

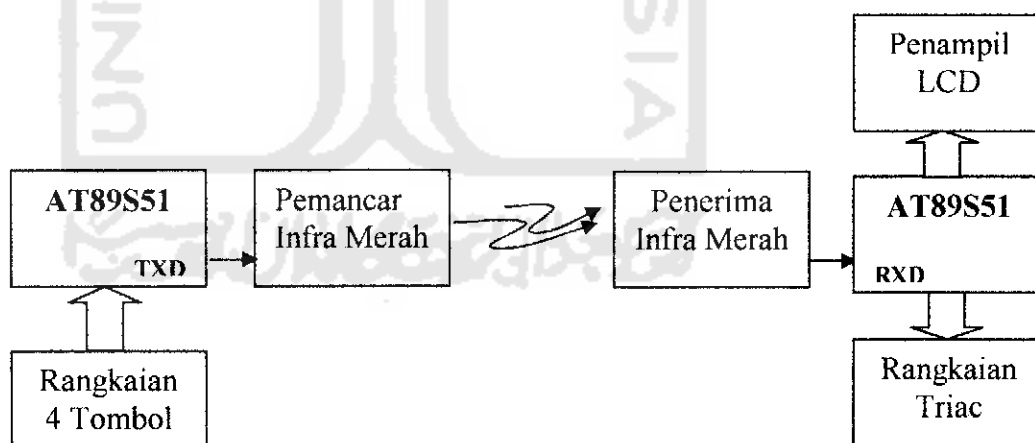
BAB III

PERANCANGAN RANGKAIAN

3.1. Perancangan Perangkat Keras

Perancangan perangkat keras untuk alat pengendali peralatan rumah tangga jarak jauh ini terdiri dari beberapa bagian yaitu :

1. Rangkaian pemancar yang terdiri dari rangkaian 4 buah tombol, mikrokontroler AT89S51, dan rangkaian pemancar infra merah
2. Rangkaian penerima yang terdiri dari rangkaian penerima infra merah, mikrokontroler AT89S51, rangkaian penampil LCD dan rangkaian Triac pengendali keluaran.



Gambar 3. 1. Diagram blok pemancar dan penerima pengendali jarak jauh

Dari gambar 3.1 diatas dapat dijelaskan bahwa mikro pertama berfungsi untuk mengirim data dari empat buah tombol, misalnya menekan tombol satu maka data dikirim 8 bit, setelah data didefenisikan oleh mikro lalu dikirim oleh pemancar infra merah dan kemudian diterima oleh rangkaian penerima, oleh mikro penerima data yang dikirim didefenisikan lagi setelah itu data di tampilkan di LCD, untuk mengaktifkan keluaran data yang disimpan tadi kemudian dicocokkan oleh data jam kalau datanya cocok maka rangkaian Triac diaktifkan.

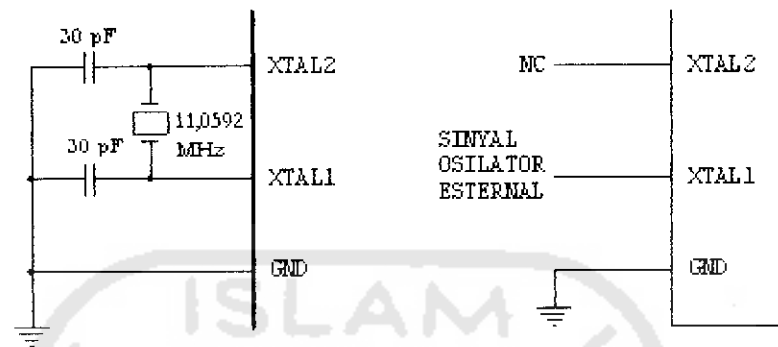
Dari gambar diagram blok diatas dapat dijelaskan bagian-bagiannya sebagai berikut:

3.1.1 Mikrokontroler AT89S51

Rangkaian dasar mikrokontroler terdiri dari rangkaian pewaktu CPU dan rangkaian *power on reset*

3.1.1.1. Pewaktuan CPU

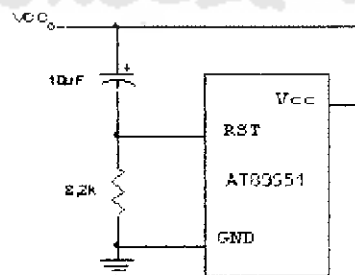
Semua mikrokontroler keluarga MCS-51 memiliki *oscillator on chip* yang dapat digunakan sebagai sumber detak (*clock*) ke CPU. Untuk menggunakan XTAL 1 dan XTAL 2 yaitu dengan menghubungkan sebuah resonator kristal atau keramik ke kaki-kaki XTAL 1 dan XTAL 2 pada mikrokontroler, selanjutnya hubungkan kapasitornya ke *ground* seperti gambar 3.2 berikut:



