

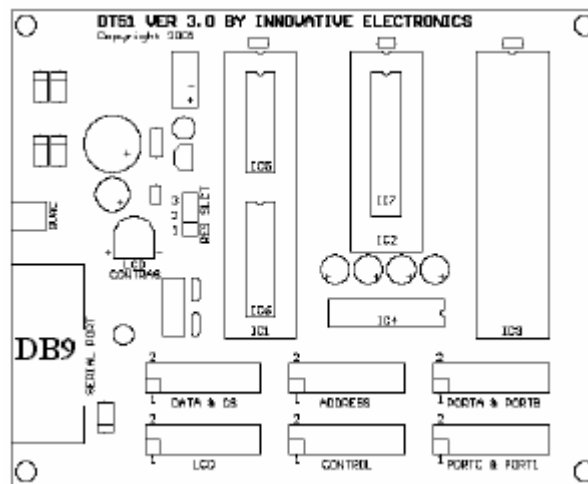
# DT-51

## DT-51 *Application Note* AN37 - How 2 Use DT-51 Minimum System ver 3.0

oleh: Tim IE

**A**pplication note yang satu ini mungkin sudah ditunggu-tunggu oleh banyak pihak. *Application Note* (AN) ini disusun untuk memberikan penjelasan tentang cara penggunaan DT-51 MinSys ver 3.0 beserta *software*-nya. Jika ada pertanyaan yang bersifat lebih spesifik, bacalah FAQ (*Frequently Asked Question*) pada bagian terakhir. Di dalamnya terdapat beberapa pertanyaan yang sering ditanyakan oleh pengguna DT-51 MinSys. Semoga AN ini dapat membantu.

### **M**EMPELAJARI KONEKTOR PADA DT-51 MINSYS VER 3.0



**Gambar 1**  
**Tata Letak Dt-51 MinSys ver 3.0**

1. Konektor 2 pin yang bertandakan “9VAC” dihubungkan dengan catu daya 9 VAC atau 12 VDC. Jika input tegangan yang digunakan adalah DC, posisi kaki positif dan negatif boleh berkebalikan karena pada DT-51 MinSys telah terdapat rangkaian dioda penyearah (*bridge*).
2. Konektor DB9 Female yang bertandakan “SERIAL PORT” dihubungkan dengan COM1 atau COM2 komputer melalui kabel serial.

**Penting:** DT-51 MinSys ver 3.0 Copyright 1999 (*obsolete*) memiliki kabel serial dengan hubungan *straight* (*pin* 2 dihubungkan dengan *pin* 2 ujung yang lain, begitu juga dengan *pin* 3). Sedangkan DT-51 MinSys ver 3.0 Copyright 2001 (hingga saat ini masih diproduksi) memiliki kabel serial dengan hubungan *cross* (*pin* 2 dihubungkan dengan *pin* 3 ujung yang lain dan sebaliknya).

3. *Port* yang bertandakan “DATA & CS” digunakan untuk ekspansi *peripheral* atau *memory*. AD0 hingga AD7 dihubungkan ke bagian “Data” pada *peripheral* yang berfungsi sebagai jalur *input* atau *output* antara 89C51 dengan *peripheral* tersebut. CS dihubungkan dengan *pin* yang berfungsi sebagai “*Chip Enable*” pada *peripheral*. Masing-masing CS dapat dihubungkan dengan *peripheral* dengan kapasitas hingga 8 KB dengan *memory mapping* seperti yang terdapat pada halaman 7 Manual DT-51 MinSys ver 3.0. CS1 telah dihubungkan ke PPI 8255 dan CS2 telah dihubungkan ke EEPROM 28C64. CS0 memang tidak dihubungkan dengan *peripheral* apapun juga namun alamat 0000h – 0FFFh telah digunakan untuk kernel pada 89C51.

4. *Port* yang bertandakan “ADDRESS” juga digunakan untuk ekspansi *peripheral*. A0 hingga A15 dihubungkan ke bagian “Address” pada *peripheral*. Beberapa *peripheral* mungkin tidak membutuhkan semua pin *address*, misalnya PPI 8255 yang hanya membutuhkan A0 dan A1 atau EEPROM 28C64 yang hanya membutuhkan A0 hingga A12 (terdapat pada skema DT-51 MinSys ver 3.0).
  5. *Port* yang bertandakan “CONTROL” berfungsi sebagai *input* untuk *external interrupt* dan *counter*. Pada *port* tersebut juga terdapat jalur kontrol WR, RD, MRD, dan RST yang digunakan untuk ekspansi *peripheral*.  
WR dihubungkan dengan *pin* yang berfungsi sebagai “Write” atau “Write Enable” pada *peripheral*.  
RD dihubungkan dengan *pin* yang berfungsi sebagai “Read” atau “Output Enable” pada *peripheral*.  
MRD dihubungkan dengan *pin* yang berfungsi sebagai “Read” atau “Output Enable” pada *peripheral* yang berfungsi sebagai *program memory* sekaligus sebagai *data memory*.  
RST dihubungkan dengan *pin* yang berfungsi sebagai “Reset” pada *peripheral*.
- Penting:** Pada *port* ini juga terdapat VCC (5V) dan GND (*ground*) yang dapat digunakan sebagai sumber tegangan untuk rangkaian tambahan yang akan dihubungkan ke DT-51 MinSys. Namun perhatikan konsumsi arus rangkaian tersebut. Pastikan kebutuhan DT-51 MinSys dan rangkaian tambahan tersebut tidak melebihi kapasitas *regulator* 7805 yang tersedia pada DT-51 MinSys.
6. *Port* yang bertandakan “PORTA & PORTB” berasal dari PPI 8255 yang berfungsi sebagai jalur *input* atau *output* data. Sebelum menggunakan *port* tersebut, harus ada proses inialisasi PPI untuk menentukan apakah *port* tersebut berfungsi sebagai *input* atau *output*. Sebagai contoh, *port* ini dapat dihubungkan dengan rangkaian saklar (sebagai *input*) atau LED (sebagai *output*).
  7. *Port* yang bertandakan “PORTC & PORT1” sebagian berasal dari PPI 8255 (*Port C*) dan sebagian berasal dari 89C51 (*Port 1*). Kedua *port* ini juga berfungsi sebagai jalur *input* atau *output* data. Sebelum menggunakan *port* C, harus ada proses inialisasi PPI untuk menentukan apakah *port* tersebut berfungsi sebagai *input* atau *output*. Penggunaan *port 1* tidak perlu inialisasi. Namun jika *port 1* ingin digunakan sebagai *input*, umumnya semua *pin* diberi nilai awal “1” (P1 bernilai 0FFh). Sebagai contoh, *port* ini dapat dihubungkan dengan rangkaian saklar (sebagai *input*) atau LED (sebagai *output*).
  8. *Port* yang bertandakan “LCD” berfungsi sebagai jalur *input output* data modul LCD. LCD yang didukung adalah LCD yang kompatibel dengan LCD Controller HD44780 dengan konfigurasi urutan kaki terdapat pada Tabel 1.

Nomor Kaki	Simbol	Fungsi
1	Vss	Ground
2	Vcc	5 V
3	Vee	Pengatur kontras
4	RS	Bernilai “1” untuk <i>input</i> berupa Data dan bernilai “0” untuk <i>input</i> berupa Instruksi
5	R/W	Bernilai “1” untuk proses Baca dan bernilai “0” untuk proses Tulis.
6	E	Sinyal <i>enable</i> bernilai “1”
7	DB0	Jalur data
8	DB1	
9	DB2	
10	DB3	
11	DB4	
12	DB5	
13	DB6	
14	DB7	
15	V+ BL	4,2 V
16	V- BL	Ground

**Tabel 1**  
**Alokasi Pin LCD**

*Back light* bersifat opsional. LCD tanpa *back light* pun dapat digunakan asalkan kompatibel. Penggunaan LCD berhubungan erat dengan VR Pengatur Kecerahan LCD yang berada di dekat *jumper* “RES SLCT”. Jika VR diputar searah jarum jam, LCD akan makin cerah (warna hitam semakin pudar). Sebaliknya,

jika VR diputar berlawanan arah jarum jam, LCD akan makin gelap (warna hitam semakin pekat). Aturlah VR untuk mendapatkan tampilan terbaik.

Perlu diperhatikan bahwa *port* LCD menggunakan hampir semua *port* 1 sehingga jika *port* LCD digunakan untuk tampilan pada modul LCD, disarankan untuk tidak menggunakan *port* 1.

9. *Jumper* yang bertanda "RES SLCT" berfungsi untuk menentukan mode operasi DT-51 MinSys. Posisi *pin* 1 dan 2 terhubung adalah untuk mode *download* sedangkan posisi *pin* 2 dan 3 terhubung adalah untuk mode *stand alone*. Pada mode *stand alone*, program tidak akan dapat di-*download* ke DT-51 MinSys.

Detil masing-masing konektor terdapat pada halaman 3 Manual DT-51 MinSys ver 3.0.

