

BAB III

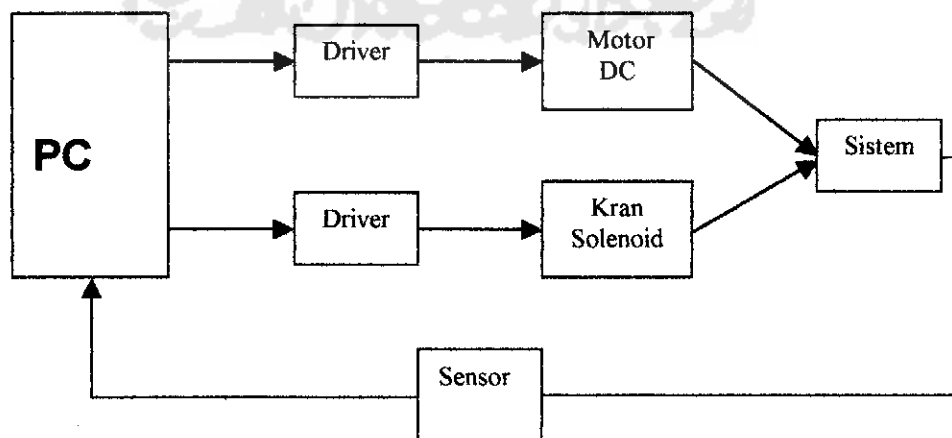
PERANCANGAN SISTEM

3.1 Pendahuluan

Antara sistem digital (sebagai pengontrol) dan sistem analog (sebagai peralatan yang dikontrol) harus terdapat suatu jembatan yang menghubungkan kedua sistem tersebut. Jembatan ini disebut sistem *interface*.

Dalam sistem Monitoring Posisi dan Pengontrolan Pengisian Air Dalam Gelas Berbasis PC ini, komputer merupakan sistem digital yang menjadi otak dari sistem secara keseluruhan. Sistem digital ini membaca kondisi dari sistem analog melalui sistem *interface* dan mengontrol sistem analog melalui sistem *interface*.

Antarmuka (*interface*) yang digunakan untuk sistem ini yaitu *port parallel* komputer dengan konektor DB-25. Secara teoritis antarmuka tersebut akan menghubungkan sistem analog dengan komputer sehingga sistem dapat dimonitor dan dikontrol.



Gambar 3.1 Diagram Blok Monitoring Posisi dan Pengisian Air Berbasis PC

Untuk mewujudkan sistem Monitoring Posisi dan Pengisian Air Dalam Gelas Berbasis PC, proses perancangannya dilakukan secara bertahap yaitu :

1. Perancangan mekanis
2. Perancangan *hardware*
3. Perancangan *software*

3.2 Perancangan Mekanis

Mekanis yang dirancang disesuaikan dengan cara kerja sistem ini. Pada mekanis ini terdapat empat dudukan gelas yang akan diisi secara *rolling* dan motor dc sebagai penggerak. Beberapa alat bantu juga digunakan pada mekanis ini seperti laker, as dan spi (ring penahan). As yang panjangnya kurang lebih 15 cm dipasang horizontal berfungsi untuk menghubungkan rotor motor dengan dudukan gelas. Untuk menjaga keseimbangan putaran motor dipasangkan laker pada as. Spi juga terpasang pada as tepat diatas laker, berfungsi untuk menahan beban terhadap motor atau dengan kata lain beban yang berupa dudukan gelas tidak sepenuhnya ditanggung oleh motor. Hal ini merupakan antisipasi untuk mencegah terjadinya kerusakan pada motor.

Empat buah dudukan gelas digabung dalam suatu media berbentuk lingkaran (piringan). Jarak tiap-tiap dudukan harus sama karena sangat berpengaruh ketika sistem beroperasi. Masing-masing dudukan diberi lubang yang ukurannya lebih kecil dari diameter gelas. Lubang tersebut berfungsi sebagai tempat keluar masuknya cahaya untuk LDR. Untuk dudukan motor

dibuat dari aluminium yang berbentuk kotak segi empat. Dudukan LDR ditempatkan pada kotak ini dan harus disesuaikan dengan dudukan gelas.

