

BAB III

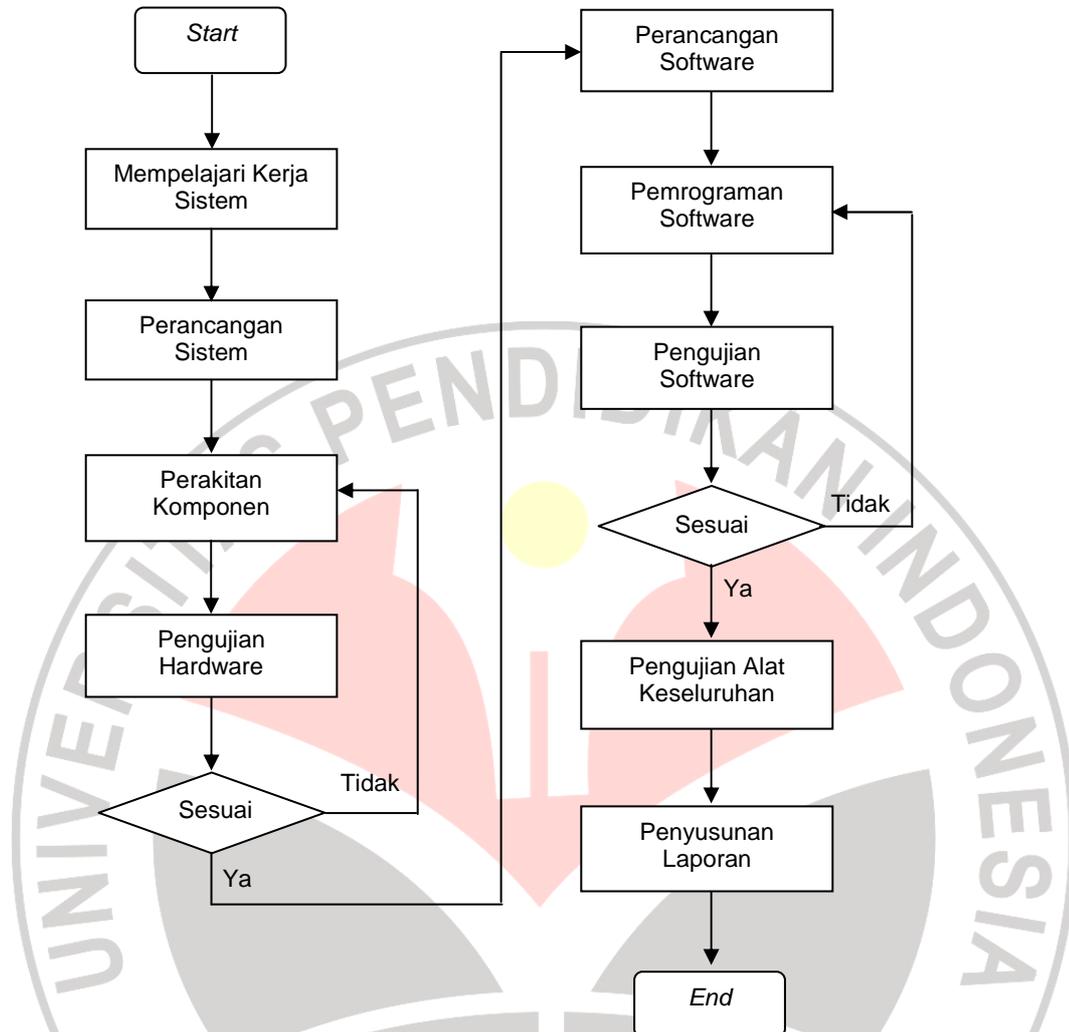
METODE PENELITIAN

3.1 Perancangan Alat

Dalam merealisasikan sebuah sistem elektronik diperlukan perancangan komponen secara tepat dan akurat. Tahap perancangan sangat penting dilakukan untuk mempermudah pada proses pembuatan dan untuk meminimalisir kesalahan pada proses perakitan komponen dan pembuatan program. Kesalahan pada proses pembuatan akan mengakibatkan sistem tidak akan berfungsi dengan baik. Oleh karena itu, sebelum masuk pada tahap pembuatan alat terlebih dahulu harus dilakukan perancangan alat.