## MENG-INSTALL SOFTWARE DT-51 MINSYS

Ada 3 cara instalasi melalui MS-DOS *prompt*:

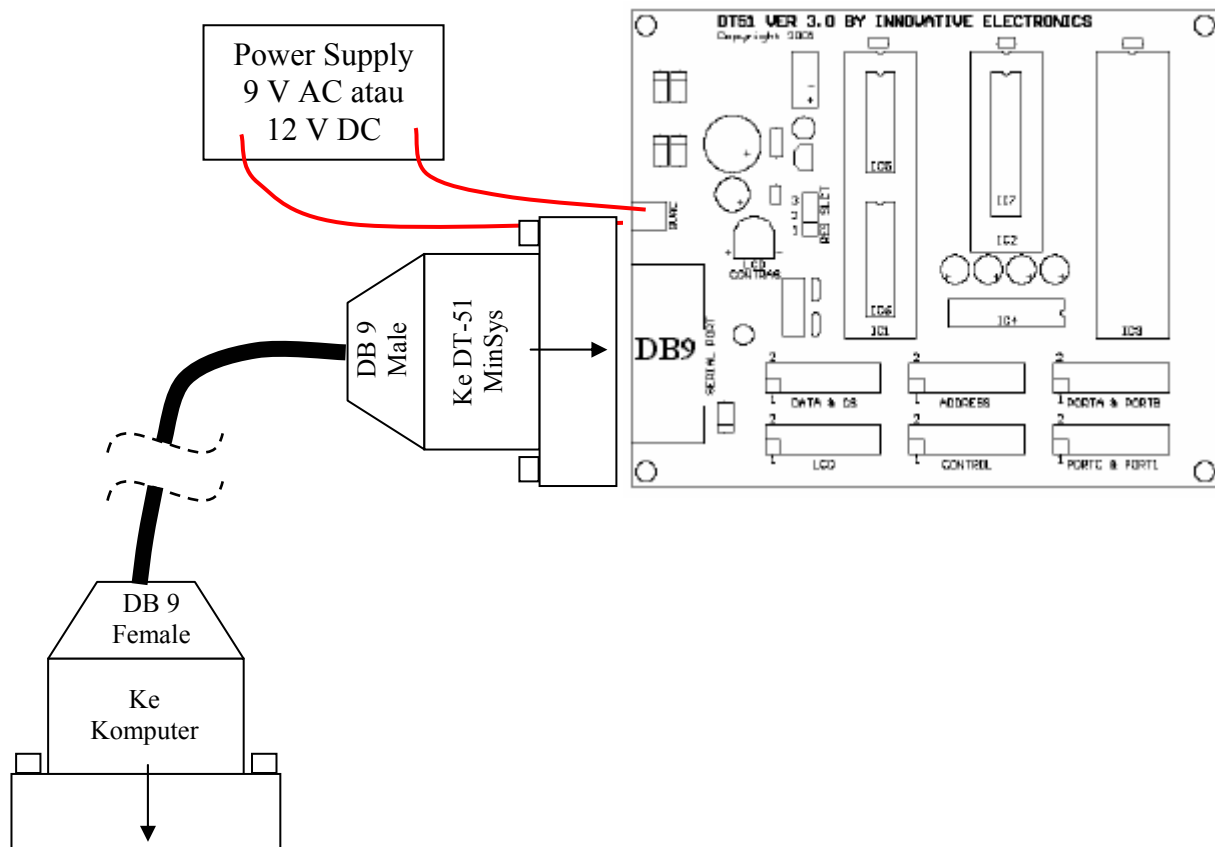
1. Untuk *software* DT-51 MinSys yang berasal dari disket (produksi sebelum Mei 2003), prosedur instalasi dilakukan dengan cara memasukkan disket tersebut ke *floppy disk drive* dan mengetikkan **install [drive disket] [drive tujuan]** pada *drive* a.  
Contoh tampilan pada monitor: `A:\>install a: c:`
2. Untuk *software* DT-51 MinSys yang berasal dari CD (produksi sekitar Mei - Juni), prosedur instalasi dilakukan dengan cara memasukkan disket tersebut ke CD-ROM dan mengetikkan **install [drive CD]\DT51MS [drive tujuan]** pada *drive* CD.  
Contoh tampilan pada monitor: `E:\>DT51MS\install e:\dt51ms c:`
3. Untuk *software* DT-51 MinSys yang berasal dari CD (hingga saat ini), prosedur instalasi dilakukan dengan cara memasukkan disket tersebut ke CD-ROM dan mengetikkan **install [drive CD] [drive tujuan]** pada *drive* CD.  
Contoh tampilan pada monitor: `E:\>DT51MS\install e: c:`

Proses instalasi akan membuat *folder* DT51R3 pada *drive* tujuan dan menambahkan beberapa baris perintah pada AUTOEXEC.BAT. Setelah komputer di-*reboot*, program DT51L dan DT51D dapat diakses dari lokasi mana saja di komputer tersebut.

**Penting:** Disarankan untuk menggunakan proses instalasi tersebut dan tidak melakukan proses *copy* secara manual. Jika Anda melakukan proses *copy*, periksa semua atribut *file* yang telah di-*copy* tersebut. Pastikan semua *file* tidak bersifat *read-only*. Program *testing* tidak akan dapat dijalankan jika *file* yang bersangkutan bersifat *read-only*.

## MEMPERSIAPKAN HARDWARE DT-51 MINSYS VER 3.0

1. Hubungkan kabel serial dengan "SERIAL PORT" DT-51 MinSys dan COM1 atau COM2 komputer.
2. Jika terdapat rangkaian atau modul lain yang akan dihubungkan ke DT-51 MinSys, disarankan untuk menghubungkan rangkaian tersebut dengan DT-51 MinSys terlebih dahulu. Perhatikan koneksi, terutama untuk jalur VCC dan GND jangan sampai terbalik.
3. Hubungkan sumber tegangan ke konektor "9VAC" (sesuai sub bab sebelumnya).



**Gambar 2**  
Hubungan DT-51 MinSys dengan Sumber Tegangan dan Komputer

## MELAKUKAN TESTING DT-51 MINSYS VER 3.0

Program *testing* yang disertakan berguna jika Anda ingin melakukan *testing* pada DT-51 MinSys. Selain versi MS-DOS, saat ini sudah tersedia program *testing* yang merangkap *downloader* untuk versi Windows. Sebelum melakukan *testing*, pastikan bahwa sumber tegangan DT-51 MinSys sudah terpasang dengan benar dan DT-51 MinSys sudah terhubung ke komputer. Selain itu, pastikan bahwa direktori yang aktif adalah direktori yang berisi semua file HEX (TESSER.HEX, TESRAM.HEX, TSPORT.HEX, TESPPI.HEX, dan TESLCD.HEX). Untuk mudahnya, operasikan program langsung dari direktori DT51R3.

### Versi MS-DOS

Program ini hanya dapat dijalankan dengan MS-DOS *prompt* atau eksekusi dari Windows (klik ke program yang bersangkutan).

1. Pengujian komunikasi serial antara komputer dengan DT-51 MinSys dapat dilakukan dengan program TSCOM1 (jika kabel serial terpasang pada COM1) atau TSCOM2 (jika kabel serial terpasang pada COM2). Komputer akan mengirimkan angka dari 0 hingga 255 ke DT-51 MinSys. DT-51 MinSys akan menerima angka tersebut dan mengirimkannya balik ke komputer.

Contoh tampilan pada monitor:

```

0=0
1=1
2=2
...
254=254
255=255
0=0
dst.
```

Tekanlah tombol "Enter" untuk mengakhiri program TSCOM.

- Pengujian EEPROM dapat dilakukan dengan program TESRAM1 (jika kabel serial terpasang pada COM1) atau TESRAM2 (jika kabel serial terpasang pada COM2).  
Program akan menuliskan angka 0 hingga 255 ke dalam EEPROM dan membacanya lagi.  
Contoh tampilan pada monitor: `EEPROM Tes Looping Succeeded`
- Pengujian *port* A, B, C dan *port* 1 dapat dilakukan dengan bantuan osiloskop atau LED. Program TESPPI akan mengeluarkan sinyal kotak (0V dan 5V) pada keempat *port* tersebut. Sedangkan program TSPORT hanya mengeluarkan sinyal kotak pada *port* 1. Bentuk sinyal dapat dilihat dengan bantuan osiloskop. Jika menggunakan LED, tampilan LED akan *blinking* (berkedip-kedip).
- Jika Anda memiliki LCD yang kompatibel, LCD tersebut dapat dihubungkan dengan *port* LCD dan jalankan TESLCD. Untuk tampilan terbaik, disarankan penggunaan LCD 16 x 2. Penggunaan LCD berukuran lain tidak menjadi masalah namun jika ukurannya lebih kecil, sebagian tulisan tidak dapat ditampilkan.  
Contoh tampilan pada LCD: `I INNOVATIVE  
E ELECTRONICS`
- Selain menggunakan program tersebut di atas, DT51L juga dapat digunakan dengan mengetikkan "DT51L -s". *Selftest* ini akan memeriksa komunikasi serial dan jenis memori.  
Contoh tampilan pada monitor: `Memory Type : AT28C64B  
Selftest Completed`

### Versi Windows (DT51LWin)

Program ini hanya dapat dijalankan dalam Windows (klik ke program yang bersangkutan). Setelah program berjalan, disarankan untuk memberi tanda centang pada bagian "Auto Detect Option" (klik pada kotak atau tulisan tersebut).

