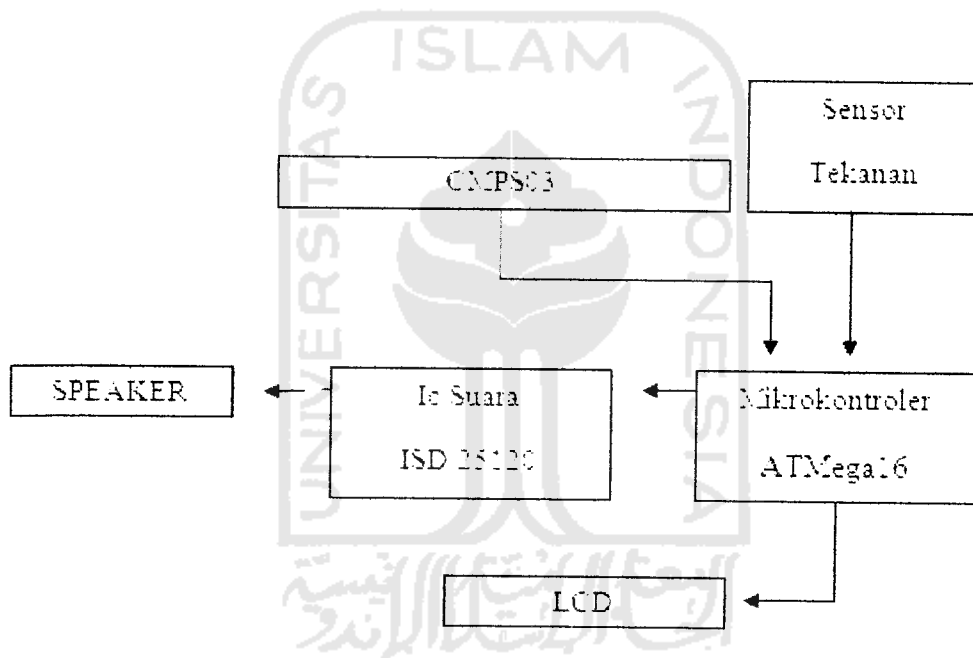


BAB III

PERANCANGAN SISTEM

3.1 Perancangan Sistem

Dalam perancangan sistem ini, sistem terdiri atas beberapa bagian penting yaitu sensor navigasi CMPS03, sensor tekanan, IC suara 25120, dan mikrokontroler ATmega16 sebagai pengolah data, dan sebagai keluaran dari sistem ini terdiri dari LCD dan Speaker.



Gambar 3.1 Bagan Alir Sistem Alat Bantu Navigasi Bagi Pendaki Gunung

Proses bekerjanya sistem diawali dengan pendektasian sensor tekanan yang dalam penelitian ini disimulasikan dengan menggunakan resirtor variabel dan pembacaan arah mata angin menggunakan CMPS03, data dari kedua sensor tersebut diolah melalui mikrokontroler ATmega16 yang berfungsi sebagai pusat kontrol sistem. *Output* dari perancangan ini berupa LCD dan IC suara ISD 25120 mengolah keluaran dari mikrokontroler untuk kemudian dihasilkan *output* berupa

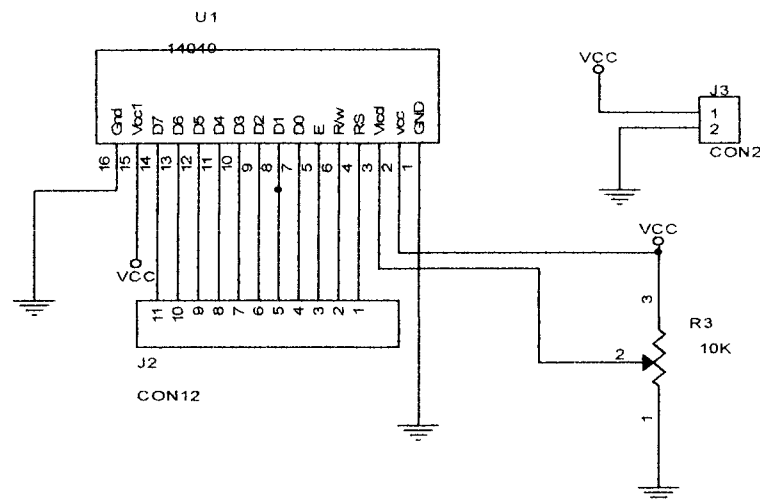
suara. LCD mampu menampilkan hasil keluaran berupa 16 arah mata angin disertai dengan derajat dan ketinggian.