3.3 Perancangan *Hardware*

3.3.1 Rangkaian *Catu Daya*

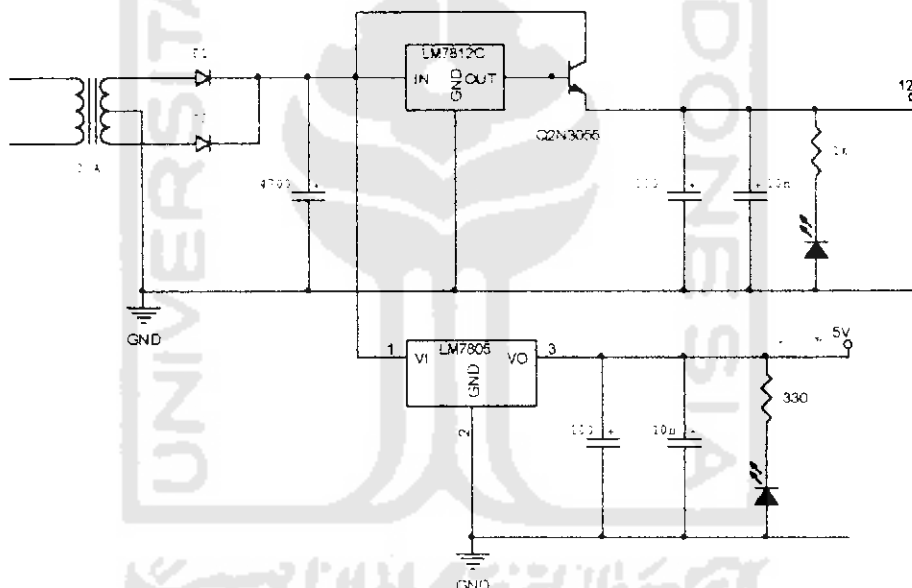
Pencatu daya berfungsi untuk mengubah daya masukan AC menjadi daya keluaran DC. Untuk menjalankan sistem analog pada tugas akhir ini dibutuhkan catu daya sebesar 12 volt, 5 volt, dan 3 volt untuk driver pengendali. Tegangan 12 volt digunakan untuk mencatu relay dan *solenoid valve*, tegangan 5 volt untuk mencatu untai sensor LDR dan optokopler, sedangkan tegangan 3 volt digunakan untuk menggerakkan motor.

3.3.1.a *Catu Daya 12V*

Untuk memperoleh tegangan yang dimaksud digunakan trafo *step down* 3A untuk menurunkan tegangan 220V menjadi 12V. Keluaran trafo yang masih bolak balik tersebut disearahkan dengan dua buah dioda penyearah sehingga menghasilkan tegangan DC. Tegangan keluaran ini perlu diberi *filter* berupa kapasitor 4700 μ F/16v untuk menghasilkan tegangan keluaran DC yang lebih rata. Agar catu daya ini lebih sempurna perlu ditambah regulator tegangan (7812) sehingga keluarannya benar-benar stabil. Karena keluaran arus dari regulator kecil, maka perlu diperkuat oleh sebuah transistor 2N3055 agar mampu mencatu arus yang lebih besar. Catu 12V ini akan digunakan untuk mencatu, relay 12V dan solenoid 12V.

3.3.1.b Catu Daya 5V

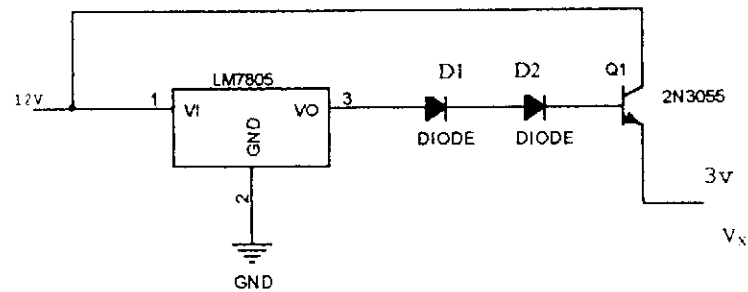
Untuk mendapatkan catu 5V, *Output* dari catu 12V langsung dihubungkan ke *input* LM 7805. Untuk menyimpan arus agar dalam pengoperasian tidak terjadi short pada tegangan, *output* dari regulator dihubungkan dengan kapasitor 100 μ F/16V dan 10n. Catu ini digunakan untuk rangkaian sensor cahaya yang menggunakan IC TTL 74LS14. Rangkaian logika TTL selalu menggunakan catu daya 5V DC.



Gambar 3.2 Untai Catu Daya 12V dan 5V

3.3.1.c Catu Daya 3V

Tegangan 3V didapat dari dua buah dioda silikon yang terhubung seri dan transistor NPN 2N3055. *Output* catu 5V langsung dihubungkan dengan dioda, sedangkan kolektor transistor terhubung dengan 12V.