Perancangan alat ini terdiri dari dua tahap perancangan, tahap pertama yaitu perancangan pada bagian perangkat keras (*hardware*) sedangkan tahap kedua yaitu perancangan pada bagian perangkat lunak (*software*). Perancangan perangkat keras meliputi tahapan rancangan untuk perakitan komponen-komponen elektronika yang akan digunakan, diantaranya yaitu untuk rangkaian osilator, konverter, op amp, mikrokontroler dan LCD. Sedangkan perancangan pada perangkat lunak yaitu berupa perancangan untuk pembuatan program. Tahap perancangan ini dapat dijadikan sebagai acuan dalam mengimplementasikan pada tahap pembuatan perangkat keras maupun perangkat lunak, sehingga sistem yang dibuat berfungsi sesuai dengan yang diharapkan.

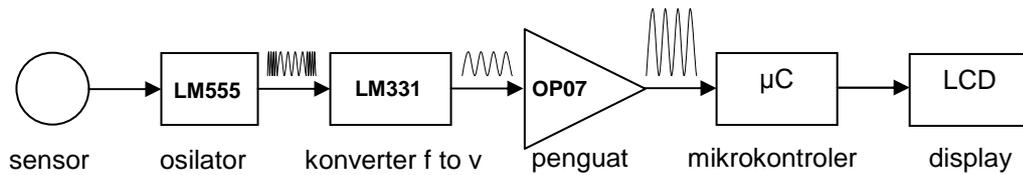
Visualisasi mengenai sistem pengidentifikasi jenis cairan dapat dilihat dalam bentuk diagram alir penelitian yang ditunjukkan pada bagan berikut.



Bagan 3.1 Diagram Alir Penelitian

3.2 Perancangan dan Pembuatan Perangkat Keras (Hardware)

Perangkat keras dibuat berdasarkan pada tahap perancangan sebelumnya. Perangkat keras yang akan dibuat terdiri dari sensor kapasitansi, osilator, konverter, op amp, mikrokontroler dan LCD. Secara garis besar perangkat keras yang akan dibuat terdiri dari bagian-bagian sebagai berikut.



Bagan 3.2 Diagram Alir Perangkat Keras

3.2.1 Sensor Kapasitansi

Sensor kapasitansi adalah sensor yang bekerja berdasarkan prinsip kerja kapasitor. Sensor kapasitansi berbentuk silinder yang baik adalah ketika selisih antara jari-jari dalam (R_1) dan jari-jari luar (R_2) adalah sekecil mungkin. Hal ini sesuai dengan perumusan kapasitansi kapasitor silinder, yaitu :

$$C = \frac{q}{V} = \frac{2\pi\epsilon_0 l}{\ln\left(\frac{b}{a}\right)} \dots \dots \dots (3.1)$$

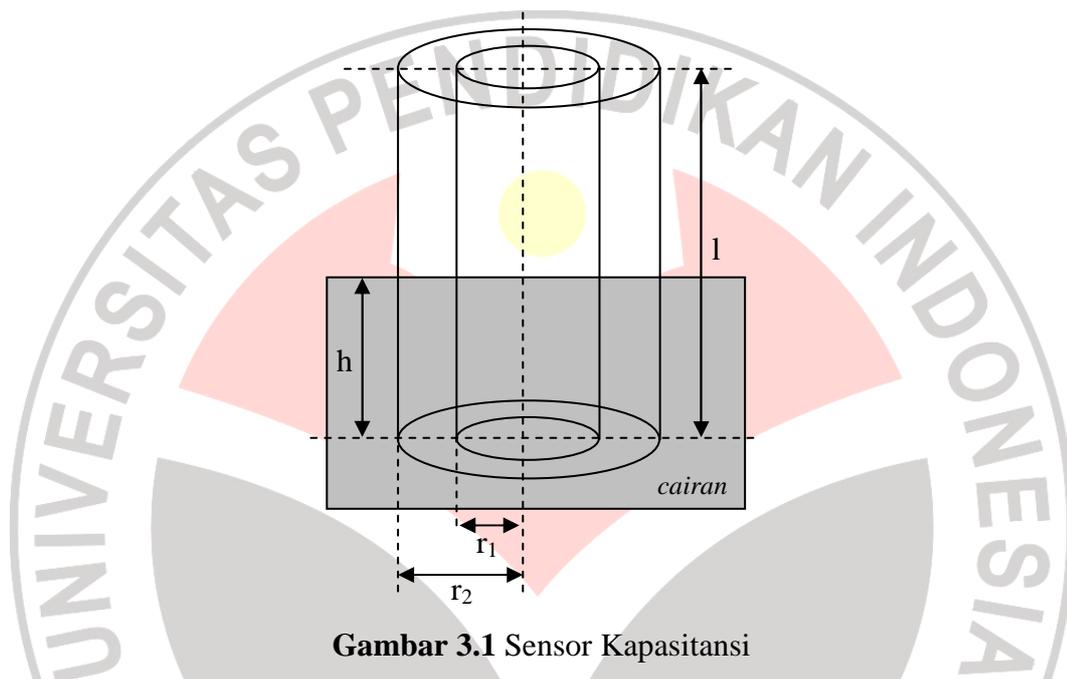
Dari persamaan di atas semakin kecil perbandingan antara kedua jari-jari maka diperoleh nilai $\ln(b/a)$ semakin kecil, sehingga nilai kapasitansi yang diperoleh akan semakin besar.