- Klik "Selftest" pada bagian "Testing Option". Lalu klik tombol "Testing".  
Akan muncul *window* dengan tampilan: `Memory : AT28C64B  
Selftest completed`
- Klik "Tes Port1" pada bagian "Testing Option". Lalu klik tombol "Testing".  
Akan muncul *window* dengan tampilan: `This test output squarewave  
signal at Port1(0..7)`
- Klik "Tes PPI" pada bagian "Testing Option". Lalu klik tombol "Testing".  
Akan muncul *window* dengan tampilan: `This test output squarewave  
signal at PortA..PortC`
- Klik "Tes Serial" pada bagian "Testing Option". Lalu klik tombol "Testing".  
Akan muncul *window* dengan tampilan: `This test sending data 0-255  
and receive echo via serial port  
  
Serial testing done`
- Klik "Tes Memory" pada bagian "Testing Option". Lalu klik tombol "Testing".  
Akan muncul *window* dengan tampilan: `This test write data 0-255 to memory  
then read and check  
  
Memory testing done`
- Klik "Tes LCD" pada bagian "Testing Option". Lalu klik tombol "Testing".  
Akan muncul *window* dengan tampilan: `This test display symbol & text  
to LCD if connected at LCD Port`

**Penting:** Jika program *testing* tidak berjalan sebagaimana mestinya, hubungi Innovative Electronics. Disarankan untuk tidak mencoba memperbaiki sendiri.

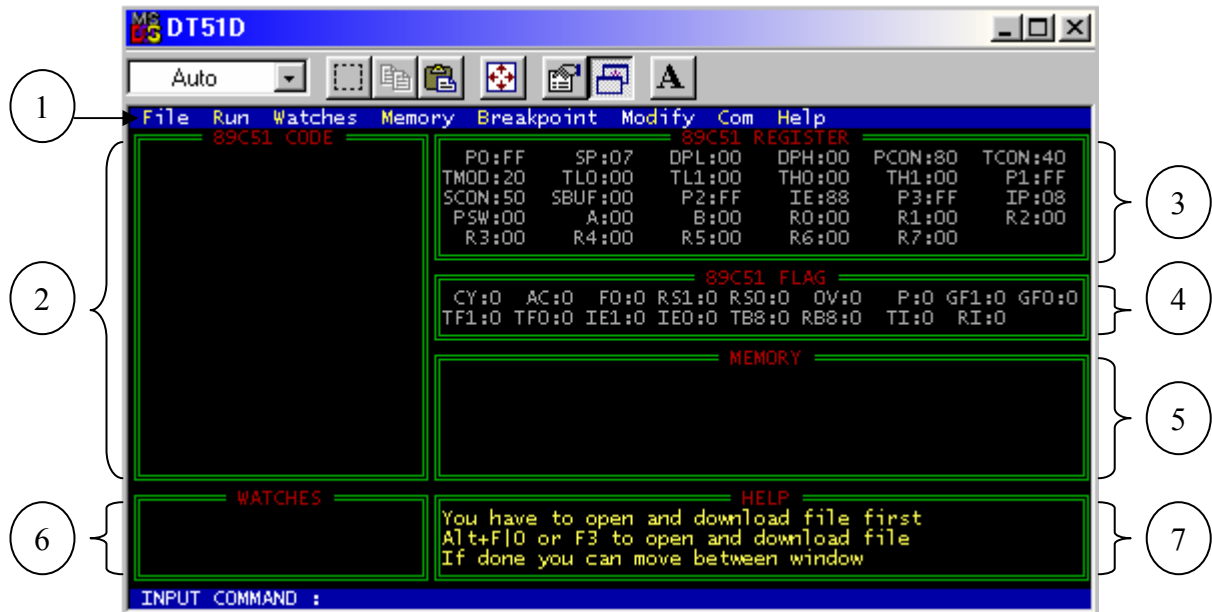
## PENGENALAN *DEBUGGER* DAN PENGGUNAAN *DOWNLOADER*

Program *downloader* untuk DT-51 MinSys ver 3.0 saat ini ada dua versi, MS-DOS (dengan nama DT51L) dan Windows (dengan nama DT51LWin). Contoh penggunaan program *debugger* DT51D terdapat di

www.innovativeelectronics.com pada bagian Tutorial dengan judul “Mengenal Instruksi MCS-51 dengan DT-51 Debugger”.

### Debugger Versi MS-DOS

Program ini dapat dijalankan dengan MS-DOS *prompt* dengan *syntax*: DT51D. Program ini berfungsi untuk meneliti jalannya program yang dibuat serta perubahan-perubahan yang terjadi pada memori dan *register*-nya.



**Gambar 3**  
**DT51D Versi MS-DOS**

1. Bagian “Menu” menunjukkan menu yang dapat diakses dengan menekan tombol “Alt” + Huruf yang berwarna kuning (*highlight*). Penekanan “Alt” hanya berlaku untuk menu utama (**F**ile, **R**un, dsb.). Setelah masuk ke menu berikutnya, tombol yang ditekan hanyalah huruf yang berwarna kuning (*highlight*).

  - File** : berisi menu untuk membuka dan men-*download* program, keluar sementara ke MS-DOS *prompt*, keluar dari DT51D, menyimpan *setting* dan menampilkan *setting*. *Setting* berisi nama file, COM *port* yang digunakan, *Baudrate* yang digunakan, kondisi *Fast Download*, dan kondisi *Verify*.
  - Run** : berisi menu untuk menjalankan program secara keseluruhan, menjalankan program secara per rutin, menjalankan program per baris, melompat ke alamat tertentu, dan reset dari awal.
  - Watches** : berisi menu untuk menambah atau mengubah alamat memori tertentu yang diawasi. Alamat ini mencakup alamat 0000h – 007Fh dan 4000h – 5FFFh.
  - Memory** : berisi menu untuk mengawasi alamat memori tertentu dan mengubah nilainya.
  - Breakpoint** : berisi menu untuk menentukan alamat *breakpoint*. Jika program dijalankan dengan perintah Run (secara keseluruhan), maka program akan berhenti pada alamat *breakpoint* yang ditentukan.
  - Modify** : berisi menu untuk mengubah nilai register atau flag 89C51.
  - Com** : berisi menu untuk memilih *baudrate*, COM *port*, serta aktivasi *Fast Download* dan *Verify*.
  - Help** : berisi penjelasan lengkap tentang menu dan *shortcut key*-nya.
2. Bagian “89C51 CODE” menunjukkan alamat dan instruksi yang terdapat dalam program yang di-*download*. Pada saat menjalankan program dengan StepOver atau TraceInto, baris alamat yang diproses akan diberi *highlight* putih.
3. Bagian “89C51 REGISTER” menunjukkan kondisi (nilai) *register* 89C51 (*byte*) meliputi *Special Function Register* dan *Register* 0 hingga 7.

Jika bagian ini diaktifkan (dengan menekan "Tab" hingga sekeliling bagian ini diberi warna hijau terang (*highlight*), kursor dapat digerakkan ke *register* yang ditampilkan. Dengan mengarahkan kursor ke *register*, tertentu dan menekan "Enter", bagian yang ditunjuk kursor tersebut dapat dimodifikasi.

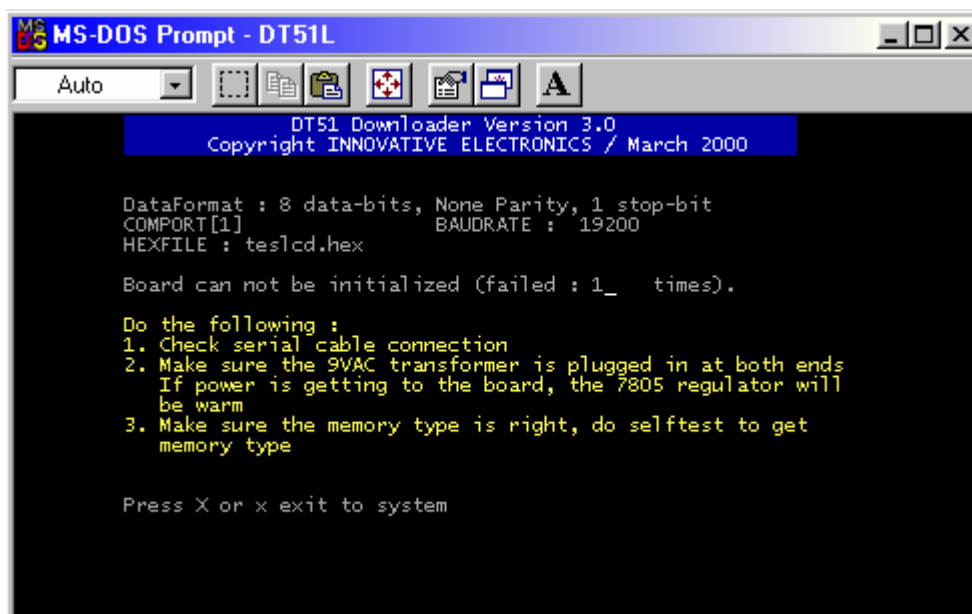
4. Bagian "89C51 FLAG" menunjukkan kondisi (nilai) *register* 89C51 (*bit*) meliputi *Carry Flag* dan *Interrupt Flag*. Jika bagian ini diaktifkan (dengan menekan "Tab" hingga sekeliling bagian ini diberi warna hijau terang (*highlight*), kursor dapat digerakkan ke *flag* yang ditampilkan. Dengan mengarahkan kursor ke *flag* tertentu dan menekan "Enter", bagian yang ditunjuk kursor tersebut dapat dimodifikasi.
5. Bagian "MEMORY" menunjukkan nilai alamat memori yang ditambahkan melalui menu **Memory**. Jika bagian ini diaktifkan (dengan menekan "Tab" hingga sekeliling bagian ini diberi warna hijau terang (*highlight*), kursor dapat digerakkan ke alamat memori yang ditampilkan. Dengan mengarahkan kursor ke alamat memori tertentu dan menekan "Enter", bagian yang ditunjuk kursor tersebut dapat dimodifikasi.
6. Bagian "WATCHES" menunjukkan nilai alamat memori yang ditambahkan melalui menu **Watches**. Jika bagian ini diaktifkan (dengan menekan "Tab" hingga sekeliling bagian ini diberi warna hijau terang (*highlight*), kursor dapat digerakkan ke alamat memori yang ditampilkan. Dengan mengarahkan kursor ke alamat memori tertentu dan menekan "Enter", bagian yang ditunjuk kursor tersebut dapat dimodifikasi.
7. Bagian "HELP" menunjukkan petunjuk tombol-tombol yang dapat ditekan (*shortcut key*) dan fungsinya.

