



**TUGAS AKHIR - TE 145561**

**MONITORING KEMIRINGAN BANGUNAN MENGGUNAKAN  
ACCELERO DENGAN MEDIA KOMUNIKASI *BLUETOOTH* DAN  
*WIFI***

**GHINAYAN DWI FIKRIYUDA  
NRP 2212039015**

**M. IBADUR RACHMAN  
NRP. 2212039039**

Dosen Pembimbing  
Suwito, ST., MT.  
Onie Meiyanto, S.Pd.

PROGRAM STUDI D3 TEKNIK ELEKTRO  
Fakultas Teknologi Industri  
Institut Teknologi Sepuluh Nopember  
Surabaya 2015



**TUGAS AKHIR - TE 145561**

**MONITORING KEMIRINGAN BANGUNAN MENGGUNAKAN  
ACCELERO DENGAN MEDIA KOMUNIKASI *BLUETOOTH* DAN  
*WIFI***

**GHINAYAN DWI FIKRIYUDA  
NRP 2212039015**

**M. IBADUR RACHMAN  
NRP. 2212039039**

**Dosen Pembimbing  
Suwito, ST., MT.  
Onie Meiyanto, S.Pd.**

**PROGRAM STUDI D3 TEKNIK ELEKTRO  
Fakultas Teknologi Industri  
Institut Teknologi Sepuluh Nopember  
Surabaya 2015**



**FINAL PROJECT - TE 145561**

***MONITORING WRYNESS OF BUILDING USING  
ACCELERO WITH BLUETOOTH AND WIFI  
COMMUNICATION MEDIA***

**GHINAYAN DWI FIKRIYUDA  
NOR 2212039015**

**M. IBADUR RACHMAN  
NOR. 2212039039**

*Advisor*  
**Suwito, ST., MT.  
Onie Meiyanto, S.Pd.**

**ELECTRICAL ENGINEERING D3 STUDY PROGRAM  
Industrial Technology Faculty  
Institut Teknologi Sepuluh Nopember  
Surabaya 2015**



**FINAL PROJECT - TE 145561**

***MONITORING WRYNESS OF BUILDING USING  
ACCELERO WITH BLUETOOTH AND WIFI  
COMMUNICATION MEDIA***

**GHINAYAN DWI FIKRIYUDA  
NOR 2212039015**

**M. IBADUR RACHMAN  
NOR. 2212039039**

*Advisor*  
**Suwito, ST., MT.  
Onie Meiyanto, S.Pd.**

**ELECTRICAL ENGINEERING D3 STUDY PROGRAM  
Industrial Technology Faculty  
Institut Teknologi Sepuluh Nopember  
Surabaya 2015**

**MONITORING KEMIRINGAN BANGUNAN MENGGUNAKAN  
ACCELERO DENGAN MEDIA KOMUNIKASI  
BLUETOOTH DAN WIFI**

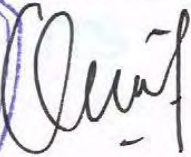
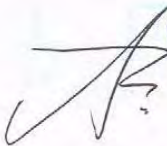
**TUGAS AKHIR**

**Diajukan Guna Memenuhi Sebagian Persyaratan  
Untuk Memperoleh Gelar Ahli Madya  
Pada  
Bidang Studi Teknik Elektro Industri  
Program Studi D3 Teknik Elektro  
Fakultas Teknologi Industri  
Institut Teknologi Sepuluh Nopember**

**Menyetujui :**

**Dosen Pembimbing I**

**Dosen Pembimbing II**

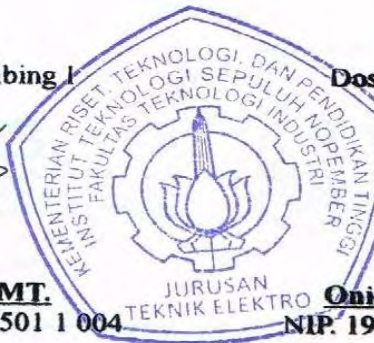


**Suwito, ST., MT.**

**NIP. 19810105 200501 1 004**

**Onie Meiyanto, S.Pd.**

**NIP. 19850501 201101 1 008**



**SURABAYA  
JUNI, 2015**

## **MONITORING KEMIRINGAN BANGUNAN MENGGUNAKAN ACCELERO DENGAN MEDIA KOMUNIKASI *BLUETOOTH* DAN *WIFI***

**Nama Mahasiswa** : Ghinayan Dwi Fikriyuda  
**NRP** : 2212 039 015  
**Nama Mahasiswa** : M. Ibadur Rachman  
**NRP** : 2212 039 039  
**Dosen Pembimbing 1** : Suwito, ST., MT.  
**NIP** : 19810105 200501 1 004  
**Dosen Pembimbing 2** : Onie Meiyanto, S.Pd.  
**NIP** : 19850501 201101 1 008

### **ABSTRAK**

Gedung-gedung pencakar langit di kota-kota besar khususnya di kota Surabaya, tanpa disadari oleh para penghuninya secara tidak langsung pasti mengalami kemiringan atau perubahan pada pondasinya karena perubahan kontur tanah, terlebih lagi di kota Surabaya yang kontur tanahnya tidak rata. Dengan semakin banyaknya gedung-gedung pencakar langit tersebut, tentunya diperlukan sebuah alat untuk dapat memonitor kemiringan dari gedung atau bangunan itu agar keberadaan dari gedung-gedung tersebut tidak membahayakan penghuni dan lingkungan sekitarnya.

Berdasarkan permasalahan tersebut tercetuslah ide “monitoring kemiringan bangunan menggunakan *accelero* dengan media komunikasi *bluetooth* dan *wifi*.” Alat ini bekerja dengan dibantu oleh aplikasi dari android, dimana nantinya para pengguna android dapat mengetahui kemiringan sebuah bangunan dengan cara mengkoneksikan *smartphone* mereka yang bersistem operasi android dengan *bluetooth* atau *wifi* yang sudah terhubung dengan *accelero*. Data yang ditampilkan pada android berupa data sudut 1 dan sudut 2. Sehingga alat ini dapat mempermudah penggunaannya.

Dari hasil pengujian kami mendapatkan hasil bahwa sensor dapat menunjukkan nilai yang baik ketika diletakkan pada bagian atap dari bangunan atau gedung. Nilai yang ditunjukkan sensor menunjukkan nilai sebenarnya, dimana untuk sudut 1 nilai presisi mencapai 100% dan sudut 2 mencapai 100%.

**Kata kunci** : Monitoring, Kemiringan, *Accelero*, Android, *Bluetooth*, *WiFi*

-----Halaman ini sengaja dikosongkan-----

**MONITORING WRYNESS OF BUILDING USING ACCELERO  
WITH BLUETOOTH AND WIFI COMMUNICATION MEDIA**

**Name of Student** : Ghinayan Dwi Fikriyuda  
**Registration Number** : 2212039015  
**Name of Student** : M. Ibadur Rachman  
**Registration Number** : 2212039039  
**Supervisor 1** : Suwito, ST., MT.  
**ID** : 19810105 200501 1 004  
**Supervisor 2** : Onie Meiyanto, S.Pd.  
**ID** : 19850501 201101 1 008

**ABSTRACT**

*Highrise buildings at metropolitan town specially at Surabaya, without be realized by their dwellers unautomatically will overcome wryness or change in their concrete that was caused by change of contour of their buildings. What ever at Surabaya that tehir buildings were not level, much of highrise buildings mentioned of course it will need tool that can monitor a wryness of those buildings, in order that a performment of those buildings can't danger the dwellers and environment.*

*Base on case mentioned there was bright idea "monitoring wryness of building using accelero with bluetooth and wifi communication media". The tool works and will be helped by android application, where later the users of android can know the wryness of the buildings by connecting the tool to smartphone that systematize android seems angle 1 and angle 2 data. So the tool can help it's users easily.*

*From the test results we get the result that the sensor can indicate a good value when placed on the roof of a building or premises. Values shown sensor indicates the actual value, which for the first angle precision value reached 100% and the angle 2 reached 100%.*

**Key words** : Monitoring, Wryness, Accelero, Android, Bluetooth, WiFi



-----Halaman ini sengaja dikosongkan-----

## **KATA PENGANTAR**

Dengan mengucapkan puji dan syukur kehadirat Allah SWT, karena atas ridho dan hidayahNya, sehingga penulis dapat menyelesaikan Tugas Akhir ini.

Maksud dan tujuan dari penulisan Tugas Akhir ini adalah untuk memenuhi persyaratan kelulusan program Studi Diploma 3 pada Jurusan D3 Teknik Elektro, Program studi Teknik Elektro Industri, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember Surabaya dengan judul:

### **MONITORING KEMIRINGAN BANGUNAN MENGGUNAKAN ACCELERO DENGAN MEDIA KOMUNIKASI *BLUETOOTH* DAN *WIFI***

Selain itu penulis juga dapat mencoba menerapkan dan membandingkan pengetahuan dan keterampilan yang diperoleh dibangku kuliah dengan kenyataan yang ada di lingkungan kerja.

Dalam Tugas Akhir ini dirancang Alat untuk mendeteksi kemiringan dari suatu bangunan menggunakan sensor *accelerometer* yang dapat diakses melalui *bluetooth* dan *wifi* menggunakan *smartphone* sehingga memudahkan bagi semua kalangan untuk menggunakannya.

Penulis mengucapkan terima kasih kepada Ibu dan Bapak penulis yang memberikan berbagai bentuk doa serta dukungan tulus tiada henti, Bapak Suwito, ST., MT. dan Bapak Onie Meiyanto, S.Pd. atas segala bimbingan ilmu, moral, dan spiritual dari awal hingga terselesaikannya Tugas Akhir ini, serta keluarga D3 Teknik Elektro Industri 2012 yang selalu memberikan doa, semangat, dan dukungannya kepada penulis. Penulis juga mengucapkan banyak terima kasih kepada semua pihak yang telah membantu baik secara langsung maupun tidak langsung dalam proses penyelesaian Tugas Akhir ini.

Penulis menyadari dan memohon maaf atas segala kekurangan pada Tugas Akhir ini. Akhir kata, semoga Tugas Akhir ini dapat bermanfaat dalam pengembangan keilmuan di kemudian hari.

Surabaya, Juni 2015

Penulis

-----Halaman ini sengaja dikosongkan-----

# DAFTAR ISI

<b>HALAMAN JUDUL</b> .....	i
<b>HALAMAN JUDUL</b> .....	iii
<b>LEMBAR PENGESAHAN</b> .....	v
<b>ABSTRAK</b> .....	vii
<b>ABSTRACT</b> .....	ix
<b>KATA PENGANTAR</b> .....	xi
<b>DAFTAR ISI</b> .....	xiii
<b>DAFTAR GAMBAR</b> .....	xv
<b>DAFTAR TABEL</b> .....	xvii

## **BAB I PENDAHULUAN**

1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	2
1.3 Batasan Masalah .....	2
1.4 Tujuan .....	2
1.5 Sistematika Laporan .....	2
1.6 Relevansi .....	3

## **BAB II TEORI PENUNJANG**

2.1 Mikrokontroler Arduino Uno .....	5
2.2 Sensor <i>Accelerometer</i> .....	6
2.3 LCD .....	9
2.4 <i>Power Supply</i> .....	11
2.4.1 LED .....	11
2.4.2 Resistor .....	12
2.4.3 Dioda .....	12
2.4.4 Kapasitor .....	13
2.4.5 IC LM7809 dan IC LM7805 .....	13
2.5 Android .....	15
2.6 <i>Ethernet Shield</i> .....	16
2.7 <i>Bluetooth</i> .....	17
2.7.1 <i>Bluetooth</i> HC-06 .....	17
2.8 <i>Router</i> .....	18

## **BAB III PERANCANGAN DAN PEMBUATAN ALAT**

3.1 Perancangan Sistem Keseuluruhan .....	21
3.2 Perancangan Mekanik .....	22

3.3	Perancangan Perangkat Keras ( <i>Hardware</i> ) .....	24
3.3.1	Perancangan Rangkaian <i>Power Supply</i> .....	24
3.3.2	Perancangan Skematik Mikrokontroler Arduino ....	25
3.3.3	Perancangan Rangkaian <i>Ethernet Shield</i> .....	26
3.3.4	Perancangan Rangkaian Sensor <i>Accelerometer</i> ADXL345 .....	27
3.3.5	Perancangan Rangkaian LCD .....	28
3.3.6	<i>Wiring Bluetooth</i> .....	29
3.4	Perancangan Perangkat Lunak ( <i>Software</i> ) .....	30
3.4.1	Pemrograman Mikrokontroler Arduino Uno .....	30
3.4.2	Pemrograman Aplikasi Android .....	33

## **BAB IV PENGUJIAN DAN ANALISA DATA**

4.1	Pengujian Rangkaian <i>Board</i> Arduino .....	39
4.2	Pengujian Rangkaian <i>Power Supply</i> .....	42
4.3	Pengujian Komunikasi <i>Bluetooth</i> .....	44
4.4	Pengujian Komunikasi <i>WiFi</i> .....	44
4.5	Pengujian Sensor <i>Accelerometer</i> ADXL345 .....	46
4.6	Pengujian <i>Software</i> .....	47
4.7	Pengujian Alat Secara Keseluruhan.....	49

## **BAB V PENUTUP**

5.1	Kesimpulan .....	53
5.2	Saran .....	53

<b>DAFTAR PUSTAKA</b> .....	55
<b>LAMPIRAN A FOTO ALAT</b> .....	A-1
<b>LAMPIRAN B LISTING PROGRAM</b> .....	B-1
<b>LAMPIRAN C DATASHEET</b> .....	C-1
<b>DAFTAR RIWAYAT HIDUP</b> .....	D-1

## DAFTAR TABEL

<b>Tabel 2.1</b>	Tabel Indeks Papan Arduino .....	6
<b>Tabel 2.2</b>	Pin LCD dan Fungsi .....	10
<b>Tabel 4.1</b>	Hasil Pengukuran per Pin saat <i>Active High</i> .....	41
<b>Tabel 4.2</b>	Hasil Pengukuran per Pin saat <i>Active Low</i> .....	41
<b>Tabel 4.3</b>	Pengujian Rangkaian <i>Power Supply 5V</i> .....	42
<b>Tabel 4.4</b>	Pengujian Rangkaian <i>Power Supply 9V</i> .....	43
<b>Tabel 4.5</b>	Pengujian Komunikasi <i>Bluetooth HC-06</i> .....	44
<b>Tabel 4.6</b>	Pengujian Koneksi <i>Router</i> .....	45
<b>Tabel 4.7</b>	Tabel Pengamatan Pengujian Sensor .....	46
<b>Tabel 4.8</b>	Hasil Pengujian Busur 1 .....	50
<b>Tabel 4.9</b>	Hasil Pengujian Busur 2 .....	51

-----Halaman ini sengaja dikosongkan-----

## DAFTAR GAMBAR

<b>Gambar 2.1</b>	<i>Board</i> Mikrokontroler Arduino Uno .....	6
<b>Gambar 2.2</b>	Kabel USB <i>Board</i> Arduino Uno .....	6
<b>Gambar 2.3</b>	Sensor <i>Accelerometer</i> Tipe ADXL345 .....	7
<b>Gambar 2.4</b>	Sudut dari Sensor <i>Accelerometer</i> ADXL345 .....	7
<b>Gambar 2.5</b>	Persamaan Sudut 1 dan Persamaan Sudut 2 .....	8
<b>Gambar 2.6</b>	<i>Library</i> Sensor <i>Accelerometer</i> ADXL345 .....	8
<b>Gambar 2.7</b>	LCD 16x4 Karakter .....	9
<b>Gambar 2.8</b>	Blok Diagram LCD .....	9
<b>Gambar 2.9</b>	Konfigurasi pada 16 Pin Konektor LCD .....	9
<b>Gambar 2.10</b>	LED ( <i>Light Emitting Diode</i> ) .....	11
<b>Gambar 2.11</b>	Resistor 470Ω .....	12
<b>Gambar 2.12</b>	Dioda 1N4007 .....	13
<b>Gambar 2.13</b>	Kapasitor 1000µF/50V .....	13
<b>Gambar 2.14</b>	Kapasitor 100µF/50V .....	13
<b>Gambar 2.15</b>	IC LM 7809 dan Skematik IC LM 7809 .....	14
<b>Gambar 2.16</b>	IC LM 7805 dan Skematik IC LM 7805 .....	14
<b>Gambar 2.17</b>	Susunan Kaki pada IC .....	15
<b>Gambar 2.18</b>	Lambang Android .....	16
<b>Gambar 2.19</b>	<i>Ethernet Shield</i> dan Kabel <i>Network</i> RJ-45 .....	17
<b>Gambar 2.20</b>	<i>Bluetooth</i> HC-06 .....	18
<b>Gambar 2.21</b>	<i>Router</i> .....	19
<b>Gambar 3.1</b>	Diagram Fungsional Sistem Alat Secara Keseluruhan .....	21
<b>Gambar 3.2</b>	Miniatur Bangunan .....	22
<b>Gambar 3.3</b>	<i>Box</i> Peralatan Elektronik .....	23
<b>Gambar 3.4</b>	Desain <i>Prototype</i> Indikator Kemiringan .....	23
<b>Gambar 3.5</b>	Bentuk <i>Prototype</i> Indikator Kemiringan .....	24
<b>Gambar 3.6</b>	Simulasi Rangkaian <i>Power Supply</i> 5V di Proteus .....	25
<b>Gambar 3.7</b>	Simulasi Rangkaian <i>Power Supply</i> 9V di Proteus .....	25
<b>Gambar 3.8</b>	Rangkaian <i>Power Supply</i> pada <i>Dote</i> PCB .....	25
<b>Gambar 3.9</b>	Skematik Mikrokontroler Arduino Uno .....	26
<b>Gambar 3.10</b>	Rangkaian <i>Ethernet Shield</i> .....	27
<b>Gambar 3.11</b>	Rangkaian Sensor <i>Accelerometer</i> ADXL345 .....	28
<b>Gambar 3.12</b>	Rangkaian LCD .....	29
<b>Gambar 3.13</b>	Skematik <i>Wiring Bluetooth</i> .....	29
<b>Gambar 3.14</b>	Inisialisasi Program Mikrokontroler Arduino .....	30
<b>Gambar 3.15</b>	Program untuk Mencari Besar Sudut .....	31
<b>Gambar 3.16</b>	Program untuk Komunikasi <i>Bluetooth</i> .....	32
<b>Gambar 3.17</b>	Program untuk Komunikasi <i>WiFi</i> .....	32



<b>Gambar 3.18</b>	<i>Flow Chart</i> Program Arduino .....	33
<b>Gambar 3.19</b>	Tampilan Layar <i>Login</i> .....	34
<b>Gambar 3.20</b>	Pengisian Identitas saat <i>Login</i> .....	34
<b>Gambar 3.21</b>	Tampilan Setelah <i>Login</i> .....	35
<b>Gambar 3.22</b>	Pemilihan <i>Bluetooth</i> untuk Media Komunikasi ....	35
<b>Gambar 3.23</b>	Pemilihan <i>WiFi</i> untuk Media Komunikasi .....	36
<b>Gambar 3.24</b>	Tampilan Monitoring Kemiringan Bangunan .....	36
<b>Gambar 3.25</b>	<i>Flow Chart</i> Sistem Aplikasi Android .....	37
<b>Gambar 4.1</b>	Program Kondisi Logika 1 .....	40
<b>Gambar 4.2</b>	Program Kondisi Logika 0 .....	41
<b>Gambar 4.3</b>	Bentuk Fisik Rangkaian <i>Power Supply</i> .....	43
<b>Gambar 4.4</b>	Pin Out <i>Bluetooth</i> HC-06 .....	44
<b>Gambar 4.5</b>	Koneksi antara <i>Ethernet Shield</i> dengan <i>Router</i> ....	46
<b>Gambar 4.6</b>	Pengujian Sensor <i>Accelerometer</i> ADXL345 .....	47
<b>Gambar 4.7</b>	Pengujian <i>Software</i> dengan Komunikasi <i>Wifi</i> .....	48
<b>Gambar 4.8</b>	Pengujian <i>Software</i> dengan Komunikasi <i>Bluetooth</i> .....	48
<b>Gambar 4.9</b>	Pengujian Keseluruhan Alat Monitoring Kemiringan Bangunan Menggunakan <i>Accelerometer</i> dengan Media Komunikasi <i>Bluetooth</i> dan <i>Wifi</i> .....	52

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Surabaya adalah salah satu kota yang menjadi pusat perekonomian dan pembangunan di Indonesia. Sebagai salah satu pusat perekonomian dan pembangunan di Indonesia, tentu banyak pengusaha yang menjalankan bisnisnya di kota pahlawan ini. Melihat semakin bertambah majunya bisnis tersebut, maka para pengusaha semakin berlomba-lomba untuk membangun bangunan-bangunan pencakar langit yang menjulang tinggi.

Melihat kontur tanah di Surabaya yang tidak rata, memungkinkan bangunan-bangunan tersebut mengalami kemiringan yang tidak bisa dilihat secara langsung. Kemiringan tersebut sedikit demi sedikit akan berakibat fatal pada kekuatan bangunan tersebut. Keadaan seperti itu tentu sangat berbahaya bagi masyarakat dan lingkungan di sekitar bangunan pencakar langit tersebut. Masyarakat pun masih belum banyak yang peduli dengan hal-hal semacam itu, mengingat teknologi yang ada sangat mahal dan tidak dapat dijangkau oleh masyarakat pada umumnya.

Melihat dari kebiasaan masyarakat Indonesia yaitu tidak akan hati-hati atau waspada bila belum terjadi kejadian yang tidak diinginkan, maka pembuatan sebuah teknologi agar dapat memantau keadaan dari suatu bangunan yang terjangkau dan tidak memerlukan biaya yang begitu mahal akan mejadi sebuah kebutuhan bagi masyarakat. Sehingga dapat memudahkan masyarakat untuk memantau keadaan dari bangunan di sekitarnya agar dapat mencegah adanya bangunan yang roboh.

Dengan melihat kondisi tersebut tercetuslah judul “Monitoring Kemiringan Bangunan Menggunakan *Accelero* dengan Media Komunikasi *Bluetooth* dan *Wifi*”, dengan harapan agar masyarakat dapat lebih peduli terhadap lingkungannya dan membantu pemerintah agar tidak terjadi bangunan yang roboh. Maka kedepannya diharapkan tidak akan terjadi bangunan roboh, karena terdapat teknologi yang terjangkau yang dapat digunakan untuk memantau kemiringan sebuah bangunan secara berkala, sehingga perbaikan terhadap bangunan-bangunan yang mengalami kemiringan dapat dilakukan sebelum terjadi kejadian yang dapat merugikan berbagai pihak.

## 1.2 Permasalahan

Melihat dari latar belakang di atas, maka dapat dirumuskan beberapa permasalahan diantaranya:

1. Kemiringan pada bangunan yang dinamis dapat membahayakan masyarakat di sekitarnya.
2. Pemantauan terhadap kondisi sebuah bangunan belum dilakukan secara berkala.
3. Belum adanya teknologi yang harganya dapat dijangkau oleh masyarakat umum.

## 1.3 Batasan Masalah

Dalam penyusunan tugas akhir ini perlu diberikan beberapa batasan permasalahan dengan tujuan agar pembahasan tidak meluas dan menyimpang dari tujuan. Adapun batasan permasalahan dari sistem yang dirancang ini adalah :

1. Parameter yang diukur adalah sudut kemiringan dari suatu bangunan.
2. Sensor *accelerometer* digunakan untuk mendeteksi kemiringan dari bangunan.
3. Pada Tugas Akhir kali ini hanya difokuskan pada hal monitoring kemiringan bangunan saja.

## 1.4 Tujuan

Tujuan yang ingin dicapai dalam penulisan Tugas Akhir ini:

1. Membuat pendeteksi kemiringan bangunan dengan sistem yang lebih mudah untuk digunakan oleh masyarakat umum dengan harga yang terjangkau
2. Mencegah terjadinya kejadian yang tidak diinginkan yaitu bangunan yang roboh khususnya di Surabaya.

## 1.5 Sistematika Laporan

Dalam penyusunan buku Tugas Akhir ini, pembahasan mengenai sistem alat yang dibuat dibagi menjadi lima bab dengan sistematika sebagai berikut :

### **BABI : PENDAHULUAN**

Dalam bab ini membahas tentang latar belakang, permasalahan, batasan masalah, maksud dan tujuan, sistematika laporan, serta relevansi penulisan pada Tugas Akhir.

**BAB II : TEORI PENUNJANG**

Menjelaskan teori penunjang yang mendukung dalam perencanaan dan pembuatan alat

**BAB III : PERANCANGAN DAN PEMBUATAN ALAT**

Membahas tentang perencanaan dan pembuatan perangkat keras yang meliputi rangkaian-rangkaian, desain bangun, dan perangkat lunak yang meliputi program yang akan digunakan untuk mengaktifkan alat tersebut.

**BAB IV : PENGUJIAN DAN ANALISA DATA**

Dalam bab ini membahas tentang pengukuran, pengujian, dan analisa terhadap prinsip kerja dan proses dari alat yang telah dibuat.

**BAB V : PENUTUP**

Dalam bab ini berisi kesimpulan yang diperoleh dari pembuatan Tugas Akhir ini dan saran-saran untuk pengembangan lebih lanjut.

**1.6 Relevansi**

Manfaat alat ini adalah untuk dapat memberikan akses kepada masyarakat umum agar dapat mengetahui kondisi suatu bangunan yang ada di lingkungan sekitarnya, karena dengan itu masyarakat dapat mendapat informasi sejak dini dan dapat membantu mencegah adanya bangunan yang roboh.

-----Halaman ini sengaja dikosongkan-----

## **BAB II**

### **TEORI PENUNJANG**

Pada bab ini membahas tentang teori dasar, teori penunjang dari peralatan-peralatan yang digunakan dalam pembuatan alat Monitoring Kemiringan Bangunan Menggunakan *Accelero* dengan Media Komunikasi *Bluetooth* dan *Wifi*.

#### **2.1 Mikrokontroler Arduino Uno**

Pengaturan sistem *input output* pada perancangan alat ini akan diatur oleh Arduino Uno. Secara umum Arduino terdiri dari dua bagian, yaitu: 1. *Hardware* papan *input/output (I/O)* . 2. *Software* Arduino meliputi IDE untuk menulis program, *driver* untuk dengan komputer, contoh program dan *library* untuk pengembangan program.