Pada penelitian ini sensor kapasitansi dirancang berbentuk silinder dengan jari-jari dalam (R_1), jari-jari luar (R_2) dan panjang tertentu. Bahan kapasitor silinder terbuat dari aluminium dengan $R_1 = 0,7$ cm, $R_2 = 1$ cm dan $l = 15$ cm.

Beberapa alasan menggunakan aluminium sebagai bahan sensor kapasitansi diantaranya adalah :

- Penghantar listrik dan panas yang baik.
- Tahan korosi.
- Tidak memiliki sifat kemagnetan, sehingga tidak terjadi induksi magnetik.

Untuk mengidentifikasi jenis cairan, cairan dimasukkan pada suatu wadah yang di dalamnya terdapat sensor kapasitansi. Pengujian dilakukan dengan posisi sensor tegak lurus terhadap permukaan cairan dan ketinggian cairan yang diuji harus sama.



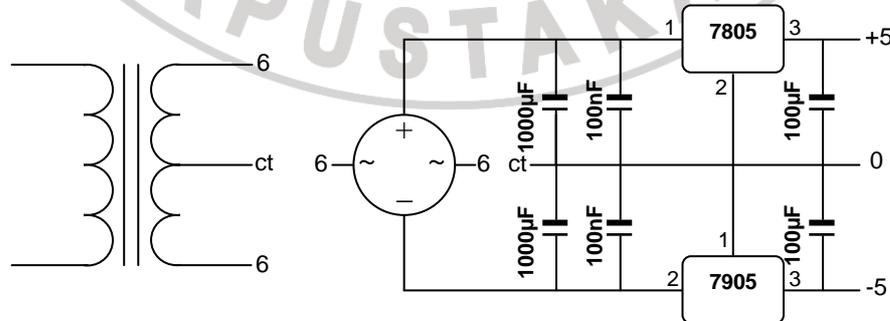
Gambar 3.1 Sensor Kapasitansi

Penyisipan bahan dielektrik antara plat silinder kapasitor akan meningkatkan nilai kapasitansinya. Sensor kapasitansi dapat mengidentifikasi jenis cairan berdasarkan nilai kapasitansi kapasitor. Kesamaan nilai kapasitansi kapasitor dari jenis cairan berbeda akan sulit untuk dibedakan jenisnya. Terkadang jenis cairan berbeda memiliki konstanta dielektrik yang sama. Untuk memperoleh nilai kapasitansi yang berbeda, jenis cairan yang digunakan harus memiliki konstanta dielektrik yang berbeda. Jenis cairan yang digunakan pada penelitian ini adalah air, alkohol, gliserin dan minyak tanah.

3.2.2 Rangkaian Power Supply

Rangkaian penyearah berfungsi sebagai penyearah arus listrik, sedangkan regulator berfungsi untuk meregulasi atau mengatur tegangan. Perangkat elektronika umumnya menggunakan input arus searah DC (*direct current*) yang stabil. Baterai dan accumulator adalah beberapa contoh sumber power supply DC yang baik. Namun untuk aplikasi yang membutuhkan power supply lebih besar, sumber tersebut tidak cukup. Power supply yang besar adalah sumber arus bolak-balik AC (*alternating current*) yang dihasilkan dari pembangkit tenaga listrik. Untuk keperluan tersebut maka diperlukan komponen yang dapat mengubah arus AC menjadi DC, diantaranya yaitu dengan menggunakan dioda *bridge*.

Rangkaian penyearah dikatakan sudah cukup bagus jika tegangan *ripple*-nya kecil, namun masih terdapat masalah pada stabilitas tegangannya. Jika tegangan PLN naik/turun, maka tegangan outputnya juga akan ikut naik/turun. Untuk beberapa aplikasi perubahan tegangan ini cukup mengganggu, sehingga diperlukan komponen aktif yang dapat meregulasi tegangan keluaran ini menjadi lebih stabil.