Gambar 3.2. Hubungan ke kristal dan konfigurasi pemberian *clock* eksternal

3.1.1.2. Power on Reset

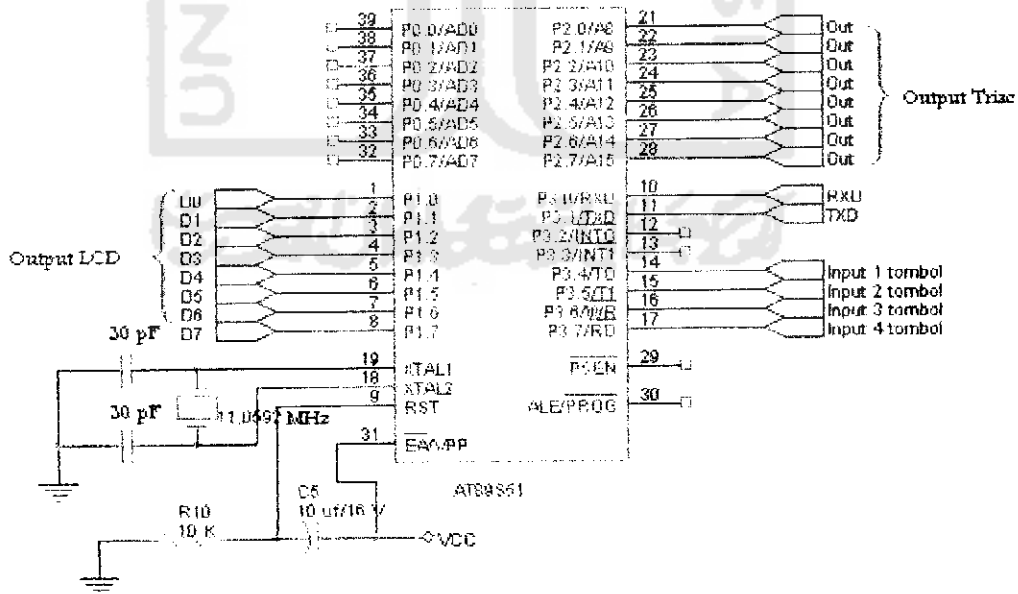
Mikrokontroler di *reset* pada transisi tegangan rendah ke tegangan tinggi dan mikrokontroler mengeksekusi program pada saat *reset* (RST) dalam logika rendah oleh karena itu pada pin RST di pasang kapasitor yang terhubung ke *Vcc* dan resistor ke *ground* yang akan menjaga RST bernilai 1 pada saat pengisian kapasitor dan bernilai 0 saat kemudian, dengan demikian mikrokontroler akan di *reset* setiap kali diberi catu daya seperti ditunjukkan pada gambar 3.3. berikut



Gambar 3.3. Rangkaian *Power on Reset*

3.1.1.3. Penggunaan pin Atmel 89S51

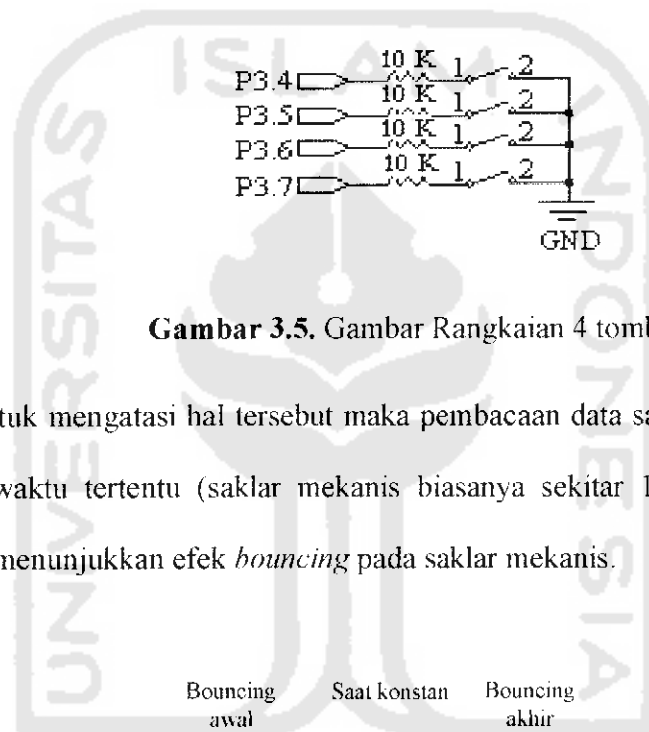
Pada gambar 3.4 dijelaskan tentang rangkaian *minimum* mikrokontroler Atmel 89S51 dan penggunaan pin-pin yang dipakai pada sistem pengendalian. Untuk rangkaian 4 tombol menggunakan *port* sebanyak empat buah, yaitu *port* 3.4, *port* 3.5, *port* 3.6 dan *port* 3.7. Untuk keluaran LCD menggunakan delapan buah *port*, yaitu *port* 1.0, *port* 1.1, *port* 1.2, *port* 1.3., *port* 1.4, *port* 1.5, *port* 1.6, dan *port* 1.7, untuk keluaran triac juga menggunakan delapan buah *port* yaitu *port* 2.0, *port* 2.1, *port* 2.2, *port* 2.3., *port* 2.4, *port* 2.5, *port* 2.6, dan *port* 2.7. yang dihubungkan ke 8 buah alat seperti setrika, hidrayer, kulkas. tape, lampu 5 watt, lampu 10 watt, lampu 40 watt dan lampu 100 watt dan untuk *port* 3.0 keluaran RXD serta *port* 3.1 untuk keluaran TXD.