Arduino Uno adalah board mikrokontroler berbasis ATmega328. Uno memiliki 14 pin digital input/output (dimana 6 pin dapat digunakan sebagai output PWM), 6 input analog, resonator keramik 16 MHz, koneksi usb, Jack listrik, header ICSP, dan tombol reset. Uno dibangun berdasarkan apa yang diperlukan untuk mendukung mikrokontroler, sumber daya bisa menggunakan power usb (jika terhubung kekomputer dengan kabel usb) dan juga dengan adaptor atau baterai. Arduino Uno berbeda dari semua papan sebelumnya dalam hal tidak menggunakan FTDI chip driver usb-to-serial. Sebaliknya, fitur ATmega16U2 (ATmega8U2 sampai versi R2) diprogram sebagai converter usb-to-serial. Revisi 2 dari Uno memiliki resistor pulling 8U2 HWB yang terhubung ketanah, sehingga lebih mudah untuk menggunakan mode DFU.

Mikrokontroler Arduino Uno memiliki keistimewaan tersendiri dibanding mikrokontroler yang lainnya. Berikut ini adalah fitur-fitur yang dimiliki oleh mikrokontroler Arduino Unoyaitu 1,0 pinout: tambah SDA dan SCL pin yang dekat ke pin aref dan dua pin baru lainnya ditempatkan dekat ke pin RESET, dengan IO REF yang memungkinkan sebagai buffer untuk beradaptasi dengan tegangan yang disediakan dari board sistem. Pengembangannya, sistem akan lebih kompatibel dengan Prosesor yang menggunakan AVR, yang beroperasi dengan 5V dan dengan Arduino Karena yang beroperasi dengan 3.3V. Yang kedua adalah pin tidak terhubung, yang disediakan untuk tujuan pengembangannya.



**Gambar 2.1** Board Mikrokontroler ArduinoUno



**Gambar 2.2** Kabel USB Board Arduino

**Tabel 2.1** Tabel Indeks Papan Arduino

Mikrokontroler	ATmega328
Tegangan pengoperasian	5V
Tegangan input yang disarankan	7-12V
Batas tegangan input	6-20V
Jumlah pin I/O digital	14 (6 di antaranya menyediakan keluaran PWM)
Jumlah pin input analog	6
Arus DC tiap pin I/O	40 mA
Arus DC untuk pin 3.3V	50 mA
Memori Flash	32 KB (ATmega328), sekitar 0.5 KB digunakan oleh bootloader
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Clock Speed	16 MHz

## 2.2 Sensor Accelerometer

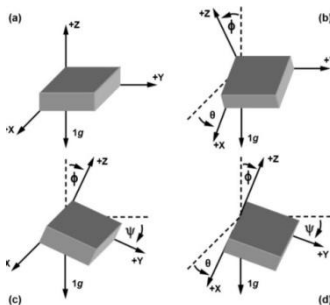
Sensor *Accelerometer* adalah sensor yang digunakan untuk mengukur percepatan suatu objek. *Accelerometer* mengukur percepatan *dynamic* dan *static*. Pengukuran *dynamic* adalah pengukuran percepatan pada objek bergerak, sedangkan pengukuran *static* adalah pengukuran terhadap gravitasi bumi. Untuk mengukur sudut kemiringan (*tilt*). Prinsip kerja yang digunakan adalah prinsip percepatan (*acceleration*).

Dalam tugas akhir ini digunakan sensor *Accelerometer* tipe ADXL345. Pada gambar 2.3 adalah bentuk fisik dari sensor *accelerometer* tipe ADXL345.



**Gambar 2.3** Sensor *Accelerometer* Tipe ADXL345

Pada gambar 2.4 menunjukkan sudut dari sensor *accelerometer*. Setelah mengetahui sudut dari sensor *accelerometer* ADXL345, selanjutnya menentukan persamaan sudut untuk mengetahui besaran sudut dari *output* sensor.



**Gambar 2.4** Sudut dari Sensor *Accelerometer* ADXL345

Persamaan sudut ini kami bagi menjadi 2 bagian yaitu persamaan sudut 1 dan persamaan sudut 2. Dalam gambar 2.5 menunjukkan rumus untuk persamaan sudut yang nantinya digunakan untuk mencari besarnya sudut dari bangunan yang akan di-*monitoring*. Berikut ini adalah persamaan sudut pada gambar 2.5.



$$\theta = \tan^{-1} \left( \frac{A_{X,OUT}}{\sqrt{A^2_{Y,OUT} + A^2_{Z,OUT}}} \right)$$

$$\psi = \tan^{-1} \left( \frac{A_{Y,OUT}}{\sqrt{A^2_{X,OUT} + A^2_{Z,OUT}}} \right)$$

$$\phi = \tan^{-1} \left( \frac{\sqrt{A^2_{X,OUT} + A^2_{Y,OUT}}}{A_{Z,OUT}} \right)$$

**Gambar 2.5** Persamaan Sudut 1 dan Persamaan Sudut 2

Agar pemrograman untuk sensor lebih mudah maka dibutuhkan *library* dari pemrograman sensor *accelerometer* tipe ADXL345 ini. *Library* yang digunakan adalah *AdafruitLibrary*.

```
sensors_event_t event;
accel.getEvent(&event);
accel.setRange(ADXL345_RANGE_8_G);
datax=((event.acceleration.x)); //fungsi x
datay=((event.acceleration.y)); //fungsi y
dataz=((event.acceleration.z)); //fungsi z
```

**Gambar 2.6** *Library* Sensor *Accelerometer* ADXL345

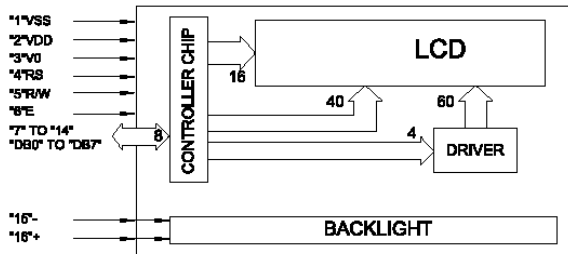
Karena *accelerometer* yang memiliki *output* percepatan ( $m/s^2$ ) akan memberikan nilai keluaran ketika terjadi perubahan posisi. Ketika *accelerometer* dalam keadaan diam atau tidak bergerak, maka tidak ada percepatan yang terjadi. Maka dari itu digunakan fungsi “*sensor\_event\_t event*” yang digunakan untuk mengambil satu dari sekian banyak nilai *output* data sensor pada 3-axis yakni x,y,z karena sifat *ouput* data dari sensor yang berubah-ubah. Kemudian fungsi dari “*accel.getEvent (&event)*” yang akan kembali mengambil data ketika terjadi perubahan posisi yang menghasilkan nilai *ouput* baru.

### 2.3 LCD

LCD (*Liquid Crystal Display*) adalah suatu display dari bahan cairan kristal yang pengoperasiannya menggunakan sistem dot matriks. LCD banyak digunakan sebagai *display* dari alat-alat elektronik seperti kalkulator, *multitester digital*, jam *digital* dan sebagainya.

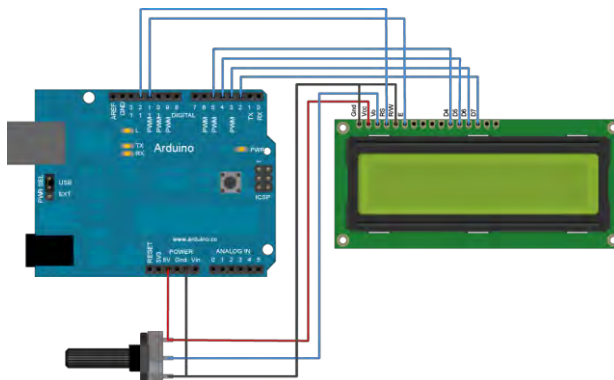


**Gambar 2.7** LCD 16x4 Karakter



**Gambar 0.8** Blok Diagram LCD

LCD dapat dengan mudah dihubungkan dengan mikrokontroler Arduino Uno. LCD yang digunakan pada Tugas Akhir ini adalah LCD 16 x 4, lebar display 4 baris 16 kolom, yang mempunyai 16 *pin* konektor. Seperti pada Gambar 2.9 :



**Gambar 0.9** Konfigurasi pada 16 Pin Konektor LCD

**Tabel 0.2** PinLCD dan Fungsi

<b>PIN</b>	<b>Deskripsi</b>	<b>Fungsi</b>	<b>Data Logika</b>	<b>Keterangan</b>
1.	VSS (Ground)	<i>Ground Voltage</i>	-	0 V
2.	VDD (VCC)	Tegangan sumber	-	+5V
3.	VEE	<i>Contrast Voltage</i>	-	0 -Vdd
4.	RS	<i>Register Select</i> 0 = <i>Instruction Register</i> 1 = <i>Data Register</i>	0/1	
5.	R/W	<i>Read/Write, to choose write or read mode</i> 0 = <i>write mode</i> 1 = <i>read mode</i>	0/1	
6.	E	<i>Enable</i> 0 = <i>start to lacht data to LCD character</i> 1 = <i>disable</i>	0/1/floating	
7.	D0	DATA	0/1	<i>bit ke-0(LSB)</i>
8.	D1		0/1	<i>bit ke-1</i>
9.	D2		0/1	<i>bit ke-2</i>
10.	D3		0/1	<i>bit ke-3</i>
11.	D4		0/1	<i>bit ke-4</i>
12.	D5		0/1	<i>bit ke-5</i>
13.	D6		0/1	<i>bit ke-6</i>
14.	D7		0/1	<i>bit ke-7 (MSB)</i>
15.	Anoda	<i>Back Plane Light, lampu background.</i>	-	
16.	Katoda	<i>Back Plane Light, lampu background.</i>	-	

## 2.4 Power Supply

*Power supply* adalah alat yang berfungsi untuk menyalurkan energi listrik atau bentuk energi jenis apapun yang sering digunakan untuk menyalurkan energi listrik. Secara prinsip rangkaian *power supply* yang digunakan pada alat ini bertugas untuk memberikan tegangan input pada arduino, sensor *accelerometer*, dan *bluetooth*. Tegangan input yang dibuat yaitu 9V dan 5V. Tegangan 9V merupakan tegangan yang masuk pada arduino, karena tegangan input yang disarankan berkisar 7V-12V. jika disuplai oleh tegangan yang lebih kecil dari 7V maka *board* arduino menjadi tidak stabil. Sehingga tegangan keluaran dari pin arduino menjadi lebih kecil dari 5V. Jika disuplai lebih dari 12V maka dapat membuat board arduino menjadi panas. Sedangkan tegangan 5V merupakan tegangan masukan pada sensor *accelero*, *bluetooth*. *Power Supply* yang dipakai pada alat ini terdiri atas, IC LM 7809, IC LM7805, dioda 1N4007 1A dan kapasitor 100 $\mu$ F/50V. IC LM 7805 dan IC LM 7809 adalah suatu komponen yang mampu membatasi tegangan *input* dengan *output* menjadi 5V dan 9V. Komponen *power supply* yang kami gunakan antara lain, yaitu:

### 2.4.1 LED

LED (*Light Emitting Diode*) yang digunakan pada rangkaian *power supply* berfungsi sebagai indikator *power supply*. Cara kerja LED pada rangkaian ini adalah menerima tegangan kemudian memancarkan cahaya. Prinsip kerja LED sebenarnya sama saja dengan prinsip kerja dioda yang hanya memerlukan tegangan tertentu untuk dapat beroperasi. Artinya apabila tidak ada tegangan yang mengalir komponen tersebut maka komponen tersebut tidak akan bekerja. Pada gambar 2.10 dibawah ini adalah gambar dari bentuk fisik LED (*Light Emitting Diode*).



**Gambar 2.10** LED (*Light Emitting Diode*)

## 2.4.2 Resistor

Resistor adalah komponen yang paling sering digunakan dalam sebuah rangkaian. Termasuk dalam rangkaian *power supply* pada perancangan alat ini. Karakteristik utama dari resistor adalah resistansinya dan daya listrik yang dapat dihantarkan. Karakteristik lain termasuk koefisien suhu, desah listrik, dan induktansi. Resistor pada *power supply* ini menggunakan resistor 470Ω. Kebutuhan daya resistor harus cukup dan disesuaikan dengan kebutuhan. Resistor pada rangkaian ini digunakan sebagai hambatan untuk LED yang digunakan sebagai indikator. Pada gambar 2.11 dibawah adalah gambar dari bentuk fisik resistor 470Ω.



**Gambar 2.11** Resistor 470Ω

## 2.4.3 Dioda

Dioda adalah komponen aktif dua kutub yang pada umumnya bersifat semikonduktor, yang memperbolehkan arus listrik mengalir ke satu arah (kondisi panjar maju) dan menghambat arus dari arah sebaliknya (kondisi panjar mundur). Dioda dapat disamakan sebagai fungsi katup di dalam bidang elektronika. Dioda sebenarnya tidak menunjukkan karakteristik kesearahan yang sempurna, melainkan mempunyai karakteristik hubungan arus dan tegangan kompleks yang tidak linier dan seringkali tergantung pada teknologi atau material yang digunakan serta parameter penggunaan. Beberapa jenis dioda juga mempunyai fungsi yang tidak ditujukan untuk penggunaan penyearahan.. Dioda yang dipakai pada rangkaian *power supply* pada alat ini adalah dioda tipe 1N4007. Spesifikasi arus : 1A. *Input* tegangan puncak :1000V. Dibawah ini pada gambar 2.12 adalah bentuk fisik dari Dioda 1N4007.



**Gambar 2.12** Dioda 1N4007

### **2.4.4 Kapasitor**

Pada perancangan rpower supply kali ini menggunakan kapasitor  $1000\mu\text{F}/50\text{V}$  dan  $100\mu\text{F}/50\text{V}$ . Fungsi kapasitor ini sebagai penyaring atau filter tegangan di rangkaian power supply. Pada gambar 2.13 adalah gambar dari bentuk fisik kapasitor  $1000\mu\text{F}/50\text{V}$  dan gambar 2.14 adalah kapasitor  $100\mu\text{F}/50\text{V}$ .



**Gambar 2.13** Kapasitor  $1000\mu\text{F}/50\text{V}$

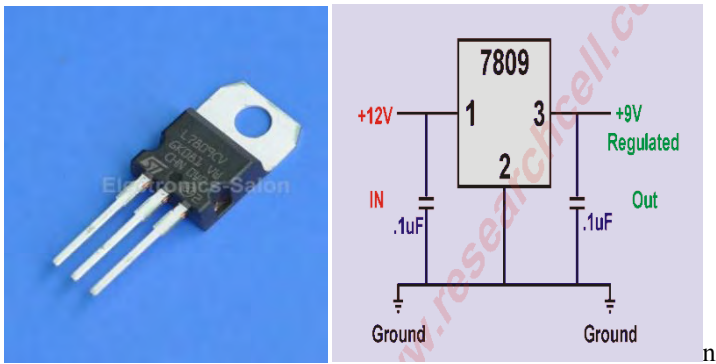


**Gambar 2.14** Kapasitor  $100\mu\text{F}/50\text{V}$

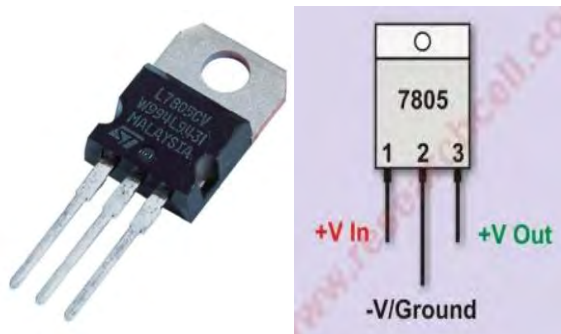
#### **1.1.1 IC LM7809 dan IC LM7805**

IC LM7809 dan IC LM7805 merupakan salah satu komponen yang diperlukan untuk pembuatan *power supply*. IC LM tersebut

mempunyai tegangan tetap dengan tiga kaki. Yaitu Vin, GND dan Vout. Angka terakhir pada tipe IC LM7809 dan IC LM7805 merupakan besaran keluaran tegangan dari IC tersebut. Karena pada rangkaian *power supply* ini menggunakan IC LM7809 dan IC LM7805 oleh karena itu tegangan keluaran yang dihasilkan oleh power supply ini juga sebesar 9V dan 5V. Pada gambar 2.15 merupakan bentuk fisik dan skematik dari IC LM7809. Sedangkan pada gambar 2.16 merupakan bentuk fisik dan pin out diagram IC LM7805. Setelah itu dijelaskan pada gambar 2.17 adalah susunan kaki pada IC.



**Gambar 2.15** IC LM7809 dan Skematik IC LM7809



**Gambar 2.16** IC LM7805 dan Pin Out Diagram IC LM7805



**Gambar 2.17** Susunan Kaki pada IC

## 2.5 Android

Android pertama kali dikembangkan oleh perusahaan bernama Android Inc., pada tahun 2005 diakuisisi oleh raksasa Internet yaitu Google. Android dibuat dengan basis kernel Linux yang telah dimodifikasi, dan untuk setiap release-nya diberi kode nama berdasarkan nama hidangan makanan. Android adalah sistem operasi *Mobile Phone* berbasis Linux. Android bersifat *open source* yang *source code*-nya diberikan secara gratis bagi para pengembang untuk menciptakan aplikasi mereka agar dapat berjalan di Android.

Antarmuka pengguna Android didasarkan pada manipulasi langsung, menggunakan masukan sentuh yang serupa dengan tindakan di dunia nyata, seperti menggesek, mengetuk, mencubit, dan membalikkan cubitan untuk memanipulasi obyek di layar. Kode dengan sumber terbuka dan lisensi perizinan pada Android memungkinkan perangkat lunak untuk dimodifikasi secara bebas dan didistribusikan oleh para pembuat perangkat, operator nirkabel, dan pengembang aplikasi. Keunggulan utama Android adalah gratis dan *open source*, yang membuat *smartphone* Android dijual lebih murah dibandingkan dengan BlackBerry atau iPhone meski fitur (hardware) yang ditawarkan Android lebih baik.

Beberapa fitur utama dari Android antara lain *WiFi* hotspot, *Multi-touch*, *Multitasking*, GPS, *accelerometers*, *support java*, mendukung banyak jaringan (GSM/EDGE, IDEN, CDMA, EV-DO, UMTS, *Bluetooth*, *Wi-Fi*, LTE & WiMAX) serta juga kemampuan dasar



*handphone* pada umumnya. Pada gambar 2.18 merupakan gambar lambing android pada umumnya.



**Gambar 2.18** Lambang Android

## **2.6 Ethernet Shield**

*Ethernet* adalah teknologi jaringan komputer berdasarkan pada kerangka jaringan area lokal (LAN). Sistem komunikasi melalui *Ethernet* membagi aliran data ke dalam paket individual yang disebut *frame*. Setiap *frame*, berisi alamat sumber dan tujuan serta pengecekan error data sehingga data yang rusak dapat dideteksi dan dikirim kembali. *Ethernet* adalah protokol LAN yang memungkinkan setiap PC berlomba untuk mengakses *network*.

*Ethernet Shield* adalah modul yang berfungsi untuk menghubungkan *board* arduino dengan jaringan internet. Untuk menghubungkan dan menggunakan modul hingga dapat terkoneksi internet caranya adalah dengan meletakkannya diatas *board* arduino dan menyambungkan dengan kabel *network* RJ45. Pada *ethernet shield* terdapat slot mikro SD yang berfungsi sebagai tempat penyimpanan file. Sedangkan untuk mengakses mikro SD card menggunakan *library\_SD*, untuk jenis *board* arduino yang bisa di pasang dengan *ethernet shield* adalah arduino uno dan mega. Pada gambar 2.19 adalah bentuk fisik dari *ethernet shield* dan kabel *network* RJ45.



**Gambar 2.19** Ethernet Shield dan Kabel Network RJ-45

## **2.7 Bluetooth**

*Bluetooth* sebuah teknologi komunikasi *wireless* (tanpa kabel) yang beroperasi dalam pita frekuensi 2,4 GHz *unlicensed ISM (Industrial, Scientific and Medical)* dengan menggunakan sebuah *frequency hopping tranceiver* yang mampu menyediakan layanan komunikasi data dan suara secara *real-time* antara *hostbluetooth* dengan jarak jangkauan layanan yang terbatas. *Bluetooth* sendiri dapat berupa *card* yang bentuk dan fungsinya hampir sama dengan *card* yang digunakan untuk *wireless local area network (WLAN)* yang menggunakan frekuensi radio standar IEEE 802.11, hanya saja pada *bluetooth* mempunyai jangkauan jarak layanan yang lebih pendek dan kemampuan transfer data yang lebih rendah. Dalam Tugas akhir ini untuk komunikasi *Bluetooth* kami menggunakan *Bluetooth HC-06*.

### **2.7.1 Bluetooth HC-06**

Modul *Bluetooth HC-06* adalah sebuah modul *Bluetooth SPP (Serial Port Protocol)* yang mudah digunakan untuk komunikasi serial *wireless* (nirkabel) yang mengkonversi port serial ke *Bluetooth*. *HC-06* menggunakan modulasi *bluetooth V2.0 + EDR (Enhanced Data Rate)* 3 Mbps dengan memanfaatkan gelombang radio berfrekuensi 2,4 GHz. Modul ini dapat digunakan sebagai *slave* maupun *master*. *HC-06* memiliki 2 mode konfigurasi, yaitu *AT mode* dan *Communication mode*. *AT mode* berfungsi untuk melakukan pengaturan konfigurasi dari *HC-06*. Sedangkan *Communication mode* berfungsi untuk melakukan komunikasi *bluetooth* dengan piranti lain. Dalam penggunaannya, *HC-06* dapat beroperasi tanpa menggunakan driver khusus. Untuk

berkomunikasi antar *Bluetooth*, minimal harus memenuhi dua kondisi berikut :

1. Komunikasi harus antara master dan slave.
2. Password harus benar (saat melakukan pairing).

Jarak sinyal dari HC-06 adalah  $\pm 10$  meter, dengan kondisi tanpa halangan. Adapun spesifikasi dari HC-06 adalah :

1. *Bluetooth protocol: Bluetooth Specification v2.0+EDR*
2. *Frequency: 2.4GHz ISM band*
3. *Modulation: GFSK(Gaussian Frequency Shift Keying)*
4. *Emission power: 4dBm, Class 2*
5. *Sensitivity: -84dBm at 0.1% BER*
6. *Speed: Asynchronous: 2.1Mbps(Max) / 160 kbps*
7. *Synchronous: 1Mbps/1Mbps*
8. *Security: Authentication and encryption*
9. *Profiles: Bluetooth serial port*
10. *Power supply: +3.3VDC 50mA*
11. *Working temperature: -20 ~ +75 Centigrade*
12. *Dimension: 3.57cm x 1.52cm*

Pada gambar 2.20 merupakan bentuk fisik modul *Bluetooth* HC-06.



**Gambar 2.20** *Bluetooth* HC-06

## **2.8 Router**

*Router* adalah perangkat jaringan yang berfungsi untuk menghubungkan beberapa jaringan atau *network*, baik jaringan yang menggunakan teknologi sama atau yang berbeda, misalnya menghubungkan jaringan topologi Bus, topologi Star atau topologi Ring. Karena *router* ini menghubungkan beberapa jaringan tentunya

**router** berbeda dengan *Switch*. *Switch* hanya perangkat yang digunakan untuk menghubungkan beberapa komputer sehingga membentuk LAN atau *local area network*. Sedangkan *router* adalah perangkat yang menghubungkan satu LAN dengan banyak LAN lainnya.

**Router** dapat digunakan untuk menghubungkan banyak jaringan kecil ke sebuah jaringan yang lebih besaryang disebut dengan *internetwork*, atau untuk membagi sebuah jaringan besar ke dalam beberapa *subnetwork* untuk meningkatkan kinerja dan juga mempermudah manajemennya. Router juga kadang digunakan untuk mengoneksikan dua buah jaringan yang menggunakan media yang berbeda atau berbeda arsitektur jaringan, seperti halnya dari *Ethernet* ke Token Ring. Pada gambar 2.21 adalah bentuk fisik dari *router* yang digunakan dalam tugas akhir ini.



**Gambar 2.21** Router

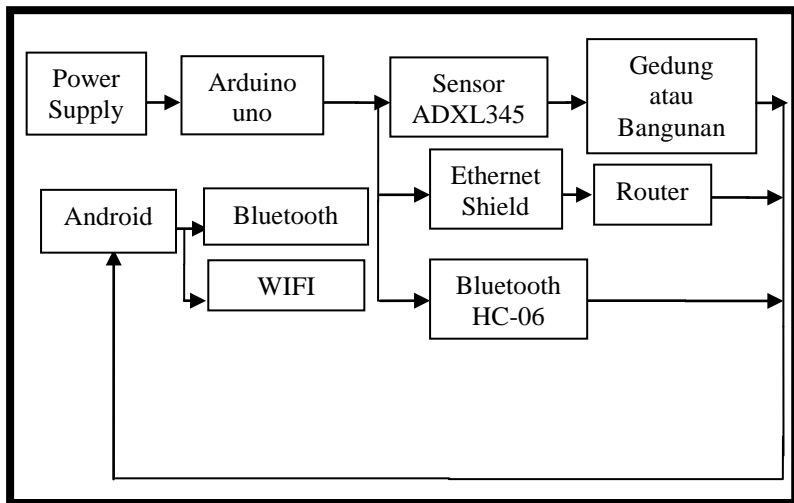
-----Halaman ini sengaja dikosongkan-----

## BAB III PERANCANGAN DAN PEMBUATAN ALAT

Melalui penjelasan Teori Penunjang pada bab sebelumnya, berikutnya dilakukan langkah dalam perancangan dan pembuatan alat yang dibutuhkan pada Tugas Akhir ini.

Perancangan dan pembuatan alat ini dibagi menjadi empat bagian, yaitu: Perancangan sistem secara keseluruhan, perancangan mekanik, perancangan *hardware* yang meliputi perancangan rangkaian elektronika dan perancangan *software* yang meliputi perancangan program dan *interface* yang digunakan.