Gambar 3.2 Rangkaian Power Supply

Beberapa komponen yang digunakan seperti osilator, konverter, op amp, mikrokontroler dan LCD akan berfungsi dengan baik jika input tegangannya stabil. Oleh karena itu diperlukan komponen tambahan untuk kebutuhan tersebut, yaitu dengan menggunakan IC regulator. IC regulator yang digunakan adalah IC LM7805 untuk output tegangan +5V dan IC LM7905 untuk output tegangan -5V. Skema rangkaian penyearah dan regulator ditunjukkan oleh gambar 3.2 di atas.

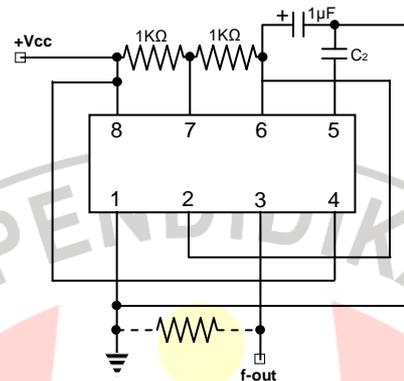
Penggunaan yang terlalu lama akan membuat regulator tersebut terus menerus bekerja, sehingga hal ini dapat menyebabkan komponen tersebut menjadi panas. Oleh karena itu IC regulator tersebut sebaiknya diberikan aluminium pendingin (*heatshink*). Penggunaan komponen pendingin ini bertujuan sebagai pelindung regulator dari kerusakan yang diakibatkan oleh panas.

3.2.3 Rangkaian Osilator

Osilator berfungsi sebagai sumber pembangkit pulsa (*pulse generator*). Osilator yang digunakan pada penelitian ini adalah IC LM555. Salah satu aplikasi dari IC LM555 adalah *astable multivibrator*, yaitu osilator yang beroperasi terus menerus untuk menghasilkan gelombang kotak dengan perbandingan mark/space (*hi/lo*). Rangkaian *astable multivibrator* biasanya dimanfaatkan sebagai sinyal clock. Untuk membuat rangkaian *astable multivibrator* adalah dengan rangkaian seperti gambar 3.3 di bawah (port 2 dan port 6 terhubung).

Pada suatu rangkaian osilator terdapat beberapa komponen pendukung didalamnya seperti resistor dan kapasitor. Untuk rangkaian *astable multivibrator* terdapat dua buah resistor dan dua buah kapasitor, namun salah satu kapasitornya

adalah menjadi variabel kapasitor. Yang berperan menjadi variabel kapasitor adalah sensor kapasitansi (C_2) seperti ditunjukkan pada gambar berikut.



Gambar 3.3 Rangkaian Astable Multivibrator

Rangkaian tersebut akan mentrigger secara sendirinya dan bekerja sebagai multivibrator. Kapasitor eksternal charges melalui $R_a + R_b$ dan discharge melalui R_b . Duty cycle diset dengan rasio dari dua buah resistor tersebut.

Pada mode operasi seperti ini, kapasitor charges dan discharges antara $1/3 V_{cc}$ dan $2/3 V_{cc}$. Pada mode trigger, waktu charges, discharge dan frekuensi tidak berhubungan dengan tegangan supply.

Perhitungan pada rangkaian astable adalah sebagai berikut :

- Waktu charge (*output high*) :

$$t_1 = 0,693 (R_a + R_b) C \dots\dots\dots (3.2)$$

- Waktu discharge (*output low*) :

$$t_2 = 0,693 (R_b) C \dots\dots\dots (3.3)$$

- Periode total :

$$T = t_1 + t_2 = 0,693 (R_a + 2R_b) C \dots\dots\dots (3.4)$$

- Frekuensi osilasi :

$$f = \frac{1}{T} = \frac{1,44}{(R_a + 2R_b)C} \dots\dots\dots (3.5)$$

Komponen yang digunakan pada penelitian ini adalah $R_1 = R_2 = 1 \text{ k}\Omega$, $C_1 = 0,1 \text{ }\mu\text{F}$ dan C_2 adalah kapasitor variabel. Maka besarnya frekuensi osilasi dapat dinyatakan sebagai berikut.

$$\begin{aligned} f &= \frac{1,44}{(R_a + 2R_b)C} \\ &= \frac{1,44}{(1 \cdot 10^3 + 2 \cdot 10^3)C} \\ &= \frac{1,44}{3 \cdot 10^3 C} \\ &= 4,8 \cdot 10^{-4} C^{-1} \end{aligned}$$

Karena nilai kapasitansi kapasitor berbeda untuk setiap jenis cairan, maka nilai frekuensi yang dihasilkannya akan berbeda pula. Berdasarkan persamaan tersebut, semakin besar nilai kapasitansi kapasitor maka frekuensi yang dihasilkan oleh rangkaian osilator akan semakin kecil, begitu pula sebaliknya.