Gambar 3.3 Untai Catu Data 3V

Dari gambar rangkaian diatas diperoleh persamaan :

$$V_X = V_0 - (D1 + D2 + V_{BE}) \quad (3.1)$$

Persamaan (3.1) diperoleh dari hasil pengukuran rangkaian diatas. Rangkaian ini merupakan rangkaian pembagi tegangan dengan menggunakan dioda silicon. Ketika tegangan melewati masing-masing dioda, akan ada penurunan tegangan bernilai 0,7V pada dioda, dan dikurangi lagi dengan V_{BE} yang bernilai 0,7V. Dengan menggunakan persamaan (3.1) didapat nilai V_0 yaitu:

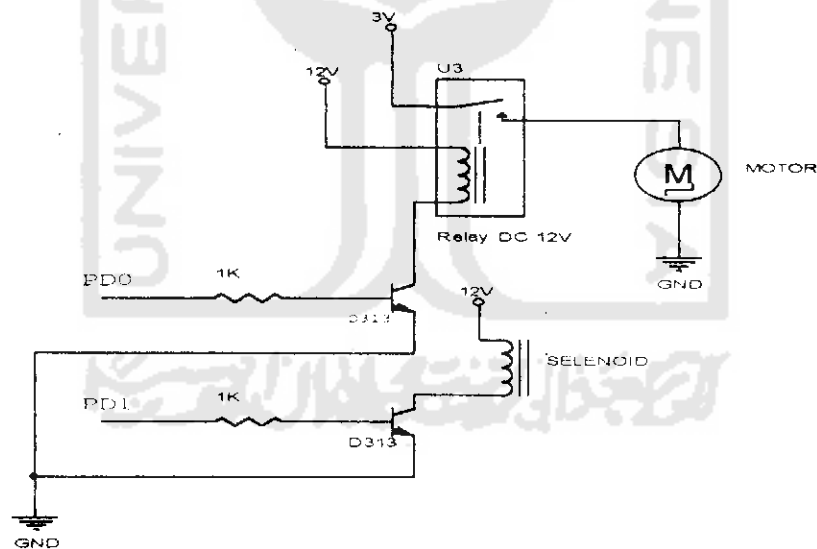
$$\begin{aligned} V_0 &= 5V - (0,7 + 0,7 + 0,7) \\ &= 5V - 2,1V \\ &= 2,9V. \end{aligned}$$

Catu ini digunakan untuk memperlambat kecepatan motor. Arus kolektor transistor yang diperoleh dari tegangan 12V DC berfungsi untuk menghindari drop tegangan pada saat motor dijalankan.

3.3.2 Untai Driver

Untai driver pengendali fungsinya sebagai *switch* otomatis, yang digunakan untuk mengendalikan motor dan *solenoid valve*. Driver ini menerima masukan dari *output port parallel*.

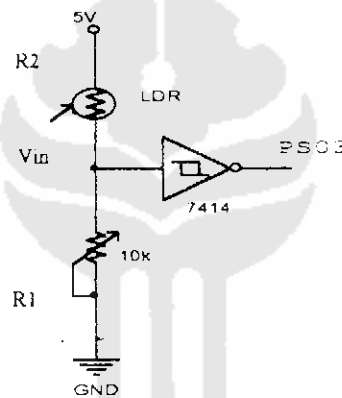
Transistor NPN pada driver ini difungsikan sebagai *switching*, dimana ketika *basis* pada transistor tersebut mendapat tegangan, maka arus *collector* masuk ke *emitor* yang mengakibatkan arus DC yang melalui kumparan relai mengalir menuju *ground* sehingga relai tersebut menjadi aktif yang kemudian dengan aktifnya relai tersebut maka beban akan mendapat tegangan, maka beban mendapat tegangan.



Gambar 3.4 Untai Driver

3.3.3 Untai Sensor LDR

Sensor LDR (*Light Dependent Resistor*) digunakan sebagai pendeteksi keberadaan gelas yang dapat dimonitor langsung dengan komputer. Pendeteksian ini dilakukan sebelum memulai pengisian dilakukan. Digunakan empat buah LDR dan VR (*Variable Resistor*) atau disebut juga trimpot. IC 74LS14 merupakan suatu IC TTL yang berisi enam gerbang NOT yang berfungsi sebagai inverter.



Gambar 3.5 Untai Sensor LDR

Berdasarkan rangkaian diatas dapat diturunkan persamaan untuk tegangan masukan (V_{in}) yaitu :

$$V_{in} = (R1/R1 + R2) \times V_{sumber} \quad (3.2)$$

Apabila LDR tidak dikenai cahaya, maka tahanan yang ada pada LDR sangat besar sekali, dapat dikatakan tidak berhingga. Dengan menggunakan persamaan (3.2) akan didapat hasil sebagai berikut :

$$\begin{aligned} V_{in} &= (1/1 + \infty) \times 5V \\ &= \approx 0 \end{aligned}$$

Karena V_{in} mendekati 0, maka V_{out} akan menjadi 1. Kalau LDR dikenai cahaya, maka tahanan LDR menjadi lebih kecil. V_{in} akan menjadi *high*. V_{in} dalam kondisi *high* harus mengikuti level TTL. Oleh karena itu untuk mendapatkan nilai tegangan V_{in} ini digunakanlah trimpot. V_{out} akan menjadi *low* jika V_{in} dalam kondisi *high*.