3.2 Perancangan *Hardware*

Pada perancangan *hardware* penelitian ini, seluruh rangkaian elektronik baik yang merupakan rangkaian kontroler maupun utilitas, semuanya terhubung dengan *wiring* secara fisik. Perancangan *hardware* ini didukung oleh rangkaian-rangkaian listrik yang membantu kerja mikrokontroler sebagai pengendali utama, seperti: sistem minimum, power supply serta rangkaian listrik lainnya yang menjalankan sistem secara keseluruhan. Berikut merupakan komponen-komponen yang digunakan:

3.2.1 Rangkaian Penampil LCD

Modul *Liquid Crystal Display* (LCD) merupakan modul display yang serbaguna, karena dapat digunakan untuk menampilkan berbagai tampilan baik berupa huruf, angka dan karakter lainnya serta dapat menampilkan berbagai macam tulisan maupun pesan – pesan pendek lainnya. Rangkaian penampil LCD pada sistem difungsikan untuk menampilkan koordinat dan ketinggian dari suatu lokasi. Didalam sistem ini LCD menampilkan keluaran dari mikrokontroler dan CMPS03 berupa tulisan, huruf yaitu arah mata angin serta derajatnya dan juga keluaran dari mikrokontroler dan simulasi dari sensor tekanan (Potensiometer) berupa ketinggian.



Gambar 3.2 Penampil rangkaian LCD

Rangkaian penampil LCD pada perancangan ini menggunakan teknik antarmuka 8 bit. Artinya teknik ini membutuhkan 8 bit jalur data dalam proses antarmukanya. Sehingga membutuhkan 8 bit I/O pada mikrokontroler yang terhubung oleh P0.0 ... P0.7. Pada gambar diatas, LCD selalu pada kondisi tulis (write) dengan menghubungkan kaki R/W ke ground. Ini dimaksudkan agar LCD tidak mengeluarkan data (kondisi Read) yang mana data tersebut akan bertabrakan dengan data komponen lain/perintah lain di jalur bus. Rangkaian Penampil LCD juga dilengkapi dengan pengatur cerah terang pada layer LCD secara langsung dengan mengatur pada VR yang terhubung pada VEE dan dipararel dengan VCC.