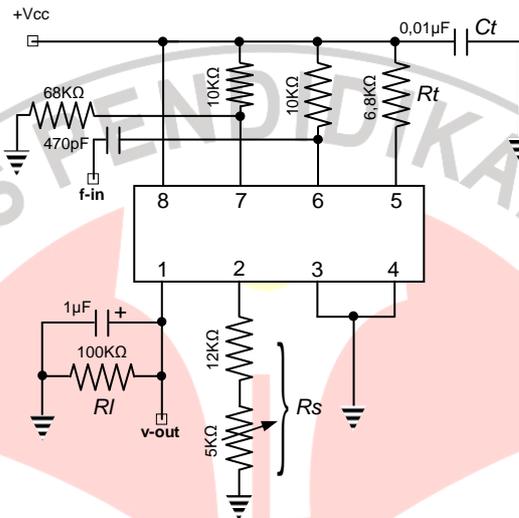
3.2.4 Rangkaian Konverter

Konverter berfungsi sebagai pengkonversi atau pengubah frekuensi menjadi tegangan yang lebih dikenal dengan istilah rangkaian F to V. Konverter yang digunakan pada penelitian ini adalah IC LM331.

IC LM331 ini mampu mengkonversi frekuensi sampai 10KHz (batas maksimum). Untuk mendapatkan tegangan yang diinginkan supaya alat tersebut

dapat bekerja dengan baik, maka diperlukan perhitungan sebagai berikut dengan persamaan :

$$V_{out} = 2,09 \cdot f_{in} \cdot \frac{R_t}{R_s} (R_t \cdot C_t) \dots\dots\dots (3.6)$$



Gambar 3.4 Rangkaian Konverter Frekuensi-Tegangan

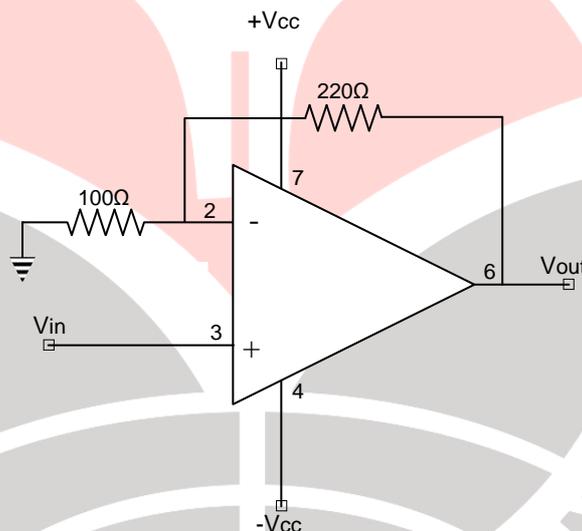
Komponen-komponen yang digunakan pada penelitian ini adalah seperti yang ditunjukkan pada gambar di atas. Untuk R_s salah satu hambatannya adalah variabel resistor, sehingga nilai hambatan pada R_s dapat divariasikan. Secara matematis nilai tegangan yang dihasilkan dari rangkaian konverter ini dapat dinyatakan sebagai berikut.

$$\begin{aligned} V_{out} &= 2,09 \cdot f_{in} \cdot \frac{R_t}{R_s} (R_t \cdot C_t) \\ &= 2,09 \cdot f_{in} \cdot \frac{100k\Omega}{12k\Omega} (6,8k\Omega \cdot 0,01\mu F) \\ &= 2,09 \cdot f_{in} \cdot 5,667 \cdot 10^{-4} \Omega F \\ &= 1,184 \cdot 10^{-3} \Omega F \cdot f_{in} \end{aligned}$$

Tegangan output dari rangkaian konverter sangat dipengaruhi oleh frekuensi yang dihasilkan oleh rangkaian osilator. Berdasarkan persamaan tersebut dapat dinyatakan bahwa besarnya tegangan output sebanding dengan dengan besarnya frekuensi input.