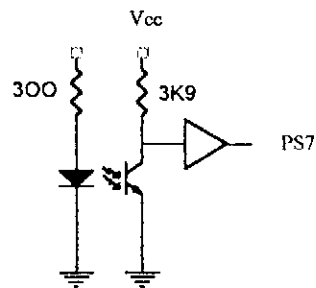
Berdasarkan level TTL untuk V_{in} yaitu :

- $V_{in} > 2,5\text{Volt}$ maka V_{out} akan 0
- $V_{in} < 2,5\text{Volt}$ maka V_{out} akan 1

Pada untai ini diberi indikator untuk masing-masing LDR yaitu LED. Ketika V_{out} sama dengan 1, LED akan menyala, begitu juga sebaliknya. Untuk pendeteksian gelas dengan monitoring, ketika V_{out} 1 gelas akan terdeteksi. Untuk V_{out} sama dengan 0, gelas tidak terdeteksi.

3.3.4 Untai Sensor Optokopler

Pada waktu pendeteksian gelas, motor berputar satu kali putaran. Agar motor berhenti secara otomatis tepat pada posisi awal pengisian gelas digunakanlah Optokopler.



Gambar 3.6 Untai Sensor Optokopler

Pada rangkaian diatas R1 digunakan untuk membatasi arus yang masuk LED. Berdasarkan data sheet arus maksimal yang dapat melewati LED adalah 20mA. Sesuai dengan karakteristik LED yaitu :

$$V_{LED} = 1,5V - 3V$$

$$I_{LED} = 10mA - 20mA$$

Untuk mencari besar nilai resistor yang digunakan, dapat menggunakan persamaan berikut :

$$R = \frac{V_{cc} - V_{LED}}{I_{LED}} \quad (3-3)$$

Misalkan $V_{LED} = 2V$ dan $I_{LED} = 10mA$, maka :

$$R = \frac{5 - 2}{10^{-3}}$$

$$R = 300 \text{ Ohm}$$

Fototransistor pada optokopler akan *On* bila cahaya IRED mengenainya. Fototransistor akan bekerja sebagai saklar sehingga tegangan pada kolektor akan bernilai nol. Berdasarkan rangkaian yang dibuat, logika untuk masukan *inverter* akan 0 bila ada cahaya yang mengenai fototransistor, sehingga keluaran logikanya menjadi 1. Kalau

fototransistor tidak dikenai cahaya, tegangan pada kolektor akan sama dengan Vcc, sehingga logika untuk keluaran akan 0. Untuk menghindari kerusakan pada fototransistor, digunakan resistor untuk membatasi arus ketika fototransistor dikenai cahaya.

3.4 Perancangan Perangkat Lunak (*Software*)

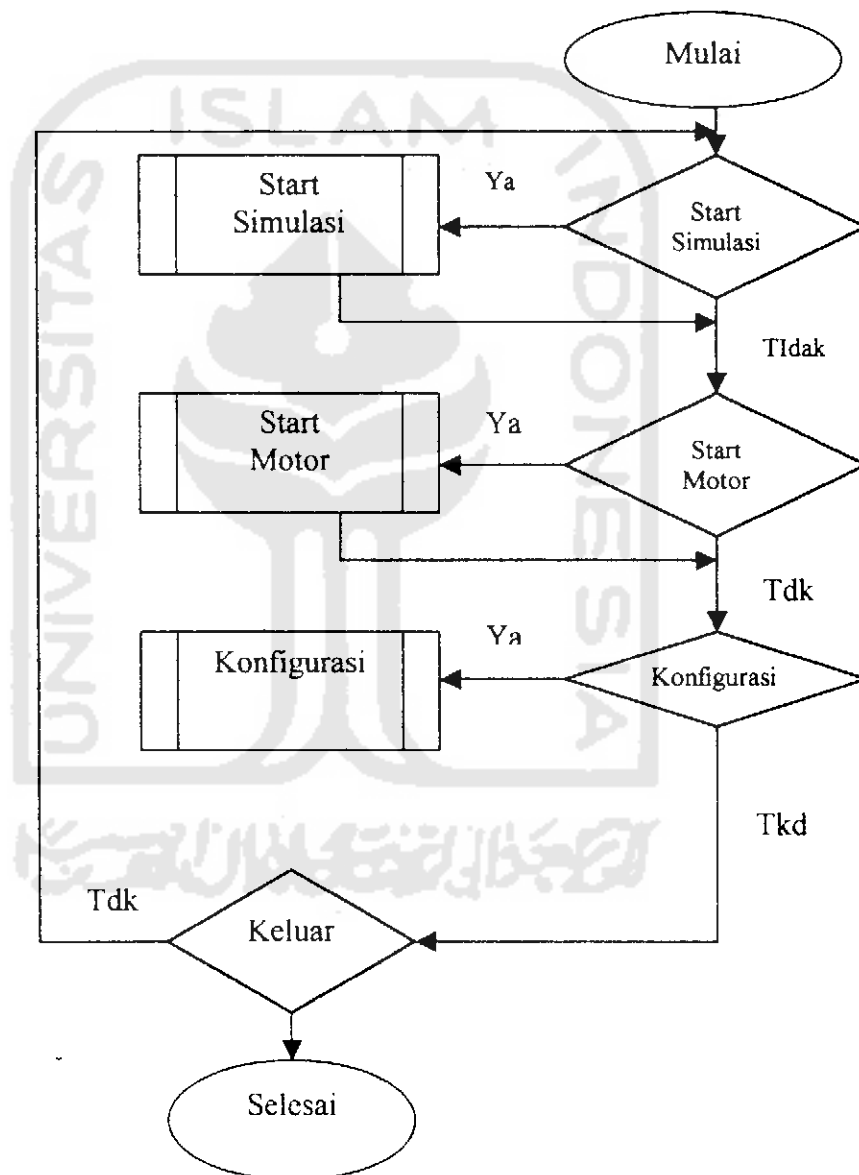
Membuat program berarti merencanakan serangkaian instruksi yang dapat dimengerti oleh komputer dan disusun menurut urutan yang logis. Pekerjaan tentang penulisan program tersebut tergantung pada pengertian tentang persoalan yang dihadapi dan pada struktur atau rencana penyelesaian-penyelesaiannya. Pekerjaan membuat program ini dipecahkan dalam beberapa tahapan, yaitu :

