

PEMROGRAMAN  
ACCELEROMETER  
BERBASIS MIKROKONTROLLER ATMEGA 8535



Oleh :

Anggit Deny Setiyoko                      2010-2-004

Rangga Wisnu Premana                    2010-2-028

LAPORAN PROYEK MIKROKONREOLLER

PROGRAM STUDI TEKNIK MEKATRONIKA

JURUSAN TEKNIK MEKATRONIKA

POLITEKNIK ATMI SURAKARTA

2012

## KATA PENGANTAR

Puji dan syukur kami panjatkan ke hadirat Tuhan yang Maha Kuasa karena atas berkat, rahmat, serta karunianya yang selalu melimpah kepada kami, akhirnya kami mampu menyelesaikan proyek Mikrokontroler dengan baik.

Laporan ini sengaja kami susun untuk memberikan penjelasan lebih mengenai komponen ACCELEROMETER, serta bagaimana mengaplikasikannya ke dalam suatu mikrokontroler. Dalam laporan ini kami akan menjelaskan pemrograman ACCELEROMETER berbasis Atmega 8535.

Penyusun sadar, bahwa dalam pembuatan proyek ini, kami mendapat bantuan serta bimbingan dari berbagai pihak. Perkenankanlah pada kesempatan ini penyusun mengucapkan terimakasih kepada pihak di bawah ini :

1. Tuhan yang Maha Esa yang memberi kami kesehatan serta kekuatan dalam menyelesaikan proyek ini.
2. Mas Laras, selaku instruktur lab. Mikrokontroler yang senantiasa memberikan kami dorongan untuk segera menyelesaikan proyek ini.
3. Keluarga yang dengan setia memotivasi kami.
4. Teman-teman yang telah bersedia membantu kami.

Pada akhirnya, kami sebagai penyusun menyadari bahwa masih banyak kekurangan kami dalam membuat proyek ini, oleh karena itu, kami menerima segala bentuk kritik dan saran yang membangun, agar di proyek mendatang, kami dapat mengusahakan yang lebih baik.

Surakarta, September 2012

Penyusun

# Daftar Isi

Kata Pengantar .....	2
Daftar Isi .....	3
BAB 1 PENDAHULUAN.....	4
1.1 Latar Belakang .....	4
1.2 Batasan Masalah .....	4
1.3 Tujuan.....	4
Bab 2 LANDASAN TEORI.....	5
2.1 Perangkat Keras .....	5
Bab 3 METODE PENGKERJAAN.....	16
3.1 Prosedur Penyelesaian Masalah .....	16
3.1.1 Rangkaian hardware.....	16
Bab 4 Program AVR .....	19
Bab 5 PENUTUP .....	33
Bab 6 Daftar Pustaka.....	34

# BAB 1

## PENDAHULUAN

### 1.1 Latar Belakang

Seiring bertambahnya waktu tidak dapat dipungkiri bahwa semakin bertambah pesat pula perkembangan dalam bidang teknologi dan elektronika. Dan berkembangnya dalam peralatan elektronika tidak terlepas dari komponen-komponen listrik yang canggih. Komponen-komponen ini dirangkai sedemikian rupa sehingga membentuk suatu sistem yang menjadi alat peralatan elektronika yang sedemikian kompleks.

Dalam makalah ini, kami akan membuktikan teori tentang ACCELEROMETER dengan mengaplikasikannya ke dalam hardware. Tak lupa kami juga akan menjelaskan tentang fungsi, kelebihan, kekurangan, cara pengaplikasian ACCELEROMETER serta komponen yang mendukung untuk membuat eksplorasi dari komponen ini.

### 1.2 Batasan Masalah

Batasan masalah yang teridentifikasi dalam laporan ini:

a) **Jumlah DC motor servo**

Jumlah DC motor yang digunakan hanya 1, sebagai penggerak axis- x.

b) **LCD**

LCD hanya menampilkan parameter angka pada axis X,Y,Z dari hasil yang dikeluarkan sensor mma7361.

### 1.3 Tujuan

Tujuan yang akan dicapai dalam makalah ini adalah :

1. Membuktikan teori tentang ACCELEROMETER dengan mengaplikasikannya ke dalam suatu hardware.
2. Menerangkan ACCELEROMETER dengan menjabarkan fungsi serta cara mengaplikasikannya.
3. Sebagai panduan pembelajaran tentang ACCELEROMETER.
4. Menyelesaikan tugas mikrokontroller.

## BAB 2

### LANDASAN TEORI

#### 2.1 Dasar Teori

##### 2.1.1 Mikrokontroler Atmega8535

Mikrokontroler Atmega8535 memiliki 32 pin bipolar yang dapat diprogram sebagai input maupun output dan dilengkapi dengan berbagai macam fitur antara lain ADC,timer,interrupt,analog input,dll.

Arsitektur Atmega8535

- 4 buah PORT dengan masing – masing 8 bit (pin)
- Analog to Digital Converter 10 bit sebanyak 8 chanel (pin)
- Timer 1, timer 2, timer 3
- 32 buah register
- Watchdog timer
- SRAM sebanyak 512 byte
- Memory flash sebesar ]8 kilo Byte
- Sumber Interrupt internal dan eksternal
- SPI (Serial Peripheral Interface)
- On board EEPROM sebesar 512 byte
- Analog comparator
- Port USART (Universal Shynchronous Ashynchronous Receive Transmitter)

(XCK/T0) PB0	1	40	PA0 (ADC0)
(T1) PB1	2	39	PA1 (ADC1)
(INT2/AIN0) PB2	3	38	PA2 (ADC2)
(OC0/AIN1) PB3	4	37	PA3 (ADC3)
(SS) PB4	5	36	PA4 (ADC4)
(MOSI) PB5	6	35	PA5 (ADC5)
(MISO) PB6	7	34	PA6 (ADC6)
(SCK) PB7	8	33	PA7 (ADC7)
RESET	9	32	AREF
VCC	10	31	GND
GND	11	30	AVCC
XTAL2	12	29	PC7 (TOSC2)
XTAL1	13	28	PC6 (TOSC1)
(RXD) PD0	14	27	PC5
(TXD) PD1	15	26	PC4
(INT0) PD2	16	25	PC3
(INT1) PD3	17	24	PC2
(OC1B) PD4	18	23	PC1 (SDA)
(OC1A) PD5	19	22	PC0 (SCL)
(ICP1) PD6	20	21	PD7 (OC2)