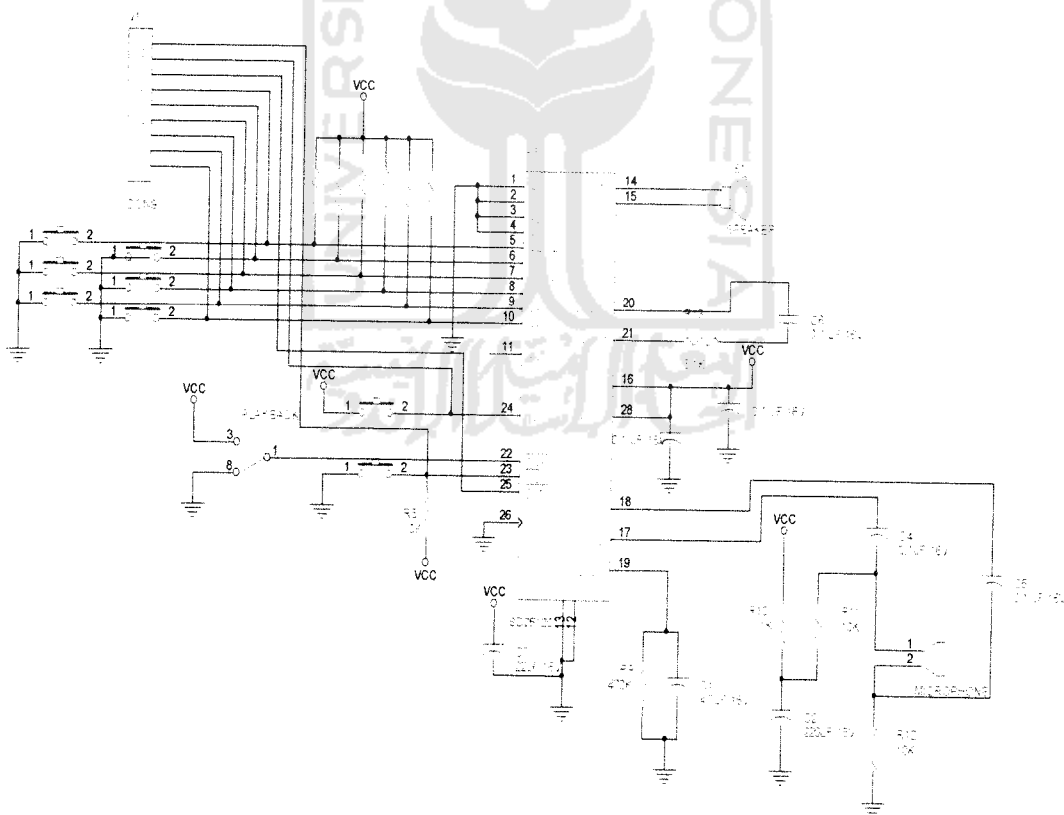
3.2.2 Modul Devantech Magnetic Compass (CMPS03)

Modul CMPS03 merupakan alat untuk mendeteksi arah dari mata angin. Pada modul ini terdapat 9 pin I/O dimana sumber tegangan 5 volt dihubungkan dengan pin 7 sebagai pembangkit frekuensi pin 2 dan 3 di hubungkan ke port PC0 dan PC1 berfungsi sebagai penerima masukan dari sensor. Pin 6 dari CMPS03 dihubungkan dengan *Push Button* yang digroundkan.

3.2.3 Rangkaian IC Suara ISD25120

Gambar 3.3 merupakan rangkaian ISD25120 yang digunakan untuk merekam suara yang akan digunakan pada proses penunjukan arah mata angin. Sistem tersebut mampu merekam 16 arah mata angin.

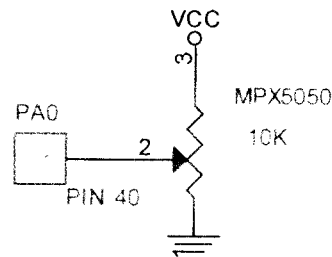
Proses perekaman dimulai dari alamat 0000 sampai 1111. Proses awal perekaman dilakukan dengan menentukan alamat terlebih dahulu kemudian menekan tombol P/R hingga dalam posisi *low*, kemudian menahan tombol CE dalam keadaan *low* diikuti dengan merekam suara, jika terjadi kesalahan dalam perekaman maka tombol PD ditekan untuk menghapus rekaman serta untuk mengembalikan data dari awal. Untuk proses perekaman selanjutnya dilakukan perubahan pada alamat memori.



Gambar 3.3 Rangkaian IC Suara ISD25120

3.2.4 Rangkaian Sensor Tekanan

Pada sistem ini, resistor variabel digunakan sebagai simulasi dari sensor tekanan. Perubahan ketinggian dapat dimisalkan dengan merubah – ubah nilai tahanan dari resistor variabel.