### 3.1 Perancangan Sistem Keseluruhan



**Gambar 3.1** Diagram Fungsional Sistem Alat Secara Keseluruhan

Dari keseluruhan sistem pada Gambar 3.1 dapat dilihat bahwa ketika ADXL345 mendeteksi bahwa sebuah bangunan mengalami kemiringan, maka secara otomatis ADXL345 mentransmisikan data ke mikrokontroler arduino uno. Data yang diterima oleh mikrokontroler arduino uno akan diolah, data akan dikirim melalui *bluetooth* dan *WiFi*.

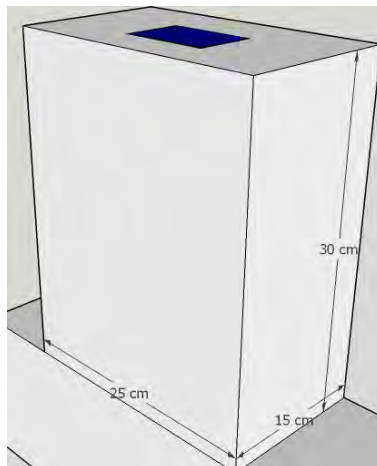
Data yang diterima oleh *bluetooth* atau *WiFi* akan ditampilkan pada LCD dan *Smartphone* melalui aplikasi dari android.

Sensor dari alat ini diletakkan pada bagian paling atas dari bangunan atau gedung karena suatu bangunan atau gedung dapat dilihat kemiringannya dari bagian paling atas, agar dapat mendeteksi kemiringan dengan data yang valid.

Pada Gambar 3.1 dapat dilihat bahwa alat ini memonitor kemiringan sebuah bangunan, dimana sumber energinya didapatkan dari *power supply* yang menyuplai mikrokontroler arduino uno. Kemudian dari mikrokontroler disambungkan ke sensor ADXL345, *ethernet shield*, dan *bluetooth* HC-06. Dari *ethernet shield* disambungkan ke *router* agar pada android selain bisa mengakses melalui *bluetooth*, tetapi juga dapat menggunakan *wifi*

### 3.2 Perancangan Mekanik

Perancangan mekanik berupa miniatur sebuah bangunan yang dibuat dalam ukuran sedang. Pemodelan miniatur ini disesuaikan dengan bentuk umum dari sebuah bangunan. Sensor dari alat tugas akhir ini yaitu sensor *accelerometer* ADXL345 diletakkan dibagian atas dari miniatur, atau terletak pada bagian atap atau *rooftop* dari bangunan. Pada gambar 3.2 dapat dilihat bentuk dari miniatur dari bangunan dan penempatan dari sensor.

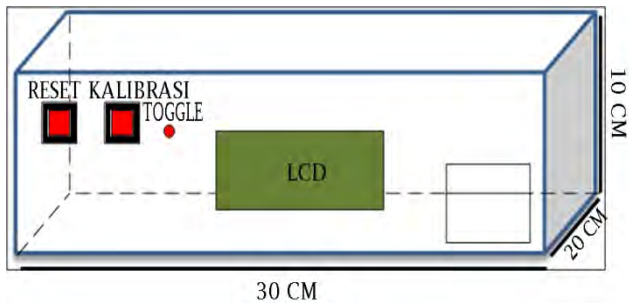


**Gambar 3.2** Miniatur Bangunan

Keterangan warna :

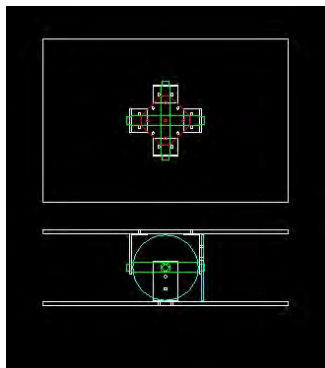
Biru : tempat sensor *accelerometer* ADXL345

Pada gambar 3.3 dibawah ini adalah perancangan mekanik sebuah kotak yang terbuat dari akrilik. Fungsinya sebagai tempat penyimpanan alat elektronik dari peralatan tugas akhir. Diantaranya mikrokontroler arduino, LCD, *power supply*, *ethernet shield*, dan *bluetooth*.



**Gambar 3.3** Box Peralatan Elektronik

Gambar 3.4 dan gambar 3.5 adalah desain *prototype* untuk indikator kemiringan dan *prototype* indikator kemiringan dari suatu bangunan. *Prototype* ini bisa dimiringkan ke kanan, kiri, depan dan belakang hingga dapat menyerupai dari kemiringan sebuah bangunan yang asli.



**Gambar 3.4** Desain *Prototype* Indikator Kemiringan





**Gambar 3.5** Bentuk *Prototype* Indikator Kemiringan

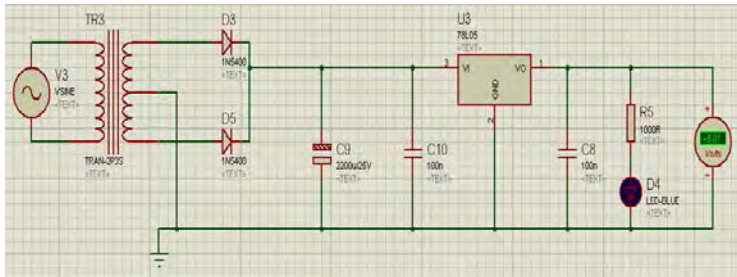
### **3.3 Perancangan *Hardware***

Perancangan *Hardware* yang dibahas terdiri dari perancangan rangkaian *power supply*, perancangan skematik mikrokontroler arduino, perancangan rangkaian *ethernet shield*, perancangan rangkaian sensor *accelerometer* ADXL345, perancangan rangkaian LCD, dan *wiring bluetooth*.

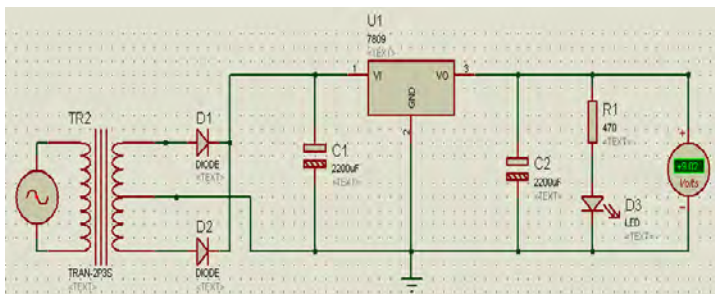
#### **3.3.1 Perancangan Rangkaian *Power Supply***

*Power supply* yang dirancang dalam tugas akhir ini menggunakan 2 *output* tegangan dengan 2 terminal. *Output* tegangan 12 Volt dari Trafo CT 1A diregulasi menggunakan *IC Regulator* 7809 dan 7805 menjadi tegangan 9 Volt dan 5 Volt. Tegangan 9 Volt di gunakan untuk mengaktifkan Arduino Uno R3, dan *Ethernet Shield* sedangkan tegangan 5 Volt digunakan untuk mengaktifkan Sensor *Accelerometer* ADXL345, dan *Bluetooth*.

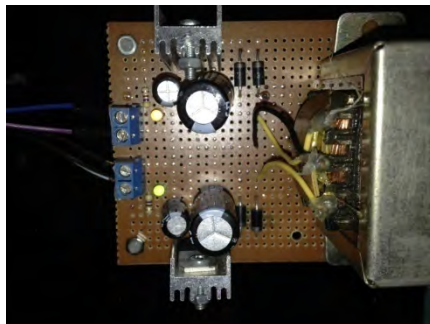
Pada gambar 3.6 dijelaskan perancangan *power supply* tegangan 5 Volt disimulasikan menggunakan proteus. Gambar 3.7 menjelaskan rangkaian *power supply* tegangan 9 Volt yang disimulasikan melalui proteus. Pada gambar 3.8 adalah rangkain *power supply* yang telah dirangkai pada *Dote PCB*.



**Gambar 3.6** Simulasi Rangkaian *Power Supply* 5V di Proteus



**Gambar 3.7** Simulasi Rangkaian *Power Supply* 9V di Proteus

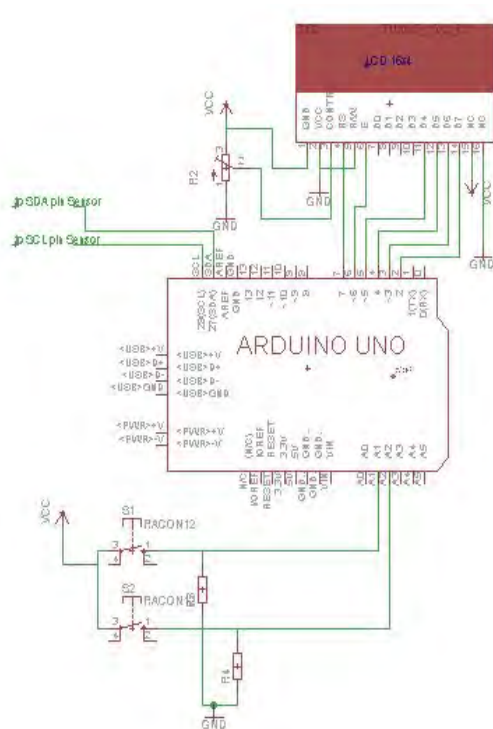


**Gambar 3.8** Rangkaian *Power Supply* pada Dote PCB

### 3.3.2 Perancangan Skematik Mikrokontroler Arduino

Rangkaian mikrokontroler arduino merupakan rangkaian mikrokontroler yang dibuat di italia dan telah didesain siap pakai oleh pembuatnya. Hal ini bertujuan agar dapat langsung digunakan

oleh penggunaanya dengan mudah. Rangkaian mikrokontroler arduino pada tugas akhir ini digunakan untuk menerima data hasil pembacaan sensor, menyalakan aktuator secara otomatis, dan mengirim seluruh data ke internet. Pin-pin mikrokontroler arduino unoyang digunakan dalam tugas akhir ini dapat dilihat pada gambar 3.9.

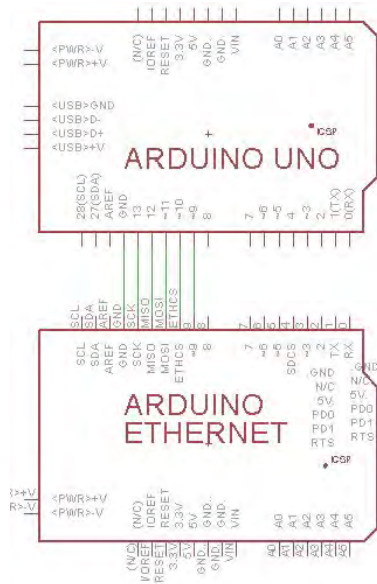


Gambar 3.9 Skematik Mikrokontroler Arduino Uno

### 3.3.3 Perancangan Rangkaian *Ethernet Shield*

Sama halnya dengan rangkaian mikrokontroler arduino. Rangkaian *ethernet shield* merupakan rangkaian buatan italia dan siap pakai. Pada tugas akhir ini rangkaian *ethernet shield* digunakan dengan tujuan agar mikrokontroler dapat terhubung

dengan internet dan berperan sebagai pemancar sinyal *wifi* dengan dihubungkan ke *router*. Cara menggunakannya adalah dengan menancapkan langsung pada mikrokontroler arduino sesuai dengan pin-pinnya. *Ethernet shield* sendiri menggunkan pin 10, 11, 12, dan 13 pada mikrokontroler. Rangkaian *ethernet shield* dapat dilihat pada Gambar 3.10.

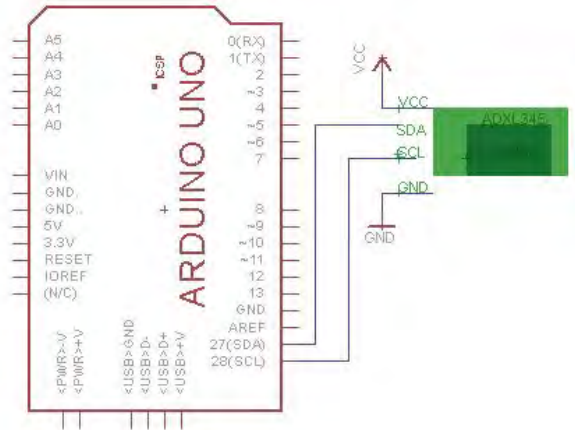


**Gambar 3.10**Rangkaian *Ethernet Shield*

### 3.3.4 Perancangan Rangkaian Sensor *Accelerometer* ADXL345

Pada perancangan rangkaian sensor *accelerometer*, sensor *accelerometer* yang dipakai adalah ADXL345. Karena pada tugas akhir ini menggunakan mikrokontroler arduino uno, maka tidak diperlukan rangkaian tambahan karena pada mikrokontroler arduino uno telah tersedia pin-pin yang digunakan untuk sensor ini. Pin-pin yang digunakan mikrokontroler arduino untuk rangkaian sensor ini adalah pin VCC 5volt, *Ground*, Analog 4 dan Analog 5. Dimana pin analog 4 disambungkan ke pin SDA dari sensor *accelerometer* ADXL345. Pin analog 5 dihubungkan ke pin SCL dari sensor. Untuk pin VCC 5 volt dihubungkan ke pin sensor 5

volt. Pin *ground* dihubungkan ke pin *ground* dari sensor ini. Pada gambar 3.11 dapat dilihat konfigurasi pin dari mikrokontroler arduino ke sensor *accelerometer* ADXL345.

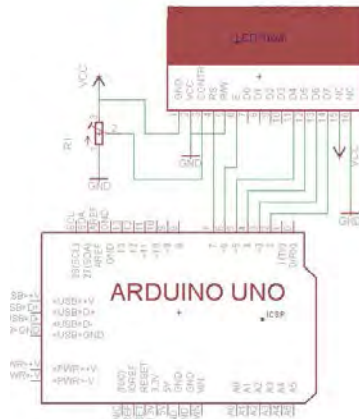


**Gambar 3.11**Rangkaian Sensor *Accelerometer* ADXL345

### 3.3.5 Perancangan Rangkaian LCD

Pada tugas akhir ini digunakan LCD 16x4, fungsi LCD pada tugas akhir ini adalah untuk menampilkan data secara serial. Karena pada tugas akhir ini menggunakan mikrokontroler arduino maka *wiring*-nya disesuaikan dengan pin-pin dari arduino. pada gambar 3.12 dapat dilihat *wiring* dari LCD 16x4 dengan mikrokontroler arduino yang digunakan pada tugas akhir ini.

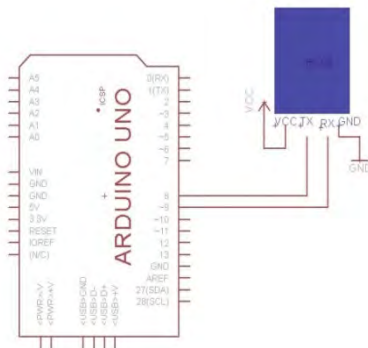
Rangkaian LCD 16x4 merupakan rangkaian yang berfungsi untuk menampilkan beberapa karakter. Pada tugas akhir ini rangkaian ini digunakan untuk menampilkan data kemiringan dari bangunan yang terukur oleh sensor. Rangkaian ini dipasang di lokal area tepatnya di bagian atap dari bangunan. Dengan tujuan mempermudah pengguna dalam membaca informasi dari kemiringan suatu bangunan. Karena pada tugas akhir ini menggunakan mikrokontroler arduino maka *wiring*-nya disesuaikan dengan pin-pin dari arduino. Rangkaian LCD ini menggunakan pin 2,3,4,5 6, dan 7 pada mikrokontroler arduino. Rangkaian LCD dapat dilihat pada Gambar 3.12.



**Gambar 3.12**Rangkaian LCD

### 3.3.6 Wiring Bluetooth

*Wiring bluetooth* pada tugas akhir ini adalah *wiringbluetooth* dengan mikrokontroler arduino uno yang mana *bluetooth* ini nantinya digunakan sebagai komunikasi *nirkabel* dengan android. Pada tugas akhir ini digunakan *bluetooth* HC-06 yang dihubungkan dengan pin-pin dari mikrokontroler arduino uno. Dalam gambar 3.13 dapat dilihat skematik *wiring* dari *bluetooth* HC-06 dengan mikrokontroler arduino uno.



**Gambar 3.13** Skematik *Wiring Bluetooth*

### 3.4 Perancangan Software

Perancangan perangkat lunak (*software*) pada Tugas Akhir ini terdiri dari pemrograman mikrokontroler arduino uno yang meliputi pemrograman sensor *accelerometer* ADXL345, pemrograman *bluetooth*, pemrograman untuk *WiFi* dan pemrograman tampilan LCD, serta pemrograman untuk aplikasi pada android,.

#### 3.4.1 Pemrograman Mikrokontroler Arduino Uno

Pemrograman Arduino dirancang dengan menggunakan *sketch* dengan bahasa Arduino yang secara sistematis sama dengan bahasa C dan C++

Generasi awal, namun tidak semua pustaka di C dan C++ dapat digunakan untuk menulis *sketch*. Bahasa pemrograman ini mudah digunakan karena dapat digunakan untuk pemula. Dalam perancangan program Arduino diperlukan program pemilahan Analog dan Digital sebelum membuat program utama. Hal ini digunakan untuk mempermudah dalam membuat program sesuai dengan yang akan dibuat.

Pemrograman mikrokontroler ini meliputi pemrograman untuk sensor *accelerometer* ADXL345, *bluetooth*, *wifi*, dan tampilan dari LCD. Berikut ini adalah beberapa gambar pemrograman dari mikrokontroler arduino uno.



```
File Edit Sketch Tools Help
arduino
#include <SPI.h>
#include <Ethernet.h>
#include <math.h>
#include <EEPROM.h>

#include <SoftwareSerial.h> // import the serial library

#include <Wire.h>
#include <Adafruit_Sensor.h>
#include <Adafruit_ADXL345_U.h>
#include <LiquidCrystal.h>
Adafruit_Crystal lcd(7, 6, 5, 4, 3, 2);

float datax, datay, dataz, sudut1, sudut2;
Adafruit_ADXL345_Unified accel = Adafruit_ADXL345_Unified(12345);

int buttonPin=1, buttonPin1=42, buttonPin2=43;
int buttonState=0, buttonState1=0, buttonState2=0;
int address = 0;
int busui1 = 0;
int busui2 = 0;

SoftwareSerial blue(5, 8); // RX, TX

byte mac[] = {
  0x00, 0x0F, 0x9C, 0x0E, 0x4C, 0x0C
};
IPAddress ip(192, 168, 1, 5);
```

Gambar 3.14 Inisialisasi Program Mikrokontroler Arduino

Pada gambar 3.14 dijelaskan bahwa gambar itu adalah program untuk inialisasi dari setiap *part* yang terhubung ke arduino agar saat pemrograman *part-part* tersebut tidak salah alamat dan sesuai dengan yang dikehendaki oleh sistem kerja dari alat ini.

```
}  
  
sensors_event_t event;  
accel.getEvent(&event);  
accel.setRange(ADXL345_RANGE_8_G);  
datax=(event.acceleration.x); //fungsi x  
datay=(event.acceleration.y); //fungsi y  
dataz=(event.acceleration.z); //fungsi z  
sudut1 = int((((asin/acos(datax/sqrt(datay^2+dataz^2))*100)-0.56)-kalibrasi1; //belum diambil datanya  
sudut2 = int((((asin/acos(datay/sqrt(datax^2+dataz^2))*100)-0.56)-kalibrasi2; //belum diambil datanya  
  
dtostrf(datax,4,2,float_datax);  
dtostrf(datay,4,2,float_datay);  
dtostrf(dataz,4,2,float_dataz);  
dtostrf(sudut1,4,2,float_sudut1);  
dtostrf(sudut2,4,2,float_sudut2);  
sprintf(buff,"%s %s %s %s %s",float_datax, float_datay, float_dataz, float_sudut1, float_sudut2);
```

**Gambar 3.15** Program untuk Mencari Besar Sudut

Pada gambar 3.15 ini adalah program untuk mencari besaran sudut yang dideteksi melalui sensor *accelerometer* ADXL345. Fungsinya mempermudah bagi pengguna alat untuk mengetahui besaran sudut kemiringan dari suatu bangunan. Pada program ini dituliskan rumus untuk menghitung besaran sudut, karena data yang diberikan oleh sensor masih berupa percepatan saja.



```

//bluetooth
blue.print(sudut1);
delay(1000);
blue.println(sudut2);
lcd.setCursor(0, 0); lcd.print("Sensor ADXL345");
//lcd.setCursor(0, 1); lcd.print("X: ");lcd.print(datax);//lcd.print("m/s^2");
//lcd.setCursor(0, 2); lcd.print("Y: ");lcd.print(datay);//lcd.print("m/s^2");
//lcd.setCursor(0, 3); lcd.print("Z: ");lcd.print(dataz);//lcd.print("m/s^2");
lcd.setCursor(0, 1); lcd.print("Sudut 1 : ");lcd.print(sudut1);
lcd.setCursor(0, 2); lcd.print("Sudut 2 : ");lcd.print(sudut2);

Serial.print(sudut1);Serial.print(" ");Serial.println(sudut2);

delay(1000);

```

**Gambar 3.16** Program untuk Komunikasi *Bluetooth*

Gambar 3.16 menjelaskan mengenai program agar alat terkoneksi menggunakan *bluetooth* dan dapat diakses menggunakan android dengan mengkoneksikan dengan *bluetooth*.

```

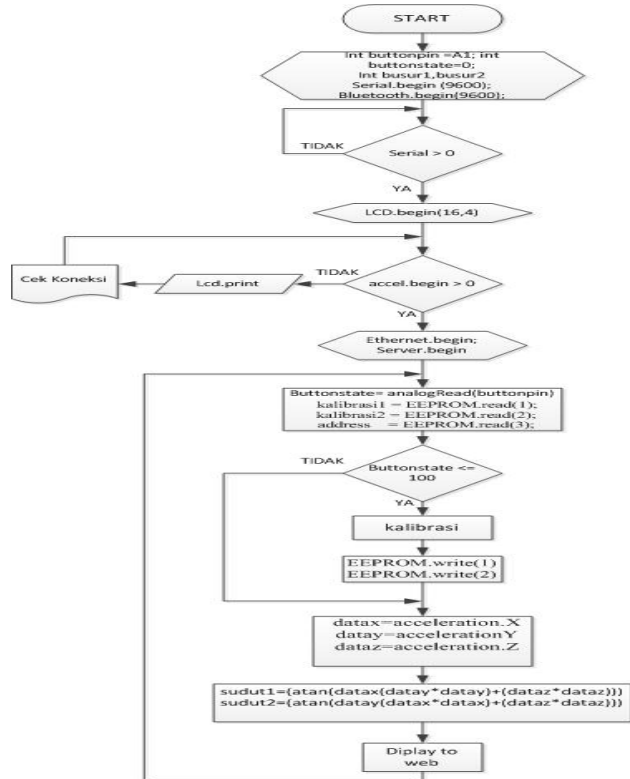
EthernetClient client = server.available();
if (client) {
  Serial.println("new client");
  // an http request ends with a blank line
  boolean currentLineIsBlank = true;
  while (client.connected()) {
    if (client.available()) {
      char c = client.read();
      Serial.write(c);
      if (c == '\n' && currentLineIsBlank) {
        // send a standard http response header
        client.println("HTTP/1.1 200 OK");
        client.println("Content-Type: text/html");
        client.println("Connection: close"); // the connection will be closed after completion of the response
        client.println("Refresh: 3");
        client.println();
        client.println("<DOCTYPE HTML>");
        client.println("<html>");
        client.print("sudut 1 = ");client.print(sudut1);client.println("<br />");
        client.print("sudut 2 = ");client.print(sudut2);client.println("<br />");
        //client.print("data z = ");client.print(dataz);client.println("<br />");
        client.println("</html>");
        break;
      }
      if (c == '\n') {
        currentLineIsBlank = true;
      }
    } else if (c != '\r') {

```

**Gambar 3.17**Program untuk Komunikasi *WiFi*

Pada gambar 3.17 adalah gambar yang menunjukkan program untuk *ethernet shield*. Pada sistem kerja alat ini, *ethernet shield* dihubungkan ke *router* menggunakan kabel RJ-45 agar dapat memancarkan sinyal *wifi*. Kemudian pengguna dapat mengakses

dengan mengkonfigurasi *IP address* dengan aplikasi dari android.



Gambar 3.18 Flow Chart Program Arduino

### 3.4.2 Pemrograman Aplikasi Android

Pemrograman aplikasi android dilakukan sebagai perancangan *interface* sensor *accelerometer* ADXL345 dan *smartphone* android yang diatur oleh mikrokontroler arduino uno. Tujuannya adalah memonitor kemiringan dari sebuah bangunan agar pengguna tidak perlu menggunakan PC saat ingin melihat keadaan dari suatu bangunan. Aplikasi ini kami buat dengan menggunakan *software basic4android*. Pada saat digunakan,

aplikasi ini dapat dihubungkan dengan *bluetooth* atau *wifi* yang sudah diprogram pada mikrokontroler arduino uno. Tampilan aplikasi ini dibuat menarik agar pengguna tidak bosan dan dapat membaca informasi yang disediakan dengan mudah. Berikut adalah beberapa desain yang dibuat dalam aplikasi android ini.



**Gambar 3.19** Tampilan Layar *Login*

Pada gambar 3.19 adalah tampilan *login* aplikasi android, dimana pengguna harus memasukkan identitas yang sesuai dengan apa yang sudah dimasukkan saat registrasi awal.



**Gambar 3.20** Pengisian Identitas saat *Login*

Gambar 3.20 menunjukkan pengisian identitas pengguna pada saat *login*. Data yang diisi harus benar agar dapat mengakses aplikasi untuk memonitor kemiringan suatu bangunan.



**Gambar 3.21** Tampilan Setelah *Login*

Dalam gambar 3.21 menunjukkan tampilan dari aplikasi setelah *login*. Terdapat tampilan simbol *bluetooth* dan *wifi*, disini pengguna diharuskan memilih salah satu dari mode komunikasi untuk mengakses data kemiringan dari suatu bangunan.