### 2.1.2 Motor Servo

Motor servo adalah sebuah motor dengan sistem closed feedback di mana posisi dari motor akan diinformasikan kembali ke rangkaian kontrol yang ada di dalam motor servo. Motor ini terdiri dari sebuah motor, serangkaian gear, potensiometer dan rangkaian kontrol. Potensiometer berfungsi untuk menentukan batas sudut dari putaran servo. Sedangkan sudut dari sumbu motor servo diatur berdasarkan lebar pulsa yang dikirim melalui kaki sinyal dari kabel motor. Tampak pada gambar dengan pulsa 1.5 mS pada periode selebar 2 mS maka sudut dari sumbu motor akan berada pada posisi tengah. Semakin lebar pulsa ON maka akan semakin besar gerakan sumbu ke arah jarum jam dan semakin kecil pulsa ON maka akan semakin besar gerakan sumbu ke arah yang berlawanan dengan jarum jam.

Motor servo biasanya hanya bergerak mencapai sudut tertentu saja dan tidak kontinu seperti motor DC maupun motor stepper. Walau demikian, untuk beberapa keperluan tertentu, motor servo dapat dimodifikasi agar bergerak kontinu. Pada robot, motor ini sering digunakan untuk bagian kaki, lengan atau bagianbagian lain yang mempunyai gerakan terbatas dan membutuhkan torsi cukup besar.

Motor servo adalah motor yang mampu bekerja dua arah (CW dan CCW) dimana arah dan sudut pergerakan rotornya dapat dikendalikan hanya dengan memberikan pengaturan duty cycle sinyal PWM pada bagian pin kontrolnya. Motor Servo tampak pada gambar 1.



**Gambar 1.** Motor Servo

Motor Servo merupakan sebuah motor DC yang memiliki rangkaian control elektronik dan internal gear untuk mengendalikan pergerakan dan sudut sudutnya. Sistem Mekanik Motor Servo tampak pada gambar 2.



**Gambar 2.** Sistem Mekanik Motor Servo

Motor servo adalah motor yang berputar lambat, dimana biasanya ditunjukkan oleh rate putarannya yang lambat, namun demikian memiliki torsi yang kuat karena internal gearnya.

Lebih dalam dapat digambarkan bahwa sebuah motor servo memiliki :

- 3 jalur kabel : power, ground, dan control
- Sinyal control mengendalikan posisi

- Operasional dari servo motor dikendalikan oleh sebuah pulsa selebar  $\pm 20$  ms, dimana lebar pulsa antara 0.5 ms dan 2 ms menyatakan akhir dari range sudut maksimum.
- Konstruksi didalamnya meliputi internal gear, potensiometer, dan feedback control.

### **JENIS-JENIS MOTOR SERVO**

- Motor Servo Standar 180°

Motor servo jenis ini hanya mampu bergerak dua arah (CW dan CCW) dengan defleksi masing-masing sudut mencapai 90° sehingga total defleksi sudut dari kanan – tengah – kiri adalah 180°.

- Motor Servo Continuous

Motor servo jenis ini mampu bergerak dua arah (CW dan CCW) tanpa batasan defleksi sudut putar (dapat berputar secara kontinyu).

### **KEGUNAAN MOTOR SERVO**

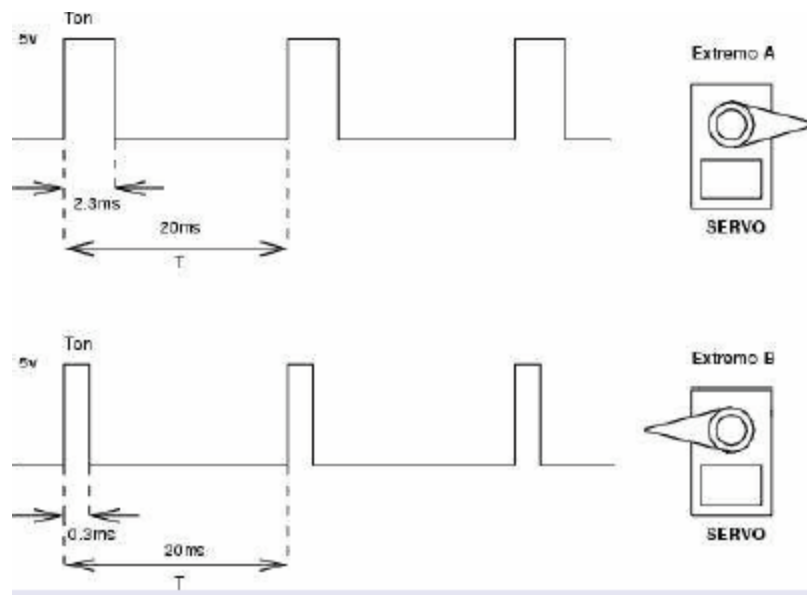
#### **Kebanyakan motor servo digunakan sebagai :**

- Manipulators.
- Moving camera's.
- Robot arms.



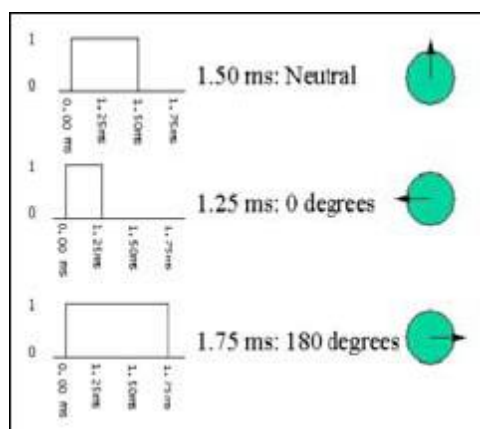
## PENSINYALAN MOTOR SERVO

Mode pensinyalan motor servo tampak pada gambar 3.