- Menyatakan persoalan yang dihadapi se jelas mungkin dan secara terperinci
- Menyusun Algoritma, yaitu prosedur penyelesaian persoalan secara bertahap
- Menyusun *Flow Chart* atau peta prosedur penyelesaian (diagram alir) yang menguraikan algoritma secara terperinci
- Menerjemahkan peta prosedur penyelesaian dalam bahasa yang dapat diproses oleh komputer

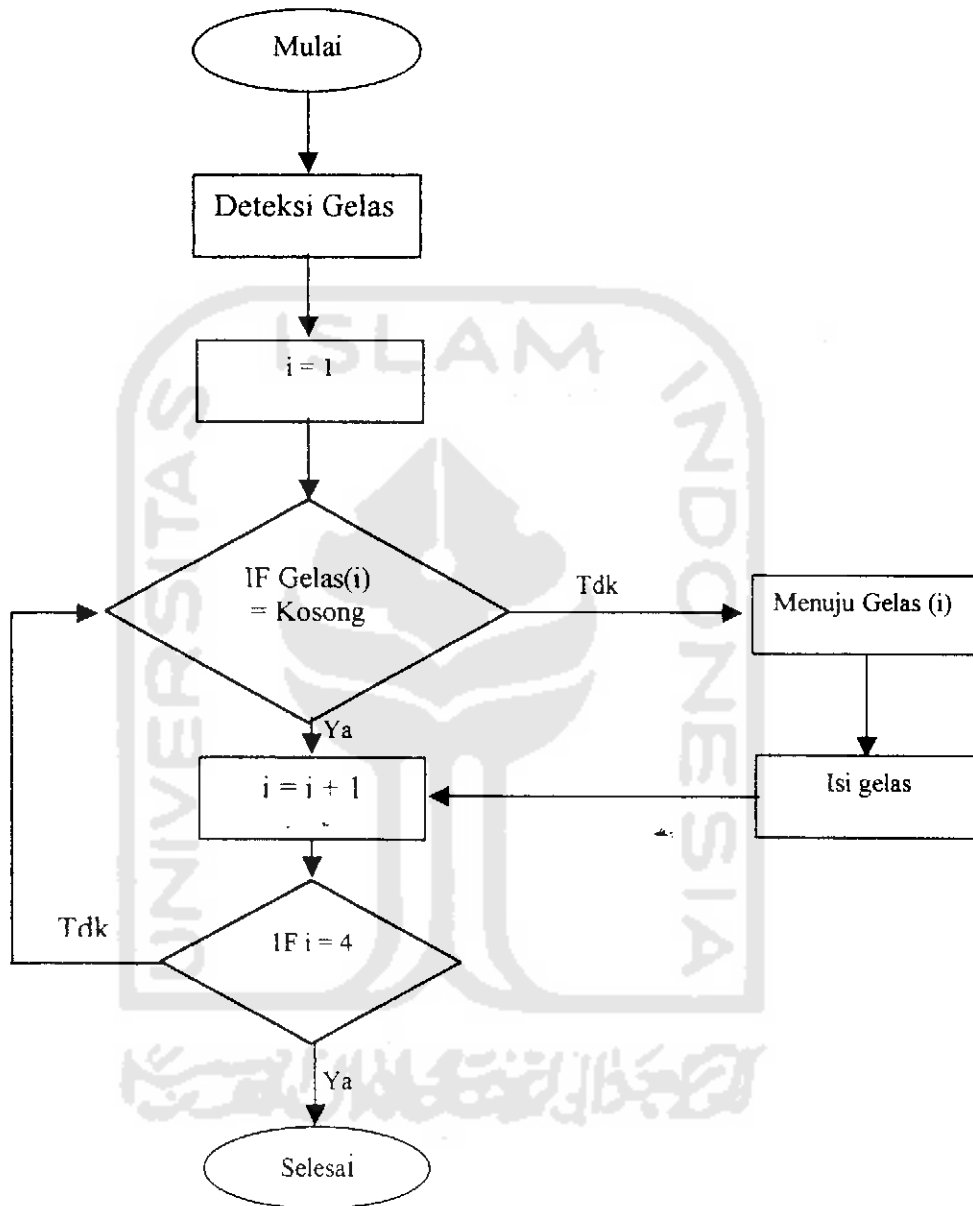
Menyusun algoritma merupakan tahapan yang penting setelah persoalan didefinisikan. Pada dasarnya algoritma ini dari langkah-langkah sederhana yang dapat diartikan sebagai penjabaran proses dari keadaan awal ke keadaan akhir. *Flow chart* sangat membantu dalam pembuatan program yang terstruktur dengan