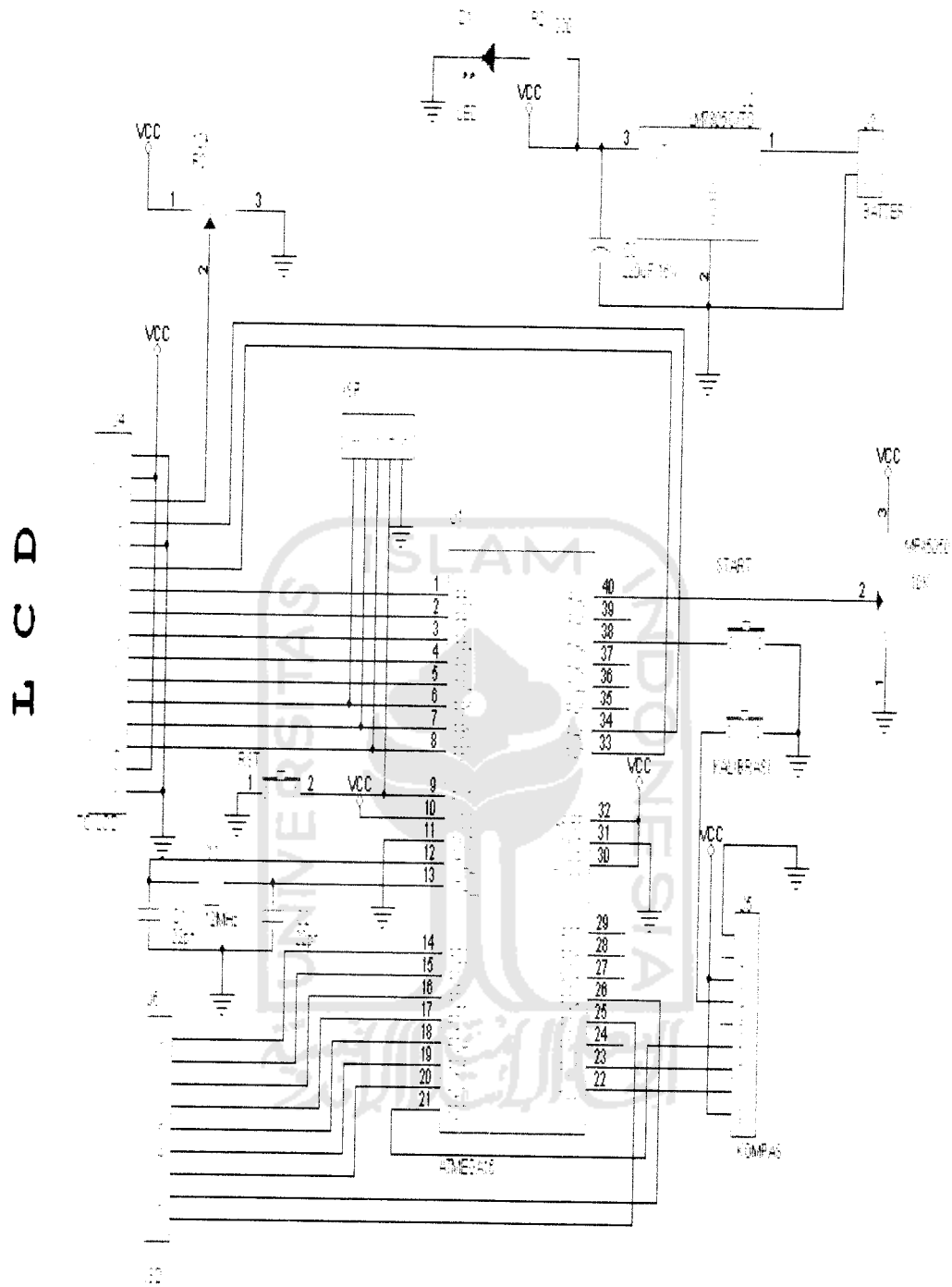
Gambar 3.4 Rangkaian Sensor tekanan

3.2.5 Rangkaian Sistem Minimum ATmega 16

Mikrokontroler ATmega16 memiliki arsitektur RISC 8 bit, dimana semua instruksi dikemas dalam kode 16-bit (16-bits word) dan sebagian besar instruksi dieksekusi dalam 1 (satu) siklus clock.