**Gambar 3.** Pensinyalan Motor Servo

Contoh dimana bila diberikan pulsa dengan besar 1.5ms mencapai gerakan 90 derajat, maka bila kita berikan data kurang dari 1.5 ms maka posisi mendekati 0 derajat dan bila kita berikan data lebih dari 1.5 ms maka posisi mendekati 180 derajat. Contoh Posisi dan Waktu Pemberian Pulsa tampak pada gambar 4.



**Gambar 4.** Contoh Posisi dan Waktu Pemberian Pulsa

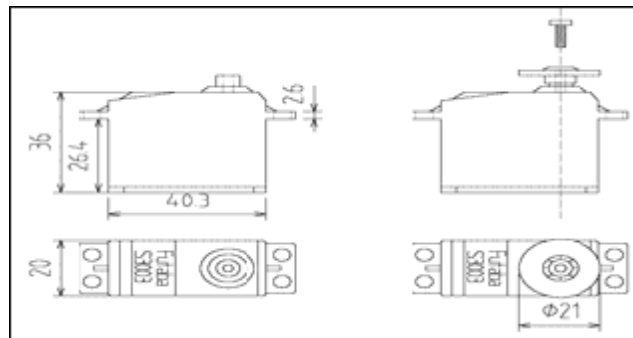
- Motor Servo akan bekerja secara baik jika pada bagian pin kontrolnya diberikan sinyal PWM dengan frekuensi 50Hz.
- Dimana pada saat sinyal dengan frekuensi 50Hz tersebut dicapai pada kondisi Ton duty cycle 1.5ms, maka rotor dari motor akan berhenti tepat di tengah-tengah (sudut 0° / netral).
- Pada saat Ton duty cycle dari sinyal yang diberikan kurang dari 1.5ms, maka rotor akan berputar ke arah kiri dengan membentuk sudut yang besarnya linier terhadap besarnya Ton duty cycle, dan akan bertahan diposisi tersebut.
- Dan sebaliknya, jika Ton duty cycle dari sinyal yang diberikan lebih dari 1.5ms, maka rotor akan berputar ke arah kanan dengan membentuk sudut yang linier pula terhadap besarnya Ton duty cycle, dan bertahan diposisi tersebut.

Bentuk-Bentuk Motor Servo tampak pada gambar 5.

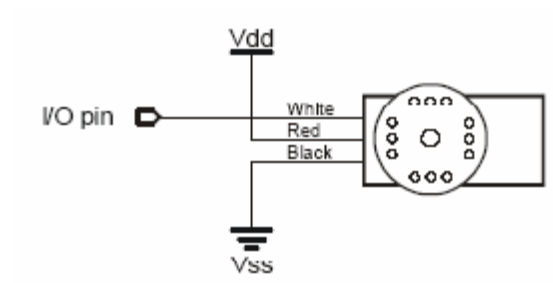


**Gambar 5.** Bentuk-Bentuk Motor Servo

Dimensi Motor Servo tampak pada gambar 6.



**Gambar 6.** Dimensi Motor Servo



**Gambar 7.** Pin-Pin dan Pengkabelan Pada Motor Servo

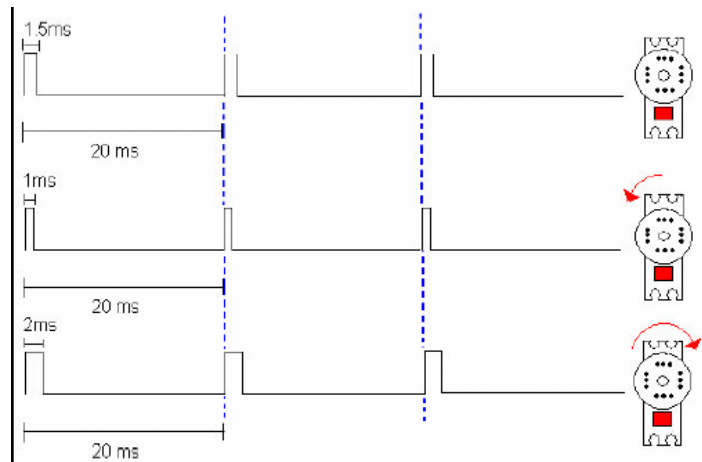
## PENGENDALIAN MOTOR SERVO

### Pengendalian sebuah motor servo dengan menggunakan mikrokontroler :

Ilustrasi pengendalian motor servo tampak pada gambar 8.

#### **Driver motor DC Servo**

Bentuk motor servo dapat dilihat pada gambar 5. Terdapat tiga utas kabel dengan warnamerah, hitam, dan kuning. Kabel merah dan hitam harus dihubungkan dengan sumber tegangan4-6 volt dc agar motor servo dapat bekerja normal. Sedangkan kabel berwarna kuning adalahkabel data yang dipakai untuk mengatur arah gerak dan posisi servo. Pergerakan motor servo terhadap perubahan lebar pulsa tampak pada gambar 8.



**Gambar 8.** Pergerakan motor servo terhadap perubahan lebar pulsa

### 2.1.3 LCD 2x16



LCD (*Liquid Crystal Display*) adalah modul penampil yang banyak digunakan karena tampilannya menarik. LCD yang paling banyak digunakan saat ini ialah tipe M1632 karena harganya cukup murah. LCD M1632 merupakan modul LCD dengan tampilan 2x16 (2 baris x 16 kolom) dengan konsumsi daya rendah. Modul tersebut dilengkapi dengan mikrokontroler yang didesain khusus untuk mengendalikan LCD. Untuk rangkaian interfacing, LCD tidak banyak memerlukan komponen pendukung. Hanya diperlukan satu variable resistor untuk memberi tegangan kontras pada matriks LCD.