baik. *Flow chart* adalah gambaran dari penyelesaian suatu masalah langkah demi langkah dengan menggunakan simbol-simbol tertentu.

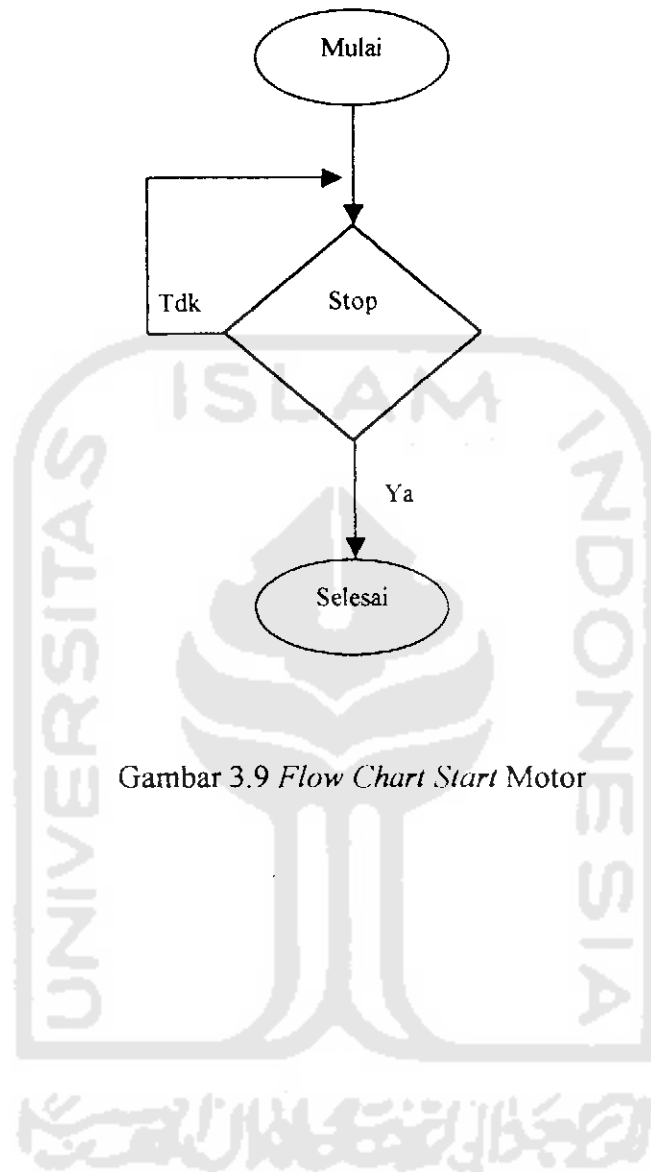
3.4.1 *Flow Chart* Program



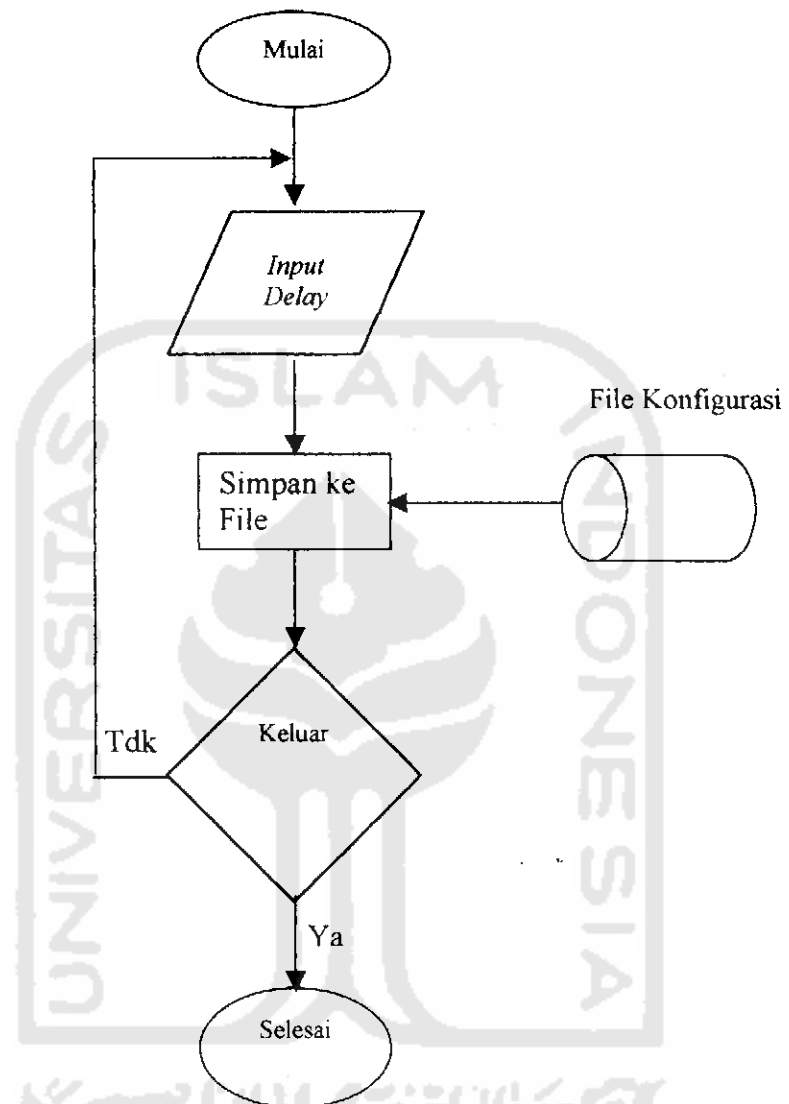
Gambar 3.7 *Flow Chart* Menu Utama



Gambar. 3.8 *Flow Chart Start Simulasi*



Gambar 3.9 *Flow Chart Start Motor*



Gambar 3.10 *Flow Chart* Konfigurasi

3.4.2 Proses Pendeteksian Gelas dan Pengisian Air

Proses pengisian air kedalam gelas dilakukan secara bergantian sesuai dengan urutan program. Sebelum proses pengisian air berjalan terlebih dahulu dilakukan pendeteksian terhadap gelas yang akan diisi. Awal proses pengisian dimulai apabila motor berhenti tepat pada optokopler. Berikut adalah listing program dari perancangan software untuk menjalankan sistem.

- **Alamat Port Untuk Input Dan Output**

Pada program ini ada dua alamat yang digunakan. Untuk *output* data menggunakan alamat h378 atau 888 dalam desimal.

```
PrinterOutAddress = &h378 ( 888 desimal)
```

Sedangkan input data menggunakan alamat h379 atau 889 dalam desimal

```
PrinterInAddress = &h379 ( 889 desimal )
```

- **Fungsi Untuk Membaca Data Dari Port Printer**

Pada program ini fungsi *inprinter* yang digunakan untuk membaca data dari *hardware* dengan memanggil fungsi in8255 bawaan dari PPI 8255. Input data dari fungsi ini berupa integer (bilangan bulat).

```
Function InPrinter() As Integer
```

```
InPrinter = In8255(PrinterInAddress)
```

```
End Function
```

- **Fungsi Untuk Mengirim Data Ke Port Printer**

Fungsi ini akan mengirim data yang *diinputkan* oleh program dengan memanggil fungsi `out8255` untuk dikirim ke *hardware* dengan alamat *printeroutaddress*, dengan data *output* sesuai kebutuhan program.

```
Sub OutPrinter(strData As String)
Dim x As Integer
x = Out8255(PrinterOutAddress, strData)
End Sub
```

- **Fungsi Untuk Menjalankan Motor**

Fungsi start motor digunakan untuk menjalankan motor dengan memanggil fungsi *outprinter* yang sudah kita definisikan sebelumnya dengan data *output* ("1").

```
Public Sub StartMotor()
OutPrinter ("1")
End Sub
```

