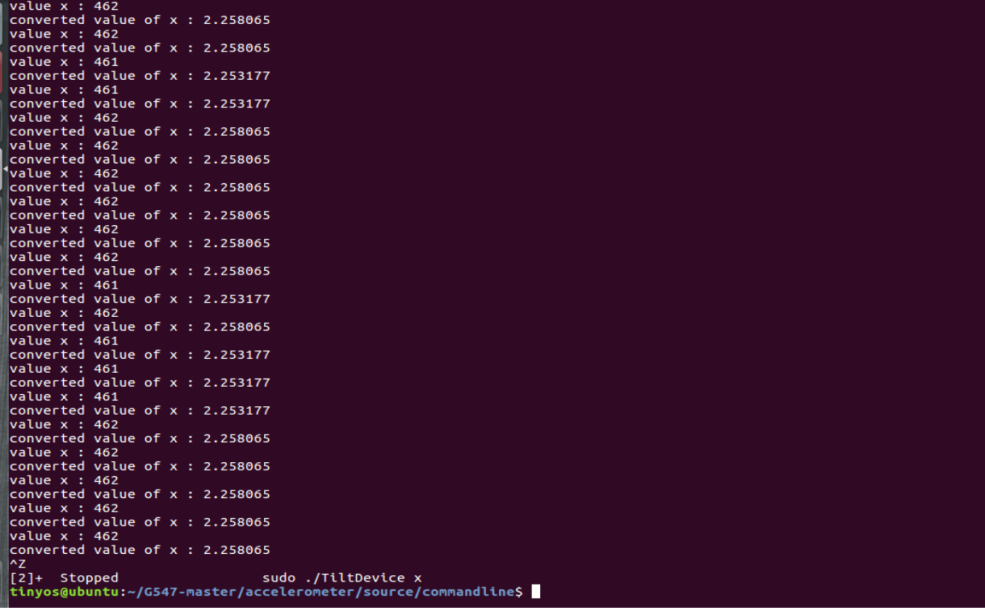
**Schematic of the circuit diagram showing connection of USB,ATMEGA8 Microcontroller and Accelerometer**

**2.ACTUAL HARDWARE CIRCUIT CONNECTION SNAP**

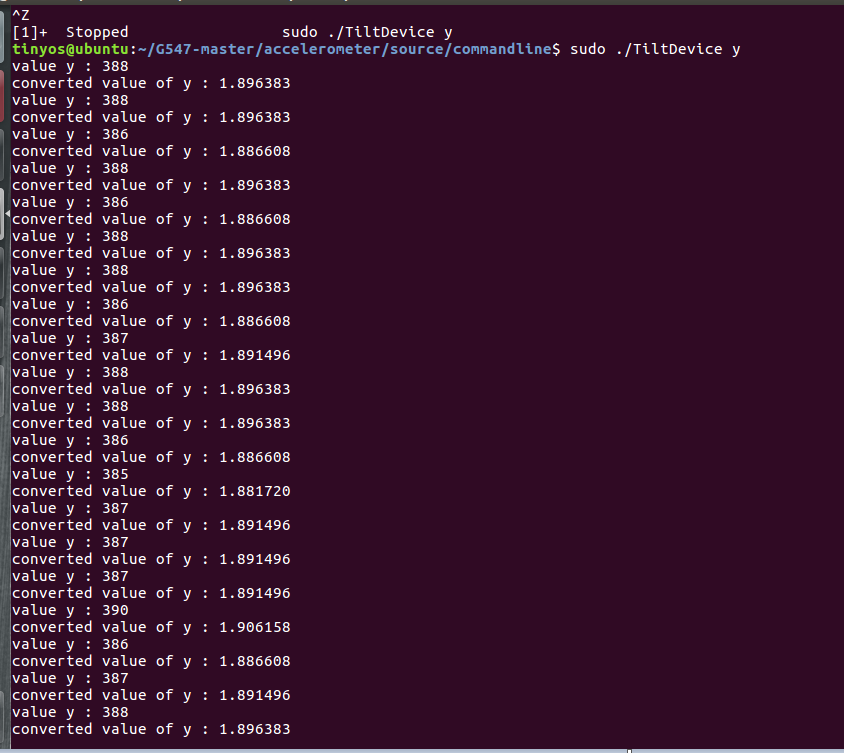


* **SAMPLE SNAP SHOTS OF THE OBTAINED RESULT**

**X-acceleration**

****

**y-acceleration**

****

* **README FILE DESCRIPTION**

1. **TILT USB Device**

Accelerometer adxl335 with atmega8 and v USB

2. **Building**

a. To Build kernel module

**make all**

b. To remove all object files

**make clean**

3**.Loading**

a. To load module with dependencies.

**modprobe input-polldev**

b. To load tilt\_usb in kernel

**insmod tilt\_usb.ko**

c. To unload tilt\_usb module from kernel

**rmmod tilt\_usb**

4. **Building and Flashing firmware**

a. Command to build firmware and generate hex file

**make hex**

b. Command to flash fuses and hex file

**make program**

c. Command to flash the fuse bits

**make fuses**

d. Command to flash the hex file

**make flash**

e. Command to delete object and hex files

**make clean**

5. **Using Command line Tool**

a. Command to build command line Tilt Device tool

**make all**

b. Command to delete all object files

**make clean**

**c.** Command to run command line Tilt Device tool to get X-axis values

**sudo ./TiltDevice x**

d. Command to run command line Tilt Device tool to get Y-axis values

**sudo ./TiltDevice y**

e. Command to run command line Tilt Device tool to get Z-axis values

**sudo ./TiltDevice z**