Dengan menggunakan CodeVision AVR, pemrograman untuk menampilkan karakter atau string ke LCD sangat mudah karena didukung library yang telah disediakan oleh CodeVision AVR itu. Kita tidak harus memahami karakteristik LCD secara mendalam, perintah tulis dan inisialisasi sudah disediakan oleh library dari CodeVision AVR.

#### 2.1.4 K-125 Downloader

K-125 downloader merupakan alat yang digunakan untuk mendownload program yang telah dibuat ke dalam chip mikrokontroler. Alat ini mengkomunikasikan data secara parallel dengan menggunakan bantuan mikrokontroler berisi ATtiny2313, resistor, dua buah Kristal (Xtal), dan sebuah kapasitor. Secara sederhana bekerja dengan cara memetakan logika tinggi bahasa C menjadi bahasa HEX yang kemudian disimpan sebagai instruksi di dalam Flash mikrokontroler sesuai dengan internal clock rate dari mikro.

K-125 downloader juga dilengkapi dengan fitur komunikasi serial yang memungkinkan mikrokontroler untuk berkomunikasi dengan PC lain melalui hubungan serial (USART) via downloader ini.

#### 2.1.5 Accelerometer

*Accelerometer* adalah sebuah transduser yang berfungsi untuk mengukur percepatan, mendeteksi dan mengukur getaran, ataupun untuk mengukur percepatan akibat gravitasi bumi. *Accelerometer* juga dapat digunakan untuk mengukur getaran yang terjadi pada kendaraan, bangunan, mesin, dan juga bisa digunakan untuk mengukur getaran yang terjadi di dalam bumi, getaran mesin, jarak yang dinamis, dan kecepatan dengan ataupun tanpa pengaruh gravitasi bumi.

Prinsip kerja dari transduser ini berdasarkan hukum fisika bahwa apabila suatu konduktor digerakkan melalui suatu medan magnet, atau jika suatu medan magnet digerakkan melalui suatu konduktor, maka akan timbul suatu tegangan induksi pada konduktor tersebut. *Accelerometer* yang diletakan di permukaan bumi dapat mendeteksi percepatan 1g (ukuran gravitasi bumi) pada titik vertikalnya, untuk percepatan yang dikarenakan oleh pergerakan horizontal maka *accelerometer* akan mengukur percepatannya secara langsung ketika bergerak secara horizontal. Hal ini sesuai dengan tipe dan jenis sensor *accelerometer* yang digunakan karena setiap jenis sensor berbeda-beda sesuai dengan spesifikasi yang dikeluarkan oleh perusahaan pembuatnya. Saat ini hampir semua sensor/transduser *accelerometer* sudah dalam bentuk digital (bukan dengan sistem mekanik) sehingga cara kerjanya hanya berdasarkan temperatur yang diolah secara digital dalam satu chip.

Berikut ini adalah gambaran bagaimana proses accelerometer analog (dengan sistem mekanik maupun digital) bekerja;

- Accelerometer digital yang bekerja berdasarkan temperatur
- Accelerometer analog yang bekerja berdasarkan sistem mekanik

Tipe Accelerometer

**Capacitive:** lempengan metal pada sensor memproduksi sejumlah kapasitansi, perubahan kapasitansi akan mempengaruhi percepatan.

**Piezoelectric:** kristal piezoelectric yang terdapat pada accelerometer jenis ini mengeluarkan tegangan yang selanjutnya dikonversi menjadi percepatan.

**Piezoresistive:** lempengan yang secara resistan akan berubah sesuai dengan perubahan percepatan.

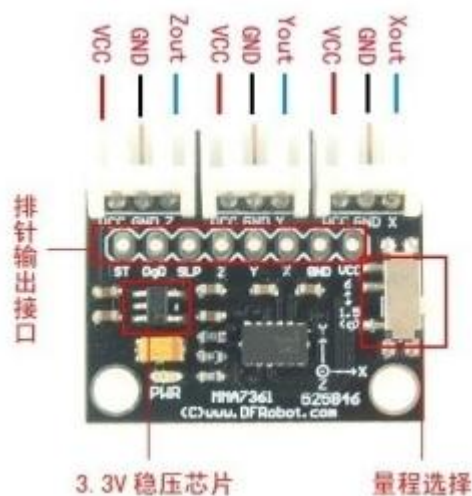
**Hall effect:** percepatan yang dirubah menjadi sinyal elektrik dengan cara mengukur setiap perubahan pergerakan yang terjadi pada daerah yang terinduksi magnet.

**Magneto-resistive:** Perubahan percepatan diketahui berdasarkan resistivitas material karena adanya daerah yang terinduksi magnet.

**Heat Transfer:** percepatan dapat diketahui dari lokasi sebuah benda yang dipanaskan dan diukur ketika terjadi percepatan dengan sensor temperatur.

Berikut adalah accelerometer yang kami gunakan:

**MMA7361**



## Features

- 3mm x 5mm x 1.0mm LGA-14 Package
- Low Current Consumption: 400  $\mu$ A
- Sleep Mode: 3  $\mu$ A
- Low Voltage Operation: 2.2 V – 3.6 V
- High Sensitivity (800 mV/g @ 1.5g)
- Selectable Sensitivity ( $\pm 1.5g$ ,  $\pm 6g$ )
- Fast Turn On Time (0.5 ms Enable Response Time)
- Self Test for Freefall Detect Diagnosis
- 0g-Detect for Freefall Protection
- Signal Conditioning with Low Pass Filter
- Robust Design, High Shocks Survivability
- RoHS Compliant
- Environmentally Preferred Product
- Low Cost