- **Fungsi Untuk Memberhentikan Motor/Menutup Selenoid**

Fungsi ini untuk mengentikan motor dan solenoid dengan memanggil fungsi *outprinter* dengan data *output* ("0")

```
Public Sub StopMotor()
OutPrinter ("0")
End Sub
```


- **Fungsi Untuk membuka solenoid**

Fungsi open solenoid digunakan untuk membuka solenoid dengan memanggil fungsi *outprinter* dengan data *output* ("2")

```
Public Sub OpenSolenoid()
    OutPrinter ("2")
End Sub
```

- **Listing Untuk deteksi gelas**

Untuk awal pendeteksian program akan memanggil fungsi *outprinter* dengan data *output* ("1") untuk menjalankan motor

deteksi_gelas:

```
OutPrinter ("1")
```

Kemudian program akan memanggil fungsi *inprinter* untuk membaca input dari hardware. Jika input kurang dari 128 maka label info sama dengan sedang deteksi gelas, kemudian menjalankan *background* proses (*DoEvents*), jika simulasi distop maka akan selesai.

```
If InPrinter < 128 Then
    lb_info.Caption = "Sedang mendeteksi
    gelas..."
    DoEvents
    If Not StatusSimulasi Then Exit Sub
    GoTo deteksi_gelas
End If
```

Setelah *inprinter* lebih besar atau sama dengan 128 maka motor akan berhenti hal ini disebabkan cahaya IRED tidak mengenai fototransistor.

Kemudian variabel *indata* diisi dengan *inprinter-7*. Setelah *inprinter* dikurangi 135 dikonversi menjadi biner. Hasil konversi ini kemungkinan bisa lebih dari empat digit padahal program hanya membutuhkan empat digit pertama. Empat digit pertama susunanya dibalik. Contoh: misalkan data 101100 maka hasilnya 1101

```
OutPrinter ("0")
inData = InPrinter - 7
strData = Dec2Bin(inData - 128)
' cek gelas
```

Digit pertama dari hasil diatas merupakan inisialisasi untuk gelas 4, digit kedua untuk gelas 3, digit ketiga untuk gelas 2 dan digit keempat untuk gelas 1.

```
strData = StrReverse(Left(strData, 4))
gl_4 = Left(strData, 1)
strData = Right(strData, 3)
gl_3 = Left(strData, 1)
strData = Right(strData, 2)
gl_2 = Left(strData, 1)
gl_1 = Right(strData, 1)
```

Setelah itu program akan menampilkan info gelas 1, 2, 3, dan 4.

```
If gl_1 = 1 Then
```

```
  AdaGelas (1)
```

```
Else
```

```
  TakAdaGelas (1)
```

```
End If
```

```
If gl_2 = 1 Then
```

```
  AdaGelas (2)
```

```
Else
```

```
  TakAdaGelas (2)
```

```
End If
```

```
If gl_3 = 1 Then
```

```
  AdaGelas (3)
```

```
Else
```

```
  TakAdaGelas (3)
```

```
End If
```

```
If gl_4 = 1 Then
```

```
  AdaGelas (4)
```

```
Else
```

```
  TakAdaGelas (4)
```

```
End If
```

- **Listing Untuk Mengisi Gelas**

Awal pengisian gelas ke i

```
isi_gelas_1:
```

```
If Not AdakahGelas(i) Then
```

Cek apakah gelas ke i ada, jika tidak ada maka label info menyatakan gelas ke i tidak ada.

```
lb_info.Caption = "Gelas i tidak ada"
```

```
Refresh
```

```
'Sleep (1000)
```

```
GoTo isi_gelas_i+1
```

```
End If
```

Sebelum gelas diisi ada *delay*.

```
Sleep (DelaySebelumIsi)
```

Setelah itu membuka solenoid, label info menyatakan sedang mengisi gelas i.

```
OutPrinter ("2")
```

```
lb_info.Caption = "Sedang mengisi gelas i"
```

Selanjutnya program akan melakukan simulasi pengisian gelas ke i dalam kurun waktu pengisian gelas yang telah ditetapkan sebelumnya.

```
' Isi Gelas i
```

```
For i = 1 To 300
```

```
DoEvents
```

```

If Not StatusSimulasi Then Exit Sub
    Sleep (DelayPengisianGelas / 300)
    DrawIsiGelas 1, Round(i / 3)
Next i

```

- **Listing Pengisian Gelas Selesai**

```

isi_gelas_selesai:
    cmdRun.Caption = "Start Simulasi"
    StatusSimulasi = False
    cmdStart.Enabled = True
    cmdConfig.Enabled = True
    MsgBox "Simulasi pengisian gelas
selesai..."

```

Msgbox merupakan suatu informasi yang menyatakan simulasi pengisian gelas selesai. Informasi ini ditampilkan diluar dari form utama.