3.2.5 Rangkaian Op Amp

Rangkaian op amp berfungsi sebagai rangkaian untuk penguat operasi. Salah satu aplikasi dari rangkaian op amp diantaranya adalah penguat non inverting.



Gambar 3.5 Penguat Non Inverting Amplifier

Rangkaian penguat non inverting hampir sama dengan rangkaian penguat inverting, perbedaannya adalah terletak pin yang menjadi tegangan inputnya. Secara matematis, persamaan untuk penguat non inverting dinyatakan sebagai berikut :

$$V_o = \frac{R_i + R_f}{R_i} V_i \dots\dots\dots (3.7)$$

atau :

$$V_o = \left(\frac{R_f}{R_i} + 1 \right) V_i \dots\dots\dots (3.8)$$

Hasil tegangan output dari penguat non inverting ini akan lebih dari satu dan selalu positif. Rangkaian penguat non inverting adalah seperti ditunjukkan pada gambar berikut ini.

Komponen yang digunakan untuk membuat rangkaian penguat non inverting adalah IC OP07 dan resistor. Resistor yang digunakan memiliki nilai hambatan $R_s = 220\Omega$ dan $R_i = 100\Omega$. Secara matematis penguatan yang dihasilkan oleh penguat non inverting tersebut adalah sebagai berikut.

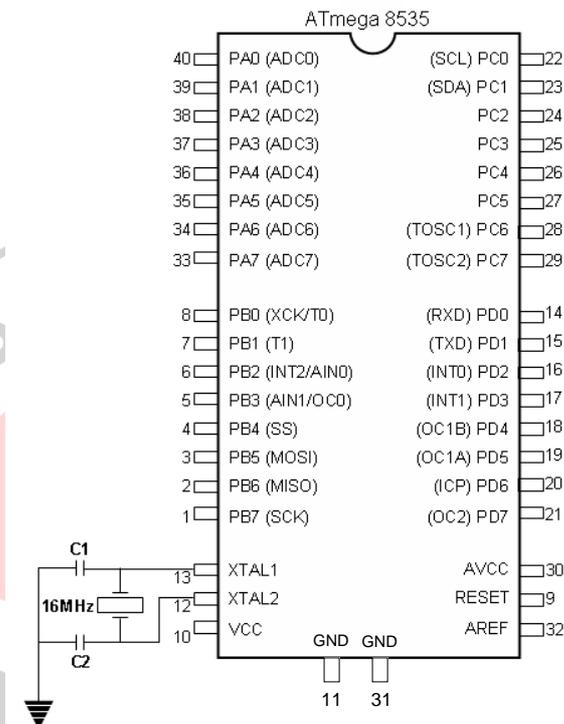
$$\begin{aligned} V_o &= \left(\frac{R_f}{R_i} + 1 \right) V_i \\ &= \left(\frac{220\Omega}{100\Omega} + 1 \right) V_i \\ &= 3,2 V_i \end{aligned}$$

Sehingga besar penguatan yang dihasilkan oleh rangkaian penguat tersebut adalah 3,2 kali penguatan.

3.2.6 Sistem Minimum Mikrokontroler ATmega8535

Rangkaian sistem minimum adalah rangkaian minimal dimana chip mikrokontroler dapat bekerja. Mikrokontroler ini bekerja dengan bantuan kristal yang besarnya 16 MHz, dan komponen lainnya. Bagian ini merupakan subsistem dari mikrokontroler yang berfungsi untuk membangkitkan clock pada mikrokontroler. Clock diperlukan untuk mensinkronkan proses yang sedang berlangsung dalam mikrokontroler, rangkaian ini tersusun atas komponen kristal dan dua buah kapasitor. Mikrokontroler bekerja dengan frekuensi clock 16 MHz

yang diperoleh dari osilator kristal. Ini berarti mikrokontroler bekerja dengan frekuensi sebesar 16 MHz untuk setiap kali menjalankan satu perintah program.

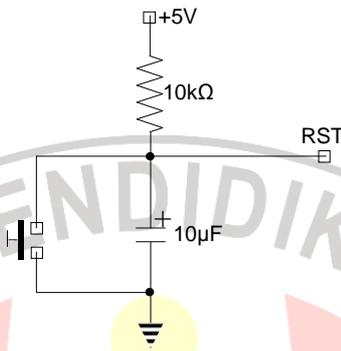


Gambar 3.6 Sistem Minimum ATmega8535

Mikrokontroler, mempunyai sistem pewaktuan CPU 12 siklus clock. Artinya setiap 12 siklus yang dihasilkan oleh resonator maka akan menghasilkan satu siklus mesin. Nilai ini yang akan menjadi acuan waktu operasi CPU. Untuk mendesain sistem mikrokontroler kita memerlukan sistem clock, sistem ini bisa dibangun dari *clock eksternal* maupun *clock internal*.