Gambar 3.4. Rangkaian Mikrokontroler AT89S51

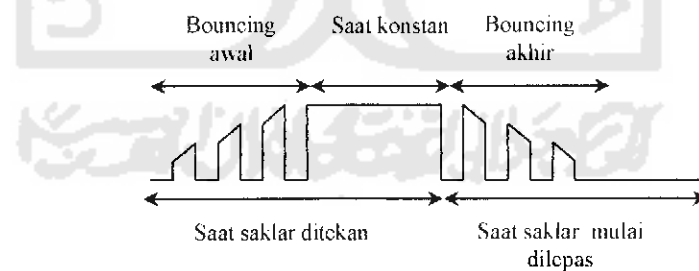
3.1.2. Rangkaian 4 Tombol

Pada saklar mekanis hal yang perlu diperhatikan adalah lentingan (*bouncing*). Saat saklar ditekan dan dilepas terjadi lentingan menyebabkan logika saklar tidak dapat diperkirakan.



Gambar 3.5. Gambar Rangkaian 4 tombol

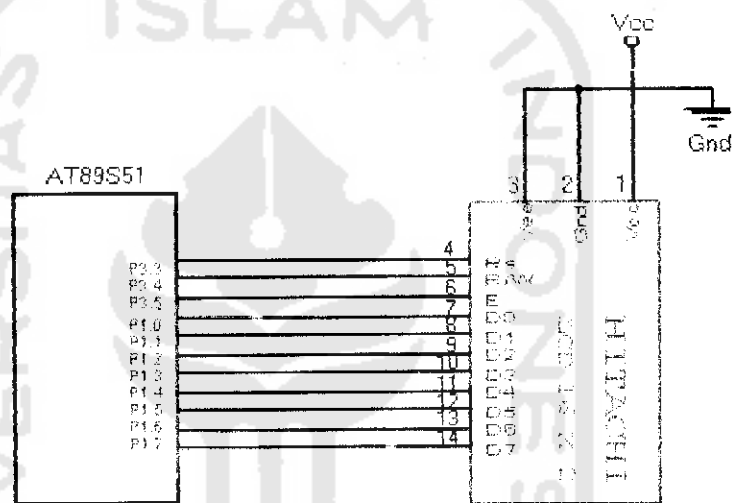
Untuk mengatasi hal tersebut maka pembacaan data saklar dilakukan setelah selang waktu tertentu (saklar mekanis biasanya sekitar 10 – 20 ms). Gambar berikut menunjukkan efek *bouncing* pada saklar mekanis.



Gambar 3. 6. Efek *bouncing* pada saklar mekanis.

3.1.3. Rangkaian penampil LCD

Rangkaian penampil LCD ditunjukkan pada gambar 3.7 dibawah, tiga bit control RS, R/W dan E dihubungkan ke *port* P_{3.3}, P_{3.4} dan P_{3.5} sedangkan 8 bit datanya (D₀ - D₇) dihubungkan ke *port* 2 (P_{1.0} - P_{1.7})



Gambar 3. 7. Rangkaian Penampil LCD

3.1.4. Pemancar dan Penerima Infra Merah

Komunikasi Inframerah dilakukan dengan menggunakan dioda infra merah sebagai pemancar dan modul penerima infra merah sebagai penerimanya. Untuk jarak yang cukup jauh, kurang lebih tiga sampai lima meter, pancaran data infra merah harus dimodulasikan terlebih dahulu untuk menghindari kerusakan data akibat *noise*. Proses modulasi dilakukan dengan mengubah kondisi logika 0 dan 1 menjadi kondisi ada dan tidak ada sinyal *carrier* infra merah yang berkisar antara 30KHz sampai 40

KHz. Logika 0 diwakili dengan adanya frekuensi 30-40 KHz, logika 1 diwakili dengan tidak adanya frekuensi 30-40 KHz.



Gambar 3.8. Modulasi Sinyal Infra Merah

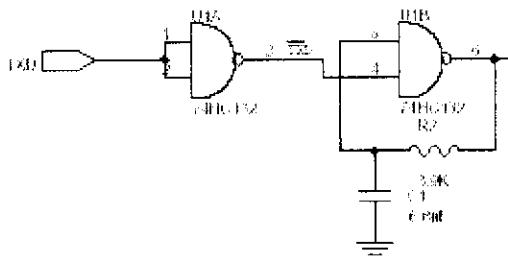
3.1.4.1. Bagian Pemancar

Pemancar terdiri dari Bagian Modulator, Bagian Penguat dan Diode Inframerah.