**Gambar 3.22** Pemilihan *Bluetooth* untuk Media Komunikasi

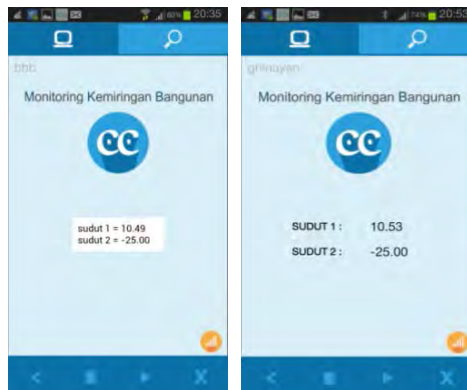
Pada gambar 3.22 menunjukkan bahwa pengguna menggunakan media komunikasi *bluetooth* untuk mengakses data kemiringan dari suatu bangunan. Sebelum menggunakan *bluetooth* untuk mengakses, terlebih dahulu pengguna harus

melakukan *pairing bluetooth* antara *bluetooth* pada *smartphone* android dengan *bluetooth* yang terhubung dengan sensor



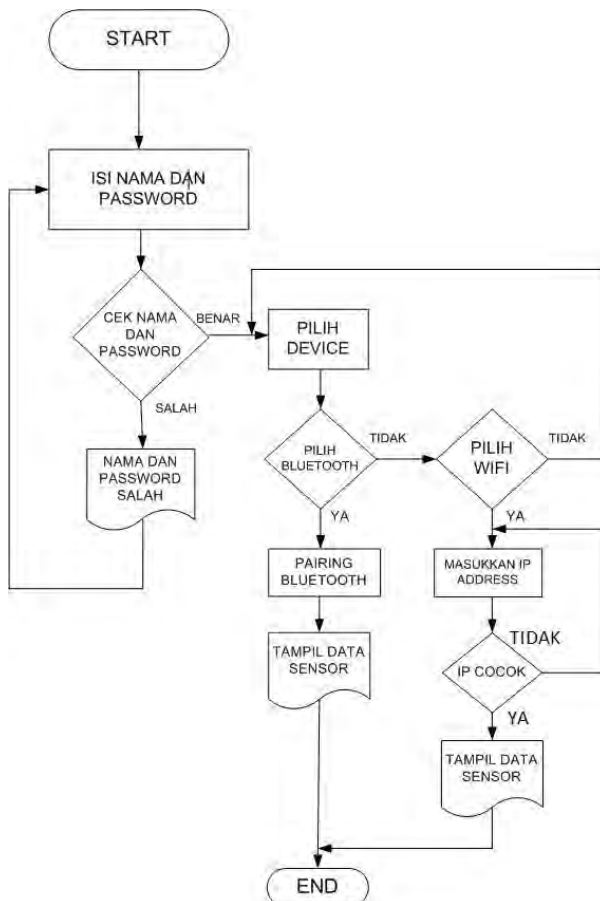
**Gambar 3.23** Pemilihan *WiFi* untuk Media Komunikasi

Pada gambar 3.23 diatas dapat dilihat bahwa pengguna dari aplikasi ini memilih *wifi* sebagai media komunikasi. Agar dapat mengakses data maka pengguna harus memasukkan *IP Address* yang sesuai dengan *IP Address* pada *router* yang terhubung dengan rangkaian pada alat monitoring.



**Gambar 3.24**Tampilan Monitoring Kemiringan Bangunan

Gambar 3.24 menunjukkan tampilan dari monitoring kemiringan bangunan baik yang menggunakan media komunikasi *Bluetooth* atau *WiFi*. Data yang ditampilkan nantinya sama dengan data yang sudah diprogram pada mikrokontroler arduino uno. Tampilan-tampilan data ini memudahkan pengguna dalam membaca data dari kemiringan suatu bangunan. Pada gambar 3.25 adalah *flow chart* dari sistem aplikasi dari android untuk memonitor kemiringan dari bangunan.



**Gambar 3.25** *Flow Chart* Sistem Aplikasi Android

-----Halaman ini sengaja dikosongkan-----

## BAB IV PENGUJIAN DAN ANALISA DATA

Dalam membuat suatu sistem, pengujian dan analisa sangat diperlukan. Pengujian dan analisa bertujuan untuk mengetahui sistem bekerja sesuai rencana atau belum. Dari hasil pengujian dan analisa dapat diketahui kelemahan-kelemahan dari sistem, sehingga dilakukan perbaikan, pengembangan, dan penyempurnaan sistem.

Bab ini membahas tentang pengujian dan analisa *hardware* maupun *software* serta sistem secara keseluruhan.

Dalam pembuatan rangkaian elektronika, sebelum dilakukan pengujian rangkaian secara keseluruhan terlebih dahulu dilakukan pengujian tiap-tiap rangkaian.

### 4.1 Pengujian Rangkaian *Board* Arduino

Pengujian rangkaian board Arduino Uno dilakukan dengan memberikan tegangan +9 Volt pada *jack* listrik yang kemudian diregulasi menjadi +5 Volt oleh regulator yang ada pada board Arduino Uno. Selanjutnya adalah membuat program dan mengunggahnya keboard Arduino Uno untuk diukur tegangan pada tiap *pin* menggunakan *voltmeter*.

Pengujian dilakukan dengan mengukur tegangan pada tiap *pin* board Arduino Uno dengan dua kondisi, yaitu kondisi logika 1 dan logika 0. Program untuk kondisi logika 1 dan logika 0 yang diberikan pada tiap *pin* mikrokontroler dapat dilihat pada Gambar 4.1 dan Gambar 4.2.

```
int PinA0 = A0; int PinD0 = 0; int PinD7 = 7 ;  
int PinA1 = A1; int PinD1 = 1; int PinD8 = 8 ;  
int PinA2 = A2; int PinD2 = 2; int PinD9 = 9 ;  
int PinA3 = A3; int PinD3 = 3; int PinD10 = 10;  
int PinA4 = A4; int PinD4 = 4; int PinD11 = 11;  
int PinA5 = A5; int PinD5 = 5; int PinD12 = 12;  
int PinD6 = 6; int PinD13 = 13;
```

```
void setup()  
{  
  pinMode(A0,OUTPUT); pinMode(0,OUTPUT); pinMode(7,OUTPUT) ;  
  pinMode(A1,OUTPUT); pinMode(1,OUTPUT); pinMode(8,OUTPUT) ;  
  pinMode(A2,OUTPUT); pinMode(2,OUTPUT); pinMode(9,OUTPUT) ;
```



```

pinMode(A3,OUTPUT); pinMode(3,OUTPUT); pinMode(10,OUTPUT);
pinMode(A4,OUTPUT); pinMode(4,OUTPUT); pinMode(11,OUTPUT);
pinMode(A5,OUTPUT); pinMode(5,OUTPUT); pinMode(12,OUTPUT);
pinMode(6,OUTPUT); pinMode(13,OUTPUT);
}

void loop()
{
digitalWrite(A0,HIGH); digitalWrite(0,HIGH); digitalWrite(7,HIGH) ;
digitalWrite(A1,HIGH); digitalWrite(1,HIGH); digitalWrite(8,HIGH) ;
digitalWrite(A2,HIGH); digitalWrite(2,HIGH); digitalWrite(9,HIGH) ;
digitalWrite(A3,HIGH); digitalWrite(3,HIGH); digitalWrite(10,HIGH);
digitalWrite(A4,HIGH); digitalWrite(4,HIGH); digitalWrite(11,HIGH);
digitalWrite(A5,HIGH); digitalWrite(5,HIGH); digitalWrite(12,HIGH);
digitalWrite(6,HIGH); digitalWrite(13,HIGH);
}

```

**Gambar 4.1** Program Kondisi Logika 1

```

int PinA0 = A0; int PinD0 = 0; int PinD7 = 7 ;
int PinA1 = A1; int PinD1 = 1; int PinD8 = 8 ;
int PinA2 = A2; int PinD2 = 2; int PinD9 = 9 ;
int PinA3 = A3; int PinD3 = 3; int PinD10 = 10;
int PinA4 = A4; int PinD4 = 4; int PinD11 = 11;
int PinA5 = A5; int PinD5 = 5; int PinD12 = 12;
int PinD6 = 6; int PinD13 = 13;

void setup()
{
pinMode(A0,OUTPUT); pinMode(0,OUTPUT); pinMode(7,OUTPUT) ;
pinMode(A1,OUTPUT); pinMode(1,OUTPUT); pinMode(8,OUTPUT) ;
pinMode(A2,OUTPUT); pinMode(2,OUTPUT); pinMode(9,OUTPUT) ;
pinMode(A3,OUTPUT); pinMode(3,OUTPUT); pinMode(10,OUTPUT);
pinMode(A4,OUTPUT); pinMode(4,OUTPUT); pinMode(11,OUTPUT);
pinMode(A5,OUTPUT); pinMode(5,OUTPUT); pinMode(12,OUTPUT);
pinMode(6,OUTPUT); pinMode(13,OUTPUT);
}

void loop(){
digitalWrite(A0,LOW); digitalWrite(0,LOW); digitalWrite(7,LOW) ;
digitalWrite(A1,LOW); digitalWrite(1,LOW); digitalWrite(8,LOW) ;
digitalWrite(A2,LOW); digitalWrite(2,LOW); digitalWrite(9,LOW) ;
digitalWrite(A3,LOW); digitalWrite(3,LOW); digitalWrite(10,LOW);
digitalWrite(A4,LOW); digitalWrite(4,LOW); digitalWrite(11,LOW);
}

```

```
digitalWrite(A5,LOW); digitalWrite(5,LOW); digitalWrite(12,LOW);
digitalWrite(6,LOW); digitalWrite(13,LOW);
}
```

**Gambar 4.2 Program Kondisi Logika 0**

Hasil pengukuran pada tiap pin dari *board* Mikrokontroler Arduino Uno ini dapat dilihat pada Tabel 4.1 dan 4.2

**Tabel 4.1** Hasil Pengukuran per Pin saat *Active High*

<i>Pin Analog</i>	Tegangan (Volt)	<b>Pin Digital</b>	Tegangan (Volt)	<b>Pin Digital</b>	Tegangan (Volt)
<i>PinA0</i>	4,8	<b>Pin 0</b>	4,8	<b>Pin 7</b>	4,8
<i>PinA1</i>	4,8	<b>Pin 1</b>	4,8	<b>Pin 8</b>	4,8
<i>PinA2</i>	4,8	<b>Pin 2</b>	4,8	<b>Pin 9</b>	4,8
<i>PinA3</i>	4,78	<b>Pin 3</b>	4,8	<b>Pin 10</b>	4,78
<i>PinA4</i>	4,78	<b>Pin 4</b>	4,8	<b>Pin 11</b>	4,78
<i>PinA5</i>	4,78	<b>Pin 5</b>	4,8	<b>Pin 12</b>	4,78
		<b>Pin 6</b>	4,8	<b>Pin 13</b>	4,78

**Tabel 4.2** Hasil Pengukuran per Pin saat *Active Low*

<i>Pin Analog</i>	Tegangan (mV)	<b>Pin Digital</b>	Tegangan (mV)	<b>Pin Digital</b>	Tegangan (mV)
<i>PinA0</i>	5	<b>Pin 0</b>	5	<b>Pin 7</b>	4
<i>PinA1</i>	5	<b>Pin 1</b>	5	<b>Pin 8</b>	4
<i>PinA2</i>	5	<b>Pin 2</b>	5	<b>Pin 9</b>	3
<i>PinA3</i>	5	<b>Pin 3</b>	5	<b>Pin 10</b>	3
<i>PinA4</i>	4	<b>Pin 4</b>	4	<b>Pin 11</b>	3
<i>PinA5</i>	4	<b>Pin 5</b>	4	<b>Pin 12</b>	3
		<b>Pin 6</b>	4	<b>Pin 13</b>	3

Dari hasil Tabel 4.1 dapat disimpulkan bahwa ketika board Arduino Uno diberi program logika 1 pada tiap *pin* maka tegangan yang dihasilkan bernilai rata-rata 4,8 Volt yang artinya tegangan yang dikeluarkan telah maksimal dan *pin* tersebut dapat berfungsi sebagai *supply* untuk beban yang diinginkan karena *output* board Arduino Uno bernilai *high* dengan *range* tegangan 2,4 – 5,5 V.

Dari hasil Tabel 4.2 dapat disimpulkan bahwa ketika board Arduino Uno diberi program logika 0 pada tiap *pin* maka tegangan yang dihasilkan bernilai rata-rata 4 mV yang artinya tegangan yang dikeluarkan sangat kecil dan *pin* tersebut dapat berfungsi sebagai *ground* karena *output* board Arduino bernilai *low* dengan *range* tegangan 0 – 800 mV.

#### 4.2 Pengujian Rangkaian Power Supply

Pada perancangan tugas akhir kali ini rangkaian *power supply* digunakan sebagai sumber tegangan yang akan diberikan pada *mikrokontroller* arduino uno R3. Rangkaian *power supply* berikut bertugas untuk memberi tegangan masukan sebesar 9V dan 5V. Jadi, pada kesimpulannya rangkaian *power supply* berikut mempunyai tegangan masukan sebesar 9V dan 5V. Yang akan diuji pada rangkaian *power supply* berikut adalah besar tegangan masukan pada *power supply* dan tegangan keluaran pada *power supply*. Serta dihitung persentase kesalahan pada rangkaian *power supply*.

Berikut pada tabel 4.3 adalah tabel pengujian rangkaian *power supply* 5V pada perancangan alat :

**Tabel 4.3** Pengujian Rangkaian Power Supply 5V

PENGUKURAN KE -	TEGANGAN MASUKAN	TEGANGAN KELUARAN	
	VIN (VOLT)	VOUT TANPA BEBAN	VOUT DENGAN BEBAN
1	12V	4.9V	4.9V
2	12V	4.9V	4.9V
3	12V	4.9V	4.9V
4	12V	4.9V	4.9V
5	12V	4.9V	4.9V
6	12V	4.9V	4.9V
7	12V	4.9V	4.9V
<b>V RATA-RATA</b>	12V	4.9V	4.9V

Pengujian tanpa beban dilakukan dengan mengukur keluaran dengan *Volt* meter, kemudian diukur persentase error.

% Kesalahan untuk *Power Supply* +5Vtanpa beban :

$$= 5 - 4,9 / 5 \times 100 \%$$

$$= 2 \%$$

% Kesalahan untuk *Power Supply* +5Vdengan beban :

$$= 5 - 4,9 / 5 \times 100 \%$$

$$= 2 \%$$

Berikut pada tabel 4.4 adalah tabel pengujian rangkaian *power supply* 9V pada perancangan alat :

**Tabel 4.4** Pengujian Rangkaian *Power Supply* 9V

PENGUKURAN KE -	TEGANGAN MASUKAN	TEGANGAN KELUARAN	
	VIN (VOLT)	VOUT TANPA BEBAN	VOUT DENGAN BEBAN
1	12V	9.2V	8.3V
2	12V	9.2V	8.5V
3	12V	9.2V	8.4V
4	12V	9.2V	8.5V
5	12V	9.2V	8.2V
<b>V RATA-RATA</b>	12V	9.2V	8.38V

Pengujian tanpa beban dilakukan dengan mengukur keluaran dengan *Volt* meter, kemudian diukur persentase error.

% Kesalahan untuk *Power Supply*+9Vtanpa beban :

$$= 9 - 9.2 / 9 \times 100 \%$$

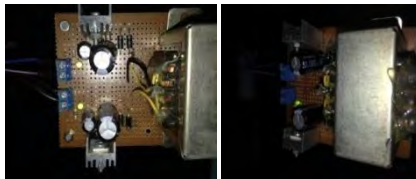
$$= 2\%$$

% Kesalahan untuk *Power Supply*+9Vdengan beban :

$$= 9 - 8.38 / 9 \times 100 \%$$

$$= 6\%$$

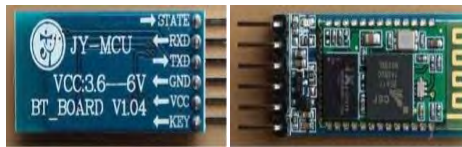
Dari beberapa kali pengambilan data, keluaran dari power supply nilainya sama sehingga dapat dikatakan bahwa rangkaian power supply ini stabil dan memiliki presentase error yang kecil. Untuk mendapatkan hasil yang sempurna sesuai tegangan masukkan dan keluaran, maka dari itu perlu diperhitungkan dengan seksama agar nanti dalam pemakaian tidak terjadi trouble atau kesalahan yang tidak diinginkan, kemungkinan akan drop tegangannya atau tidak stabil outputnya bila setelah terbebani.



**Gambar 4.3** Bentuk Fisik Rangkaian *Power Supply*

### 4.3 Pengujian Komunikasi *Bluetooth*

Bluetooth merupakan media komunikasi data yang kami gunakan untuk jarak pendek karena pada *bluetooth* HC-06 hanya memiliki jarak maksimal 10 meter antara device dan *Bluetooth* itu sendiri. Berikut adalah gambar yang akan sebagai penjelasan fungsi dari pinout Bluetooth HC-06 yang digunakan untuk menjadi penghubung atau media komunikasi antara mikrokontroler Arduino uno R3 dengan smartphone android :



**Gambar 4.4** Pin Out *Bluetooth* HC-06

RXD merupakan pin untuk menerima data dari mikrokontroler Arduino uno R3, TXD merupakan pin untuk mengirimkan data dari mikrokontroler Arduino uno R3, sedangkan GND dan VCC merupakan pin untuk sumber tegangan masukan 5V yang diberikan oleh *power supply* kepada *bluetooth*.

**Tabel 4.5** Pengujian Komunikasi *Bluetooth* HC-06

Pengujian Ke-	Jarak	Status
1.	1 meter	Terbaca
2.	2 meter	Terbaca
3..	4 meter	Terbaca
4.	6 meter	Terbaca
5.	8 meter	Terbaca
6.	10 meter	Terbaca
7.	12 meter	Tidak Terbaca
8.	> 12 meter	Tidak Terbaca

### 4.4 Pengujian Komunikasi *WiFi*

Pengujian komunikasi *WiFi* pada tugas akhir ini adalah pengujian *ethernet shield* yang telah dihubungkan dengan mikrokontroler arduino dan disambungkan dengan *router* menggunakan kabel RJ-45. Terlebih

dahulu kami melakukan pengujian terhadap router sebagai pemancar dari sinyal *WiFi*.

**Tabel 4.6** Pengujian Koneksi *Router*

<b>Pengujian Ke-</b>	<b>Jarak</b>	<b>Status</b>
1.	1 meter	Terbaca
2.	3 meter	Terbaca
3.	5 meter	Terbaca
4.	7 meter	Terbaca
5.	9 meter	Terbaca
6.	11 meter	Terbaca
7.	13 meter	Terbaca
8.	15 meter	Terbaca
9.	17 meter	Terbaca
10.	19 meter	Terbaca
11.	21 meter	Terbaca
12.	23 meter	Terbaca
13.	25 meter	Terbaca
14.	27 meter	Terbaca
15.	29 meter	Tidak Terbaca
16.	> 29 meter	Tidak Terbaca

Dari hasil pengujian didapatkan hasil bahwa *router* dapat terbaca koneksinya hingga jarak  $\pm 27$  meter. Sehingga pengguna dapat mengakses melalui komunikasi *wifi* hingga jarak tersebut. Jarak tersebut dapat bertambah lebih jauh apabila diganti dengan *router* yang dapat memancarkan sinyal lebih jauh dan kemampuan dari *smartphone* sebagai penangkap sinyal *WiFi* harus baik. Agar pengguna dapat terhubung dengan alat ini maka pengguna harus memasukkan *IP Address* yang sama dengan yang telah diatur pada *router*. Padagambar 4.5 dapat dilihat koneksi antara arduino *ethernet shield* dengan *router* dalam memancarkan sinyal *WiFi*.



**Gambar 4.5** Koneksi antara *Ethernet Shield* dengan *Router*

#### 4.5 Pengujian Sensor *Accelerometer* ADXL345

Pada tugas akhir kali ini, kami menggunakan adxl345 sensor pendeteksi kemiringan. Keluaran dari sensor adxl345 ini digital yang berupa percepatan ( $a$ ). Untuk mengubah dari percepatan menjadi sudut, kami menggunakan rumus yang sudah kami dari referensi yang kami baca. Agar mendapatkan nilai yang mudah dibaca, maka persamaan sudut ini kami bagi menjadi 2 bagian yaitu persamaan sudut 1 dan persamaan sudut 2.

Karena hasil sudut dari persamaan tersebut belum linear, sehingga kita perlu melinearkan terlebih dahulu dengan dikalikan nilai  $l$  (nilai linearisasi) yang didapat dari derajat busur ( $b$ ) dibagi dengan nilai keluaran sensor ( $s$ ). Berikut ini adalah tabel pengambilan data dari hasil pengujian sensor *accelerometer* ADXL345.

**Tabel 4.7** Tabel Pengamatan Pengujian Sensor

Data Ke -	Derajat Busur ( $b$ )	Keluaran Sensor ( $s$ )
1.	90°	47,2 $m/s^2$
2.	85°	44,5 $m/s^2$
3.	80°	43,6 $m/s^2$
4.	75°	39,6 $m/s^2$
5.	70°	38,1 $m/s^2$
6.	65°	34,7 $m/s^2$
7.	60°	31,9 $m/s^2$
8.	55°	30,5 $m/s^2$
9.	50°	27,7 $m/s^2$
10.	45°	25,3 $m/s^2$

11.	40°	22,9m/s <sup>2</sup>
12.	35°	19,6m/s <sup>2</sup>
13.	30°	18,5m/s <sup>2</sup>
14.	25°	14,8m/s <sup>2</sup>
15.	20°	13m/s <sup>2</sup>
16.	15°	9,8m/s <sup>2</sup>
17.	10°	8,5m/s <sup>2</sup>
18.	5°	5m/s <sup>2</sup>
19.	0°	3,6m/s <sup>2</sup>
Jumlah	855	478,8

Untuk mendapatkan sudut, maka program yang digunakan sebagai berikut :

1. Sudut1=int((((atan(datax/(sqrt((datay\*datay)+(dataz\*dataz))))\*100))
2. Sudut2=int((((atan(datay/(sqrt((datax\*datax)+(dataz\*dataz))))\*100))

Pada pemrograman di mikrokontroler arduino uno merupakan *float* sehingga perlu dikalikan 100 agar mendapat nilai satuan yang telah terlinearisasi dan ditampilkan pada LCD serta *smartphone* android.



**Gambar 4.6** Pengujian Sensor *Accelerometer* ADXL345

#### 4.6 Pengujian *Software*

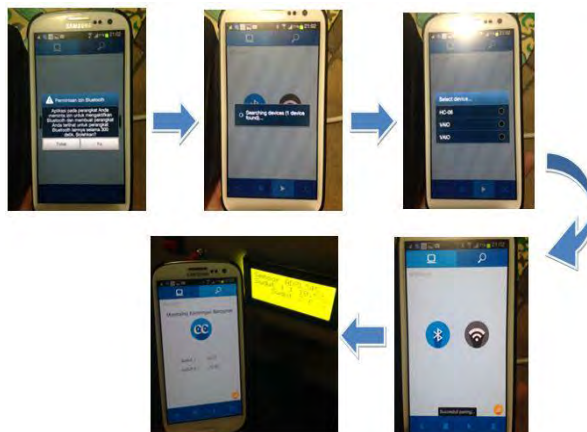
Pada pengujian *software* dilakukan pengujian terhadap *software* pada aplikasi android yang telah dibuat untuk melakukan monitoring kemiringan bangunan. Pengujian ini dilakukan untuk mengetahui apakah data yang ditampilkan oleh aplikasi android sama dengan data yang ditampilkan secara serial oleh LCD. Dari hasil pengujian didapatkan hasil sebagai berikut.





**Gambar 4.7** Pengujian *Software* dengan Komunikasi *WiFi*

Pada gambar 4.7 menunjukkan pengujian *software* menggunakan komunikasi *wifi*. Dapat dilihat pada gambar diatas bahwa pengguna memilih komunikasi *wifi*, kemudian setelah memilih komunikasi maka pengguna memasukkan *IP Address* yang sesuai dengan yang ditentukan. Setelah mengisi *IP Address* pengguna terhubung dengan mode komunikasi *wifi* sehingga pengguna dapat mengakses data kemiringan dari suatu bangunan. Setelah melakukan pengujian dapat disimpulkan bahwa data yang ditampilkan dengan media komunikasi *wifi* datanya tidak berbeda dengan data yang ditampilkan oleh LCD.



**Gambar 4.8** Pengujian *Software* dengan Komunikasi *Bluetooth*

Gambar 4.8 menunjukkan pengujian *software* menggunakan komunikasi *bluetooth*. Pada gambar tersebut menunjukkan bahwa

awalnya pengguna harus mengaktifkan *bluetooth* yang ada pada *smartphone* android. Setelah *bluetooth* aktif, pengguna harus melakukan *pairing device* agar terhubung dengan *bluetooth* HC-06 yang terhubung dengan sensor. Ketika *bluetooth* dari *smartphone* telah terhubung dengan *bluetooth* HC-06 maka akan muncul simbol sinyal berwarna *orange* pada bagian pojok kanan bawah dari layar *smartphone* android, menandakan pengguna dapat mengakses data yaitu monitoring kemiringan dari suatu bangunan atau gedung. Dapat dilihat pada gambar bahwa data monitoring yang ditampilkan pada layar *smartphone* android hasilnya tidak ada perbedaan dengan data yang ditampilkan pada LCD. Hal ini menandakan bahwa *software* siap digunakan untuk melakukan monitoring.

#### 4.7 Pengujian Alat Keseluruhan

Pengujian ini berfungsi untuk mengetahui alat dapat bekerja dengan baik secara keseluruhan. Adapun langkah-langkah dalam pengujian alat ini adalah sebagai berikut.

1. Siapkan seluruh perangkat yang akan digunakan, diantaranya : *power supply* dan miniatur gedung yang telah dipasang alat pendeteksi kemiringan pada gedung.
2. Sambungkan kabel *power supply* pada stop kontak.
3. Posisikan toggle pada posisi atas untuk mengaktifkan sensor ADXL345 pada atap miniatur gedung.
4. Sambungkan output *power supply* pada input tegangan mikrokontroler arduino uno.
5. Tentukan posisi gedung sebagai posisi awal sudut (sudut kalibrasi).=
6. Tekan tombol kalibrasi untuk menentukan sudut kalibrasi.
7. Buka aplikasi pada *smartphone* android untuk melakukan monitoring.
8. Tentukan komunikasi yang digunakan untuk melakukan monitoring, terdapat 2 pilihan yaitu *bluetooth* dan *wifi*.
9. Pilih satu mode komunikasi untuk dapat mengakses data dari sensor.
10. Amati perubahan sudut dari bangunan.