## Typical Applications

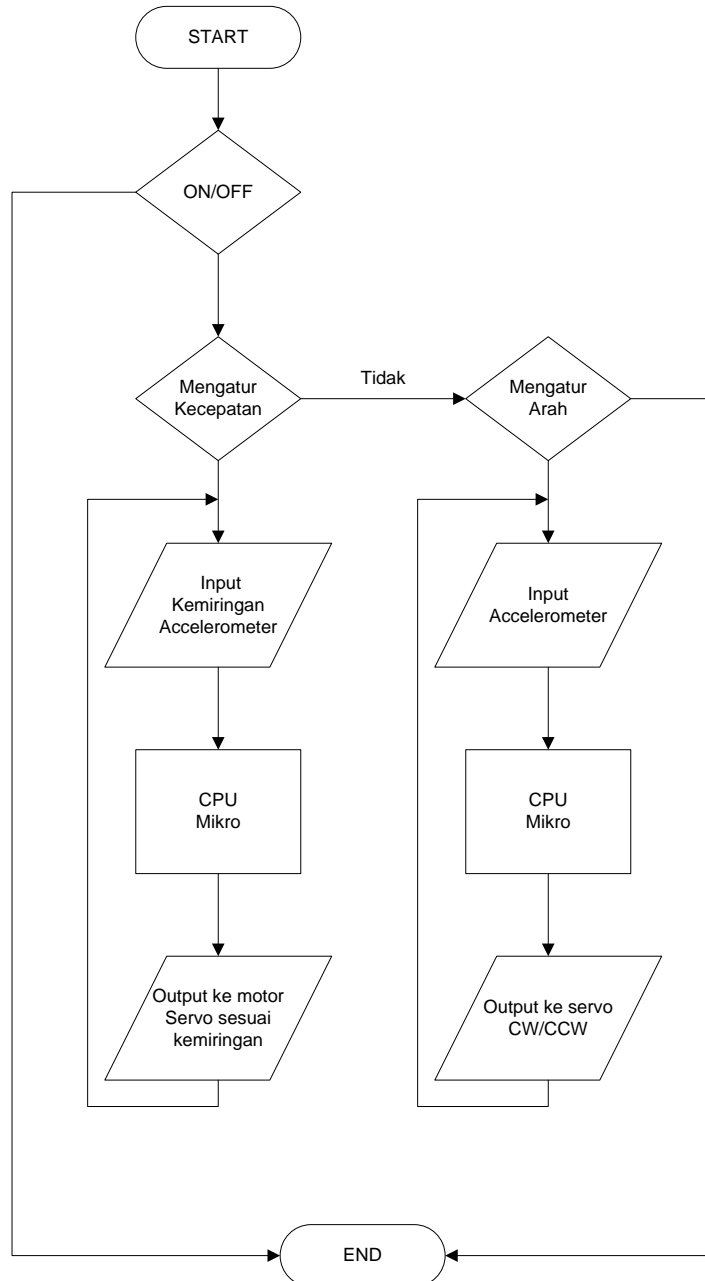
- 3D Gaming: Tilt and Motion Sensing, Event Recorder
- HDD MP3 Player: Freefall Detection
- Laptop PC: Freefall Detection, Anti-Theft
- Cell Phone: Image Stability, Text Scroll, Motion Dialing, E-Compass
- Pedometer: Motion Sensing
- PDA: Text Scroll
- Navigation and Dead Reckoning: E-Compass Tilt Compensation
- Robotics: Motion Sensing

# BAB 3

## METODE Pengerjaan

### 3.1 Prosedur Penyelesaian Masalah

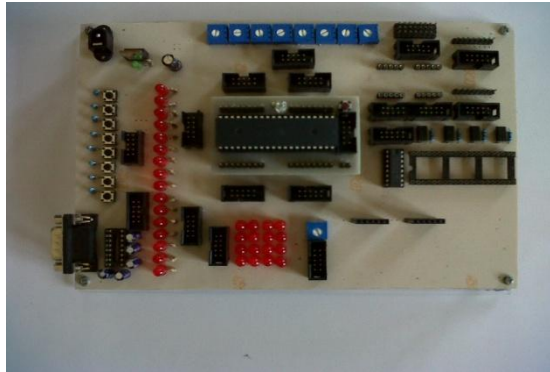
#### FLOW CHART DIAGRAM





### 3.1.1 Rangkaian Hardware

Rangkaian menggunakan training-kit yang telah di buat dengan desain dan system pengembangan sendiri yang telah dikerjakan oleh mahasiswa Politeknik ATMI Surakarta.

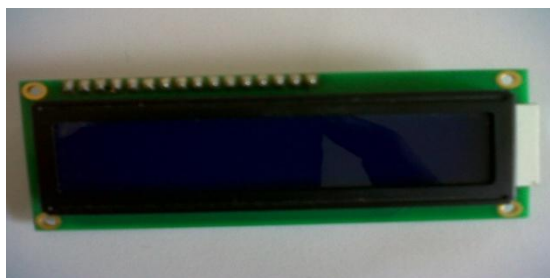


Komponen yang digunakan sebagai tambahan :

#### ACCELEROMETER



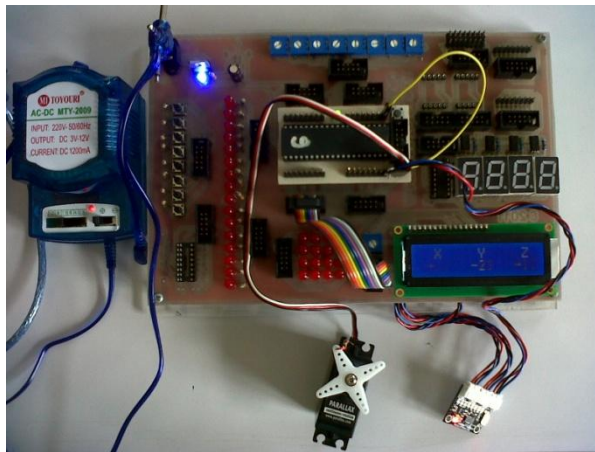
#### LCD 2x16



## Motor Servo Kontinuous



Berikut adalah rangkaian hardware yang kami gunakan sebagai eksplorasi komponen accelerometer MMA7361:



Port yang digunakan pada mikro:

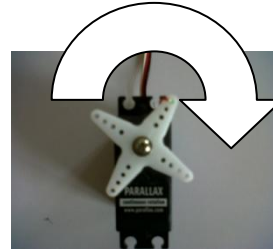
- Port A digunakan sebagai input sinyal axis-X accelerometer.
- Port B digunakan sebagai input sinyal dari motor servo.
- Port C digunakan sebagai output LCD 2x16.