Dalam menggunakan Debugger, perhatikanlah petunjuk-petunjuk yang ada pada layar. Petunjuk penggunaan DT51D dan hal-hal yang perlu diperhatikan terdapat pada halaman 13 dan 14 dalam Manual DT-51 MinSys ver 3.0.

**Penting:** Sesuai dengan petunjuk pada halaman 14 dalam Manual DT-51 MinSys ver 3.0, disarankan agar tidak melakukan proses *debug* pada program yang menggunakan komunikasi serial.

#### Downloader Versi MS-DOS

Program ini dapat dijalankan dengan MS-DOS *prompt* dengan *syntax*: DT51L [File] [Parameter]. Program ini berfungsi untuk men-*download file* berekstensi HEX.



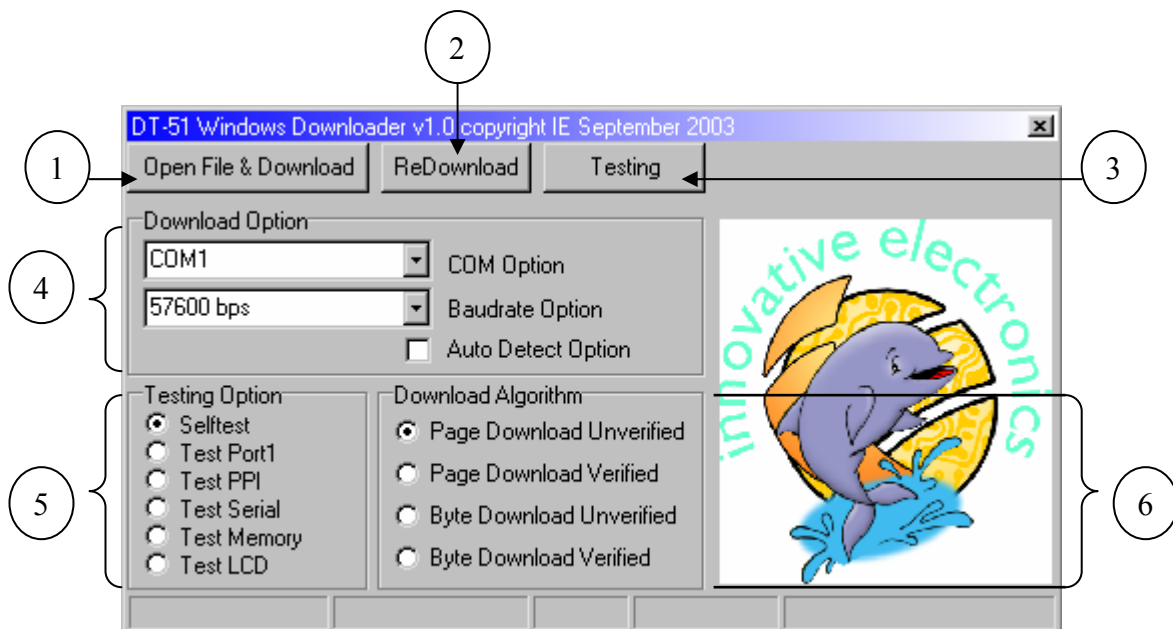
**Gambar 4**  
**DT51L Versi MS-DOS**

- File : nama *file* yang akan di-*download*. Ekstensi HEX boleh tidak dicantumkan.

- Parameter : jika tidak ada *parameter* yang dicantumkan, secara *default* DT51L akan melakukan download dengan *setting*:
  - Baudrate* dan *COM port* otomatis
  - Fast download* aktif
  - Verify* tidak aktif
 Jika ingin mengubah *setting* tersebut, cantumkan *parameter* yang bersangkutan:
  - *-b/-Bxxxxx* : *parameter baudrate* dimana *xxxxx* diisi dengan nilai kecepatan yang diinginkan yaitu: 2400 (bps), 4800 (bps), 9600 (bps), dan 19200 (bps). Contoh: *-b19200*.
  - *-c/-Cx* : *parameter COM port*, dimana *x* diisi dengan “1” jika menggunakan *COM1* dan diisi dengan “2” jika menggunakan *COM2*. contoh: *-c1*.
  - *-p/-P* : *parameter fast download* yang jika dicantumkan akan mematikan *fast download* dan membuat proses download lebih lama. *Fast download* hanya dapat digunakan untuk *SRAM* seri *62Cxxx* dan *EEPROM* seri *28CxxB*. *EEPROM* seri *28Cxx* tidak memiliki kemampuan *fast download*.
  - *-v/-V* : *parameter verify* yang jika dicantumkan akan mengaktifkan proses verifikasi dan membuat proses *download* lebih lama namun data yang di-*download* bebas dari kesalahan.
  - *-h/-H/-?* : *parameter help* yang jika dicantumkan akan menampilkan semua *parameter*. Nama *file* tidak perlu dicantumkan. Selain dengan *parameter* ini, eksekusi *DT51L* tanpa ada *File* ataupun *Parameter* juga akan menampilkan hasil yang sama.

**Downloader Versi Windows**

Program ini hanya dapat dijalankan dalam Windows (klik ke program yang bersangkutan). Kondisi *default* pada saat program dieksekusi tampak seperti pada gambar 5. Program ini berfungsi untuk men-*download file* berekstensi *HEX* dan sekaligus menjadi program *tester*.



**Gambar 5**  
**DT51L Versi Windows**

1. Tombol “Open File & Download” digunakan untuk membuka *file* dan langsung men-*download* ke DT-51 MinSys ver 3.0.
2. Tombol “ReDownload” digunakan untuk men-*download* ulang *file* terakhir yang telah di-*download*.



3. Tombol "Testing" digunakan untuk melakukan tes sesuai pilihan pada bagian "Testing Option".
4. Bagian "Download Option" digunakan untuk menentukan COM *port* dan *Baudrate* yang digunakan. Untuk mudahnya, beri tanda pada kotak "Auto Detect Option" dan program akan mendeteksi COM *port* dan *baudrate* secara otomatis.
5. Bagian "Testing Option" digunakan untuk memilih jenis tes. Pilihlah dengan cara klik pada jenis tes yang diinginkan lalu tekan tombol "Testing".
6. Bagian "Download Algorithm" digunakan untuk memilih proses *download*. *Page Download* serupa dengan *Fast Download* pada versi MS-DOS. Sedangkan *Byte Download* prosesnya lebih lama daripada *Page Download*. Proses *Verified* lebih lama daripada *Unverified* namun data yang di-*download* bebas dari kesalahan.

## S SOFTWARE DEVELOPMENT CYCLE

Setelah semua *hardware* terpasang dan teruji dan penggunaan *software* telah dipahami, lantas bagaimana dengan pembuatan program? Berikut ini adalah petunjuk pembuatan program menggunakan DT-51 MinSys ver 3.0 dan *assembler* ASM51 yang disertakan dalam paket DT-51 MinSys ver 3.0.

