

BAB III

PERANCANGAN ALAT

3.1. Konsep Perancangan

Dalam perancangan peralatan ada beberapa perangkat pendukung antara lain :

- a. Unit komputer.
- b. Rangkaian komunikasi serial MAX-232
- c. Rangkaian *master* mikrokontroler.
- d. Rangkaian *slave* mikrokontroler 3 buah.
- e. Rangkaian input berupa tombol *push button*.
- f. Perangkat lunak "*Visual Basic*"

Secara keseluruhan diagram blok dapat ditunjukkan pada Gambar 3.1.

3.2. Cara Kerja Alat.

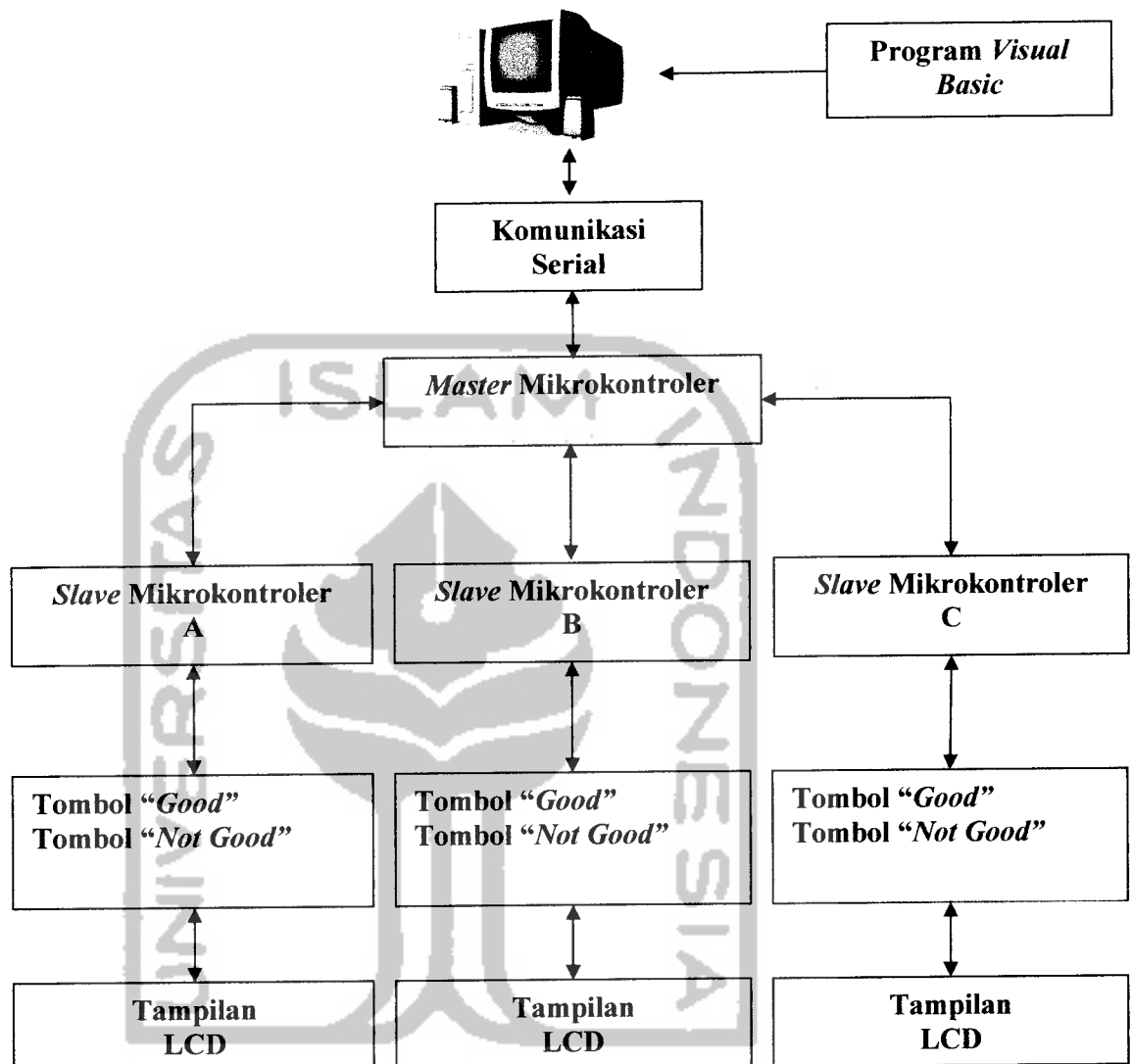
3.2.1. Rangkaian LCD.