Proses Kerja dari percobaan :

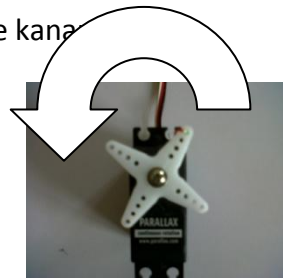
Sensor accelerometer digerakkan sesuai dengan arah gerakan axis-x (lihat gerakan axis pada datasheet accelerometer MMA7361), gerakan axis-x digunakan sebagai navigator untuk motor servo, dalam proyek ini kami mengasumsikan sebagai remot control RC dengan sensor gerak. Output dapat dilihat pula pada layar LCD 2x16 sebagai parameter angka (sebagai referensi/indicator dari accelerometer MMA6731).

Cara kerja dari rangkaian :

- Apabila sensor MMA7361 dimiringkan ke kanan maka motor akan berputar CW. dan sebaliknya apabila dimiringkan ke kiri maka motor akan berputar CCW.



Gambar accelerometer miring ke kanan

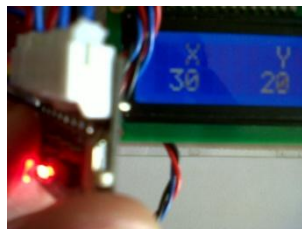


Gambar accelerometer miring ke kiri.

- Untuk tampilan pada LCD apabila accelerometer dimiringkan ke kanan maka angka akan bernilai negative dan sebaliknya apabila accelerometer dimiringkan ke kiri maka angka akan bernilai positif.



Gambar accelerometer miring ke kanan.



Gambar accelerometer miring ke kiri.

## BAB 4

### Program AVR

```
#include <mega8535.h>
#include <stdio.h>
#include <delay.h>

// Alphanumeric LCD functions
#include <alcd.h>
int a,b,s,c,i,j,k,ii, welcome=0;
char tempx[16],tempy[16],tempz[16];
int kecepatan, kec;

#define ADC_VREF_TYPE 0x60
// Read the 8 most significant bits
// of the AD conversion result
unsigned char read_adc(unsigned char adc_input)
{
    ADMUX=adc_input | (ADC_VREF_TYPE & 0xff);
    // Delay needed for the stabilization of the ADC input voltage
    delay_us(10);
    // Start the AD conversion
    ADCSRA|=0x40;
    // Wait for the AD conversion to complete
    while ((ADCSRA & 0x10)==0);
    ADCSRA|=0x10;
    return ADCH;
}
```

Fungsi  
mendeklarasikan  
variable-variabel  
yang digunakan

```

void lcd()
{
    a=read_adc(0);
    b=read_adc(1);
    c=read_adc(2);

    lcd_gotoxy(1,1);
    sprintf(tempx,"%d",a-83);
    lcd_puts(tempx);
    lcd_gotoxy(2,0);
    lcd_putsf("X");

    lcd_gotoxy(7,1);
    sprintf(tempy,"%d",b-90);
    lcd_puts(tempy);
    lcd_gotoxy(8,0);
    lcd_putsf("Y");

    lcd_gotoxy(13,1);
    sprintf(tempz,"%d",c-80);
    lcd_puts(tempz);
    lcd_gotoxy(14,0);
    lcd_putsf("Z");
    delay_ms(25);
    lcd_clear();
}

```

Program  
Penampil pada  
LCD 2x16

```

void timer(int delay)
{
    int satu,dua;
    for (satu=0;satu<delay;satu++)
    {
        for(dua=0;dua<5;dua++)
        {}
    }
}

```

Program untuk  
pengganti delay

```

void led()
{
    if(a>=75&&a<=94)
    {
        PORTB.1=1;
        goto aa;
    }
    PORTB.1=0;
    aa:
    if(b>=84&&b<=92)
    {
        PORTB.3=1;
        goto bb;
    }
    PORTB.3=0;
    bb:
    if(c>=75&&c<=83)
    {

```

Program untuk  
menyalakan LED  
pada training-kit

```
        PORTB.5=1;
        goto cc;
    }
    PORTB.5=0;
    cc:
}
```

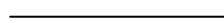
```
void main(void)
{
```

```
    PORTA=0x00;
    DDRA=0x00;
    PORTB=0x00;
    DDRB=0x0F;
    PORTC=0x00;
    DDRC=0x08;
    PORTD=0x00;
    DDRD=0x00;
```



Untuk mengatur input output port micro

```
TCCR0=0x00;
TCNT0=0x00;
OCR0=0x00;
TCCR1A=0x00;
TCCR1B=0x00;
TCNT1H=0x00;
TCNT1L=0x00;
ICR1H=0x00;
ICR1L=0x00;
OCR1AH=0x00;
OCR1AL=0x00;
```



Untuk setting Timer

```
OCR1BH=0x00;  
OCR1BL=0x00;  
ASSR=0x00;  
TCCR2=0x00;  
TCNT2=0x00;  
OCR2=0x00;  
MCUCR=0x00;  
MCUCSR=0x00;  
TIMSK=0x00;
```

```
// USART initialization  
// USART disabled  
UCSRB=0x00;
```



Untuk  
setting  
USART

```
ACSR=0x80;  
SFIOA=0x00;  
ADMUX=ADC_VREF_TYPE & 0xff;  
ADCSRA=0x84;  
SFIOA&=0xEF;
```