Dalam pengujian alat secara keseluruhan kami mengambil 20 data tetapi hanya beberapa *sample* datayang akan dihitung dan didapatkan hasil pengujian dan perhitungan data sebagai berikut :

**Tabel 4.8** Hasil Pengujian Busur 1

POSISI	DATA SUDUT 1	
	Derajat Busur	LCD
1	90°	0°
2	85°	-5°
3	80°	-10°
4	75°	-15°
5	70°	-20°
6.	65°	-25°
7.	95°	5°
8.	100°	10°
9.	105°	15°
10.	110°	20°
11.	115 °	25 °

Dari data diatas dapat dilakukan perhitungan untuk menganalisa nilai kepresisian dari alat monitoring kemiringan bangunan. Rumus perhitungannya adalah :

- Data Sudut 1

% Presisi Data : (Data yang diambil sesuai dengan tabel)

$$\% \text{ Presisi Data} = \frac{\sum \text{SELISIH DATA LCD dan SETELAH KALIBRASI}}{\text{Banyak data yang diambil}} \times 100 \%$$

Keterangan :

$$\text{Selisih Data} = \frac{\text{Data yang kecil}}{\text{Data yang besar}}$$

%PresisiData =

$$\frac{\left(\frac{0}{0}\right) + \left(\frac{5}{5}\right) + \left(\frac{10}{10}\right) + \left(\frac{15}{15}\right) + \left(\frac{20}{20}\right) + \left(\frac{25}{25}\right) + \left(\frac{5}{5}\right) + \left(\frac{10}{10}\right) + \left(\frac{15}{15}\right) + \left(\frac{20}{20}\right) + \left(\frac{25}{25}\right)}{11} \times 100 \%$$

$$\% \text{ Presisi Data} = \frac{(1+1+1+1+1+1+1+1+1+1+1)}{11} \times 100 \%$$

$$\% \text{ Presisi Data} = \frac{11}{11} \times 100 \%$$

$$\% \text{ Presisi Data} = 100 \%$$

Pada perhitungan data diatas dapat disimpulkan bahwa data sudut 1 pada alat monitoring ini diatas 50% yaitu 83%, maka data yang ditampilkan adalah benar dan mendekati nilai presisi atau stabil.

**Tabel 4.9** Hasil Pengujian Busur 2

POSISI	DATA SUDUT 2	
	Derajat Busur	LCD
1	90°	0°
2	85°	-5°
3	80°	-10°
4	75°	-15°
5	70°	-20°
6.	65°	-25°
7.	95°	5°
8.	100°	10°
9.	105°	15°
10.	110°	20°
11.	115 °	25 °

Dari data diatas dapat dilakukan perhitungan untuk menganalisa nilai kepresisian dari alat monitoring kemiringan bangunan. Rumus perhitungannya adalah :

- Data Sudut 1

% Presisi Data : (Data yang diambil sesuai dengan tabel)

$$\% \text{ Presisi Data} = \frac{\sum \text{SELISIH DATA LCD dan SETELAH KALIBRASI}}{\text{Banyak data yang diambil}} \times 100 \%$$

Keterangan :

$$\text{Selisih Data} = \frac{\text{Data yang kecil}}{\text{Data yang besar}}$$

%PresisiData=

$$\frac{\left(\frac{0}{0}\right) + \left(\frac{5}{5}\right) + \left(\frac{10}{10}\right) + \left(\frac{15}{15}\right) + \left(\frac{20}{20}\right) + \left(\frac{25}{25}\right) + \left(\frac{5}{5}\right) + \left(\frac{10}{10}\right) + \left(\frac{15}{15}\right) + \left(\frac{20}{20}\right) + \left(\frac{25}{25}\right)}{11} \times 100 \%$$

11

$$\% \text{ Presisi Data} = \frac{(1+1+1+1+1+1+1+1+1+1)}{11} \times 100 \%$$

$$\% \text{ Presisi Data} = \frac{11}{11} \times 100 \%$$

$$\% \text{ Presisi Data} = 100 \%$$

Pada perhitungan data diatas dapat disimpulkan bahwa keakuratan data dari sudut 2 pada alat monitoring ini sebesar 100% maka data yang ditampilkan oleh sudut 2 dapat dikatakan benar dan data termasuk dalam nilai presisi atau stabil.

Pada pengujian alat secara keseluruhan ini dapat diketahui bahwa data yang ditampilkan oleh alat adalah benar dan sesuai dengan yang diharapkan. Data yang ditampilkan berupa perubahan sudut kemiringan dari suatu bangunan atau gedung. Selain itu kita dapat mengetahui kemiringan dari suatu bangunan dengan *realtime* menggunakan teknologi *androidsmartphone* yang bisa diakses melalui sambungan *bluetooth* maupun *wifi*. Sehingga pengguna, baik pengelola maupun penghuni dapat mengetahui kondisi dari suatu bangunan apakah layak huni ataupun tidak. Diharapkan alat ini dapat bekerja sesuai dengan sistem kerja yang ditentukan agar dapat memberikan peringatan dini kepada para penghuni dan pengelola bangunan atau gedung pencakar langit dalam mengantisipasi adanya gedung atau bangunan yang roboh.



**Gambar 4.9** Pengujian Keseluruhan Alat Monitoring Kemiringan Bangunan Menggunakan *Accelerodengan* Media Komunikasi *Bluetooth* dan *Wifi*

## **BAB V PENUTUP**

Bab penutup ini berisi tentang kesimpulan yang diperoleh selama proses pembuatan monitoring kemiringan bangunan menggunakan *accelero* dengan media komunikasi *bluetooth* dan *wifi*, kesimpulan dari hasil pengujian dan analisa data, serta saran untuk monitoring kemiringan bangunan menggunakan *accelero* dengan media komunikasi *bluetooth* dan *wifi* ini kedepannya.

### **5.1 Kesimpulan**

1. Pemantauan kemiringan bangunan fisik menggunakan *accelero* dengan media komunikasi *bluetooth* dan *wifi* dapat memberikan informasi perubahan sudut kemiringan posisi bangunan. Informasi yang diterima merepresentasikan perubahan sudut kemiringan bangunan fisik yang bermanfaat untuk mengetahui perubahan secara fisik secara aktual sebagai basis peringatan dini bahaya keruntuhan gedung dan bangunan fisik.
2. Rancang purwarupa pada Tugas Akhir monitoring kemiringankali ini memiliki rata-rata kebenaran sebesar 100% sehingga *accelerometer* ini dapat digunakan sebagai sensor kemiringan yang cukup akurat.
3. Komunikasi data antara *bluetooth* dan *wifi* sama-sama memiliki kelebihan dan kekurangan. Kelebihan dari *bluetooth* adalah saat *refreshing data* lebih mudah dibandingkan dengan menggunakan *wifi*, karena komunikasi *wifi* bergantung dari koneksi dari jaringan internet. Namun menggunakan *wifi* jangkauannya lebih luas daripada menggunakan *bluetooth* yang hanya dapat diakses sampai  $\pm 10$  meter.

### **5.2 Saran**

1. Jika ingin menghasilkan data yang lebih baik lagi, maka sebaiknya menggunakan mikrokontroler yang *port* komunikasi I to C-nya lebih dari 1, karena mikrokontroler arduino uno *port*-nya terbatas.
2. Rancang purwarupa pada Tugas Akhir kali ini memiliki kekurangan pada hal akurasi sehingga ketika akan

direalisasikan pada kondisi riil, maka harus dilakukan perbaikan terlebih dahulu.

3. Perlu ditambahkan data *logger* agar data yang diambil dapat dibandingkan dan dapat diketahui perbedaan dari setiap data yang diambil, sehingga dapat diketahui sebuah bangunan itu mengalami kemiringan atau tidak.

## DAFTAR PUSTAKA

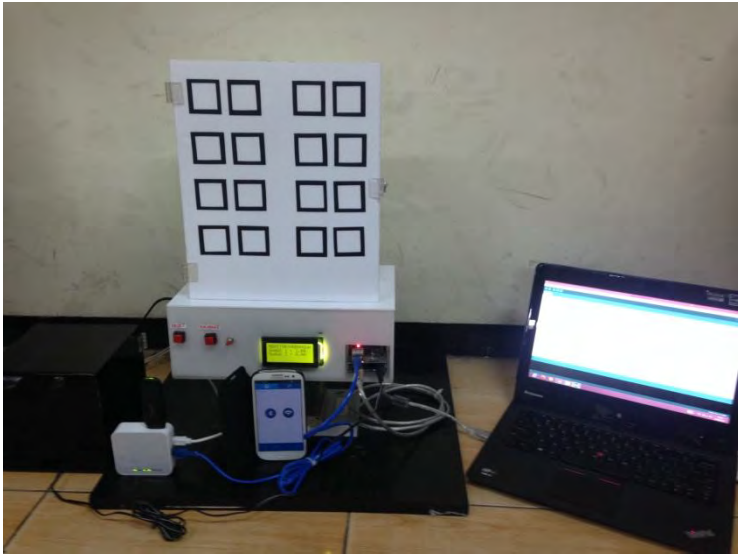
- [1] Susilo, Tri Bagus, *Pengukuran Sudut Kemiringan Benda dengan Sensor Percepatan*, Tugas Akhir Teknik Elektro Universitas Diponegoro, Semarang, 2011.
- [2] Banzi, Massimo. 2011. *Getting Started with Arduino*. O'Reilly Media Inc. USA.
- [3] J. Alam, M. Agus. 2002. *Mengenal Wifi, Hotspot, LAN, dan Sharing Internet*. Jakarta : PT Elex Media Komputindo.
- [4] Agusli, Rachmat. 2007. *Panduan Praktis Koneksi INTERNET Dengan Handphone*. Jakarta Selatan : mediakita
- [5] Hantoro, Dwi, Gunadi, 2009, *WIFI (Wireless LAN) Jaringan Komputer Tanpa Kabel*, Penerbit, INFORMATIKA, Bandung.
- [6] Haryanto, Agus. 2014. *Android Fast Track: Belajar Membuat Aplikasi Android dengan Mudah dan Cepat*, Hal 9.
- [7] Hardana, 2011, *Konfigurasi wireless routerbord mikrotik*, Andi, Yogyakarta
- [8] J. Kim, Kuinam, 2015, *Information Science and Applications*. Springer. London, Hal 22.



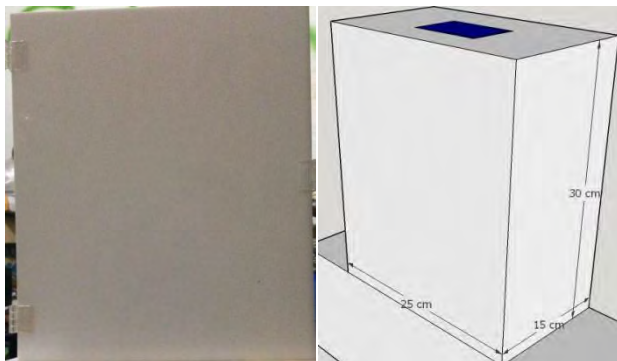
-----Halaman ini sengaja dikosongkan-----

## LAMPIRAN A FOTO ALAT

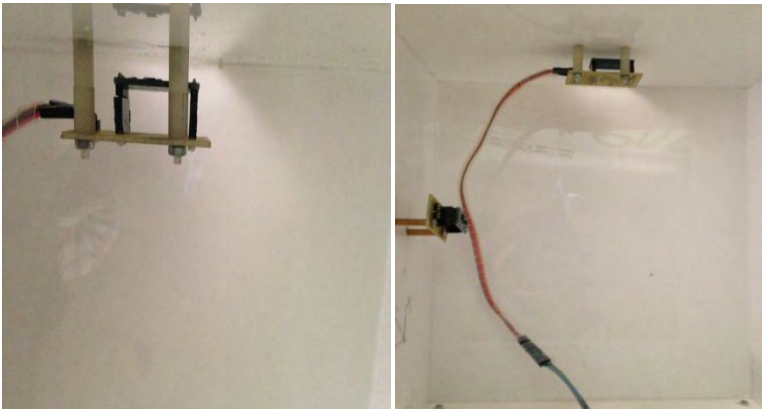
### Alat Monitoring Kemiringan Bangunan



### Miniatur Bangunan



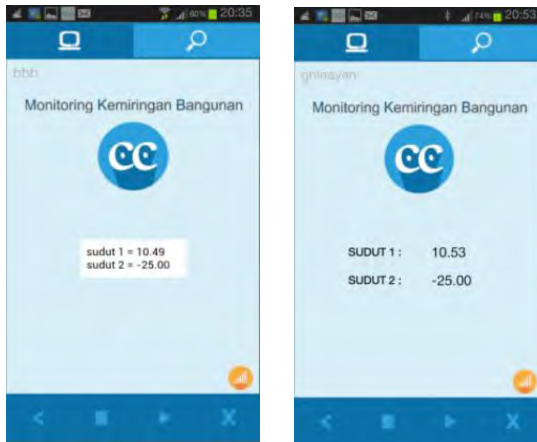
## Penempatan Sensor *Accelerometer* ADXL345 di bagian atap gedung



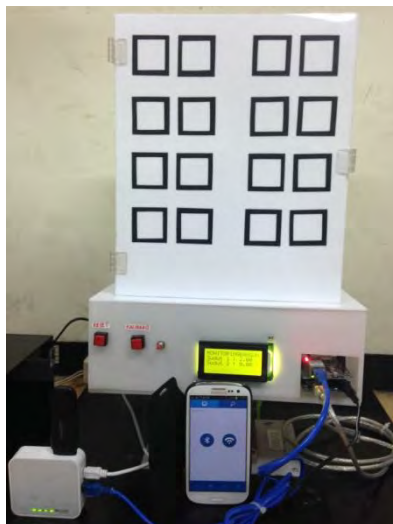
## Hasil Monitoring yang Ditampilkan oleh LCD



## Tampilan Monitoring pada Android



## Hasil Pengambilan Data Monitoring Kemiringan Bangunan Keseluruhan



-----Halamaninisengajadikosongkan-----

## LAMPIRAN B

### LISTING PROGRAM

#### **Listing Program Mikrokontroler Arduino Uno**

```
#include <SPI.h>
#include <Ethernet.h>
#include <math.h>
#include <EEPROM.h>

#include <SoftwareSerial.h> // import the serial library

#include <Wire.h>
#include <Adafruit_Sensor.h>
#include <Adafruit_ADXL345_U.h>
#include <LiquidCrystal.h>
LiquidCrystal lcd(7, 6, 5, 4, 3, 2);

float datax, datay, dataz, sudut1, sudut2;
Adafruit_ADXL345_Unified accel = Adafruit_ADXL345_Unified(12345);

int buttonPin=A1,buttonPin1=A2,buttonPin2=A3;
int buttonState=0,buttonState1=0,buttonState2=0;
int address = 0;
int busur1 = 0;
int busur2 = 0;

SoftwareSerial blue(9, 8); // RX, TX

byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED }; //physical mac
address
byte ip[] = { 192, 168, 0, 5 }; // ip in lan (that's what you need
to use in your browser. ("192.168.1.178")
byte gateway[] = { 192, 168, 0, 254 }; // internet access via
router
byte subnet[] = { 255, 255, 255, 0 }; //subnet mask
//IPAddress ip(192, 168, 1, 5);
EthernetServer server(80); //server port
String readString;
```

```
float kalibrasix = 0,kalibrasiy=0, kalibrasiz = 0,kalibrasi1= 0,kalibrasi2=0;
char buff[255]="";
char float_datax[16] = "";
char float_datay[16] = "";
char float_dataz[16] = "";
char float_sudut1[16] = "";
char float_sudut2[16] = "";
```

```
void setup(void)
{
  pinMode(10, INPUT_PULLUP);
  Serial.begin(9600);
  blue.begin(9600);
  while (!Serial) {}
  lcd.begin(16, 4);
  if(!accel.begin())
  {
    Serial.println("Cek Koneksi ANda");
    lcd.setCursor(0, 0); lcd.print("Cek Sensor");
    while(1);
  }
}
```

```
  Ethernet.begin(mac, ip, gateway, subnet);
  server.begin();
  Serial.print("server is at ");
  Serial.println(Ethernet.localIP());
```

```
}
```

```
void loop(void)
{
  lcd.clear();
  buttonState = analogRead(buttonPin);
  kalibrasi1 = EEPROM.read(1);
  kalibrasi2 = EEPROM.read(2);
  address = EEPROM.read(3);
```

```

if (buttonState <= 100) {
    kalibrasi1 = kalibrasi1 + sudut1;
    kalibrasi2 = kalibrasi2 + sudut2;
    EEPROM.write(1, kalibrasi1);
    EEPROM.write(2, kalibrasi2);

    lcd.clear();
    lcd.setCursor(0, 0); lcd.print("kalibrasi");
    lcd.setCursor(0, 1); lcd.print("Dont Shake!!");
    delay(1000);
    lcd.setCursor(0, 0); lcd.print("kalibrasi .");
    lcd.setCursor(0, 1); lcd.print("Dont Shake!!");
    delay(1000);
    lcd.setCursor(0, 0); lcd.print("kalibrasi . .");
    lcd.setCursor(0, 1); lcd.print("Dont Shake!!");
    delay(1000);
    lcd.setCursor(0, 0); lcd.print("kalibrasi . . .");
    lcd.setCursor(0, 1); lcd.print("Dont Shake!!");
    lcd.clear();
}
else {
    //nothing
}

sensors_event_t event;
accel.getEvent(&event);
accel.setRange(ADXL345_RANGE_8_G);
datax=((event.acceleration.x)); //fungsi x
datay=((event.acceleration.y)); //fungsi y
dataz=((event.acceleration.z)); //fungsi z

sudut1 = (atan(datax/(sqrt((datay*datay)+(dataz*dataz))))*100);
sudut2 = (atan(datay/(sqrt((datax*datax)+(dataz*dataz))))*100);

//sudut1 = int((((acos(datax/dataz)*100))*0.56)-kalibrasi1;//belum
diambil datanya//
//sudut2 = int((((acos(datay/dataz)*100))*0.56)-kalibrasi2;//belum
diambil datanya//

```



```

dtostrf(datax,4,2,float_datax);
dtostrf(datay,4,2,float_datay);
dtostrf(dataz,4,2,float_dataz);
dtostrf(sudut1,4,2,float_sudut1);
dtostrf(sudut2,4,2,float_sudut2);
sprintf(buff,"%s _ %s _ %s _ %s _ %s",float_datax, float_datay,
float_dataz, float_sudut1, float_sudut2);

```

```

EthernetClient client = server.available();
if (client) {
  Serial.println("new client");
  // an http request ends with a blank line
  boolean currentLineIsBlank = true;
  while (client.connected()) {
    if (client.available()) {
      char c = client.read();
      Serial.write(c);
      if (c == '\n' && currentLineIsBlank) {
        // send a standard http response header
        client.println("HTTP/1.1 200 OK");
        client.println("Content-Type: text/html");
        client.println("Connection: close"); // the connection will be closed
        after completion of the response
        client.println("Refresh: 3");
        client.println();
        client.println("<!DOCTYPE HTML>");
        client.println("<html>");
        client.print("sudut 1 = ");client.print(sudut1);client.println("<br />");
        client.print("sudut 2 = ");client.print(sudut2);client.println("<br />");
        //client.print("data z = ");client.print(dataz);client.println("<br />");
        client.println("</html>");
        break;
      }
    }
    if (c == '\n') {
      currentLineIsBlank = true;
    }
    else if (c != '\r') {
      currentLineIsBlank = false;
    }
  }
}

```

```

    }
  }
}
  delay(1);
  client.stop();
  Serial.println("client disconnected");
}

```

```

//bluetooth
blue.print(sudut1);
delay(1000);
blue.println(sudut2);
lcd.setCursor(0, 0); lcd.print("MONITORINGBANGUN");
//lcd.setCursor(0, 1); lcd.print("X: ");lcd.print(datax);//lcd.print("m/s^2");
//lcd.setCursor(0, 2); lcd.print("Y: ");lcd.print(datay);//lcd.print("m/s^2");
//lcd.setCursor(0, 3); lcd.print("Z: ");lcd.print(dataz);//lcd.print("m/s^2");
lcd.setCursor(0, 1); lcd.print("Sudut 1 : ");lcd.print(sudut1);
lcd.setCursor(0, 2); lcd.print("Sudut 2 : ");lcd.print(sudut2);

Serial.print(sudut1);Serial.print(" ");Serial.println(sudut2);

  delay(1000);

}

```

## Listing Program Aplikasi Android

Version=3.82

IconFile=

NumberOfModules=1

Module1=Smart

Build1=Default,b4a.example

DoNotOverwriteManifest=False

ManifestCode=This code will be applied to the manifest file during compilation.~\n~You do not need to modify it in most cases.~\n~'See this link for for more information:

<http://www.basic4ppc.com/forum/showthread.php?p=78136>~\n~AddManife

stText(~\n~<uses-sdk android:minSdkVersion="4"

android:targetSdkVersion="14"/>~\n~<supports-screens

android:largeScreens="true" ~\n~ android:normalScreens="true" ~\n~

android:smallScreens="true" ~\n~

android:anyDensity="true"/>)~\n~SetApplicationAttribute(android:icon,

"@drawable/icon")~\n~SetApplicationAttribute(android:label,

"\$LABEL\$")~\n~'End of default text.~\n~

NumberOfFiles=22

File1=2.bal

File2=BACK.png

File3=BACK1.png

File4=CLOSE.png

File5=CLOSE1.png

File6=first.bal

File7=HOME.png

File8=HOME1.png

File9=HWK.png

File10=MENU.png

File11=MENU1.png

File12=PLAY.png

File13=PLAY1.png

File14=setting.bal

File15=STOP.png

File16=STOP1.png

File17=TOOTH.png

File18=TOOTH1.png

File19=TOOTH2.png

File20=WIFI.png

```

File21=WIFI1.png
File22=WIFI2.png
NumberOfLibraries=12
Library1=accessibility
Library2=camera
Library3=core
Library4=http
Library5=httplib2
Library6=livewallpaper
Library7=network
Library8=randomaccessfile
Library9=serial
Library10=stringutils
Library11=xmlsax
Library12=phone
@EndOfDesignText@
#Region Project Attributes
    #ApplicationLabel: Accelero
    #VersionCode: 1
    #VersionName:
    'SupportedOrientations possible values: unspecified, landscape or
portrait.
    #SupportedOrientations: unspecified
    #CanInstallToExternalStorage: False
#End Region

#Region Activity Attributes
    #FullScreen: False
    #IncludeTitle: False
#End Region

Sub Process_Globals
    Dim time As Timer
    Dim speed As Int
    Dim jeneng As String
End Sub

Sub Globals
    Dim user, pass As EditText

```

```

        Dim enter As Button
        Dim a As Int
        Private HWK As Button
End Sub

Sub Activity_Create(FirstTime As Boolean)
    speed = 200
    Activity.LoadLayout ("first")
    time.Initialize ("time",speed)
    time.Enabled =True
End Sub

Sub Activity_Resume

End Sub

Sub Activity_Pause (UserClosed As Boolean)
    'time.Enabled = False
End Sub

Sub time_tick()
    a = a + 1
    If a=1 Then
        HWK.SetBackgroundImage(LoadBitmap (File.DirAssets
, "HWK.png"))
    Else
        HWK.SetBackgroundImage(LoadBitmap (File.DirAssets
, "HWK1.png"))
    a=0
    End If
End Sub

Sub enter_Click
    Dim b, c As String
    b = user.Text
    c = pass.Text
    If b = "" AND c = "" Then
        MsgBox("Plese insert Username and Password !!!", "WRONGWAY")
        user.RequestFocus()
    End If
End Sub

```

```

Else If b <> "" AND c = "" Then
    MsgBox("Plese insert your Password !!!", "WRONGWAY")
    pass.RequestFocus()
Else If b = "" AND c <> "" Then
    MsgBox("Plese insert your Name !!!", "WRONGWAY")
    user.RequestFocus()
Else If b <> "" AND c <> ("ghinayan") AND b <> ("" ) Then
    MsgBox("Wrng Password !!!", "WRONGWAY")
    pass.Text = ""
    pass.RequestFocus()
Else If b <> "" AND c = ("ghinayan") Then
    jeneng = b
    StartActivity("smart")

End If
End Sub

Sub HWK_Click
    Activity.LoadLayout("2")
    HWK.Visible = False
    time.Enabled =False
End Sub

```

-----Halaman ini sengaja dikosongkan-----

# LAMPIRAN C DATASHEET

## 1. Datasheet Arduino Uno

### Features

- High Performance, Low Power AVR<sup>®</sup> 8-Bit Microcontroller
- Advanced RISC Architecture
  - 131 Powerful Instructions – Most Single Clock Cycle Execution
  - 32 x 8 General Purpose Working Registers
  - Fully Static Operation
  - Up to 20 MIPS Throughput at 20 MHz
  - On-chip 2-cycle Multiplier
- High Endurance Non-volatile Memory Segments
  - 4/8/16/32K Bytes of In-System Self-Programmable Flash program memory (ATmega48PA/88PA/168PA/328P)
  - 256/512/1K/2K Bytes EEPROM (ATmega48PA/88PA/168PA/328P)
  - 612/1K/1K/2K Bytes Internal SRAM (ATmega48PA/88PA/168PA/328P)
  - Write/Erase Cycles: 10,000 Flash/100,000 EEPROM
  - Data retention: 20 years at 85°C/100 years at 25°C<sup>(1)</sup>
  - Optional Boot Code Section with Independent Lock Bits
  - In-System Programming by On-chip Boot Program
  - True Read-While-Write Operation
    - Programming Lock for Software Security
- Peripheral Features
  - Two 8-bit Timer/Counters with Separate Prescaler and Compare Mode
  - One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
  - Real Time Counter with Separate Oscillator
  - 3x PWM Channels
  - 8-channel 10-bit ADC in TQFP and QFN/MLF package
    - Temperature Measurement
  - 8-channel 10-bit ADC in PDIP Package
    - Temperature Measurement
  - Programmable Serial USART
  - Master/Slave SPI Serial Interface
  - Byte-oriented 2-wire Serial Interface (Philips I<sup>2</sup>C compatible)
  - Programmable Watchdog Timer with Separate On-chip Oscillator
  - On-chip Analog Comparator
  - Interrupt and Wake-up on Pin Change
- Special Microcontroller Features
  - Power-on Reset and Programmable Brown-out Detection
  - Internal Calibrated Oscillator
  - External and Internal Interrupt Sources
  - 3x Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby, and Extended Standby
- I/O and Packages
  - 25 Programmable I/O Lines
  - 28-pin PDIP, 32-lead TQFP, 28-pad QFN/MLF and 32-pad QFN/MLF
- Operating Voltage:
  - 1.8 - 6.5V for ATmega48PA/88PA/168PA/328P
- Temperature Range:
  - -40°C to 85°C
- Speed Grade:
  - 0 - 20 MHz @ 1.8 - 6.5V
- Low Power Consumption at 1 MHz, 1.8V, 25°C for ATmega48PA/88PA/168PA/328P:
  - Active Mode: 0.2 mA
  - Power-down Mode: 0.7 µA
  - Power-save Mode: 0.75 µA (including 32 kHz RTC)