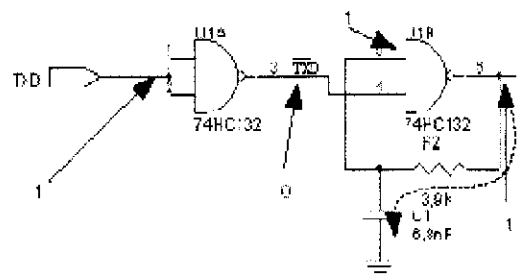


Gambar 3.9. Blok Diagram Pemancar

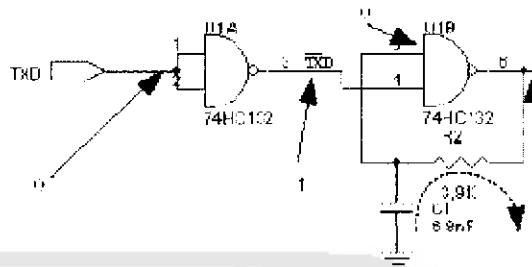
Proses modulasi dilakukan dengan mencampur sinyal *carrier* inframerah dengan sinyal data yang diperoleh dari port serial mikrokontroler. Rangkaian modulator yang terdiri dari sebuah gerbang dan rangkaian R-C sebagai *oscillator*.



Gambar 3.9a. Rangkaian Modulator



Gambar 3.9b. Saat Data Logika 1 atau Idle



Gambar 3.9c. Saat Data Logika 0

TXD Gerbang tersebut menggunakan IC 74HC132 di mana pada saat pin TXD berkondisi logika 1 dan berkondisi logika 0 maka output dari IC ini sesuai dengan tabel kebenaran yang ada pada data sheet adalah logika 1. Namun bila sebaliknya TXD berkondisi logika 0 maka sesaat output dari IC ini (pin 6) berubah ke logika 0. sehingga kapasitor C1 akan membuang muatannya melalui R2. Bila tegangan C1 terbuang hingga di bawah tegangan ambang 74HC132 maka input pin nomor 5 dari IC ini akan dianggap berkondisi logika 0 sehingga outputnya (pin 6) berubah menjadi logika 1. Arus akan mengalir dari pin 6 menuju ke C1 melalui R1 hingga tegangan pada kapasitor ini melebihi tegangan ambang dan input pin nomor 5 dianggap berkondisi logika 1 (gambar 3.9b). Bila pada saat itu TXD masih berkondisi 1 maka output dari gerbang ini yaitu pin nomor 6 akan berkondisi 0 dan C1 akan membuang muatannya melalui R1 sehingga tegangan pada pin 5 berada di bawah tegangan ambang atau berlogika 0 (gambar 3.9c). Output pada pin 6 kembali berlogika 1 dan arus mengalir lagi mengisi C1 melalui R1.

Proses pembuangan dan pengisian capacitor C1 ini menyebabkan adanya osilasi pada pin 6 IC 74HC132 sehingga terbentuk sinyal *carrier* dengan frekuensi yang dapat dihitung dengan menggunakan rumus berikut:

$T = \text{Waktu Pengisian } C1 + \text{Waktu Pembuangan } C1$

Waktu Pengisian C1 = Waktu Pembuangan C1 maka

$T = 2 * \text{Waktu Pengisian } C1$

$$T = 2RC \left[\ln \left[\frac{V_s - (VT^-)}{V_s} \right] - \ln \left[\frac{V_s - (VT^+)}{V_s} \right] \right] \quad (3.1)$$

Dimana :

VT^- = batas bawah tegangan ambang (Volt)

VT^+ = batas atas dari tegangan ambang (Volt)

V_s = tegangan power supply (Volt)

R = resistansi (Ohm)

C = kapasitor (farat)

di mana VT^- adalah batas bawah tegangan ambang 74HC132 yaitu sekitar 2 Volt dan VT^+ adalah batas atas dari tegangan ambang 74HC132 yaitu sekitar 3 Volt. Dengan R sebesar 3,9K, $C10nF$ dan $V_{supply} = 5$ Volt

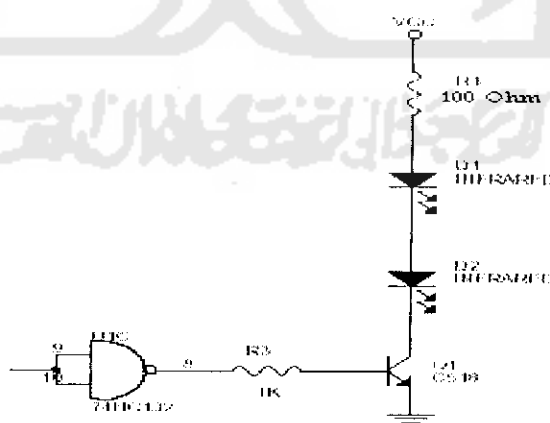
$$\begin{aligned} T &= 2.3.9.10 \left[\ln \left[\frac{5-2}{5} \right] - \ln \left[\frac{5-3}{5} \right] \right] \\ &= 78[-0,5108] - [-0,9162] \\ &= 31,63\mu S \end{aligned}$$