Untuk  
setting  
ADC

```
// SPI initialization  
// SPI disabled  
SPCR=0x00;
```

```
// TWI initialization  
// TWI disabled  
TWCR=0x00;  
lcd_init(16);
```

```
PORTC.3=1;      →
```


Program untuk menyalakan lampu LCD 2x16



```

while(1)
{
  PORTC.3=0;
  mati:
  if (PIND.0==1)
  {
    goto nyala;
  }
}

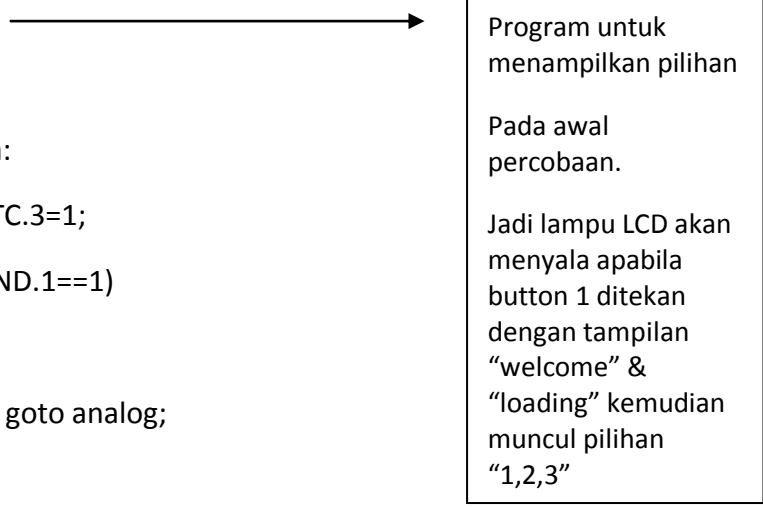
```



```

while (1)
{
  nyala:
  PORTC.3=1;
  if (PIND.1==1)
  {
    goto analog;
  }
  if (PIND.2==1)
  {
    i=0;
    j=0;
    goto derajat;
  }
  if (PIND.7==1)
  {
    welcome=0;
    goto mati;
  }
}

```



```

if (welcome==0)
{
    lcd_clear();
    lcd_gotoxy(5,0);
    lcd_putsf("WELCOME");
    delay_ms(1000);
    for (j=0;j<=5;j++)
    {
        for (s=0;s<=25;s++)
        {
            if(s<=3)
            {
                lcd_clear();
                lcd_gotoxy(3,0);
                lcd_putsf("LOADING");
            }
            if(s>3 && s<=8)
            {
                lcd_clear();
                lcd_gotoxy(3,0);
                lcd_putsf("LOADING.");
            }
            if(s>8 && s<=12)
            {
                lcd_clear();
                lcd_gotoxy(3,0);
                lcd_putsf("LOADING..");
            }
        }
    }
}

```

```

        if(s>13)
        {
            lcd_clear();
            lcd_gotoxy(3,0);
            lcd_putsf("LOADING...");
        }
        delay_ms(50);
    }
}
lcd_clear();
lcd_gotoxy(3,0);
lcd_putsf(" ");
delay_ms(1000);
welcome=1;
}
lcd_gotoxy(0,0);
lcd_putsf("1 2 3 ");
}
while (1)
{
    if (PIND.7==1)
    {
        welcome=0;
        goto mati;
    }
    analog:
    PORTC.3=1;
    atas:

```

```

i=0;
k=0;
led();
lcd();
if (a<=70)
{
    kec=(a+239); //kecepatan=kec/100;
    while(i<=5)
    {
        k=0;
        led();
        lcd();
        PORTB.0=1 ;
        timer(kec); //309 atau 310 stop
        PORTB.0=0;
        timer(2200);
        i++;
    }
    goto atas;
}
if (a>=95)
{
    kec=(a+215); //kecepatan=kec/100;
    while(k<=5)
    {
        i=0;
        led();
        lcd();

```

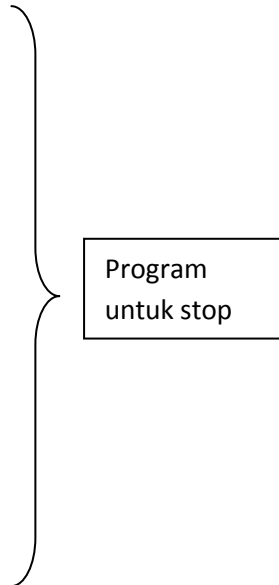
Program  
putar kanan

Program  
putar kiri

```

        PORTB.0=1 ;
        timer(kec); //309 atau 310 stop
        PORTB.0=0 ;
        timer(2200) ;
        k++;
    }
    goto atas;
}
if (a>=73 && a<=92)
{
    led();
    lcd();
    PORTB.0=1 ;
    timer(309); //309 atau 310 stop
    PORTB.0=0 ;
    timer(2200);
}
}
while(1)
{
    if (PIND.7==1)
    {
        welcome=0;
        goto mati;
    }
    derajat:
    PORTC.3=1;
    led();

```



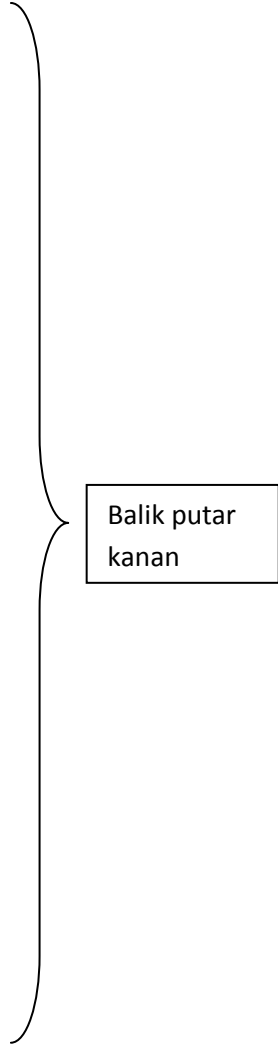
```
lcd();
if (a<=60 && j==0)
{
    for(ii=0;ii<=30;ii++)
    {
        led();
        lcd();
        PORTB.0=1 ;
        delay_us(1470);
        PORTB.0=0;
        delay_us(11000);
    }
    PORTB.0=1 ;
    delay_us(1500);
    PORTB.0=0 ;
    delay_us(15000) ;
    j=1;
}
if (a>=105 && j==0)
{
    for(ii=0;ii<=30;ii++)
    {
        led();
        lcd();
        PORTB.0=1 ;
        delay_us(1530);
        PORTB.0=0 ;
        delay_us(11000) ;
    }
}
```