8-bit AVR<sup>®</sup>  
Microcontroller  
with 4/8/16/32K  
Bytes In-System  
Programmable  
Flash

ATmega48PA  
ATmega88PA  
ATmega168PA  
ATmega328P

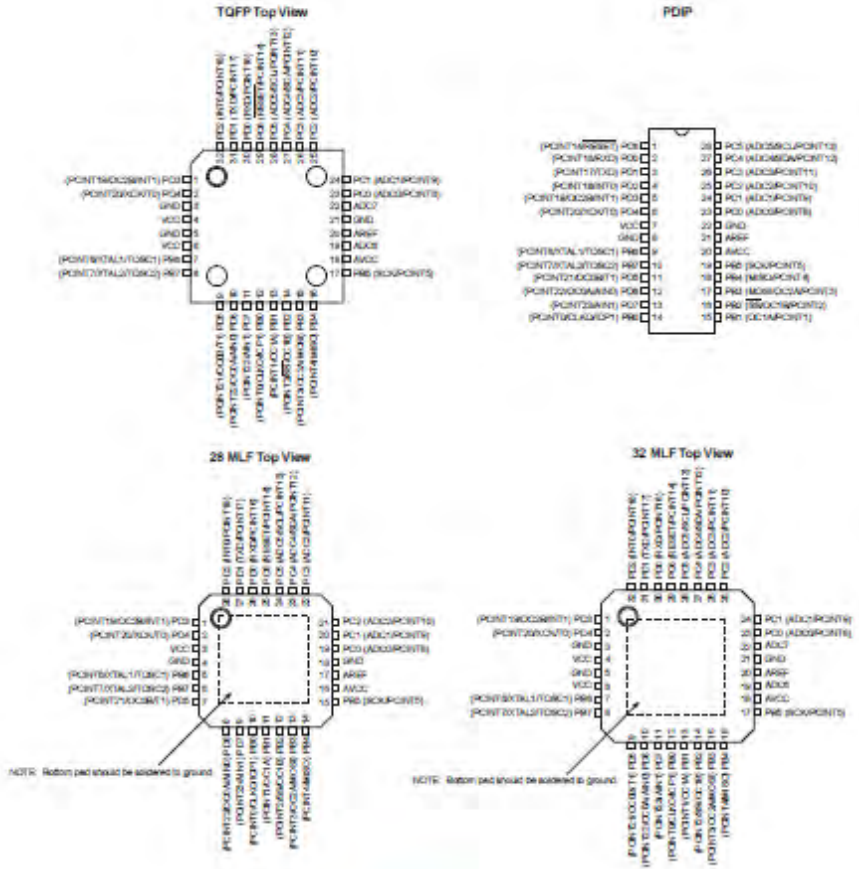
(Rev. 9/01) (D-AVR)-1029





## 1. Pin Configurations

Figure 1-1. Pinout ATmega48PA/88PA/168PA/328P



## 1.1 Pin Descriptions

### 1.1.1 VCC

Digital supply voltage.

### 1.1.2 GND

Ground.

### 1.1.3 Port B (PB7:0) XTAL1/XTAL2/TOSC1/TOSC2

Port B is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port B output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port B pins that are externally pulled low will source current if the pull-up resistors are activated. The Port B pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Depending on the clock selection fuse settings, PB6 can be used as input to the Inverting Oscillator amplifier and input to the internal clock operating circuit.

Depending on the clock selection fuse settings, PB7 can be used as output from the Inverting Oscillator amplifier.

If the Internal Calibrated RC Oscillator is used as chip clock source, PB7...6 is used as TOSC2...1 input for the Asynchronous Timer/Counter2 if the AS2 bit in ASSR is set.

The various special features of Port B are elaborated in: ["System Clock and Clock Options" on page 26.](#)

### 1.1.4 Port C (PC5:0)

Port C is a 7-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The PC5...0 output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port C pins that are externally pulled low will source current if the pull-up resistors are activated. The Port C pins are tri-stated when a reset condition becomes active, even if the clock is not running.

### 1.1.5 PC6/RESET

If the RSTDISBL Fuse is programmed, PC6 is used as an I/O pin. Note that the electrical characteristics of PC6 differ from those of the other pins of Port C.

If the RSTDISBL Fuse is unprogrammed, PC6 is used as a Reset input. A low level on this pin for longer than the minimum pulse length will generate a Reset, even if the clock is not running. The minimum pulse length is given in [Table 29-12 on page 323.](#) Shorter pulses are not guaranteed to generate a Reset.

The various special features of Port C are elaborated in: ["Alternate Functions of Port C" on page 88.](#)

### 1.1.6 Port D (PD7:0)

Port D is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port D output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port D pins that are externally pulled low will source current if the pull-up resistors are activated. The Port D pins are tri-stated when a reset condition becomes active, even if the clock is not running.



## ATmega48A/48PA/88A/88PA/168A/168PA/328/328P

The various special features of Port D are elaborated in "Alternate Functions of Port D" on page 89.

- 1.1.7 AV<sub>CC</sub>**  
AV<sub>CC</sub> is the supply voltage pin for the A/D Converter, PC3:0, and ADC7:6. It should be externally connected to V<sub>CC</sub>, even if the ADC is not used. If the ADC is used, it should be connected to V<sub>CC</sub> through a low-pass filter. Note that PC6...4 use digital supply voltage, V<sub>CC</sub>.
- 1.1.8 AREF**  
AREF is the analog reference pin for the A/D Converter.
- 1.1.9 ADC7:6 (TQFP and QFN/MLF Package Only)**  
In the TQFP and QFN/MLF package, ADC7:6 serve as analog inputs to the A/D converter. These pins are powered from the analog supply and serve as 10-bit ADC channels.



## 2. Datasheet ADXL345



# 3-Axis, $\pm 2$ g/ $\pm 4$ g/ $\pm 8$ g/ $\pm 16$ g Digital Accelerometer

Data Sheet

ADXL345

### FEATURES

Ultralow power: as low as 23  $\mu$ A in measurement mode and 0.1  $\mu$ A in standby mode at  $V_{CC} = 2.5$  V (typical)

Power consumption scales automatically with bandwidth  
User-selectable resolution

Fixed 10-bit resolution

Full resolution, where resolution increases with g range, up to 13-bit resolution at  $\pm 16$  g (maintaining 4 mg/LSB scale factor in all g ranges)

Embedded memory management system with FIFO technology minimizes host processor load

Single tap/double tap detection

Activity/inactivity monitoring

Free-fall detection

Supply voltage range: 2.0 V to 3.6 V

I/O voltage range: 1.7 V to  $V_{CC}$

SPI (3- and 4-wire) and I<sup>2</sup>C digital interfaces

Flexible interrupt modes mappable to either interrupt pin

Measurement ranges selectable via serial command

Bandwidth selectable via serial command

Wide temperature range (-40°C to +85°C)

10,000 g shock survival

Pb free/RoHS compliant

Small and thin: 3 mm  $\times$  5 mm  $\times$  1 mm LGA package

### APPLICATIONS

Handsets

Medical instrumentation

Gaming and pointing devices

Industrial instrumentation

Personal navigation devices

Hard disk drive (HDD) protection

### GENERAL DESCRIPTION

The ADXL345 is a small, thin, ultralow power, 3-axis accelerometer with high resolution (13-bit) measurement at up to  $\pm 16$  g. Digital output data is formatted as 16-bit two's complement and is accessible through either a SPI (3- or 4-wire) or I<sup>2</sup>C digital interface.

The ADXL345 is well suited for mobile device applications. It measures the static acceleration of gravity in tilt-sensing applications, as well as dynamic acceleration resulting from motion or shock. Its high resolution (3.9 mg/LSB) enables measurement of inclination changes less than 1.0°.

Several special sensing functions are provided. Activity and inactivity sensing detect the presence or lack of motion by comparing the acceleration on any axis with user-set thresholds. Tap sensing detects single and double taps in any direction. Free-fall sensing detects if the device is falling. These functions can be mapped individually to either of two interrupt output pins. An integrated memory management system with a 32-level first in, first out (FIFO) buffer can be used to store data to minimize host processor activity and lower overall system power consumption.

Low power modes enable intelligent motion-based power management with threshold sensing and active acceleration measurement at extremely low power dissipation.

The ADXL345 is supplied in a small, thin, 3 mm  $\times$  5 mm  $\times$  1 mm, 14-lead, plastic package.

### FUNCTIONAL BLOCK DIAGRAM

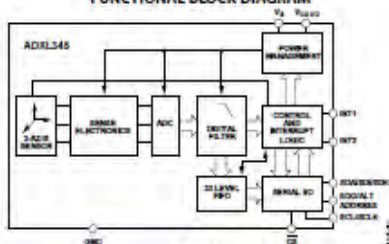


Figure 1.

Rev. E

Information furnished by Analog Devices is believed to be accurate and reliable. However, no responsibility is assumed by Analog Devices for its use, nor for any infringements of patents or other rights of third parties that may result from its use. Specifications subject to change without notice. No license is granted by implication or otherwise under any patent or patent rights of Analog Devices. Trademarks and registered trademarks are the property of their respective owners.

### Document Feedback

One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106, U.S.A.  
Tel: 781-329-4700 ©2009–2015 Analog Devices, Inc. All rights reserved.  
Technical Support [www.analog.com](http://www.analog.com)

## SPECIFICATIONS

$T_A = 25^\circ\text{C}$ ,  $V_D = 2.5\text{ V}$ ,  $V_{\text{DIGIO}} = 1.8\text{ V}$ , acceleration = 0 g,  $C_0 = 10\ \mu\text{F}$  tantalum,  $C_{\text{INT}} = 0.1\ \mu\text{F}$ ; output data rate (ODR) = 800 Hz, unless otherwise noted. All minimum and maximum specifications are guaranteed. Typical specifications are not guaranteed.

Table 1.

Parameter	Test Conditions	Min	Typ <sup>1</sup>	Max	Unit
<b>SENSOR INPUT</b>					
Measurement Range	Each axis User selectable			$\pm 2, \pm 4, \pm 8, \pm 16$	g
Nonlinearity	Percentage of full scale		$\pm 0.5$		%
Inter-Axis Alignment Error			$\pm 0.1$		Degrees
Cross-Axis Sensitivity <sup>2</sup>			$\pm 1$		%
<b>OUTPUT RESOLUTION</b>					
All g Ranges	Each axis 10-bit resolution		10		Bits
$\pm 2$ g Range	Full resolution		10		Bits
$\pm 4$ g Range	Full resolution		11		Bits
$\pm 8$ g Range	Full resolution		12		Bits
$\pm 16$ g Range	Full resolution		13		Bits
<b>SENSITIVITY</b>					
Sensitivity at $X_{\text{OUT}}, Y_{\text{OUT}}, Z_{\text{OUT}}$					
	All g-ranges, full resolution	230	256	282	LSB/g
	$\pm 2$ g, 10-bit resolution	230	256	282	LSB/g
	$\pm 4$ g, 10-bit resolution	115	128	141	LSB/g
	$\pm 8$ g, 10-bit resolution	57	64	71	LSB/g
	$\pm 16$ g, 10-bit resolution	29	32	35	LSB/g
Sensitivity Deviation from Ideal					
	All g-ranges		$\pm 1.0$		%
Scale Factor at $X_{\text{OUT}}, Y_{\text{OUT}}, Z_{\text{OUT}}$					
	All g-ranges, full resolution	3.5	3.9	4.3	mg/LSB
	$\pm 2$ g, 10-bit resolution	3.5	3.9	4.3	mg/LSB
	$\pm 4$ g, 10-bit resolution	7.1	7.8	8.7	mg/LSB
	$\pm 8$ g, 10-bit resolution	14.1	15.6	17.5	mg/LSB
	$\pm 16$ g, 10-bit resolution	28.6	31.2	34.5	mg/LSB
Sensitivity Change Due to Temperature					
			$\pm 0.01$		%/°C
<b>0 g OFFSET</b>					
0 g Output for $X_{\text{OUT}}, Y_{\text{OUT}}$					
	Each axis	-150	0	+150	mg
0 g Output for $Z_{\text{OUT}}$					
		-250	0	+250	mg
0 g Output Deviation from Ideal, $X_{\text{OUT}}, Y_{\text{OUT}}$					
			$\pm 35$		mg
0 g Output Deviation from Ideal, $Z_{\text{OUT}}$					
			$\pm 40$		mg
0 g Offset vs. Temperature for X-, Y-Axis					
			$\pm 0.4$		mg/°C
0 g Offset vs. Temperature for Z-Axis					
			$\pm 1.2$		mg/°C
<b>NOISE</b>					
$X$ -, $Y$ -Axis					
	ODR = 100 Hz for $\pm 2$ g, 10-bit resolution or all g-ranges, full resolution		0.75		LSB rms
$Z$ -Axis					
	ODR = 100 Hz for $\pm 2$ g, 10-bit resolution or all g-ranges, full resolution		1.1		LSB rms
<b>OUTPUT DATA RATE AND BANDWIDTH</b>					
Output Data Rate (ODR) <sup>3,4,5</sup>					
	User selectable	0.1		3200	Hz
<b>SELF-TEST<sup>6</sup></b>					
Output Change in X-Axis					
		0.20		2.10	g
Output Change in Y-Axis					
		-2.10		-0.20	g
Output Change in Z-Axis					
		0.30		3.40	g
<b>POWER SUPPLY</b>					
Operating Voltage Range ( $V_D$ )					
		2.0	2.5	3.6	V
Interface Voltage Range ( $V_{\text{DIGIO}}$ )					
		1.7	1.8	$V_L$	V
Supply Current					
	ODR $\geq$ 100 Hz		140		$\mu\text{A}$
	ODR < 10 Hz		30		$\mu\text{A}$
Standby Mode Leakage Current					
			0.1		$\mu\text{A}$
Turn-On and Wake-Up Time <sup>7</sup>					
	ODR = 3200 Hz		1.4		ms

Rev. E | Page 4 of 40

Parameter	Test Conditions	Min	Typ <sup>1</sup>	Max	Unit
TEMPERATURE					
Operating Temperature Range		-40		+85	°C
WEIGHT					
Device Weight			30		mg

<sup>1</sup> The typical specifications shown are for at least 68% of the population of parts and are based on the worst case of mean  $\pm 1\sigma$ , except for 0 g output and sensitivity, which represents the target value. For 0 g offset and sensitivity, the deviation from the ideal describes the worst case of mean  $\pm 1\sigma$ .

<sup>2</sup> Cross-axis sensitivity is defined as coupling between any two axes.

<sup>3</sup> Bandwidth is the  $-3$  dB frequency and is half the output data rate, bandwidth = ODR/2.

<sup>4</sup> The output format for the 3300 Hz and 1600 Hz ODRs is different than the output format for the remaining ODRs. This difference is described in the Data Formatting of Upper Data Rates section.

<sup>5</sup> Output data rates below 0.25 Hz exhibit additional offset shift with increased temperature, depending on selected output data rate. Refer to the Offset Performance at Lowest Data Rates section for details.

<sup>6</sup> Self-test charge is defined as the output (g) when the SELF\_TEST bit = 1 in the DATA\_FORMAT register, Address 0x1F minus the output (g) when the SELF\_TEST bit = 0. Due to device filtering, the output reaches its final value after  $4 \times \tau$  when enabling or disabling self-test, where  $\tau = 1/(\text{data rate})$ . The part must be in normal power operation (LOW\_POWER bit = 0 in the BW\_RATE register, Address 0x2C) for self-test to operate correctly.

<sup>7</sup> Turn-on and wake-up times are determined by the user-defined bandwidth. At a 100 Hz data rate, the turn-on and wake-up times are each approximately 11.1 ms. For other data rates, the turn-on and wake-up times are each approximately  $\tau + 1.1$  in milliseconds, where  $\tau = 1/(\text{data rate})$ .

## ABSOLUTE MAXIMUM RATINGS

Table 2.

Parameter	Rating
Acceleration	
Any Axis, Unpowered	10,000 g
Any Axis, Powered	10,000 g
$V_i$	-0.3 V to +3.9 V
$V_{DDIO}$	-0.3 V to +3.9 V
Digital Pins	-0.3 V to $V_{DDIO} + 0.3$ V or 3.9 V, whichever is less
All Other Pins	-0.3 V to +3.9 V
Output Short-Circuit Duration (Any Pin to Ground)	Indefinite
Temperature Range	
Powered	-40°C to +105°C
Storage	-40°C to +105°C

Stresses at or above those listed under Absolute Maximum Ratings may cause permanent damage to the product. This is a stress rating only; functional operation of the product at these or any other conditions above those indicated in the operational section of this specification is not implied. Operation beyond the maximum operating conditions for extended periods may affect product reliability.

## THERMAL RESISTANCE

Table 3. Package Characteristics

Package Type	$\theta_{JA}$	$\theta_{JC}$	Device Weight
14-Terminal LGA	150°C/W	85°C/W	30 mg

## PACKAGE INFORMATION

The information in Figure 2 and Table 4 provide details about the package branding for the ADXL345. For a complete listing of product availability, see the Ordering Guide section.

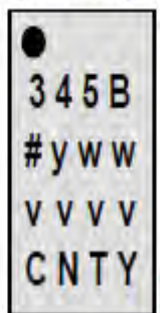


Figure 2. Product information on Package (Top View)

Table 4. Package Branding Information

Branding Key	Field Description
345B	Part Identifier for ADXL345
#	RoHS-compliant designation
yww	Date code
vvv	Factory lot code
CNTY	Country of origin

## ESD CAUTION



**ESD (electrostatic discharge) sensitive device.** Charged devices and circuit boards can discharge without detection. Although this product features patented or proprietary protection circuitry, damage may occur on devices subjected to high energy ESD. Therefore, proper ESD precautions should be taken to avoid performance degradation or loss of functionality.

## PIN CONFIGURATION AND FUNCTION DESCRIPTIONS

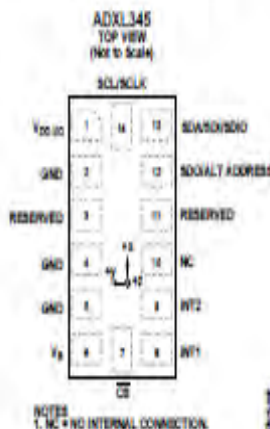


Figure 3. Pin Configuration (Top View)

Table 5. Pin Function Descriptions

Pin No.	Mnemonic	Description
1	V <sub>DDIO</sub>	Digital Interface Supply Voltage.
2	GND	This pin must be connected to ground.
3	RESERVED	Reserved. This pin must be connected to V <sub>I</sub> or left open.
4	GND	This pin must be connected to ground.
5	GND	This pin must be connected to ground.
6	V <sub>I</sub>	Supply Voltage.
7	$\overline{CS}$	Chip Select.
8	INT1	Interrupt 1 Output.
9	INT2	Interrupt 2 Output.
10	NC	Not Internally Connected.
11	RESERVED	Reserved. This pin must be connected to ground or left open.
12	SDA/ALT ADDRESS	Serial Data Output (SPI 4-Wire)/Alternate I <sup>2</sup> C Address Select (I <sup>2</sup> C).
13	SDA/SCL/SDIO	Serial Data (I <sup>2</sup> C)/Serial Data Input (SPI 4-Wire)/Serial Data Input and Output (SPI 3-Wire).
14	SCL/SCLK	Serial Communications Clock. SCL is the clock for I <sup>2</sup> C, and SCLK is the clock for SPI.



## TILT/INCLINATION CALCULATION

### Single-Axis Tilt Calculation

In applications where inclination sensing is needed only over a limited angle and with a somewhat coarse resolution, a single-axis device (or a single axis of a multiple-axis device) can be used.

For example, in Figure 1 a single axis (the x-axis in this example) is rotated through gravity. Because this approach uses only a single axis and requires the gravity vector, the calculated angle of inclination is accurate only when the device is oriented such that the x-axis is always in the plane of gravity. Any rotation about the other axes reduces the magnitude of the acceleration on the x-axis and results in error in the calculated angle of inclination.

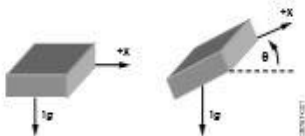


Figure 1. Single Axis Used for Tilt Sensing

Referring to basic trigonometry, the projection of the gravity vector on the x-axis produces an output acceleration equal to the sine of the angle between the accelerometer x-axis and the horizon. The horizon is typically taken to be the plane orthogonal to the gravity vector. For an ideal value of 1 g for gravity, the output acceleration is

$$A_{\text{axis}} [g] = 1 g \times \sin(\theta) \quad (1)$$

When using a single-axis solution, note that the sensitivity—that is, the change in output for some change in input—of the inclination calculation decreases as the angle between the horizon and the x-axis increases, approaching 0 as the angle approaches  $\pm 90^\circ$ . This can be seen in Figure 2, where the output acceleration, in g, is plotted against the angle of inclination. Near  $\pm 90^\circ$ , a large change in inclination angle results in a small change in output acceleration.

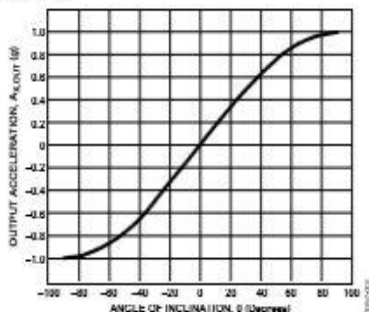


Figure 2. Output Acceleration vs. Angle of Inclination for Single-Axis Inclination Sensing

Because the inclination calculation is done digitally, the output acceleration is presented as a constant acceleration per least significant bit (LSB), or code, obtained either from an analog-to-digital converter (ADC) or directly from a digital output part. Because the output resolution is a constant acceleration, the resolution in degrees of inclination is variable, with the best resolution close to  $0^\circ$  and the worst resolution at  $\pm 90^\circ$ .

Figure 3 and Figure 4 show the incremental sensitivity for  $1^\circ$  and  $0.25^\circ$  inclination angle steps. The incremental sensitivity is the output change, shown in mg, per inclination angle step, or

$$S [g] = 1 g \times (\sin(N+P) - \sin(N)) \quad (2)$$

where:

$N$  is the current angle.

$P$  is the step size.

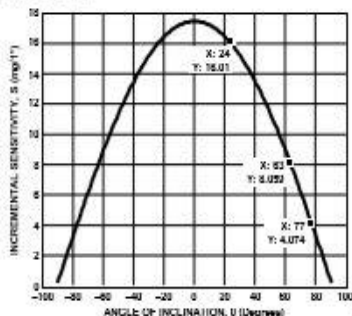


Figure 3. Incremental Inclination Sensitivity for  $1^\circ$  Steps

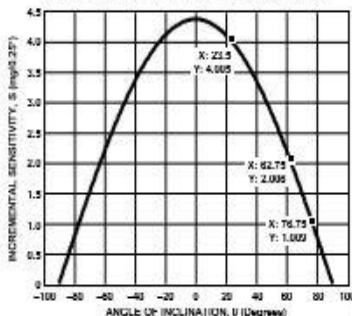


Figure 4. Incremental Inclination Sensitivity for  $0.25^\circ$  Steps

These curves can be used to determine the minimum necessary resolution when measuring the output acceleration in order to meet the desired inclination resolution for the entire range of an application. For example, designing for a maximum step size of  $1^\circ$ , a resolution of at least 8 mg/LSB is necessary for a range of  $\pm 63^\circ$ . Similarly, to achieve a maximum step size of  $0.25^\circ$  for a range of  $\pm 63$  requires a resolution of at least 2 mg/LSB. Note that, if enough dither is present, oversampling can be used to achieve better resolution.

Because the output of the accelerometer obeys a sinusoidal relationship as it is rotated through gravity, conversion from acceleration to angle is done using the inverse sine function.

$$\theta = \sin^{-1}\left(\frac{A_{\text{LOTT}}(k)}{1g}\right) \quad (3)$$

where the inclination angle,  $\theta$ , is in radians.

If a narrow range of inclination is required, a linear approximation can be used in place of the inverse sine function. The linear approximation relates to the approximation of sine for small angles.

$$\sin(\theta) \approx \theta, \quad \theta \ll 1 \quad (4)$$

where the inclination angle,  $\theta$ , is in radians.

An additional scaling factor,  $k$ , can be included in the linear approximation for inclination angle, which allows the valid range for the approximation to be increased if the allowable error is increased.

$$\theta \approx k \times \left(\frac{A_{\text{LOTT}}(k)}{1g}\right) \quad (5)$$

where the inclination angle,  $\theta$ , is in radians.

Conversion to degrees is done by multiplying the result of Equation 5 by  $(180/\pi)$ . Figure 5 shows a comparison between using the inverse sine function and the linear approximation with  $k$  equal to 1. As the magnitude of the inclination angle increases, the linear approximation begins to fail, and the calculated angle deviates from the actual angle.