Rangkaian tampilan berfungsi untuk menampilkan karakter yang akan ditampilkan. Rangkaian ini berupa modul tampilan kristal cair matriks titik. Modul ini terdiri dari beberapa bagian, yaitu :

- Bagian pengendali
- Bagian penggerak segment
- Bagian tampilan kristal cair

Modul ini memiliki 20 kaki masukan yang terdiri dari : 8 bit masukan data (DB0 – DB7), masukan pilihan register (RS) sebanyak 1 bit, masukan pengendali sinyal baca dan tulis (RW) 1 bit, masukan sinyal enable (E) 1 bit, masukan tegangan

catu positif (+ 5 V) yaitu masukan V_{DD} , masukan catu tegangan 0 volt (V_{SS}), masukan tegangan pengatur kecerahan tampilan (V_{ee}).



Gambar 3.1. Diagram blok alat kontrol produksi

Saluran data 8 bit DBO – DB7 merupakan sarana untuk memasukkan data alamat ROM dan RAM internal serta data tampilan kedalam modul. Masukan sinyal RS digunakan untuk memilih register-register yang berada dalam modul yaitu register perintah dan register data tampilan. Masukan sinyal E digunakan untuk mengaktifkan modul agar siap menerima data yang dikirimkan kepada modul. Bagian pengendali

berfungsi mengendalikan bekerjanya modul ini yaitu mengatur penerima data dari luar dan menampilkan data pada tampilan.

Untuk Cara kerja modul tampilan kristal cair adalah sebagai berikut :

1. Untuk proses masukkan data perintah internal :

Data 8 bit ditempatkan pada saluran data $DB_0 - DB_7$, masukan sinyal RS diberi masukan logika rendah untuk memilih register perintah internal, kemudian masukan sinyal R/W diberi logika rendah untuk melaksanakan operasi penulisan terhadap modul. Selanjutnya masukan sinyal E diberi masukan perubahan dari logika tinggi ke rendah. Sinyal ini berfungsi untuk memulai modul untuk segera beroperasi sesuai perintah yang dimasukkan.

2. Untuk proses memasukkan data tampilan :

Data 8 bit ditempatkan pada saluran data $DB_0 - DB_7$, masukan sinyal RS diberi masukan logika tinggi untuk memilih register tampilan, kemudian masukan sinyal R/W diberi logika rendah untuk melaksanakan operasi penulisan terhadap modul. Selanjutnya masukan sinyal E diberi masukan perubahan dari logika tinggi ke rendah. Sinyal E diberi masukan perubahan dari logika tinggi ke rendah. Sinyal E ini berfungsi untuk memulai kerja sesuai perintah yang telah dituliskan pada modul. Adapun hubungan mikrokontroler dengan LCD ditunjukkan pada Gambar 3.2.

Sedangkan alur inisialisasi penulisan LCD diperlihatkan pada Gambar 3.3.