$$f = \frac{1}{T} \quad (3.2)$$

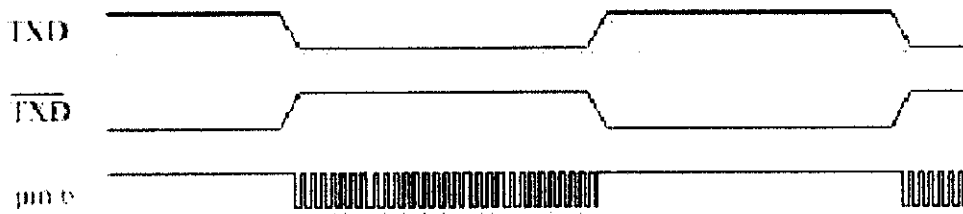
$$= \frac{1}{31,63 \cdot 10^{-3}}$$

$$f = 31,616 \text{ KHz}$$

Jadi pada intinya apabila input TXD berkondisi 1 maka frekuensi *oscillator* sebesar 31,616 KHz yang terjadi pada pin nomor 4 akan dilewatkan ke outputnya dengan frekuensi yang sama persis, namun bila TXD berkondisi 0 maka osilasi pada pin nomor 4 akan berhenti dan output dari gerbang adalah 1. Ayunan sinyal berfrekuensi 31,6 KHz ini diperkuat lagi oleh gerbang lain dari 74HC132 yang dibentuk menjadi inverter dan diteruskan ke transistor BC548 yang mengalirkan sinyal-sinyal frekuensi hasil dari modulator tersebut ke Diode *Infrared*.



Gambar 3.10. Bagian Penguat Inframerah

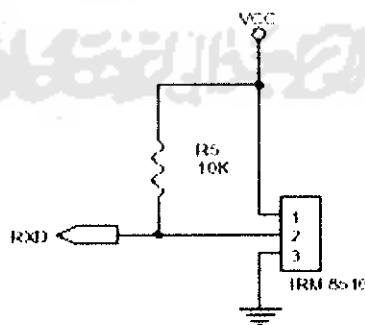


Gambar 3.11. Timing Diagram Pemancar Inframerah

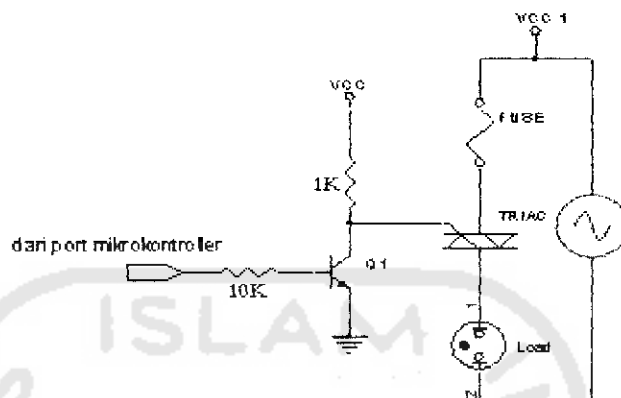
Osilasi dengan frekuensi 31,6 KHz yang telah dikuatkan oleh bagian penguat ini akan mengakibatkan adanya pancaran sinyal inframerah dengan frekuensi seperti pada gambar 3.11.

3.1.4.2. Bagian Penerima

Bagian Penerima seperti telah disebutkan sebelumnya, adalah merupakan modul penerima inframerah yang terdiri dari photodiode yang sudah dilengkapi dengan rangkaian *filter* yang hanya melewatkan frekuensi antara 30KHz sampai 40 KHz saja. Modul (IRM-8510) ini terbungkus dengan plat yang terhubung dengan *ground* rangkaian untuk melindungi rangkaian dari interferensi *noise*.



Gambar 3.12. Modul Penerima Inframerah



Gambar 3.14. Rangkaian Triac.

Rangkaian transistor NPN yang dipasang pada *gate* triac berfungsi sebagai saklar dari mikrokontroler sehingga apabila keluaran mikrokontroler berlogika '1' berarti basis transistor NPN mendapatkan tegangan V_{bb} sebesar 5 volt yang membuat transistor tersebut aktif (ON) saat transistor aktif arus kolektor I_c maksimum, dengan demikian triac mendapatkan arus *gatenya* yang menyebabkan triac aktif.