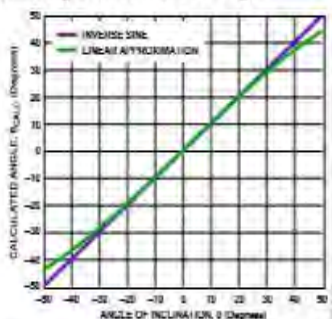


Figure 5. Comparison of Inverse Sine Function and Linear Approximation for Inclination Angle Calculation

Because the calculated angle is plotted against the actual angle of inclination, the linear approximation appears to bend near the ends. This is because the linear approximation is linear only when compared to the output acceleration and, as shown in Figure 2, the output acceleration behaves similarly as the actual angle of inclination is increased. However, the inverse sine function should produce an output that is one-to-one with the actual angle of inclination, causing the calculated angle to be a straight line when plotted against the actual angle of inclination.

As an example, if the desired resolution of inclination sensing is  $1^\circ$ , an error of  $\pm 0.5^\circ$  is acceptable because it is below the rounding error of the calculation. If the error between the actual angle of inclination and the calculated angle of inclination is plotted for  $k$  equal to 1, as shown in Figure 6, the valid range for the linear approximation is only  $\pm 20^\circ$ . If the scaling factor is adjusted such that the error is maximized but kept within the calculation rounding limits, the valid range of the linear approximation increases to greater than  $\pm 30^\circ$ .

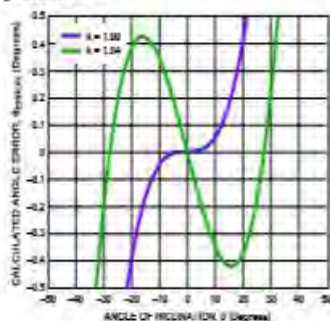


Figure 6. Calculated Angle Error for Different Scaling Factors

### Dual-Axis Tilt Calculation

One limitation of single-axis inclination sensing is the need for a high resolution ADC or digital output to achieve a large range of valid inclination angles, as shown in Figure 3 and Figure 4. Another limitation is that a single-axis measurement cannot provide a  $360^\circ$  measurement, because the acceleration generated at an inclination of  $N^\circ$  is the same as the acceleration generated at an inclination of  $180^\circ - N^\circ$ . For some applications, this is acceptable, but for applications that require higher resolution or the ability to distinguish angles of inclination in a complete  $360^\circ$  arc, a second axis, as shown in Figure 7, or a second sensor is necessary. If a second sensor is used, it should be oriented such that the sensing axis of the second sensor is orthogonal to the sensing axis of the first sensor.

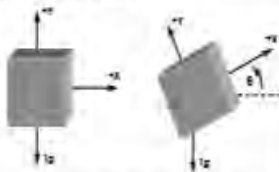


Figure 7. Two Axes Used for Tilt Sensing

There are three major benefits to including a second axis in determining the angle of inclination. These benefits are described in the following sections.

### Constant Sensitivity

The first major benefit of using a second axis is due to the orthogonality of the axes. As in the single-axis solution, the acceleration detected by the x-axis is proportional to the sine of the angle of inclination. The y-axis acceleration, due to the orthogonality, is proportional to the cosine of the angle of inclination (see Figure 8). As the incremental sensitivity of one axis is reduced, such as when the acceleration on that axis approaches  $\pm 1 g$  or  $-1 g$ , the incremental sensitivity of the other axis increases.

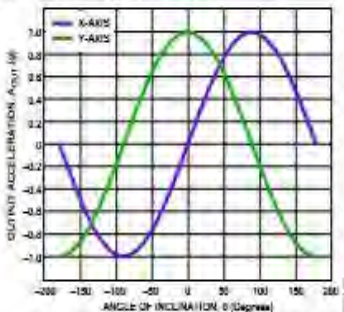


Figure 8. Output Acceleration vs. Angle of Inclination for Dual-Axis Inclination Sensing

One method to convert the measured acceleration to an inclination angle is to compute the inverse sine of the x-axis and the inverse cosine of the y-axis, similar to the single-axis solution. However, an easier and more efficient approach is to use the ratio of the two values, which results in the following:

$$\frac{A_{xOUT}}{A_{yOUT}} = \frac{1 g \times \sin(\theta)}{1 g \times \cos(\theta)} = \tan(\theta) \quad (6)$$

$$\theta = \tan^{-1}\left(\frac{A_{xOUT}}{A_{yOUT}}\right) \quad (7)$$

where the inclination angle,  $\theta$ , is in radians.

Unlike the single-axis example, using the ratio of the two axes to determine the angle of inclination makes determining an incremental sensitivity very difficult. Instead, it is more useful to determine the minimum necessary accelerometer resolution, given a desired inclination resolution. Given that the incremental sensitivity of one axis increases as the other decreases, the net result is an effective incremental sensitivity that is roughly constant. This means that the selection of an accelerometer with sufficient resolution to achieve the desired inclination step size at one angle is sufficient for all angles.

To determine the minimum necessary accelerometer resolution, Equation 6 is examined to determine where the resolution limitations are. Because the output of each axis relies on the sine or cosine of the angle of inclination, and the angle of inclination for each function is the same, the minimum resolvable angle corresponds to the minimum resolvable acceleration.

As shown in Figure 3 and Figure 4, the sine function has the greatest rate of change near  $0^\circ$ , and it can be shown that the cosine function has the least rate of change at this point. For this reason, the change in acceleration on the x-axis due to a change in inclination is recognized before a change in acceleration on the y-axis. Therefore, the resolution of the system near  $0^\circ$  depends primarily on the resolution of the x-axis. To detect an inclination change of  $P^\circ$ , the accelerometer must be able to detect a change of approximately

$$\Delta A_{OUT} [g] \approx 1 g \times \sin(P) \quad (8)$$

Figure 9 can be used to determine the minimum necessary accelerometer resolution—or maximum accelerometer scale factor—for a desired inclination step size. Note that increased accelerometer resolution corresponds with a reduction in accelerometer scale factor and with the ability to detect a smaller change in output acceleration. Therefore, when selecting an accelerometer with the appropriate resolution, the scale factor should be less than the limit shown in Figure 9 for the intended inclination step size.

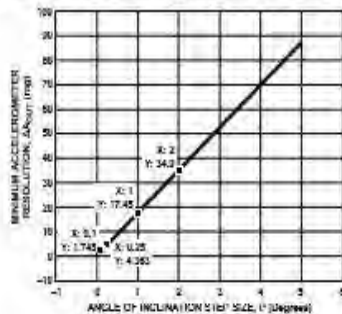


Figure 9. Minimum Accelerometer Resolution for a Desired Angle of Inclination Resolution

### Reduced Dependence on Alignment with Plane of Gravity

The second major benefit of using at least two axes is that unlike the single-axis solution, where tilt in any axis other than the x-axis can cause significant error, the use of a second axis allows for an accurate value to be measured even when inclination in the third axis is present. This is because the effective incremental sensitivity is proportional to the root-sum-square (RSS) value of gravity on the axes of interest.

When gravity is completely contained in the xy-plane, the RSS value of acceleration detected on those axes is ideally equal to  $1 g$ . If tilt is present in the xz- or yz-plane, the total acceleration due to gravity is reduced, which also reduces the effective incremental sensitivity. This, in turn, increases the inclination step size for a given accelerometer resolution, but still provides an accurate measurement. The resulting angle from the inclination calculation corresponds to the rotation in the xy plane.

If the system is tilted enough, such that very little acceleration due to gravity is present in the xy-plane, the inclination angle step size will be too coarse to be useful; therefore, it is recommended that tilt in the xz- or yz-plane be limited.

### Complete 360° Tilt Sensing

The third major benefit of using a second axis is the ability to distinguish between each quadrant and to measure angles throughout the entire 360° arc. As shown in Figure 10, each quadrant has a different combination of signs associated with the x- and y-axis acceleration.

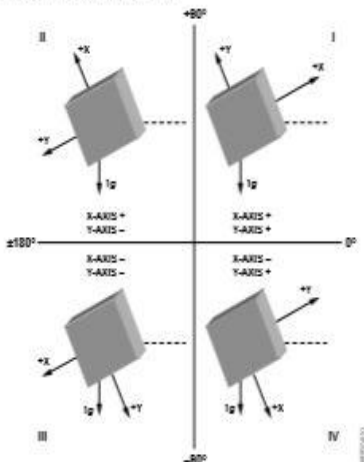


Figure 10. Angle of Inclination and Sign of Acceleration for Quadrant Detection

The inverse tangent function returns a value in Quadrant I if the operand,  $A_{X,OUT}/A_{Y,OUT}$ , is positive; if the operand is negative, the inverse tangent function returns a value in Quadrant IV. Because the operand in Quadrant II is negative, a value of 180° should be added to the result of the calculation when the angle is in that quadrant. Because the operand in Quadrant III is positive, a value of 180° should be subtracted from the result of the calculation when the angle is in that quadrant. The correct quadrant of the calculated angle can be determined by examining the sign of the measured acceleration on each axis.

### Triple-Axis Tilt Calculation

When a third axis is introduced, the orientation of the sensor can be determined in a complete sphere. The classical method of rectangular (x, y, z) to spherical ( $\rho, \theta, \phi$ ) conversion can be used to relate the angle of tilt in the xy-plane,  $\theta$ , and the angle of inclination from the gravity vector,  $\phi$ , to the measured acceleration in each axis, as follows:

$$\theta = \tan^{-1} \left( \frac{A_{X,OUT}}{A_{Y,OUT}} \right) \quad (9)$$

$$\phi = \cos^{-1} \left( \frac{A_{Z,OUT}}{\sqrt{A_{X,OUT}^2 + A_{Y,OUT}^2 + A_{Z,OUT}^2}} \right) \quad (10)$$

Given the assumption that the only measured acceleration is due to gravity, the denominator of the operand in Equation 10 can be replaced with a constant, ideally 1, because the RSS value of all the axes is constant when the only acceleration is gravity. The angles are shown in Figure 11, where Figure 11c shows  $\theta$  only in the xy-plane, and Figure 11d shows  $\phi$  as the angle between the z-axis and the gravity vector.

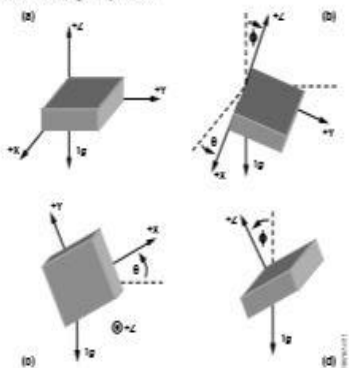


Figure 11. Angles of Spherical Coordinate System

Due to the similarities between the equations for the triple-axis method and the equations for the single- and dual-axis methods, the analysis of the triple-axis solution is the same as for the single- and dual-axis methods combined. The measurement of  $\theta$  benefits from the ratio of two orthogonal axes, and a desired inclination resolution requires a minimum accelerometer resolution as described by Equation 8.

The measurement of  $\phi$  corresponds to the measurement of the inclination angle for the single-axis solution, along with the method for determining the minimum accelerometer resolution needed for a specific inclination angle resolution over a desired range. The difference is that the use of the inverse cosine function to determine  $\phi$  results in a maximum incremental sensitivity when  $\phi$  is 90° and a minimum incremental sensitivity at 0° and 180°.

A plot similar to Figure 3 and Figure 4 can be generated by substituting cosine for sine in Equation 2. It is important to note that although  $\theta$  ranges from -180° to +180°,  $\phi$  ranges only from 0° to 180°. A negative angle for  $\phi$  causes the angle of  $\theta$  to become negative.

An alternative method for inclination sensing with three axes is to determine the angle individually for each axis of the accelerometer from a reference position. The reference position is taken as the typical orientation of a device with the x- and y-axes in the plane of the horizon (0 g field) and the z-axis orthogonal to the horizon (1 g field). This is shown in Figure 12 with  $\theta$  as the angle between the horizon and the x-axis of the accelerometer,  $\psi$  as the angle between the horizon and the y-axis of the accelerometer, and  $\phi$  as the angle between the gravity vector and the z-axis. When in the initial position of 0 g on the x- and y-axes and 1 g on the z-axis, all calculated angles would be 0°.

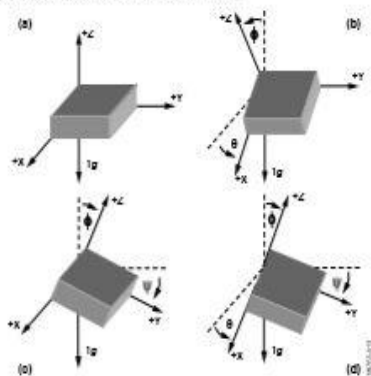


Figure 12. Angles for Independent Inclination Sensing

Basic trigonometry can be used to show that the angles of inclination can be calculated using Equation 11, Equation 12, and Equation 13.

$$\theta = \tan^{-1} \left( \frac{A_{x,OUT}}{\sqrt{A_{y,OUT}^2 + A_{z,OUT}^2}} \right) \quad (11)$$

$$\psi = \tan^{-1} \left( \frac{A_{y,OUT}}{\sqrt{A_{x,OUT}^2 + A_{z,OUT}^2}} \right) \quad (12)$$

$$\phi = \tan^{-1} \left( \frac{\sqrt{A_{x,OUT}^2 + A_{y,OUT}^2}}{A_{z,OUT}} \right) \quad (13)$$

The apparent inversion of the operand in Equation 13 is due to the initial position being a 1 g field. If the horizon is desired as the reference for the z-axis, the operand can be inverted. A positive angle means that the corresponding positive axis of the accelerometer is pointed above the horizon, whereas a negative angle means that the axis is pointed below the horizon.

Because the inverse tangent function and a ratio of accelerations is used, the benefits mentioned in the dual-axis example apply, namely that the effective incremental sensitivity is constant and that the angles can be accurately measured for all points around the unit sphere.

## CALIBRATION FOR OFFSET AND SENSITIVITY MISMATCH ERROR

The analysis in this application note was done under the assumption that an ideal accelerometer was used. This corresponds to a device with no 0 g offset and with perfect sensitivity (expressed as mV/g for an analog sensor or LSB/g for a digital sensor). Although sensors come trimmed, the devices are mechanical in nature, which means that any static stress on the part after assembly of the system may affect the offset and sensitivity. This, combined with the limits of factory calibration, can result in error beyond the allowable limits for the application.

### Effects of Offset Error

To demonstrate how large the error can be, imagine first a dual-axis solution with perfect sensitivity but with a 50 mg offset on the x-axis. At 0° the x-axis reads 50 mg and the y-axis reads 1 g. The resulting calculated angle would be 2.9°, resulting in an error of 2.9°. At ±180° the x-axis would report 50 mg, whereas the y-axis would report -1 g. This would result in a calculated angle and error of -2.9°. The error between the calculated angle and the actual angle is shown in Figure 13 for this example. The error due to an offset may not only be large compared to the desired accuracy of the system, but it can vary, thus making it difficult to simply calibrate out an error angle. This becomes more complicated when an offset for multiple axes is included.

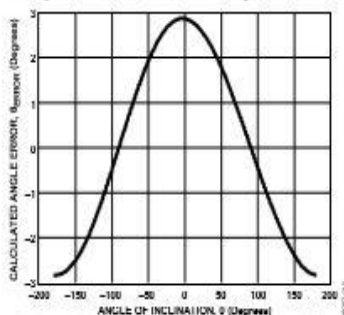


Figure 13. Calculated Angle Error Due to Accelerometer Offset

### Effects of Sensitivity Mismatch Error

The main error component due to accelerometer sensitivity in a dual-axis inclination sensing application is when a difference in sensitivity exists between the axes of interest (as opposed to a single-axis solution, where any deviation between the actual sensitivity and the expected sensitivity results in an error). Because the ratio of the x- and y-axes is used, most of the error is cancelled if the sensitivities are the same.

As an example of the effect of accelerometer sensitivity mismatch, assume that a dual-axis solution is used with perfect offset trim, perfect sensitivity on the y-axis, and +5% sensitivity on the x-axis. This means that in a 1 g field, the y-axis reports 1 g, whereas the x-axis reports 1.05 g. Figure 14 shows the error in the calculated angle due to this sensitivity mismatch. Similar to offset error, the error due to accelerometer sensitivity mismatch varies over the entire range of rotation, making it difficult to compensate for the error after calculation of the inclination angle. Skewing the mismatch further by varying the y-axis sensitivity results in even greater error.

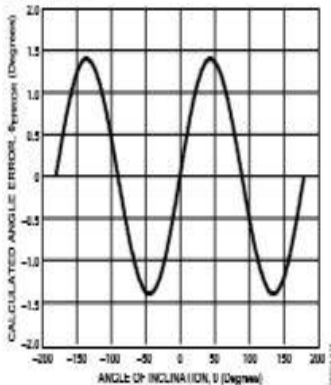


Figure 14. Calculated Angle Error Due to Accelerometer Sensitivity Mismatch

### Basic Calibration Techniques

When the errors due to offset and sensitivity mismatch are combined, the error can become quite large and well beyond the acceptable limits in an inclination sensing application. To reduce this error, the offset and sensitivity should be calibrated and the calibrated output acceleration used to calculate the angle of inclination. When including the effects of offset and sensitivity, the accelerometer output is as follows:

$$A_{OUT}[g] = A_{OFF} + (Gain \times A_{ACTUAL}) \quad (14)$$

where:

$A_{OFF}$  is the offset error, in g.

$Gain$  is the gain of the accelerometer, ideally a value of 1.

$A_{ACTUAL}$  is the real acceleration acting on the accelerometer and the desired value, in g.

A simple calibration method is to assume that the gain is 1 and to measure the offset. This calibration then limits the accuracy of the system to the uncalibrated error in sensitivity. The simple calibration method can be done by placing the axis of interest into a 0 g field and measuring the output, which would be equal to the offset. That value should then be subtracted from the output of the accelerometer before processing the signal. This is often referred to as a no-turn or single-point calibration, because the typical orientation of a device puts the x- and y-axes in a 0 g field. If a three-axis device is used, at least one turn or a second point should be included for the z-axis.

A more accurate calibration method is to use two points per axis of interest (up to six points for a three-axis design). When an axis is placed into a +1 g and -1 g field, the measured outputs are as follows:

$$A_{+1g}[g] = A_{OFF} + (1g \times Gain) \quad (15)$$

$$A_{-1g}[g] = A_{OFF} - (1g \times Gain) \quad (16)$$

where the offset,  $A_{OFF}$ , is in g.

These two points can be used to determine the offset and gain as follows:

$$A_{OFF}[g] = 0.5 \times (A_{+1g} + A_{-1g}) \quad (17)$$

$$Gain = 0.5 \times \left( \frac{A_{+1g} - A_{-1g}}{1g} \right) \quad (18)$$

where the +1 g and -1 g measurements,  $A_{+1g}$  and  $A_{-1g}$ , are in g.

This type of calibration also helps to minimize cross-axis sensitivity effects as the orthogonal axes are in a 0 g field when making the measurements for the axis of interest. These values would be used by first subtracting the offset from the accelerometer measurement and then dividing the result by the gain.

$$A_{ACTUAL}[g] = \frac{A_{OUT} - A_{OFF}}{Gain} \quad (19)$$

where  $A_{OUT}$  and  $A_{OFF}$  are in g.

The calculations of  $A_{OFF}$  and  $Gain$  in Equation 15 through Equation 19 assume that the acceleration values,  $A_{+1g}$  and  $A_{-1g}$ , are in g. If acceleration in mg is used, the calculation of  $A_{OFF}$  in Equation 17 remains unchanged, but the calculation of  $Gain$  in Equation 18 should be divided by 1000 to account for the change in units.

### 3. Datasheet Bluetooth HC-06

## HC Serial Bluetooth Products User Instructional Manual

### 1 Introduction

HC serial Bluetooth products consist of Bluetooth serial interface module and Bluetooth adapter, such as:

(1) Bluetooth serial interface module:

Industrial level: HC-03, HC-04(HC-04-M, HC-04-S)

Civil level: HC-05, HC-06(HC-06-M, HC-06-S)

HC-05-D, HC-06-D (with baseboard, for test and evaluation)

(2) Bluetooth adapter:

HC-M4

HC-M6

This document mainly introduces Bluetooth serial module. Bluetooth serial module is used for converting serial port to Bluetooth. These modules have two modes: master and slaver device. The device named after even number is defined to be master or slaver when out of factory and can't be changed to the other mode. But for the device named after odd number, users can set the work mode (master or slaver) of the device by AT commands.

HC-04 specifically includes:

Master device: HC-04-M, M=master

Slave device: HC-04-S, S=slaver

The default situation of HC-04 is slave mode. If you need master mode, please state it clearly or place an order for HC-04-M directly. The naming rule of HC-06 is same.

When HC-03 and HC-05 are out of factory, one part of parameters are set for activating the device. The work mode is not set, since user can set the mode of HC-03, HC-05 as they want.

The main function of Bluetooth serial module is replacing the serial port line, such as:

I. There are two MCUs want to communicate with each other. One connects to Bluetooth master device while the other one connects to slave device. Their connection can be built once the pair is made. This Bluetooth connection is equivalently liked to a serial port line connection including RXD, TXD

fluctuant in the range of 30-40mA. The mean current is about 25mA. After paring, no matter processing communication or not, the current is 8mA. There is no sleep mode. This parameter is same for all the Bluetooth modules.	fluctuant in the range of 30-40 m. The mean current is about 25mA. After paring, no matter processing communication or not, the current is 8mA. There is no sleep mode. This parameter is same for all the Bluetooth modules.
Reset: PIN11, active if it's input low level. It can be suspended in using.	Reset: PIN11, active if it's input low level. It can be suspended in using.
Level: Civil	Level: Civil

The table above that includes main parameters of two serial modules is a reference for user selection.

HC-03/HC-05 serial product is recommended.

### 3. Information of Package

The PIN definitions of HC-03, HC-04, HC-05 and HC-06 are kind of different, but the package size is the same: 28mm \* 15mm \* 2.35mm.

The following figure 1 is a picture of HC-06 and its main PINs. Figure 2 is a picture of HC-05 and its main PINs. Figure 3 is a comparative picture with one coin. Figure 4 is their package size information. When user designs the circuit, you can visit the website of Guangzhou HC Information Technology Co., Ltd. ([www.wavesen.com](http://www.wavesen.com)) to download the package library of protle version.



Figure 1 HC-06



Figure 2 HC-05





Figure 3 Comparative picture with one coin

LINOR BLUE I  
www.linor.com

LV-8C-2.0

单位: mm

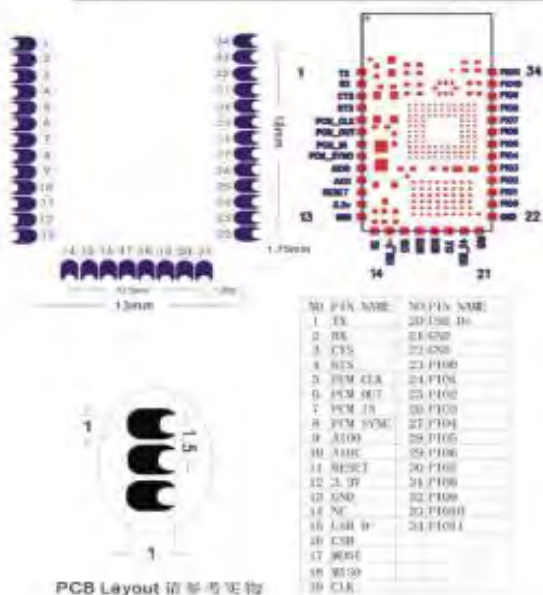


Figure 4 Package size information

#### 4. The Using and Testing Method of HC-06 for the First Time

This chapter will introduce the using method of HC-06 in detail. User can test the module according to this chapter when he or she uses the module at the first time.

PINs description:

PIN1	UART_TXD , TTL/CMOS level, UART Data output
PIN2	UART_RXD, TTL/COMS level, s UART Data input
PIN11	RESET, the reset PIN of module, inputting low level can reset the module, when the module is in using, this PIN can connect to air.
PIN12	VCC, voltage supply for logic, the standard voltage is 3.3V, and can work at 3.0-4.2V
PIN13	GND
PIN22	GND
PIN24	LED, working mode indicator Slave device: Before paired, this PIN outputs the period of 102ms square wave. After paired, this PIN outputs high level. Master device: On the condition of having no memory of pairing with a slave device, this PIN outputs the period of 110ms square wave. On the condition of having the memory of pairing with a slave device, this PIN outputs the period of 750ms square wave. After paired, this PIN outputs high level.
PIN26	For master device, this PIN is used for emptying information about pairing. After emptying, master device will search slaver randomly, then remember the address of the new got slave device. In the next power on, master device will only search this address.

(1) The circuit 1 (connect the module to 3.3V serial port of MCU) is showed by figure 5.

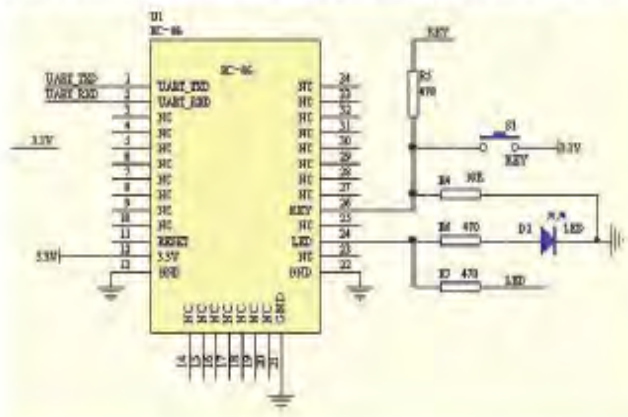


Figure 5 The circuit 1

In principle, HC-06 can work when UART\_TXD, UART\_RXD, VCC and GND are connected. However, for better testing results, connecting LED and KEY are recommended (when testing the master).

Where, the 3.3V TXD of MCU connects to HC-06's UART\_RXD, the 3.3V RXD of MCU connects to HC-06's UART\_TXD, and 3.3V power and GND should be connected. Then the minimum system is finished.

Note that, the PIN2:UART\_RXD of Bluetooth module has no pull-up resistor. If the MCU TXD doesn't have pull-up function, then user should add a pull-up resistor to the UART\_RXD. It may be easy to be ignored.

If there are two MCU which connect to master and slave device respectively, then before paired(LED will flicker) user can send AT commands by serial port when the system is power on. Please refer to HC-04 and HC-06's data sheet for detailed commands. In the last chapter, the command set will be introduced. Please pay attention to that the command of HC-04/HC-06 doesn't have terminator. For example, consider the call command, sending out AT is already enough, need not add the CRLF (carriage return line feed).