Sebagai contoh dalam penulisan LCD untuk memunculkan tulisan "Nanang Hidayat" dalam bahasa *assembly* adalah dengan cara :

```
rs      bit    p1.2
rw      bit    p1.1
e       bit    p1.0
atas    bit    24h
```

```

;+++++ Insialisasi LCD ++++++
;+++++ Insialisasi LCD ++++++
;+++++ Insialisasi LCD ++++++

        lcall    tunda
        clr      rs
        mov      a,#00110000b
        lcall    tulis
        lcall    tunda
        mov      a,#00110000b
        lcall    tulis
        lcall    tunda
        mov      a,#00110000b
        lcall    tulis
        lcall    tunda

        mov      a,#00111111b    ; function set
        lcall    tulis
        mov      a,#00000001b    ; display clear
        lcall    tulis
        mov      a,#00000010b    ; return home position
        lcall    tulis
        mov      a,#00001100b    ; display off
        lcall    tulis
        mov      a,#00000110b    ; entry mode set

        clr      rs
        mov      a,#80h
        acall    tulis
        setb     rs
        mov      dptr,#good
        mov      a,#00h
        mov      atas,#16
rep:    clr      a
        movc     a,@a+dptr
        lcall    tulis
        inc     dptr
        djnz    atas,rep

tulis:  clr      rw    ;sub rutin untuk program tulis
        setb     e
        mov      p0,a
        acall    tunda
        clr      e
        setb     rw
        ret

tunda:  mov      20h,#0ffh
ul1:    mov      21h,#0ffh
ul2:    nop
        djnz    21h,ul2
        djnz    20h,ul1
        ret

good:   db      'NANANG HIDAYAT'
end.

```


3.2.2. Rangkaian *input push button*.

Pada rangkaian ini tombol *push button* yang digunakan terdiri atas tombol *GOOD* dan tombol *NOT GOOD*. Tombol *GOOD* terhubung pada port P3.0 rangkaian *Slave* Mikrokontroler sedangkan tombol *NOT GOOD* terhubung pada port P3.1 rangkaian *Slave* Mikrokontroler. Tombol ini bersifat *active low* dimana ketika berlogika 0 maka tombol akan aktif dan ketika berlogika 1 maka tombol tidak aktif.

3.2.3. Rangkaian *slave* mikrokontroler.

Rangkaian ini adalah rangkaian mikrokontroler dengan input berupa tombol dan output berupa tampilan dot matriks (LCD). Cara kerja rangkaian ini adalah ketika salah satu tombol *GOOD* atau *NOT GOOD* ditekan maka program akan menjalankan perintah penambahan. Data penambahan selanjutnya ditampilkan pada rangkaian LCD dimana baris pertama untuk data *GOOD* dan baris kedua untuk data *NOT GOOD*. Selanjutnya data-data tersebut dikirimkan ke komputer melalui rangkaian *Master* Mikrokontroler. Proses ini berjalan berulang-ulang ketika ada aktifitas tombol ditekan. Adapun rangkaian skematik dari *Slave* Mikrokontroler ditunjukkan pada Gambar 3.4.

Sedangkan program utama *slave* mikrokontroler dalam bahasa *assembler* adalah sebagai berikut :

```

START:
MOV  DATABUS, #0FFH
MOV  IE, #0
CLR  LCDCE
MOV  SP, #STACKHERE
MOV  R0, #GD
MOV  A, #0
MOV  B, #4
CALL STRFILL
MOV  CHGFLAG, #1
CALL LCDINIT
MOV  A, #0
MOV  DPTR, #LCDMESSAGE1
CALL LCDPRINT
MOV  A, #40H
MOV  DPTR, #LCDMESSAGE2