Sehingga didapat arus basis (I_B) dengan persamaan 2.2 seperti berikut :

$$I_B = \frac{V_{BB} - V_{BE}}{R_B}$$

Besarnya V_{BE} pada transistor silicon 0,6 – 0,7 Volt

$$I_B = \frac{5 - 0,7}{10 \cdot 10^3} = 0,43 \text{ mA}$$

Untuk membuat arus I_c maksimum maka transistor dibuat jenuh, transistor akan jenuh apabila tegangan $V_{CE} \approx 0$ pada kondisi ini I_c sebesar

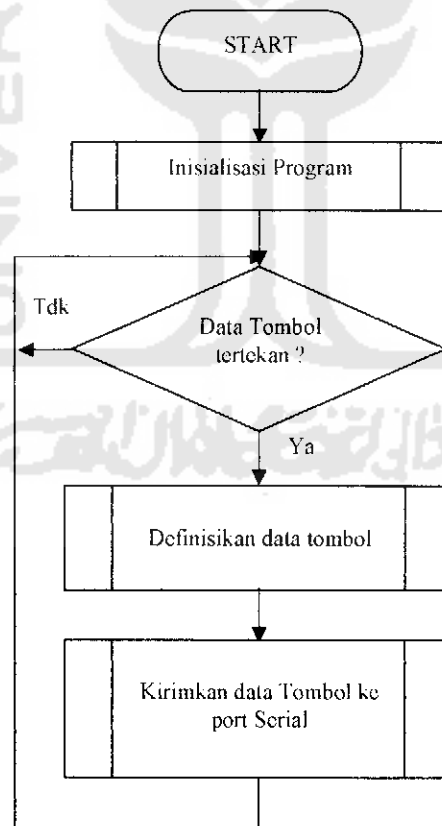
$$I_c = \frac{V_{cc}}{R_c} = \frac{5}{1 \cdot 10^3} = 5mA$$

3.2. Perancangan Perangkat Lunak

Perancangan perangkat lunak merupakan perancangan pemrograman mikrokontroler AT89S51 yang menggunakan bahasa *assembler* yang meliputi :

3.2.1. Perancangan Program

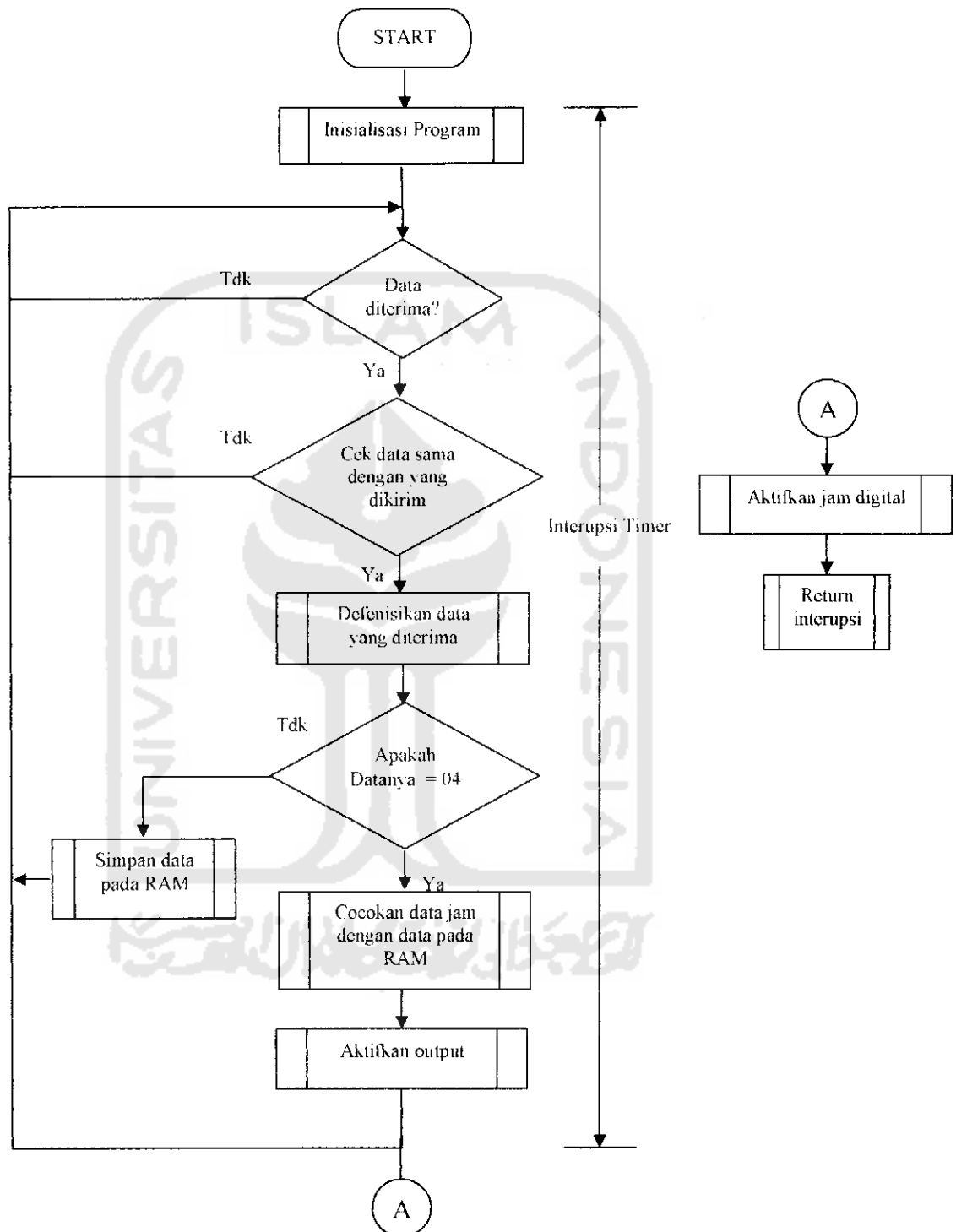
Perancangan program dapat dilihat dalam diagram alir berikut :