If the LED is constant lighting, it indicates the pairing is finished. The two MCUs can communicate with each other by serial port. User can think there is a serial port line between two MCUs.

(2) The circuit 2 (connect the module to 5V serial port of MCU) is showed by figure 6.

Figure 6 is the block diagram of Bluetooth baseboard. This kind of circuit can amplify Bluetooth module's operating voltage to 3.1-6.5V. In this diagram, the J1 port can not only be connected with MCU system of 3.3V and 5V, but also can be connected with computer serial port.

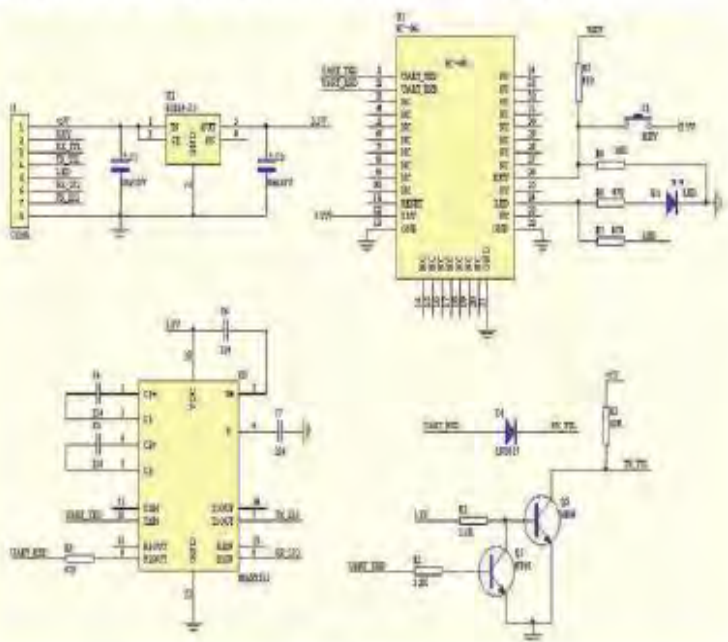


Figure 6 The circuit 2

### (3) AT command test

Before paired, the mode of HC-04 and HC-06 are AT mode.

On the condition of 9600N81, OK will be received when user send the two letters AT. Please refer to the last chapter of datasheet for other commands of HC-06. Please pay attention to that sending out AT is already enough, need not add the CRLF (carriage return line feed).

The command set of Version V1.4 doesn't include parity. The version V1.5 and its later version have parity function. Moreover, there are three more commands of V1.5 than V1.4. They are:

No parity (default)      AT+PN

### (7) Name introduction

If the device has no name, it's better that user doesn't try to change the master device name. The name should be limited in 20 characters.

Summary: The character of HC-06: 1 not many command 2 easy for application 3 low price. It's good for some specific application. HC-04 is very similar with HC-06. Their only one difference is HC-04 is for industry, HC-06 is for civil. Except this, they don't have difference.

The following reference about HC-04 and HC-06 can be downloaded from company website [www.wavesin.com](http://www.wavesin.com):

HC-06 datasheet.pdf	(the command set introduction is included)
HC-04 datasheet.pdf	(the command set introduction is included)
IVT BlueSoleil-2.6	(IVT Bluetooth drive test version)
Bluetooth FAQ.pdf	
HC-04-D(HD-06-D)datasheet(English).pdf	
HC-06-AT command software (test version)	(some commands in V1.5 is not supported by V1.4)
PCB package of Bluetooth key modules	(PCB package lib in protel)
IVT software manual.pdf	(introduce how to operate the module and make pair with Bluetooth module)
PDA serial test helper.exe	(serial helper used for WM system)

### 5 manual for the first use of HC-05

This chapter will introduce how to test and use the HC-05 if it's the first time for user to operate it.

#### (1) PINs description

PIN1	UART_TXD, Bluetooth serial signal sending PIN, can connect with MCU's RXD PIN
PIN2	UART_RXD, Bluetooth serial signal receiving PIN, can connect with the MCU's TXD PIN, there is no pull-up resistor in this PIN. But it needs to be added an external pull-up resistor.
PIN11	RESET, the reset PIN of module, inputting low level can reset the module, when the module is in using, this PIN can connect to air.
PIN12	VCC, voltage supply for logic, the standard voltage is 3.3V, and can work at 3.0-4.2V
PIN13	GND

PIN31	<p>LED1, indicator of work mode. Has 3 modes:</p> <p>When the module is supplied power and PIN34 is input high level, PIN31 output 1Hz square wave to make the LED flicker slowly. It indicates that the module is at the AT mode, and the baud rate is 38400;</p> <p>When the module is supplied power and PIN34 is input low level, PIN31 output 2Hz square wave to make the LED flicker quickly. It indicates the module is at the pairable mode. If PIN34 is input high level, then the module will enter to AT mode, but the output of PIN31 is still 2Hz square wave.</p> <p>After the pairing, PIN31 output 2Hz square wave.</p> <p>Note: if PIN34 keep high level, all the commands in the AT command set can be in application. Otherwise, if just excite PIN34 with high level but not keep, only some command can be used. More information has provided at chapter 2.</p>
PIN32	<p>Output terminal. Before paired, it output low level. Once the pair is finished, it output high level.</p>
PIN34	<p>Mode switch input. If it is input low level, the module is at paired or communication mode. If it's input high level, the module will enter to AT mode. Even though the module is at communication, the module can enter to the AT mode if PIN34 is input high level. Then it will go back to the communication mode if PIN34 is input low level again.</p>

(2) Application circuit 1 (connect to the 3.3V system)

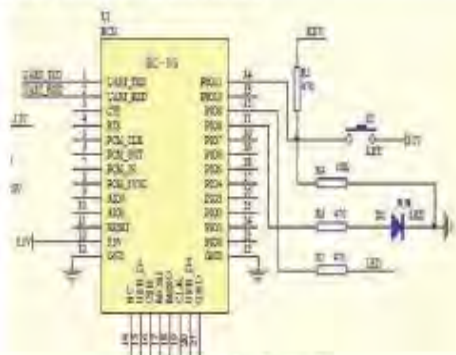


Figure 7 Application 1

-----Halaman ini sengaja dikosongkan-----

## DAFTAR RIWAYAT HIDUP



Nama : Ghinayan Dwi Fikriyuda  
TTL : Gresik, 26 Juli 1994  
Jenis Kelamin : Laki-laki  
Agama : Islam  
Alamat : Jalan Veteran Vc/no. 12  
Gresik  
Telp/HP : 082232898952  
E-mail : ghinayandwi@gmail.com

### RIWAYAT PENDIDIKAN

1. 2000 – 2006 : SD Negeri 2 Sidokumpul Gresik
2. 2006 – 2009 : SMP Negeri 1 Gresik
3. 2009 – 2012 : SMA Negeri 1 Gresik
4. 2012 – 2015 : D3 Teknik Elektro, Program studi Teknik Elektro Industri - FTI Institut Teknologi Sepuluh Nopember (ITS)

### PENGALAMAN KERJA

1. Kerja Praktek di PT. PJB Unit Pembangkitan Gresik

### PENGALAMAN ORGANISASI

1. Staff Dewan Perwakilan Mahasiswa ITS Periode 2013/2014
2. Staff Dewan Perwakilan Mahasiswa Jurusan D3 Teknik Elektro Periode 2013/2014
3. Kabiro Komunitas Divisi Minat dan Bakat Periode 2014/2015 HIMAD3TEKTRO, FTI – ITS



-----Halaman ini sengaja dikosongkan-----

## DAFTAR RIWAYAT HIDUP



Nama : M Ibadur Rachman  
TTL : Gresik, 19 Januari 1994  
Kelamin: Laki - Laki  
Agama : Islam  
Alamat : Jalan Melati RT/RW.01/01 Desa  
Racitengah Kec.Sidayu  
Kab.Gresik  
Telp/HP : 085748962629  
E-mail : santriedan94@gmail.com

### RIWAYAT PENDIDIKAN

1. 2000 – 2006 : SD Negeri Racitengah Gresik
2. 2006 – 2009 : SMP Plus Ar rahmat Bojonegoro
3. 2009 – 2012 : MAN 3 Malang
4. 2012 – 2015 : D3 Teknik Elektro, Program studi Teknik Elektro Industri - FTI Institut Teknologi Sepuluh Nopember (ITS)

### PENGALAMAN KERJA

1. Kerja Praktek di PT. PJB Unit Pembangkitan Gresik

### PENGALAMAN ORGANISASI

1. Staff Dept Hublu HIMAD3TEKTRO, FTI – ITS
2. Pengurus LMB Badminton HIMAD3TEKRO, FTI - ITS

-----Halaman ini sengaja dikosongkan-----



Arek ITS  
*cak!*  
Cerdas, Amanah, Kreatif

# MONITORING KEMIRINGAN BANGUNAN MENGGUNAKAN *ACCELERO* DENGAN MEDIA KOMUNIKASI *BLUETOOTH* DAN *WIFI*



**Ghinayan Dwi Fikriyuda**  
**2212039015**

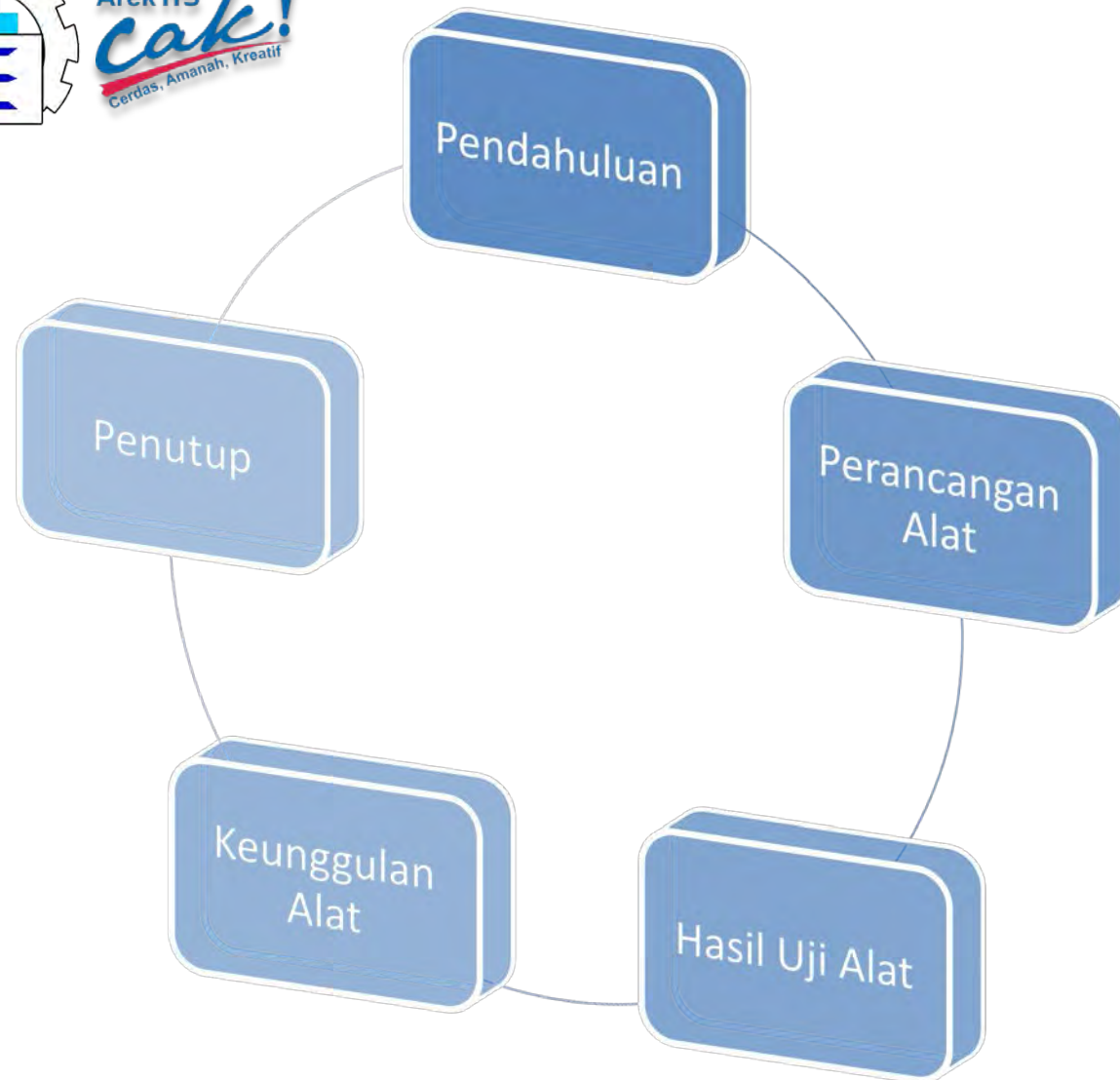


**M. Ibadur Rachman**  
**2212039039**

DIBIMBING OLEH :  
SUWITO, ST., MT.  
ONIE MEIYANTO, S.Pd.



Arek ITS  
**cak!**  
Cerdas, Amanah, Kreatif





Arek ITS  
*cak!*  
Cerdas, Amanah, Kreatif

# PENDAHULUAN

# LATAR BELAKANG

**KURANG WASPADA**



**KONTUR TANAH TIDAK RATA**



**BANYAK GEDUNG**



**TEKNOLOGI YANG MAHAL**



## BATASAN MASALAH

- Sudut kemiringan bangunan
- Sensor *accelerometer* digunakan untuk mendeteksi kemiringan
- Hanya dilakukan monitoring





## MAKSUD dan TUJUAN

- Membuat pendeteksi kemiringan bangunan atau gedung dengan sistem yang lebih mudah untuk digunakan oleh masyarakat umum dengan harga yang terjangkau
- Mencegah terjadinya kejadian gedung atau bangunan yang roboh khususnya di Surabaya.





Arek ITS  
*cak!*  
Cerdas, Amanah, Kreatif

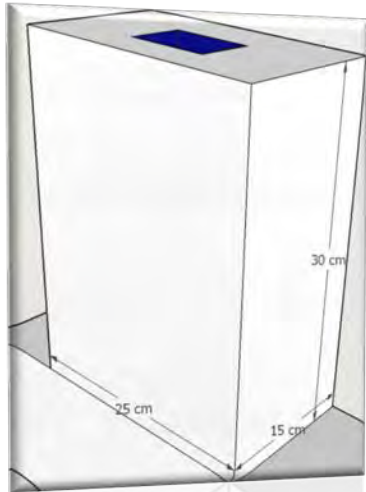
# PERANCANGAN



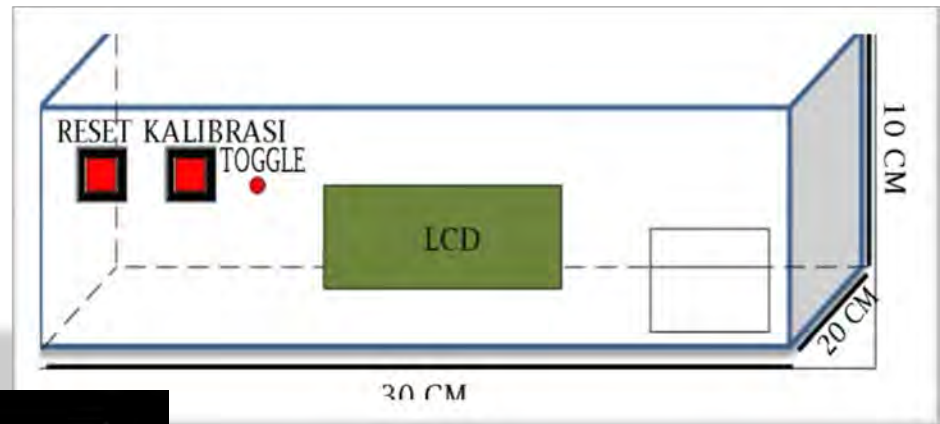
# Perancangan Sistem:



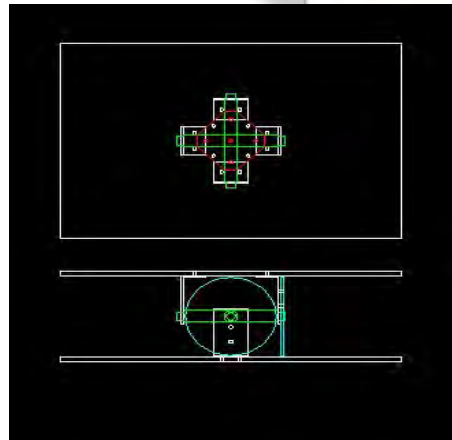
# Perancangan Mekanik



**MINIATUR  
GEDUNG**



**BOX ELEKTRONIK**



**PROTOTIPE INDIKATOR KEMIRINGAN**



# Perancangan *Hardware*



**POWER SUPPLY**



**ARDUINO UNO**



**ETHERNET SHIELD**



**LCD 16x4**



**SENSOR  
ACCELEROMETER  
ADXL345**



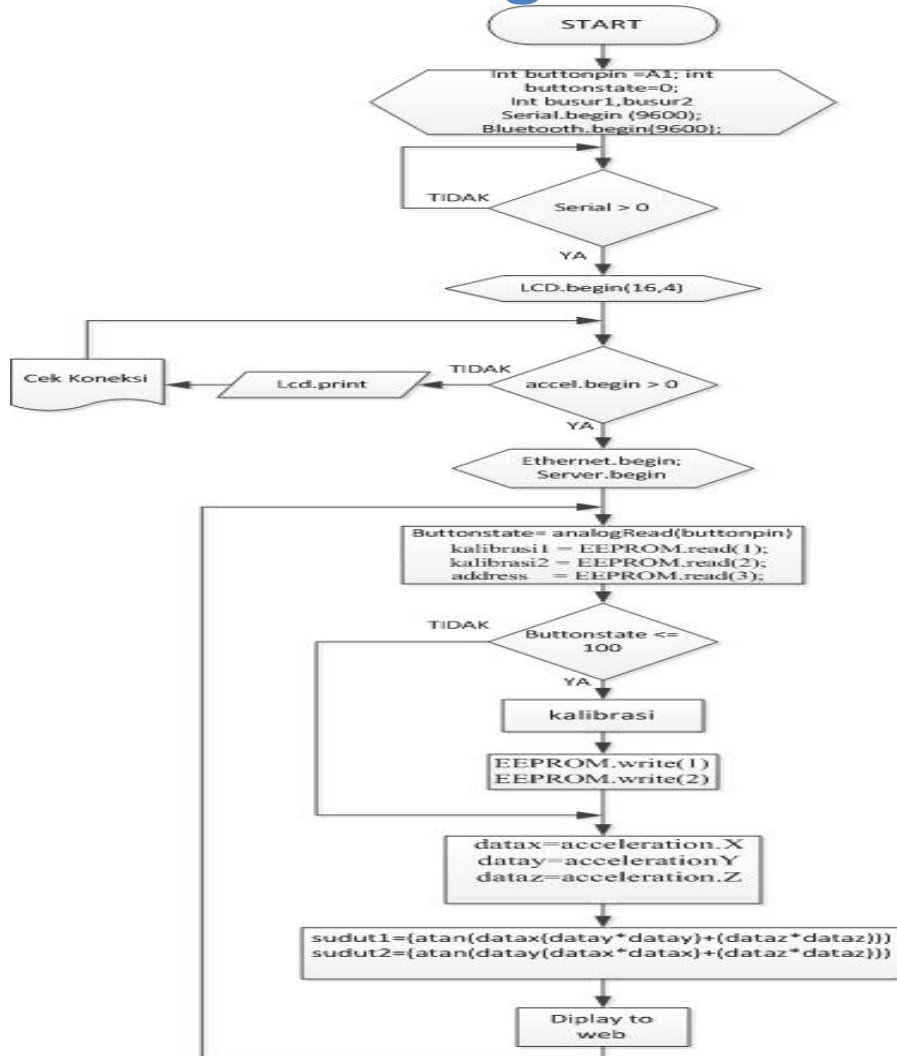
**BLUETOOTH HC-  
06**



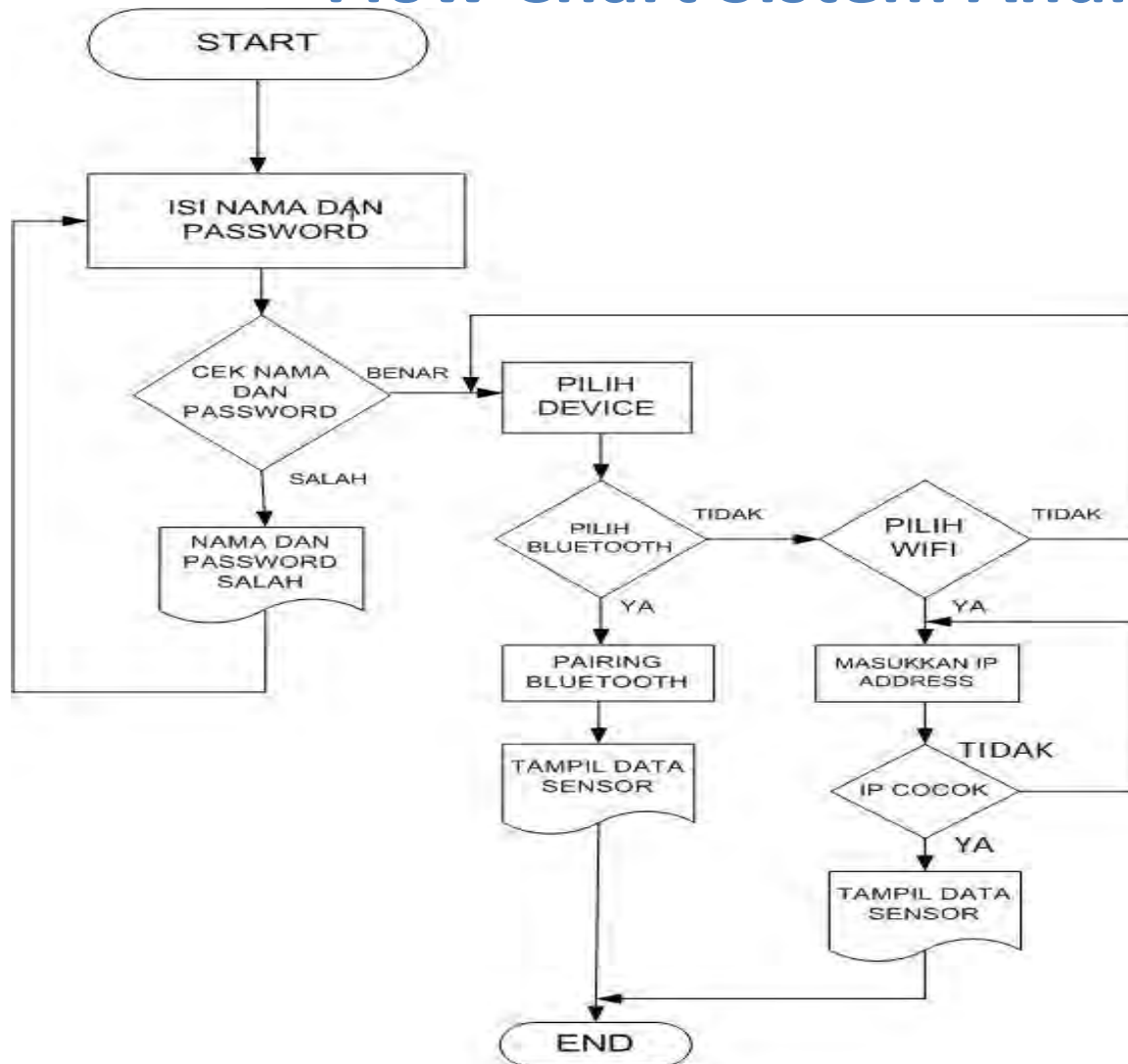
**ROUTER**



# Flow Chart Program Arduino



# Flow Chart Sistem Android





Arek ITS  
*cak!*  
Cerdas, Amanah, Kreatif

# HASIL UJI ALAT



# Hasil Uji Alat

POSISI	DATA SUDUT 1	
	Derajat Busur	LCD
1	90°	0°
2	85°	-5°
3	80°	-10°
4	75°	-15°
5	70°	-20°
6.	65°	-25°
7.	95°	5°
8.	100°	10°
9.	105°	15°
10.	110°	20°
11.	115 °	25 °

POSISI	DATA SUDUT 2	
	Derajat Busur	LCD
1	90°	0°
2	85°	-5°
3	80°	-10°
4	75°	-15°
5	70°	-20°
6.	65°	-25°
7.	95°	5°
8.	100°	10°
9.	105°	15°
10.	110°	20°
11.	115 °	25 °



Arek ITS  
*cak!*  
Cerdas, Amanah, Kreatif

# KEUNGGULAN ALAT

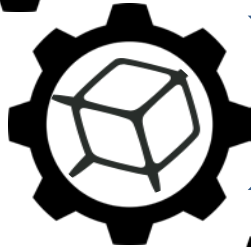
## Keunggulan Alat:



Sistem pengoperasiannya sederhana



Low Cost Budget



Dapat diakses melalui *smartphone*



Digunakan dengan *bluetooth* atau *wifi*





Arek ITS  
*cak!*  
Cerdas, Amanah, Kreatif

# PENUTUP

## Kesimpulan:

Dalam pembuatan alat ini dapat di simpulkan bahwa :

- Rancang purwarupa pada Tugas Akhir monitoring kemiringankali ini memiliki rata-rata kebenaran sebesar 100% sehingga *accelerometer* ini dapat digunakan sebagai sensor kemiringan yang cukup akurat.
- Komunikasi data antara *bluetooth* dan *wifi* sama-sama memiliki kelebihan dan kekurangan. Kelebihan dari *bluetooth* adalah saat *refreshing data* lebih mudah dibandingkan dengan menggunakan *wifi*, karena komunikasi *wifi* bergantung dari koneksi dari jaringan internet. Namun menggunakan *wifi* jangkauannya lebih luas daripada menggunakan *bluetooth* yang hanya dapat diakses sampai  $\pm 10$  meter.



Pendahuluan

Perancangan

Uji

Keunggulan

Penutup

# TERIMA KASIH



Arek ITS  
*cak!*  
Cerdas, Aktif, Kreatif