System minimum mikrokontroler ATmega8535 sangat sederhana di mana hanya dengan menghubungkan VCC dan AVCC ke +5V serta GND dan AGND ke ground. Selain itu terdapat rangkaian reset yang berfungsi untuk mereset sistem hingga proses bisa dijalankan mulai dari awal program. Chip akan reset jika

tegangan catu nol atau pin RST dipaksa nol. Tombol reset dapat ditambahkan dengan rangkaian sebagai berikut.



Gambar 3.7 Rangkaian Reset

3.2.7 Analog to Digital Converter (ADC)

Mikrokontroler ATmega8535 memiliki fasilitas Analog to Digital Converter (ADC) yang sudah tersimpan dalam chip. Fitur ADC internal inilah yang menjadi salah satu kelebihan mikrokontroler ATmega8535 bila dibandingkan dengan beberapa jenis mikrokontroler yang lain. Dengan adanya ADC internal ini kita tidak akan direpotkan lagi dengan kompleksitas hardware saat membutuhkan proses perubahan sinyal dari analog ke digital seperti yang harus dilakukan jika kita memakai IC ADC eksternal.

ATmega8535 memiliki resolusi ADC 10 bit dengan 8 channel input. Rangkaian internal ADC memiliki catu daya tersendiri yaitu pada pin AVCC. Tegangan AVCC harus sama dengan tegangan VCC $\pm 0,3V$.

Data hasil konversi ADC dirumuskan sebagai :

- Konversi tunggal :

$$ADC = \frac{V_{IN} \cdot 1024}{V_{REF}}$$

V_{in} : tegangan masukan pada pin yang dipilih

V_{ref} : tegangan referensi yang dipilih

- Penguat beda :

$$ADC = \frac{(V_{POS} - V_{NEG}) \cdot Gain \cdot 512}{V_{REF}}$$

V_{pos} : tegangan masukan pada pin positif

V_{neg} : tegangan masukan pada pin negatif

$Gain$: faktor penguatan

V_{ref} : tegangan referensi yang dipilih

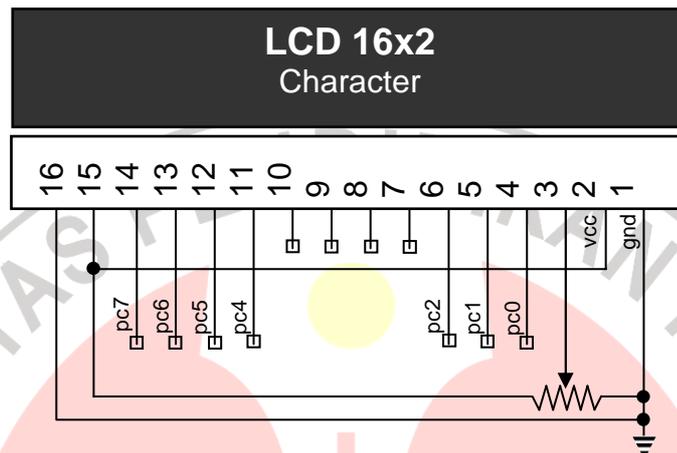
3.2.8 Rangkaian Pemrograman (Downloader)

Mikrokontroler dapat bekerja sesuai dengan tujuan yang diharapkan, yaitu dengan memrogram atau memberikan program tertentu pada chipnya. Program yang dibuat harus didownload atau dimasukkan ke dalam chip tersebut, yaitu dengan cara mendownload dari komputer ke chip. Agar program yang dibuat di komputer dapat dipindahkan ke chip dibutuhkan sambungan yang dinamakan *Downloader*. Pada penelitian ini downloader yang digunakan yaitu *Downloader ISP (In System Programming)*.

3.2.9 LCD

LCD digunakan untuk menampilkan besaran yang terukur. Mikrokontroler memerintahkan LCD untuk menampilkan karakter-karakter berupa huruf, angka

atau juga simbol matematis lainnya. LCD yang digunakan pada penelitian ini yaitu LCD 2×16. Artinya LCD ini pada sekali tampilan, maksimal dapat menampilkan 16 kolom karakter sebanyak 2 baris.