```

```

CALL LCDPRINT
MOV B, #10H
CALL DELAYLONG2
SJMP SENDQUALITYDATA

REPEATPROCESS:
MOV CHGFLAG, #0
JNB GDBTN, GDBUTTON
JNB NGBTN, NGBUTTON
SJMP SENDQUALITYDATA

GDBUTTON:
MOV R0, #GD
SJMP BUTTONPRESSED

NGBUTTON:
MOV R0, #NG

BUTTONPRESSED:
MOV B, #10H
CALL DELAYLONG
JNB GDBTN, $
JNB NGBTN, $
MOV CHGFLAG, #1
CALL INCWORD

SENDQUALITYDATA:
CALL SENDQLTDATA

SHOWQLTDATA:
MOV A, CHGFLAG
JZ LCDBUFFERREADY
MOV R1, #GD
MOV R0, #TMPDWORD
MOV B, #4
CALL STRCPY
MOV R0, #LCDR1+11
MOV R1, #TMPDWORD+0
MOV B, #5
CALL DCMLWORD
MOV R0, #LCDR2+11
MOV R1, #TMPDWORD+2
MOV B, #5
CALL DCMLWORD

LCDBUFFERREADY:
CALL LCDRFSH
MOV B, #20H
CALL DELAYLONG
SJMP REPEATPROCESS
END

```

3.2.4. Rangkaian *master* mikrokontroler.

Rangkaian ini adalah rangkaian mikrokontroler dengan ditambah rangkaian komunikasi serial dengan menggunakan IC MAX-232. Cara kerja rangkaian ini adalah ketika mendapatkan data dari salah satu *Slave* Mikrokontroler maka rangkaian *master* akan mengirimkan data tersebut menuju komputer untuk ditampilkan pada perangkat lunak Visual Basic. Pada perangkat lunak *visual basic* sendiri data-data *GOOD* dan

NOT GOOD akan ditampilkan secara langsung dan berulang-ulang. Adapun rangkaian skematik dari *Master* Mikrokontroler ditunjukkan pada Gambar 3.5.

Sedangkan program utama *master* mikrokontroler dalam bahasa *assembler* adalah sebagai berikut :

```

START:
MOV IE, #0
MOV SP, #STACKHERE
MOV R0, #GD1
MOV A, #0
MOV B, #12
CALL STRFILL
MOV SLVNR, #0
MOV TMOD, #20H
MOV TH1, #0F4H
MOV SCON, #50H
CALL DELAY256
SETB TR1
SETB EA
SETB TI

REPEATPROCESS:
MOV B, #10H

REPRCVSLVDATA:
CALL RECEIVESLAVEDATA
JNC SENDCRTSLVDATA
DJNZ B, REPRCVSLVDATA

SENDCRTSLVDATA:
CALL SENDQUALITYDATA
MOV B, #8
CALL DELAYLONG

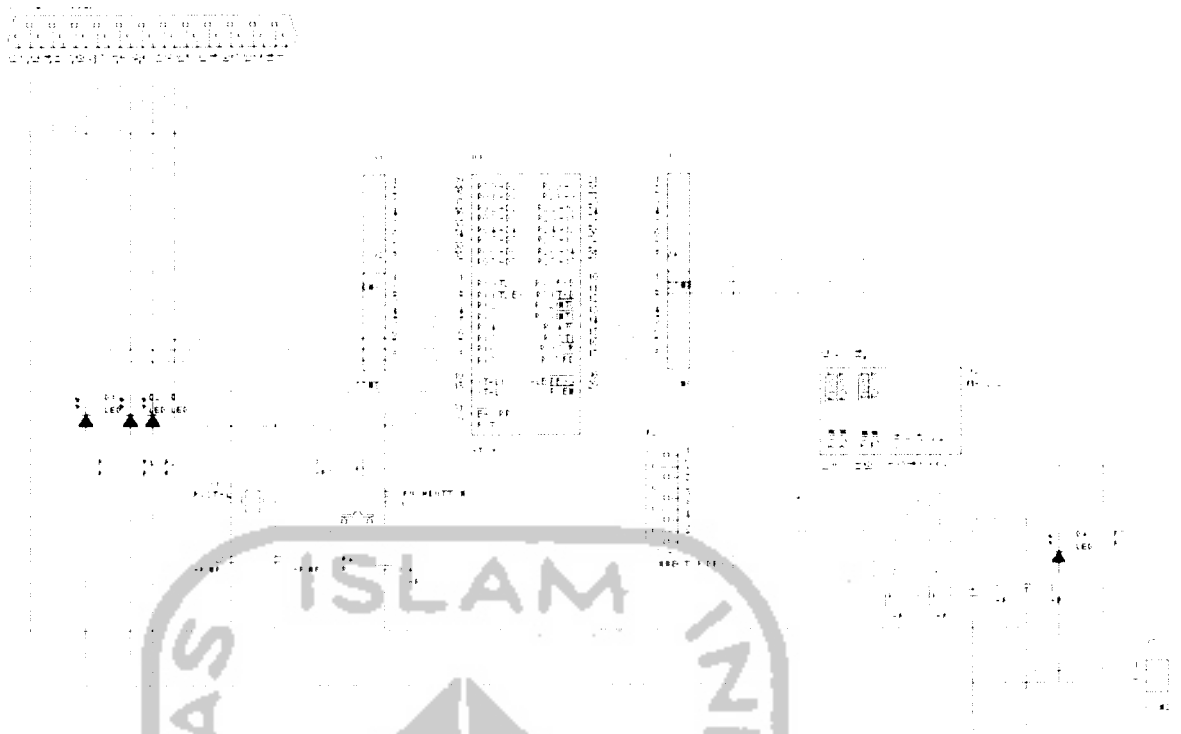
QUALITYDATASENT:
INC SLVNR
MOV A, SLVNR
CLR C
SUBB A, #3
JC SLVNRINCREMENTED
MOV SLVNR, #0

SLVNRINCREMENTED:
SJMP REPEATPROCESS
END