Gambar 3.15. Diagram Alir Pengirim

Dari diagram alir diatas dapat dijelaskan bahwa program pertama kali diawali dengan inisialisasi yang berfungsi untuk memberikan keterangan tentang pemetaan alamat memori sehingga memudahkan dalam pembuatan program. Setelah inisialisasi program kemudian program melakukan pengecekan tombol, apakah ada yang ditekan atau tidak. Jika ada, maka data tombol didefinisikan kemudian data dikirimkan ke *port serial*. Jika tidak kembali menginisialisasi program dari awal.





Gambar 3.16. Diagram Alir Penerima

Pada gambar 3.6 merupakan diagram alir untuk program penerima dimulai dengan start kemudian inisialisasi program. Jika data diterima, cek data apakah sama dengan yang dikirim. Jika tidak, kembali menginisialisasi program. Setelah data sama dengan yang dikirim, definisikan data yang diterima. Apakah data yang diterima = 04. Jika ya, cocokkan data jam dengan data pada RAM kemudian aktifkan output. Setelah pengaktifan output selesai, aktifkan jam digital dan kembali ke interupsi. Jika tidak, simpan data pada RAM. Proses ini akan berulang terus menerus.

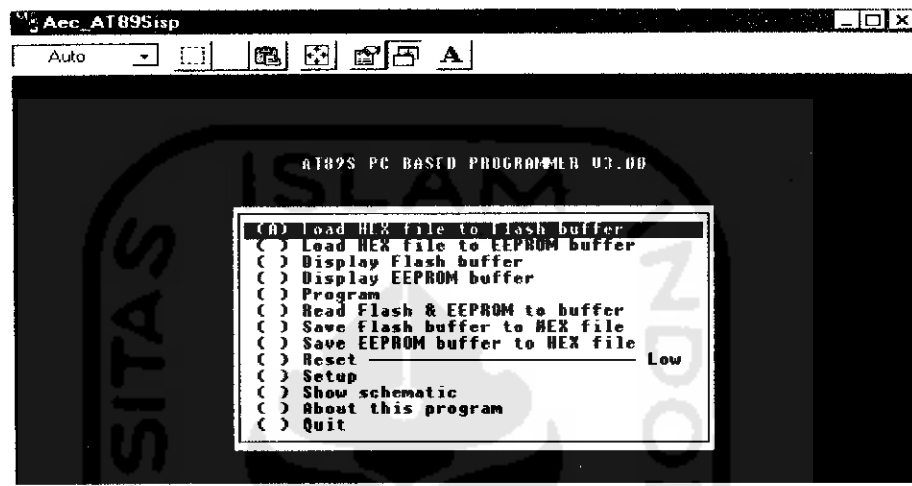
3.2.2. Downloader ke mikrokontroler AT89S51

Mikrokontroler AT89S51 merupakan mikrokontroler teknologi baru karena dilengkapi dengan fasilitas SPI (*Serial Peripheral Interface*) yang terdiri dari SCK/P1.7 (*serial clock*), MOSI/P1.5 (*input*) dan MISO/P1.6 (*output*) sehingga memungkinkan untuk mengisikan/*mendownload* program secara langsung. Hubungan pin SPI mikrokontroler dengan *port printer* ditunjukkan pada tabel 3.1 berikut :

Tabel 3. 1. Hubungan Pin mikrokontroler dengan port printer

Port Printer	Pin mikrokontroler
6 (D4)	9 Reset
7 (D5)	6 (P1.5 -MOSI)
8 (D6)	8 (P1.7 - SCK)
10 (ACK)	7 (P1.6 - MISO)
18-25 (GND)	20 (GND)

Software untuk serial *programming* menggunakan *software* *acc_isp* yang memiliki tampilan seperti gambar 3.17 berikut:



Gambar 3.17. Tampilan *software* *Acc_isp*

Software *Acc_isp* versi V3.00 merupakan dapat digunakan untuk memprogram beberapa mikrokontroler keluarga AT89Sxx dengan memilih setup seperti gambar berikut :



Gambar 3. 18. Tampilan setup

3.2.3. Inisialisasi

Inisialisasi Program rancang bangun peralatan rumah tangga jarak jauh berbasis timer mikrokontroler sebagai berikut :