1. Tahap pertama dalam membuat program adalah menuliskan *listing*. *Listing* ini dapat dituliskan dengan dua cara: menggunakan program EDIT dalam MS-DOS atau NOTEPAD dalam Windows. Ketik EDIT pada MS-DOS *prompt* atau jalankan NOTEPAD pada Windows.
2. Karena *assembler* yang digunakan adalah ASM51, maka penulisan program harus sesuai dengan peraturan ASM51, antara lain:
  - di awal program harus dituliskan "\$MOD51" (tanpa tanda petik) yang menandakan bahwa program ini menggunakan *register* 89C51.
  - Penulisan label harus diberi tanda baca "titik dua".
  - Di akhir program harus dituliskan "END" (tanpa tanda petik) dan tidak boleh ada karakter lagi setelah END.
3. *Memory mapping* DT-51 MinSys mensyaratkan penggunaan alamat 4000h hingga 5FFFh untuk program sehingga program harus dimulai dari 4000h.

```

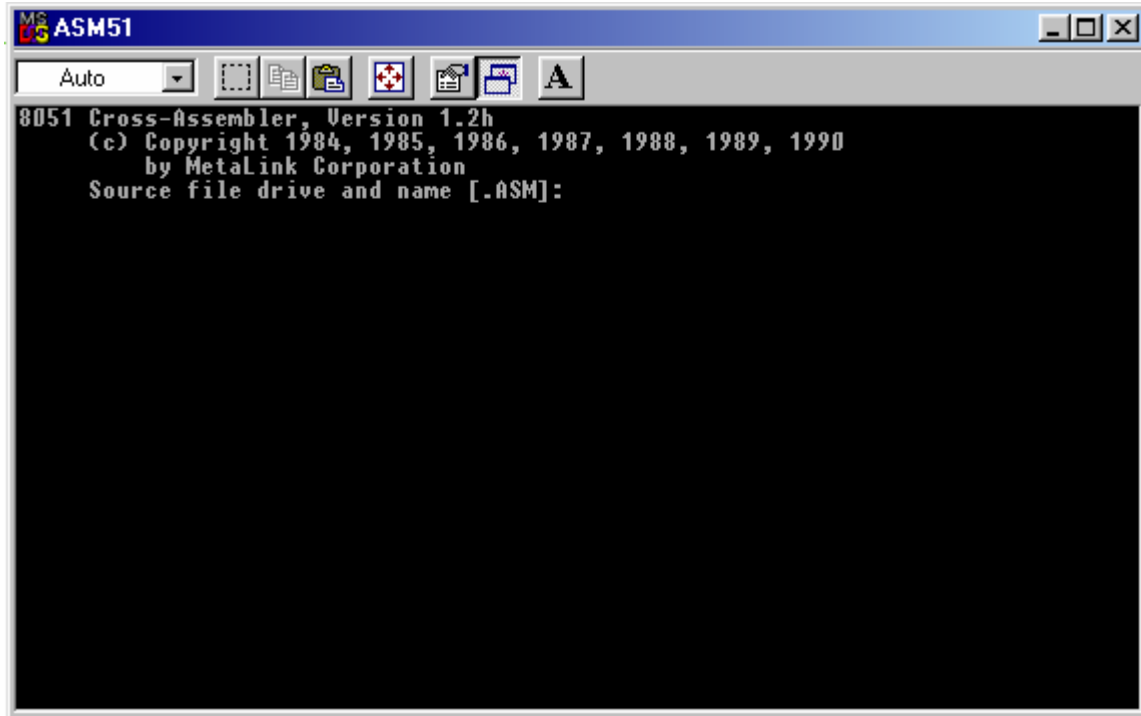
Untitled - Notepad
File Edit Search Help
$MOD51
      ORG      4000h
      LJMP    START
      ORG      4100h
START: MOV     SP, #30h
      MOV     P1, #01010101b
      SJMP    $
      END
  
```

**Gambar 6**  
**Contoh Penulisan Program dengan Notepad dan Hal-hal yang Diperhatikan**

4. Setelah selesai menuliskan program, periksalah sekali lagi apakah ada kesalahan pengetikan atau kesalahan logika. Lalu simpanlah program dengan ekstensi ASM dan nama *file* yang tidak lebih dari 8 karakter.
5. Untuk lebih mudahnya, *copy*-kan program ASM51 dan *file* MOD51 pada direktori yang sama dengan program yang dibuat sebelum proses selanjutnya dijalankan. Jika Anda bekerja pada MS-DOS *prompt*, untuk memudahkan proses, ketikkan DOSKEY terlebih dahulu. Dengan meng-*install* DOSKEY, Anda tidak perlu

mengetikkan ulang baris yang telah Anda eksekusi sebelumnya (misalkan Anda pernah menuliskan DT51L COBA.HEX. Pada *prompt* berikutnya, baris ini dapat Anda pilih dengan menekan tombol panah atas dan bawah).

6. Eksekusi/jalankan program ASM51. Saat nama *file* program diminta, ketikkan *file* berekstensi ASM yang akan di-*assemble*. Ekstensi ASM boleh tidak dicantumkan. Proses ini akan menghasilkan dua *file* bernama sama dengan *file* yang dibuat, namun berekstensi HEX dan LST.



**Gambar 7**  
**Program ASM51**

7. Perhatikan pesan di akhir proses *assembling*. Jika tidak ada kesalahan (0 Error), maka proses *download* dapat dilakukan. Jika terdapat pesan kesalahan, bukalah *file* berekstensi LST dan carilah lokasi kesalahan tersebut. Bukalah manual ASM51 jika mengalami kesulitan dalam mengartikan pesan kesalahan. Koreksilah *file* berekstensi ASM dan ulangi lagi proses *assembling* hingga tidak ada kesalahan (0 Error).
8. Setelah program di-*assemble* dengan sempurna tanpa kesalahan, *download*-lah *file* HEX dengan *downloader* (versi MS-DOS ataupun Windows) atau dengan *debugger*. Dengan *debugger*, Anda dapat melihat jalannya program tahap demi tahap. Dengan *downloader*, Anda dapat langsung menyaksikan aplikasi yang Anda buat.
9. Setelah proses *download* (dengan *downloader*) selesai, Anda dapat menyimpan program tersebut dengan memindah *jumper* RES SLCT ke mode *stand alone* agar pada saat DT-51 MinSys dinyalakan kembali, program tersebut tidak terhapus.

## MENGGUNAKAN BULIT-IN ROUTINE

Pada *kernel* DT-51 MinSys ver 3.0 telah terdapat beberapa rutin untuk menulis dan membaca memori dan mengakses LCD. Dengan rutin-rutin ini, Anda tidak perlu mempelajari prosedur baca tulis memori atau format instruksi dan data LCD. Daftar rutin ini terdapat pada halaman 15 hingga 18 dalam Manual DT-51 MinSys ver 3.0. Sebelum menggunakan rutin-rutin tersebut, lokasi rutin tersebut harus didefinisikan di awal program. Contohnya:

```
Write      EQU    0700H
CBF        EQU    0715H
dst.
```

Adapun contoh penggunaan masing-masing rutin adalah sebagai berikut:

- **Write**

```
Contoh:    MOV    A, #0AFH
           MOV    DPTR, #5F00H
           LCALL WRITE
```

Contoh di atas akan menuliskan nilai AFh ke dalam EEPROM alamat 5F00h.

- **CBF**

```
Contoh:    LCALL CBF
           AJMP  LCDREADY
```

Contoh di atas akan memeriksa *Busy Flag* LCD. Jika LCD tidak *busy*, program akan diteruskan ke baris berikutnya yaitu (dimisalkan) `AJMP LCDREADY`. Jika `InitLCD` dan rutin lain digunakan, rutin ini tidak perlu dipanggil.

- **InitLCD**

```
Contoh:    LCALL INITLCD
```

Contoh di atas akan melakukan proses inisialisasi LCD untuk mempersiapkan LCD sebelum digunakan. Jika ingin menggunakan rutin LCD yang lain, rutin ini harus dipanggil terlebih dahulu.

- **CommandLCD**

```
Contoh:    MOV    A, #01H
           LCALL COMMANDLCD
```

Contoh di atas akan menghapus semua tampilan LCD. Hal ini dikarenakan nilai 01h yang dimasukkan ke *accumulator* adalah nilai *Command Word* untuk *Display Clear*. Cara yang lebih mudah adalah dengan memberi nama pada nilai 01h agar mudah digunakan. Contohnya:

```
DisplayClear    EQU    01H
                .
                .
                .
                MOV    A, #DisplayClear
                LCALL COMMANDLCD
```

- **WriteLCD**

```
Contoh:    MOV    A, #41H
           LCALL WRITELCD
```

Contoh di atas akan menuliskan (bukan berarti menampilkan) nilai 41h (kode ASCII huruf "A") ke LCD. Jika alamat yang dipilih sebelumnya adalah alamat DDRAM (alamat karakter pada layar LCD), maka pada LCD akan tampil huruf "A".

Jika alamat yang dipilih sebelumnya adalah alamat CGRAM (alamat pada memori karakter LCD), maka pada layar LCD tidak akan tampil apa-apa karena nilai tersebut masuk ke dalam CGRAM.

- **ReadLCD**

```
Contoh:    LCALL READLCD
```

Contoh di atas akan membaca nilai yang terdapat pada alamat yang ditunjuk.

Jika alamat yang dipilih sebelumnya adalah alamat DDRAM (alamat karakter pada layar LCD), maka karakter yang ditunjuk oleh alamat tersebut akan dibaca dan (kode ASCII-nya) dimasukkan ke *accumulator*.

Jika alamat yang dipilih sebelumnya adalah alamat CGRAM (alamat pada memori karakter LCD), maka nilai pada alamat tersebut akan dibaca dan dimasukkan ke *accumulator*.

- **ReadAddrLCD**

```
Contoh:    LCALL READADDRLCD
```

Contoh di atas akan membaca lokasi alamat yang ditunjuk.

Jika alamat yang dipilih sebelumnya adalah alamat DDRAM (alamat karakter pada layar LCD), maka lokasi alamat yang ditunjuk tersebut akan dibaca dan dimasukkan ke *accumulator*.

Jika alamat yang dipilih sebelumnya adalah alamat CGRAM (alamat pada memori karakter LCD), maka lokasi alamat tersebut akan dibaca dan dimasukkan ke *accumulator*.

#### - **SetDDRAM**

Contoh:       MOV    A, #40H  
              LCALL SETDDRAM

Contoh di atas akan memindah (menunjuk) alamat DDRAM ke lokasi 40h (karakter pertama baris LCD kedua).