```

3.2.5. Rangkaian komunikasi serial MAX-232.

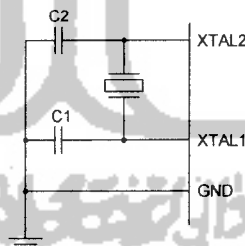
IC MAX-232 ini digunakan sebagai konektor komunikasi data serial dari komputer ke *Master* Mikrokontroler dan sebaliknya. Pada saat komputer mengeluarkan data/sinyal melalui fasilitas keluaran serialnya (port serial), maka RS-232 ini akan mengirimkan sinyal lewat pin TXD (pin kirim data) ke mikrokontroler. Sedangkan



Gambar 3.5. Rangkaian *master* mikrokontroler.

3.2.6. Rangkaian mikrokontroler AT89S52

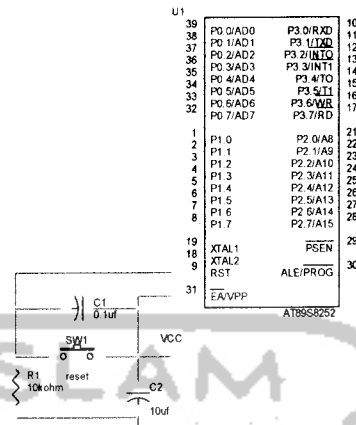
Pada rangkaian Mikrokontroler AT89S52 terdapat rangkaian osilator. Rangkaian osilator seperti yang diperlihatkan pada Gambar 3.6 menggunakan kristal frekuensi sebesar 11,059 MHz



Gambar 3.6. Rangkaian osilator

Selain itu juga terdapat rangkaian reset. Mikrokontroler direset pada transisi tegangan rendah ke tegangan tinggi dan mengeksekusi program pada saat reset (RST) dalam keadaan logika rendah. Oleh karena itu pada pin RST dipasang kapasitor yang terhubung ke VCC dan resistor ke ground yang akan menjaga RST bernilai 1 pada

saat pengisian kapasitor dan akan kembali sesaat kemudian, dengan demikian mikrokontroler akan direset setiap kali diberi catu daya. Adapun rangkaian reset ditunjukkan pada Gambar 3.7 :



Gambar 3.7. Rangkaian reset.