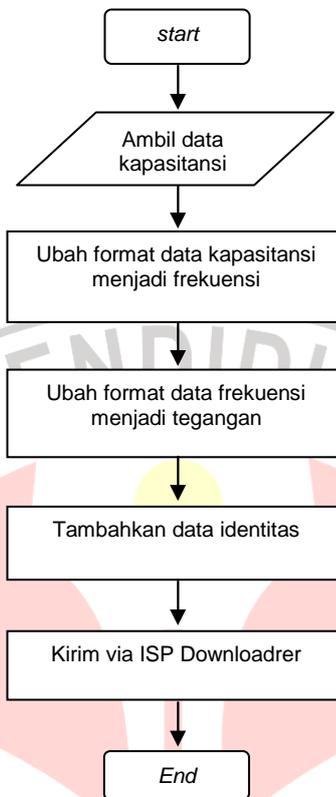


Gambar 3.8 Rangkaian interface mikrokontroler - LCD

Untuk dapat menampilkan hasil dari mikrokontroler pada LCD maka diperlukan rangkaian interface mikrokontroler-LCD. Rangkaian interface tersebut ditunjukkan pada gambar 3.9 di atas, dengan pin-pin pada LCD tersebut terhubung pada port C mikrokontroler. Untuk membuat tampilan LCD yang baik maka diperlukan rangkaian pendukung yang dapat mengatur kekontrasan matriks-matriks pada LCD, yaitu dengan menggunakan variabel resistor. Variabel resistor yang digunakan adalah sebesar 50 k Ω .

3.3 Perancangan dan Pembuatan Perangkat Lunak (Software)

Perangkat lunak berfungsi untuk mengendalikan sistem pada perangkat keras. Pada tahap awal pembuatan perangkat lunak, diperlukan rancangan diagram alur sebagai panduan dalam hal penyusunan instruksi dan pembuatan program.



Bagan 3.3 Diagram Alir Perangkat Lunak

Perangkat lunak dibuat untuk menunjang kerja dari perangkat keras yang telah dibuat untuk tahap pemrosesan data pada mikrokontroler. Bahasa pemrograman yang digunakan adalah bahasa pemrograman C dengan pertimbangan, bahasa C ini mudah untuk dimengerti dan sederhana.

Program yang dipakai untuk mendownload software yang telah dirancang menggunakan CAVR (*Code Vision AVR*). Digunakan program ini berdasarkan pertimbangan bahwa dalam mendownload dari *.hex* atau *.dec* ke dalam mikrokontroler tidak diperlukan program penunjang lainnya.

Bahasa pemrograman yang digunakan adalah bahasa C dan compiler yang digunakan adalah Code Vision AVR. Dalam struktur penulisan bahasa C, terdiri

atas empat blok, yaitu *header*, deklarasi konstanta global dan variabel, fungsi dan prosedur serta program utama. Pemrograman bahasa C yang didasarkan pada algoritma untuk sistem sensor kapasitansi dapat diuraikan sebagai berikut.

a. Bahasa pemrograman C blok *header*

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

// Alphanumeric LCD Module functions
#asm
    .equ_lcd_port=0x15 ;PORTC
#endasm
#include <lcd.h>

// LCD module initialization
lcd_init(16);
```

b. Bahasa pemrograman C blok deklarasi konstanta dan variabel

Konstanta dan variabel merupakan sebuah tempat untuk menyimpan data yang berada dalam memori. Konstanta berisi data yang nilainya tetap dan tidak dapat diubah selama program dijalankan, sedangkan variabel berisi data yang bisa berubah nilainya pada saat program dijalankan.

```
char buf[33];
```

c. Bahasa pemrograman C blok fungsi dan prosedur

```
void main(void)
{
    // Declare your local variables here
    unsigned char dtadc;
```

d. Bahasa pemrograman C blok utama

```
while (1)
{
    // Place your code here
    dtadc=read_adc(0);
    PORTB=dtadc;
    lcd_gotoxy(0,1);
```

```
sprintf(buf, "%3d", dtadc);  
lcd_puts(buf);  
delay_ms(100);  
};  
  
if (dtadc>x1&&dtadc<y1);  
{  
lcd_gotoxy(0,0);  
lcd_putsf("jenis cairan 1");  
}  
  
if (dtadc>x2&&dtadc<y2);  
{  
lcd_gotoxy(0,0);  
lcd_putsf("jenis cairan 2");  
}  
  
if (dtadc>x3&&dtadc<y3);  
{  
lcd_gotoxy(0,0);  
lcd_putsf("jenis cairan 3");  
}  
//dan seterusnya...
```