The diagram illustrates two code blocks. The first block, labeled 'Putar kanan', is enclosed in a large right-facing curly brace. The second block, labeled 'Putar kiri', is indicated by a horizontal arrow pointing from the text 'Putar kiri' to the start of the code block.

```

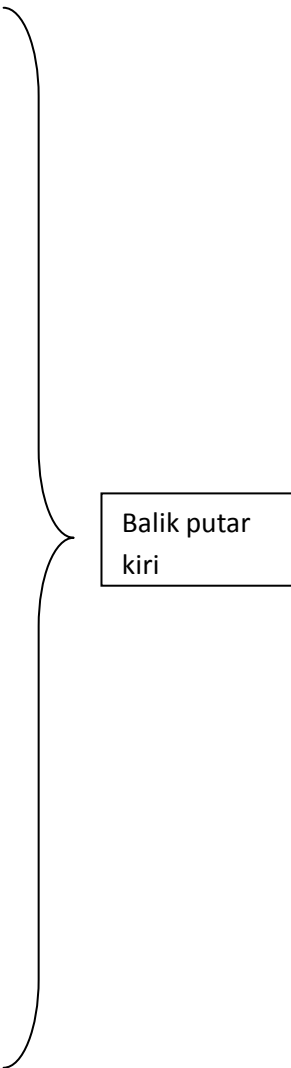
    }
    PORTB.0=1 ;
    delay_us(1500);
    PORTB.0=0 ;
    delay_us(15000) ;
    j=2;
}
if (a>60 && j==1 && a<105)
{
    for(ii=0;ii<=14;ii++)
    {
        led();
        lcd();
        PORTB.0=1 ;
        delay_us(1573);
        PORTB.0=0 ;
        delay_us(11000) ;
    }
    PORTB.0=1 ;
    delay_us(1500);
    PORTB.0=0 ;
    delay_us(15000) ;
    j=0;
}

```



Balik putar  
kanan

```
if (a>60 && j==2 && a<105)
{
    for(ii=0;ii<=14;ii++)
    {
        led();
        lcd();
        PORTB.0=1 ;
        delay_us(1437);
        PORTB.0=0 ;
        delay_us(11000) ;
    }
    PORTB.0=1 ;
    delay_us(1500);
    PORTB.0=0 ;
    delay_us(15000) ;
    j=0;
}
}
}
```



Balik putar  
kiri



## **BAB 5**

### **PENUTUP**

Bab ini adalah bab terakhir dari keseluruhan laporan proyek tentang accelerometer yang telah kami susun. Dalam bab ini kami akan memaparkan tentang kelebihan, kekurangan, dan kesimpulan dari hasil penelitian kami tentang sensor accelerometer yang di aplikasikan dengan tambahan hardware LCD 16x2 dan motor servo kontinuous. Proses penelitian ini tentunya tidak lepas dari kendala – kendala yang dihadapi kelompok, antara lain adalah :

1. Waktu terbuang cukup lama dalam menganalisa dan mempelajari hardware tambahan yaitu mencari sudut dan timing yang tepat untuk memprogram motor servo.
2. Keterbatasan sarana pendukung seperti laptop cukup mempersulit dan memperlambat kinerja dalam proses penelitian sensor accelerometer.

#### **Kelebihan**

- a. Berdasarkan pengamatan tentang penelitian terhadap Digital Tilt Sensor, sensor accelerometer memiliki kelebihan. Sensor accelerometer dilengkapi dengan 3 axis yaitu x,y,z.
- b. Dari data yang dihasilkan dari conversi ADC, data dapat dimanfaatkan sesuai dengan keinginan programmer.

#### **Kekurangan**

- a. Komponen hardware terlalu terbuka, tidak safety.

#### **Kesimpulan**

Pada penelitian sensor accelerometer yang belum sepenuhnya sempurna ini kami dapat menemukan banyak sekali pengembangan yang bisa dilakukan, yang pada saat ini tidak bisa kami lakukan karena keterbatasan teknologi. Akan tetapi pengerjaan penelitian kami telah menghasilkan perangkat dan fungsi yang sesuai dengan harapan dan rencana kami. Perangkat sensor accelerometer ini dapat digunakan sebagai pengontrol remot RC yaitu dengan menggerakkan sesuai axis yang telah dimanfaatkan untuk program kontrolnya. Disini kami memanfaatkan sebagai penggerak roda depan mobil RC dan hasilnya sesuai dengan program.

## BAB 6

### Daftar Pustaka

<http://www.alldatasheet.com/view.jsp?Searchword=MMA7361>

<http://www.google.co.id/url?sa=t&rct=j&q=&esrc=s&source=web&cd=8&cad=rja&ved=0CD0QFjAH&url=http%3A%2F%2Fiswanto.staff.umy.ac.id%2Ffiles%2F2012%2F05%2FAPLIKA-SI-MOTOR-SERVO-DENGAN-MIKROKONTROLER.doc&ei=KyptUKLFL5DkrAfcsoDQDA&usg=AFQjCNHHGxSCHl1Dw38tdFgNZuSBtWDwQQ>

[http://www.geeetech.com/wiki/index.php/MMA7361\\_Triple\\_Axis\\_Accelerometer\\_Breakout](http://www.geeetech.com/wiki/index.php/MMA7361_Triple_Axis_Accelerometer_Breakout)