Rangkaian osilator pada sistem ini digunakan oleh *mikrokontroler* sebagai sinyal denyut (*clock*). Frekuensi sinyal denyut inilah yang menentukan kecepatan eksekusi yang akan dijalankan. Frekuensi denyut maksimum yang diperbolehkan adalah 33 MHz. Tetapi pada perancangan sistem ini menggunakan XTAL 12 MHz, dan 2 buah kapasitor 22 pF.

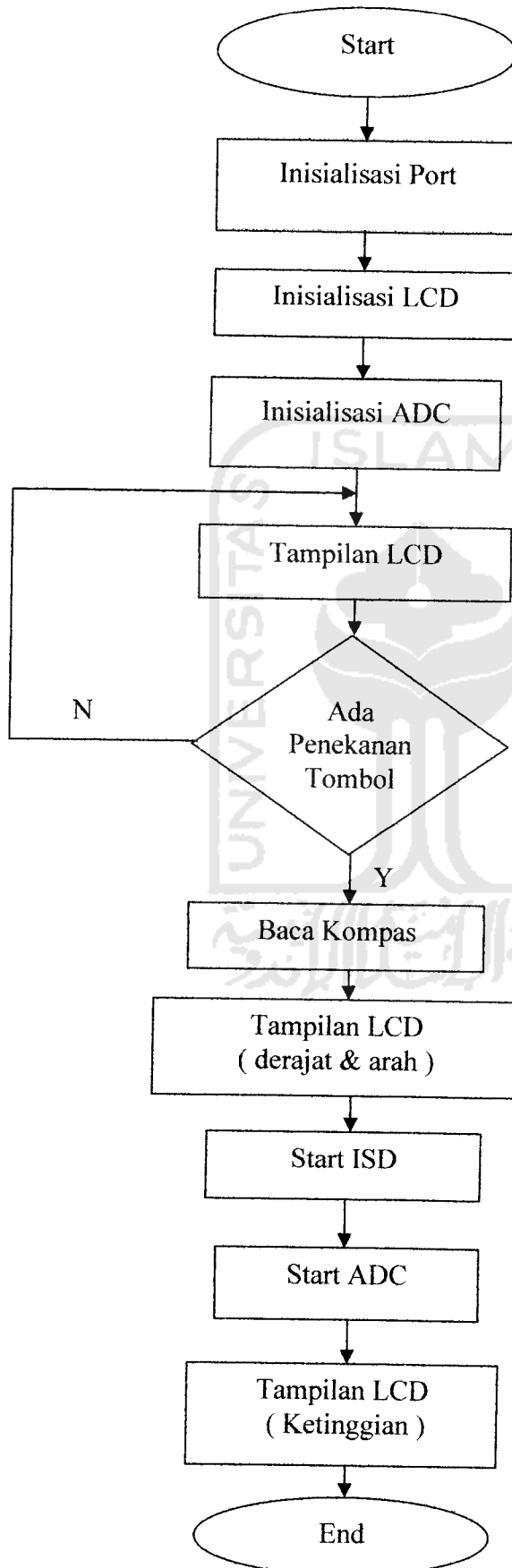
Sedangkan untuk membangkitkan sinyal reset, maka tombol reset langsung dihubungkan dengan ground .



Gambar 3.5 Rangkaian Sistem Minimum ATmega 16

3.3 Perancangan Software

Bagan Alir Program



3.4 Proses Pemrograman pada Mikrokontroler ATmega16

3.4.1 Proses Inisialisasi Program

```
#include<avr/io.h>
#include<avr/interrupt.h>
#include<compat/deprecated.h>
#define LCD_RS 6
#define LCD_CS 7
#define RESET 1
#define START 0
#define EOM 2
#define SCL 0
#define SDA 1
#define nop() __asm__("NOP");
```

3.4.2 Proses Deklarasi Program

```
unsigned char tulisan[]=" KOMPAS DIGITAL ";
unsigned char tulisan1[]=" UII 2009";
unsigned char tulisan2[]="Arah: ";
unsigned char tulisan3[]="Ketinggian: m";
```