#### - **SetCGRAM**

Contoh:       MOV    A, #08H  
              LCALL SETCGRAM

Contoh di atas akan memindah (menunjuk) alamat CGRAM ke lokasi 08h (baris pertama karakter CGRAM kedua).

Untuk keterangan lebih lanjut tentang LCD (terutama mengenai DDRAM dan CGRAM), bacalah manual LCD yang Anda miliki.

## **F**REQUENTLY ASKED QUESTIONS

Berikut ini merupakan beberapa pertanyaan (tentu saja beserta jawabannya) yang sering ditanyakan oleh pengguna DT-51 MinSys.

### **Software**

- Q: **Bagaimana cara membuat program hingga siap di-download?**
- A: Bacalah bagian *Software Development Cycle*.
  
- Q: **Dapatkah menggunakan Assembler MCS-51 selain ASM51 untuk digunakan dengan DT-51 MinSys?**
- A: Dapat, selama program tersebut mampu mengenali alamat *register* 89C51.
  
- Q: **Dapatkah Turbo Assembler 2 digunakan untuk mengubah ASM ke HEX?**
- A: Tidak. Turbo Assembler yang dikenal dengan program TASM dan TLINK menggunakan bahasa Assembly untuk tipe mikroprosesor 8088, bukan mikrokontroler MCS-51.
  
- Q: **Dapatkah membuat program dengan bahasa C untuk DT-51 MinSys?**
- A: Dapat. Saat ini sudah terdapat *compiler/assembler* bahasa C untuk mikrokontroler MCS-51, antara lain SDCC.
  
- Q: **Bagaimana cara membuat program dengan assembler ASM51 agar tidak ada error?**
- A: Bacalah bagian *Software Development Cycle* untuk petunjuk dasarnya. Keterangan lebih lengkap terdapat pada Manual ASM51.
  
- Q: **Apa itu Stack Pointer dan bagaimana cara menentukan nilainya?**
- A: Stack Pointer (SP), sesuai dengan namanya, adalah nilai yang menunjukkan alamat *stack memory* teratas.  
Jika dalam suatu program terdapat perintah untuk memanggil rutin (misalkan dengan CALL atau adanya *interrupt*) atau menyimpan nilai dengan PUSH, SP akan bertambah sesuai dengan lebar data yang disimpan (CALL atau *interrupt* menambah SP dengan 2, PUSH menambah SP dengan 1).  
Jika dalam suatu program terdapat perintah untuk keluar dari rutin (misalkan dengan RET atau RETI) atau mengambil data dengan POP, SP akan berkurang sesuai dengan lebar data yang diambil (RET atau RETI mengurangi SP dengan 2, POP mengurangi SP dengan 1).  
Cara yang paling aman dalam menentukan nilai SP adalah dengan meletakkan variabel pada alamat rendah dan meletakkan SP setelah variabel tersebut. Misalkan jumlah variabel yang digunakan adalah 5 dan diletakkan pada alamat 22h – 26h, maka SP dapat diletakkan pada alamat 27h. Dengan begitu perubahan nilai SP tidak akan mengganggu nilai variabel yang digunakan.
  
- Q: **Bagaimana jika SP menembus alamat 7Fh?**
- A: Program dapat berhenti (*hang*). *Internal program memory* untuk *scratch pad* hanya sampai 7Fh. Alamat 80h – FFh digunakan untuk menyimpan *Special Function Register* (SFR) yang isinya antara lain adalah *port* 0 – *port* 3 dan *register* untuk *interrupt*, *timer*, dan komunikasi serial (IE, IP, TCON, SCON, dsb.).

Meski kami belum pernah mencobanya, namun secara logika program akan kacau saat SP menembus alamat 7Fh. Hal ini dikarenakan adanya perubahan nilai SFR yang tidak “seharusnya”.

- Q: **Bagaimana jika DT51D tidak mau *download* atau DT51L tidak mau *download* dan *counter* pada “Failed : times” bertambah terus?**
- A: Hal itu menandakan DT51L/DT51D tidak dapat membuka COM *port* untuk komunikasi. Umumnya hal ini disebabkan adanya program lain yang menggunakan COM *port* (MS-DOS *prompt* lain yang terbuka, program Hyper Terminal yang terhubung, dsb.). Tutuplah program-program tersebut dan *download*-lah ulang. Jika masalah ini masih berlanjut, cobalah *boot* ulang komputernya.
- Q: **Bagaimana jika muncul pesan “Download Failed”?**
- A: Ada kemungkinan hardware DT-51 MinSys mengalami kerusakan. Hubungi Innovative Electronics untuk lebih jelasnya.
- Q: **Bagaimana cara menghapus program yang sudah di-*download*?**
- A: Program yang di-*download* dapat dihapus dengan dua cara. Cara pertama adalah dengan men-*download* program baru untuk menumpanginya. Pada cara kedua, matikan sumber tegangan, pindah *jumper* RES SLCT ke posisi 1-2 lalu nyalakan sumber tegangan. Dengan begitu isi EEPROM akan dihapus.
- Q: **Bagaimana cara *download* ke *internal program memory* 89C51 dengan DT-51 MinSys?**
- A: DT-51 MinSys tidak dapat digunakan untuk men-*download* ke *internal program memory* dan tidak disarankan untuk melakukannya dengan *programmer/writer* karena hal itu akan menghapus *kernel code*. Sedangkan *internal data memory*-nya masih dapat digunakan sepenuhnya.
- Q: **Bagaimana cara menggunakan *port A*, *B*, dan *C* sebagai jalur *input/output* seperti *port 1*?**
- A: PPI tidak dapat langsung digunakan seperti *port 1*. Masing-masing *port* harus ditentukan terlebih dahulu apakah berfungsi sebagai *input* atau *output*. Bacalah *data sheet* 82C55 untuk lebih jelasnya.
- Q: **Dapatkah DT-51 MinSys melakukan komunikasi serial? Kapan dan bagaimana?**
- A: Dapat, yaitu pada saat proses *download* dan *debug*, serta jika program yang dibuat menggunakan komunikasi serial. Pada proses *download* dan *debug*, komunikasi serial langsung oleh *kernel code*. Jika ingin membuat program yang mampu berkomunikasi secara serial, ada beberapa *register* yang harus diatur terlebih dahulu. Bacalah “Mikrokontroler 89C51” pada bagian Tutorial di [www.innovativeelectronics.com](http://www.innovativeelectronics.com) sebagai referensinya.
- Q: **Di mana dan bagaimana cara mendapatkan contoh aplikasi DT-51 MinSys?**
- A: Aplikasi untuk semua produk Innovative Electronics terdapat pada [www.innovativeelectronics.com](http://www.innovativeelectronics.com) dan dapat di-*download* dengan bebas secara cuma-cuma.

## Hardware

- Q: **Dapatkah DT-51 MinSys ver 3.0 dihubungkan dengan menggunakan kabel serial 25 pin (DB25) untuk komputer lama?**
- A: Dapat. bacalah Manual DT-51 MinSys ver 3.0 halaman 23 untuk lebih jelasnya.
- Q: **Apa beda DT-51 MinSys ver 3.0 dengan DT-51 PetraFuz?**
- A: Secara fisik keduanya memang sama. Namun pada DT-51 PetraFuz telah tertanam algoritma *Fuzzy Logic*. Pendek kata, DT-51 PetraFuz merupakan DT-51 MinSys yang diberi kemampuan *Fuzzy Logic*.
- Q: **Bagaimana jika 89C51 pada DT-51 MinSys rusak dan diganti 89C51 yang baru?**
- A: 89C51 (yang baru) tidak dapat langsung dipasang ke DT-51 MinSys untuk menggantikan yang lama. Hal ini dikarenakan dalam 89C51 DT-51 MinSys telah tertanam *kernel code*. *Kernel code* ini berfungsi dalam proses *download* program dan menjalankan program yang di-*download* tersebut.
- Q: **Dapatkah EEPROM 28C64B digantikan dengan memori lain semisal RAM 62C64 atau EPROM 27C64?**
- A: Pada dasarnya penggantian EEPROM 28C64B pada DT-51 MinSys dengan memori tipe lain tidak menjadi masalah asalkan semua kakinya memiliki fungsi dan *level* tegangan yang sama serta timing diagram yang sesuai. Namun RAM bersifat *volatile* dimana data yang tersimpan akan hilang jika sumber tegangan dimatikan. Sedangkan EEPROM akan tetap menyimpan data meski tidak ada sumber tegangan. Penggunaan EPROM sedikit sulit dilakukan karena EPROM 27C64 adalah UVEPROM yang membutuhkan sinar ultraviolet untuk menghapus datanya.