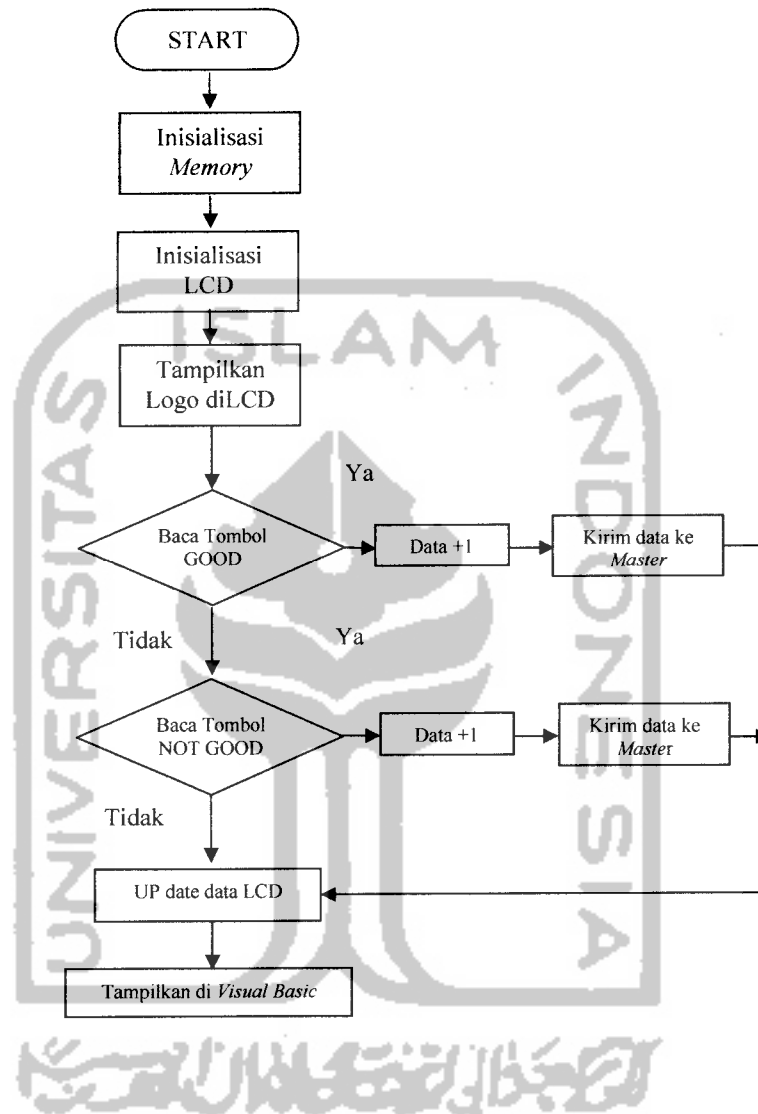
3.3. Alur Kerja Rangkaian.

Alur kerja rangkaian dapat dilihat pada Gambar 3.8.

Penjelasan *flow chart* proses kerja rangkaian adalah sebagai berikut :

1. Pertama kali alat dihidupkan.
2. Setelah itu dilanjutkan dengan proses inialisasi *memory*.
3. Setelah itu dilanjutkan dengan proses inialisasi LCD.
4. Proses selanjutnya adalah menampilkan logo pada LCD.
5. Proses dilanjutkan dengan pembacaan tombol *GOOD* dan *NOT GOOD*.
6. Apabila ada masukan maka dilakukan proses *counter GOOD +1* dan *counter NOT GOOD +1*.
7. Setelah ada masukan tiap-tiap tombol maka proses selanjutnya adalah pengiriman data.

8. Data yang dikirim melalui port serial menuju komputer. Data yang didapat oleh komputer lalu ditampilkan pada program *quality monitor*.



Gambar 3.8. Alur kerja rangkaian.