3.4.3 Proses Fungsi Delay Program

```
void delay(unsigned int nilai_delay)
{
  unsigned int i;
  unsigned int j;
  for(i=0;i<nilai_delay;i++)
  {
    for(j=0;j<=100;j++)
    {
      ;
    }
  }
}
```

3.4.4 Proses Instruksi pada LCD :

```
void WR_INST(int data_)
{
  cbi(PORTA,LCD_RS);
  PORTB=data_;
  sbi(PORTA,LCD_CS);
  delay(10);
  cbi(PORTA,LCD_CS);
  delay(10);
```


3.4.5 Proses akses pembacaan nilai I2C :

```

unsigned char i2c_read(unsigned char address, unsigned char reg)
{
    unsigned char read_data = 0;
    TWCR = ((1<<TWINT)|(1<<TWSTA)|(1<<TWEN));
    while(!(TWCR & 0x80));
    TWDR = address;
    TWCR = ((1<<TWINT)|(1<<TWEN));
    while(!(TWCR & 0x80));
    TWDR = reg;
    TWCR = ((1<<TWINT)|(1<<TWEN));
    while(!(TWCR & 0x80));
    TWCR = ((1<<TWINT)|(1<<TWSTA)|(1<<TWEN));
    while(!(TWCR & 0x80));
    TWDR = address+1;
    TWCR = ((1<<TWINT)|(1<<TWEA)|(1<<TWEN));
    while(!(TWCR & 0x80));
    TWCR = ((1<<TWINT)|(1<<TWEN));
    while(!(TWCR & 0x80));
    read_data = TWDR;
    TWCR = ((1<<TWINT)|(1<<TWSTO)|(1<<TWEN));
    return read_data;
}

```

3.4.6 Proses Program Utama :

```

int main(void)
{
    unsigned char k;
    DDRA=0xc0;
    DDRB=0xff;
    DDRC=0xff;
    DDRD=0xfb; //1111 1011
    PORTA=0xff;
    PORTC=0xf3; //1111 0011
    PORTD=0x6;
    init_LCD();
    ADMUX=0;
    ADCSRA=0xca;
    SFIOR=0;
    //while(1);
    for(k=0;k<=15;k++)
    {
        WR_DATA(tulisan[k]);
    }
    WR_INST(0xc0);
    for(k=0;k<=15;k++)
    {
        WR_DATA(tulisan1[k]);
    }
}

```

```

}
delay(5000);
WR_INST(1);
for(k=0;k<=15;k++)
{
    WR_DATA(tulisan3[k]);
}
WR_INST(0xc0);
for(k=0;k<=15;k++)
{
    WR_DATA(tulisan2[k]);
}
setup();
sei();
//i2c_transmit(0xc0,2,119);
//delay(100);
while(1)
{
    sei();
    ADCSRA=0xca;
    delay(100);
    WR_INST(0x8b);
    display_LCD(nilai_ADC*10);
    cli();
    //delay(7500);
    if(bit_is_clear(PINA,3))
    {
        delay(250);
        k=i2c_read(0xc0,2);
        hasil16=k*256;
        delay(100);
        k=i2c_read(0xc0,3);
        hasil16=hasil16+k;
        delay(100);
        hasil16=hasil16/450;
        k=PORTD;
        k=(k&7);
        k=(k|(ISD[hasil16]));
        // k=(ISD[hasil16]);
        //0x0=utara; 0x8=timur laut; 0x10=timur; 0x18=tenggara;

        0x20=selatan; 0x28=barat daya; 0x30=barat; 0x38=barat

        laut

        PORTD=k;
        ISD_start();
        WR_INST(0xc6);
    }
}

```

```
for(k=0;k<=9;k++)
{
    WR_DATA(utara[k+(hasil16*10)]);
}
while(bit_is_set(PIND,2));
}
}
return 0;
}
```