- Q: **Bagaimana jika hanya menggunakan 89C51 pada DT-51 MinSys tanpa EEPROM 28C64B dan PPI 82C55?**
- A: Tanpa adanya PPI 82C55, Port A, B, dan C tidak dapat digunakan lagi sedangkan CS1 dapat digunakan untuk *peripheral* lain. Jika EEPROM 28C64B tidak digunakan, maka proses *download* atau *debug* dengan DT51L atau DT51D tidak dapat dilakukan sedangkan CS2 dapat digunakan untuk *peripheral* lain. Dalam kondisi seperti itu, 89C51 akan beroperasi dalam mode *single chip* dimana program yang akan dijalankan harus dituliskan ke dalam 89C51 tersebut.
- Q: **Dapatkah DT-51 MinSys digunakan sebagai *writer/programmer* untuk 89C51 dan 28C64B?**
- A: DT-51 MinSys merupakan *development tool* dan tidak dapat digunakan sebagai *programmer* 89C51. Namun DT-51 MinSys masih dapat digunakan sebagai *writer* 28C64B. Hal ini dikarenakan proses *download* dengan DT51L pada dasarnya adalah menuliskan serangkaian data ke EEPROM. Contoh jika DT-51 MinSys digunakan untuk menuliskan serangkaian data ke EEPROM adalah sebagai berikut:

```

ORG    4000H
DB     00H
DB     01H
DB     02H
END

```

Contoh tersebut akan menuliskan data 00h – 02h masing-masing pada alamat 4000h – 4002h (3 *byte* pertama EEPROM).

- Q: **Bagaimana penjelasan *memory mapping* DT-51 MinSys?**
- A: DT-51 MinSys mampu mengakses memori hingga 64 KB yang dibagi menjadi 8 bagian (masing-masing berkapasitas 8 KB dan memiliki 1 *Chip Select/CS*).

Alamat yang diakses	CS yang aktif
0000 – 1FFFh	CS0
2000 – 3FFFh	CS1
4000 – 5FFFh	CS2
6000 – 7FFFh	CS3
8000 – 9FFFh	CS4
A000 – BFFFh	CS5
C000 – DFFFh	CS6
E0000 – FFFFh	CS7

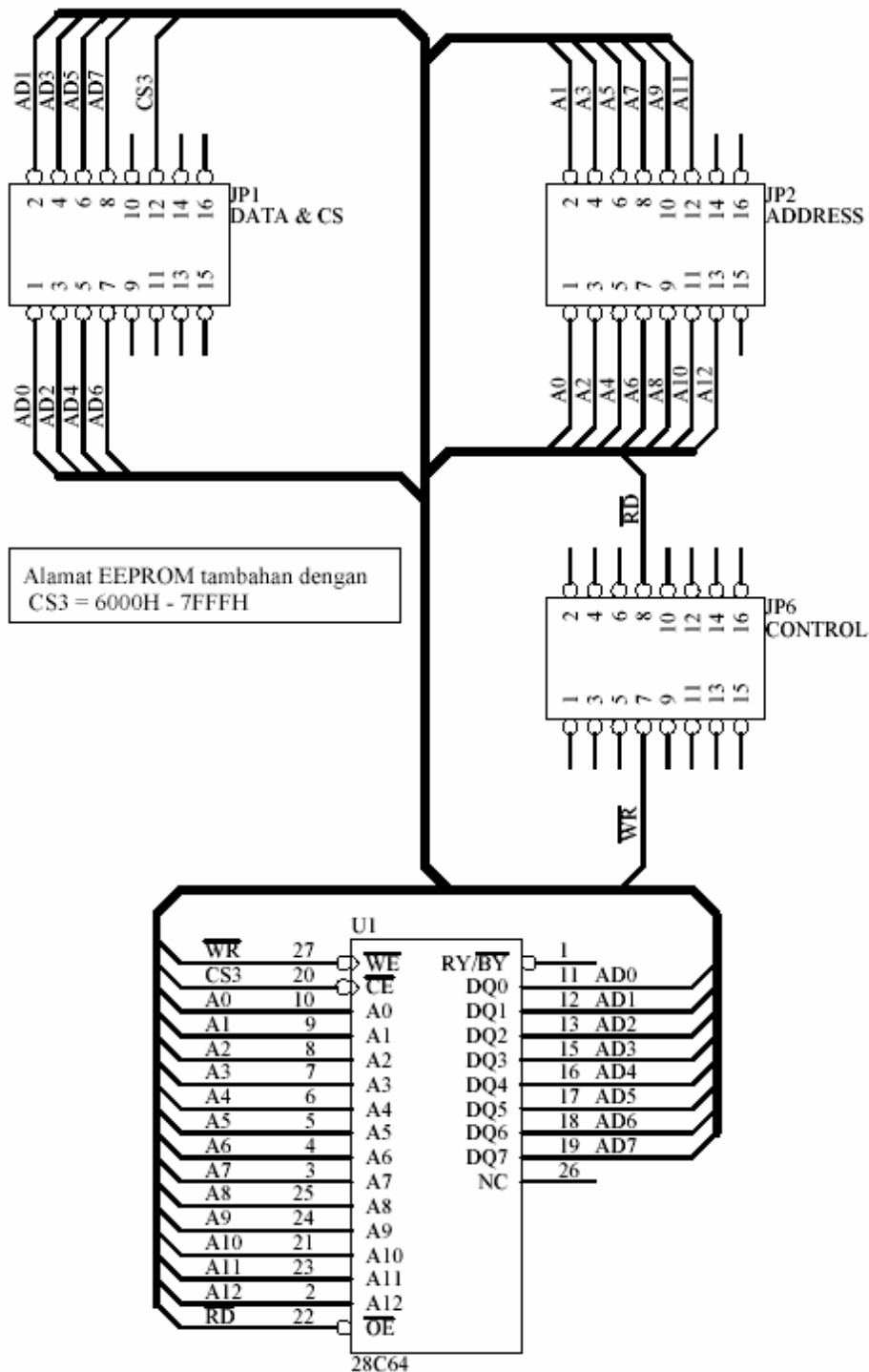
**Tabel 2**  
**Hubungan Alamat dan CS**

PPI 82C55 pada DT-51 MinSys dihubungkan dengan CS1 sehingga IC tersebut menempati lokasi 2000h – 3FFFh. Namun karena PPI hanya menggunakan A0 dan A1, maka alamat yang digunakan hanyalah 2000h – 2003h.

EEPROM 28C64B pada DT-51 MinSys dihubungkan dengan CS2 sehingga IC tersebut menempati lokasi 4000h – 5FFFh.

CS3 hingga CS7 masih belum terpakai dan dapat digunakan untuk ekspansi memori atau *peripheral* yang lain.

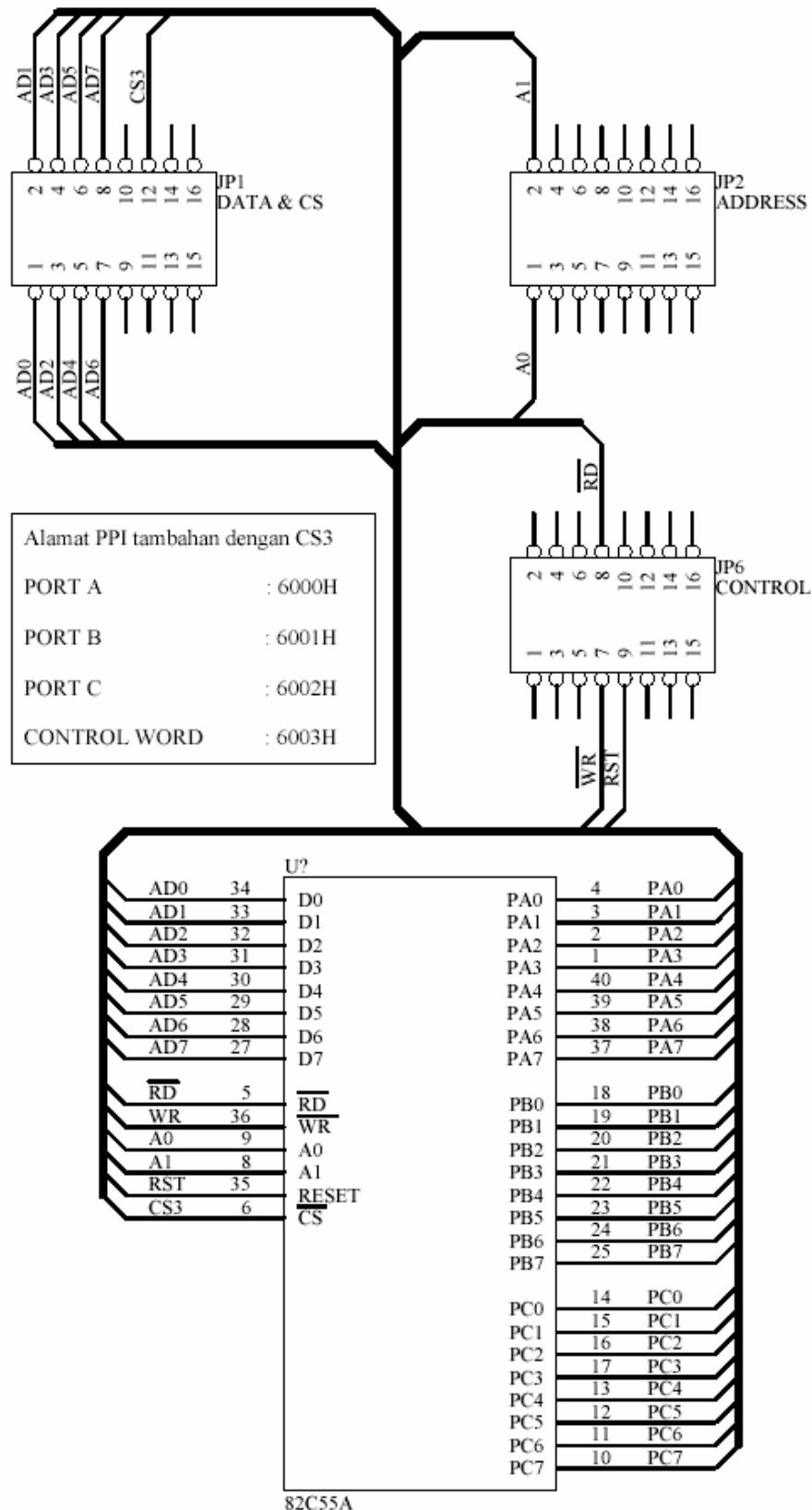
- Q: **Dapatkah menggunakan alamat di bawah 2000h? Bukankah CS0 tidak dihubungkan ke manapun juga?**
- A: Alamat di bawah 2000h dapat digunakan sebagai *data memory*. Cara mengaksesnya adalah dengan perintah MOVX. Rangkaiannya serupa dengan penambahan memori 28C64B (lihat contoh pertanyaan selanjutnya). Sedangkan penggunaan *program memory* selain pada 4000h – 5FFFh tidak didukung DT-51 MinSys.
- Q: **Bagaimana jika ingin menambah memori 8 KB tambahan misalnya 28C64B?**
- A: Seperti yang sudah dijelaskan pada bagian “Mempelajari Konektor pada DT-51 MinSys ver 3.0”, penambahan memori atau *peripheral* dilakukan dengan cara menghubungkan DATA, ADDRESS, CS, dan sinyal kontrol RD dan WR. Perhatikan bahwa memori 8 KB membutuhkan 13 *pin* alamat, A0 – A12. Salah satu contoh rangkaiannya terdapat pada gambar 8.