Program dimulai pada alamat flash 0000h

```
org 0h
```

pemetaan alamat RAM untuk menyimpan data yang dapat dimodifikasi pada alamat

30h - 52h

```

aduh      Equ    -50000
data_select Equ    30h
Detik     Equ    31h
Jam       Equ    32h
Menit     Equ    33h
jam1      Equ    34h
menit1    Equ    35h
jam2      Equ    36h
menit2    Equ    37h
out       Equ    38h
titip     Equ    39h
datako    Equ    3Ah
RuasJam10 Equ    3Bh
RuasJam1  Equ    3Ch
RuasMenit10 Equ    3Dh
RuasMenit1 Equ    3Eh
RuasDetik10 Equ    3Fh
RuasDetik1 Equ    40h
digitjam10 Equ    41h
digitjam1 Equ    42h

```

```

digitmenit10Equ 43h
digitmenit1 Equ 44h
baitjam10 Equ 45h
baitjam1 Equ 46h
baitmenit10 Equ 47h
baitmenit1 Equ 48h
Ruasout10 Equ 49h
Ruasout1 Equ 4Ah
Pencacah2 EQU 4Ch
Pencacah10 EQU 4Dh
Pencacah60 EQU 4Eh
RuasJam101 Equ 4Fh
RuasJam11 Equ 50h
RuasMenit101Equ 51h
RuasMenit11 Equ 52h
ROM EQU 0h

```

Konfigurasi sistem menentukan hal-hal yang harus dikerjakan AT89S51 saat terjadi *RESET* atau *INTERUPSI*. Saat terjadi *RESET* program menjalankan rutin *Start*, saat interupsi *Timer 0* jalankan rutin *TimerInterrupt* dan interupsi lain diabaikan dengan instruksi *RETI*

```

ORG ROM ;Reset Vector
AJMP Start
;
org 3h ;External Interrupt 0 Vector
RETI
;
org 0Bh ;Timer 0 interrupt Vector
Ajmp Timerlinter
;
org 13h ;External Interrupt 1 Vector
RETI
;

```

```

    org    1Bh                ;Timer 1 Interrupt Vector
    reti

;

    org    23h                ;Serial Interrupt Vector
    reti

;

    CSEG

Init_LCD sebagai penampil 16 x 2 karakter
    Mov    R1,#01h
    Lcall  write_inst
    mov    R1,#00111000b      ;inisialisasi LCD untuk 8 bit
    acall  write_inst         ;operasi dan dua line
    mov    R1,#00001110b      ;mengaktifkan kursor
    acall  write_inst
    Mov    R1,#00000010b
    Lcall  write_inst
    mov    R1,00000110b       ;set mode untuk menaikan alamat
    acall  write_inst
    ret

Init_serial
    MOV    SCON,#52H          ; Mode 1, Receive Enable aktif
    MOV    TMOD,#20H          ; Timer 1 Mode 2
    MOV    TH1,#0D0H          ; Baudrate 600 bps
    MOV    TCON,#040H         ; Timer 1 On
    Clr    RI                  ; Hapus Flag Receive
    Clr    TI                  ; Hapus Flag Transmit
    RET

```

Komunikasi pengiriman dan penerimaan data serial dilakukan dengan mengirimkan data ke register SBUF atau menerima data dari register SBUF saja seperti pada listing berikut ini.

```

Serial Out:
    CLR T1
    MOV SBUF,A
    JNB T1,$
    RET

```

```

serial_in:
    clr    RI
    JNB   RI,test1
    mov   A,SBUF
    ret

```

3.2.4. Menampilkan karakter pada LCD

Untuk menampilkan karakter pada LCD terlebih dulu dilakukan inisialisasi LCD seperti yang dijelaskan diatas. Untuk menuliskan instruksi pada LCD pin RS dan R/W diset pada kondisi '0' sedangkan untuk menuliskan data pin RS diset '1' dan pin R/W diset '0'. Setiap selesai menuliskan instruksi atau data ke LCD, pin E harus diberi masukan pulsa (*clock*). LCD akan mulai membaca data setelah 15 ms setelah pertama kali LCD diberi tegangan catu. Contoh program berikut adalah program untuk menampilkan karakter '*' pada LCD :

```

write_inst:
    clr    P3.3        ;untuk menulis instruksi
    clr    P3.4        ;pada LCD
    Mov    P1,R1
    setb   P3.5        ;memberi clock pada pin E LCD
    acall delay
    clr    P3.5
    acall delay
    ret

```

```

write_data:
    setb P3.3      ;untuk menulis data
    clr  P3.4      ;pada LCD
    mov  P1,R1
    setb P3.5
    acall delay
    clr  P3.5
    acall delay
    ret

```

3.2.5. Pembacaan rangkaian 4 tombol

Pembacaan *tombol* dilakukan dengan sistem *scanning* yang mana P_{3.4}– P_{3.7} sebagai *input*. Contoh programnya sebagai berikut :

```

bounc      Equ 4Ah

ambil_keypad:

    JNB P3.4,satu
    JNB P3.5,dua
    JNB P3.6,tiga
    JNB P3.7,empat

    ajmp ambil keypad

```