**Gambar 8**  
**Rangkaian EEPROM 8 KB Tambahan**

Perhatikan bahwa memori yang ditambahkan menempati alamat 6000h – 7FFFh dan hanya dapat berfungsi sebagai *data memory*. Hal ini dikarenakan DT-51 MinSys ver 3.0 hanya mendukung penempatan *program memory* pada alamat 4000h – 5FFFh.

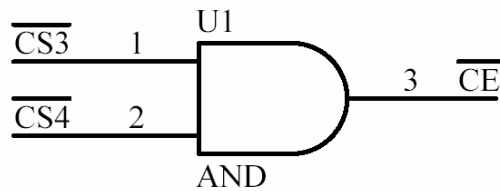
- Q: **Bagaimana jika ingin menambah PPI 82C55 lagi?**
- A: Hampir sama halnya dengan penambahan EEPROM dimana DATA, ADDRESS, CS, dan sinyal kontrol RD dan WR harus dihubungkan. Perhatikan bahwa PPI hanya membutuhkan dua *pin* alamat, A0 dan A1 (seolah-olah PPI sama dengan memori berkapasitas 4 byte). Salah satu contoh rangkaiannya terdapat pada gambar 9.



**Gambar 9**  
**Rangkaian PPI Tambahan**

- Q: **Bagaimana jika memori yang ditambahkan berkapasitas lebih dari 8 KB?**
- A: Jika memori yang berkapasitas lebih dari 8 KB dan dihubungkan hanya dengan 1 CS, maka hanya 8 KB pertama yang dapat diakses. Sisa kapasitas memori tidak dapat digunakan. Jika ingin menggunakan semua kapasitas memori, maka harus ada rangkaian tambahan berupa gerbang AND yang menghubungkan CS dengan *Chip Enable* (CE). Contoh rangkaian terdapat pada gambar 10.

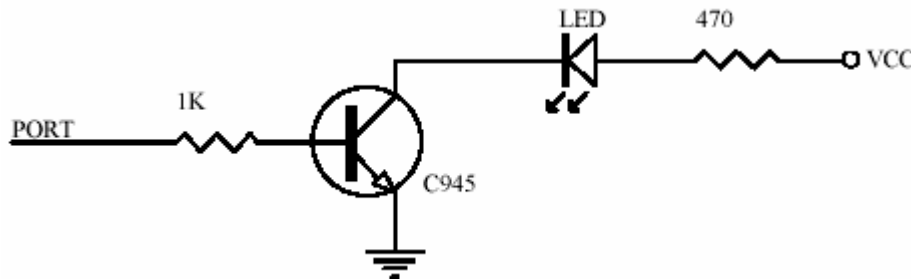




**Gambar 10**  
**Rangkaian AND untuk Memori Berkapasitas 16 KB**

Pada gambar 10 dimisalkan terdapat memori 16 KB yang diletakkan pada alamat 6000h – 9FFFh. Perhatikan bahwa jika salah satu CS aktif (berlogika '0'), maka CE juga akan aktif. Sedangkan *Address* yang dihubungkan mengikuti jumlah *pin Address* yang tersedia pada memori atau *peripheral* tersebut. Misalkan pada memori 16 KB akan terdapat 14 *pin Address*, maka *pin* yang dihubungkan adalah A0 – A13.

- Q: **Bagaimana cara menghubungkan dengan rangkaian lain?**
- A: Pelajari terlebih dahulu rangkaian tersebut. Teliti apakah rangkaian tersebut memiliki *level* tegangan yang sama dengan DT-51 MinSys. Jika tidak maka harus dibuat semacam konverter *level* tegangan. Jika rangkaian tersebut siap dihubungkan ke DT-51 MinSys, tentukan ke *port* mana rangkaian tersebut harus dihubungkan. Setelah semua jalur data (dan alamat) terhubung, hubungkan juga referensi *ground* rangkaian tersebut (dengan asumsi bahwa rangkaian tersebut sudah memiliki standar *level* tegangan TTL).
- Q: **Bagaimana cara membuat rangkaian *input output* sederhana dan membuat programnya?**
- A: Komponen *output* yang sering digunakan adalah LED sedangkan komponen *input* yang sering digunakan adalah saklar baik *tactile switch* (*push-button*) ataupun *toggle switch*. Contoh rangkaian terdapat pada gambar 11 dan 12.



**Gambar 11**  
**Rangkaian LED**

Hubungkan jalur berlabel PORT ke masing-masing *pin* pada *port* 1, A, B, atau C. Berikut ini merupakan contoh program jika LED dihubungkan ke *port* 1.

```

$mod51

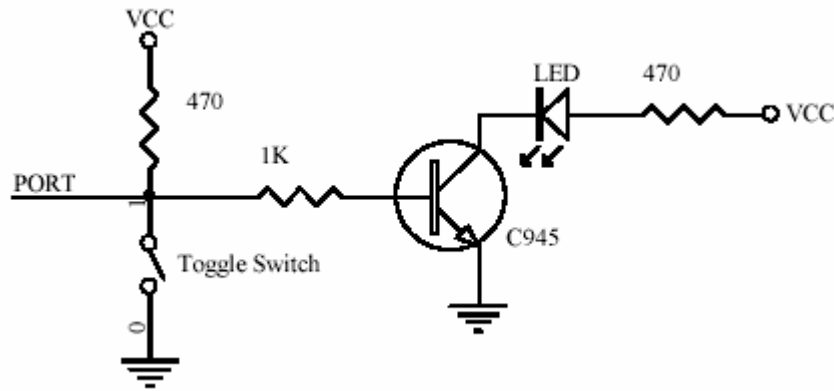
                ORG    4000h
                LJMP   START

START:          MOV    SP, #30H
                MOV    P1, #10101010B           ;output ke LED
                SJMP   $

END

```

Sesuai contoh tersebut, LED yang akan menyala adalah LED pada P1.7, P1.5, P1.3, dan P1.1 (yang diwakili oleh angka "1").



**Gambar 12**  
**Rangkaian Toggle Switch**

Hubungkan jalur berlabel PORT ke masing-masing *pin* pada *port* 1, A, B, atau C. Berikut ini merupakan contoh program jika *toggle switch* dihubungkan ke *port* 1.

```

$mod51

                ORG    4000h
                LJMP   START

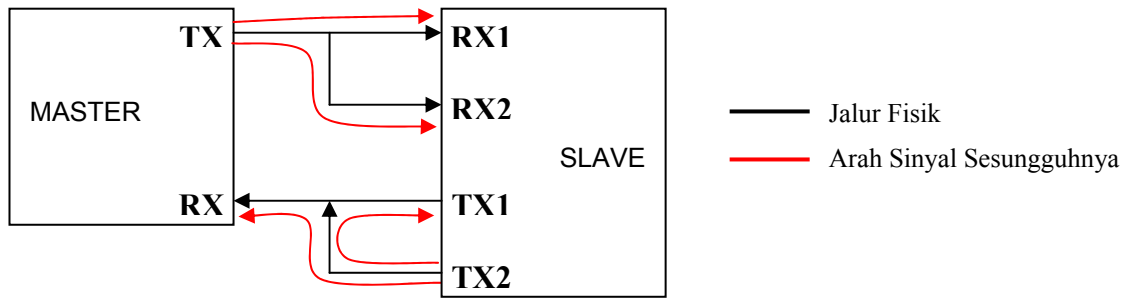
START:         MOV    SP, #30H
                MOV    A, P1                ;input dari switch
                SJMP   $

END

```

Sesuai contoh tersebut, kondisi/nilai *toggle switch* akan dipindah ke *accumulator*. Jika terdapat *toggle switch* yang menghasilkan logika “1”, maka *bit* yang bersangkutan akan bernilai “1”. Begitu juga sebaliknya.

- Q: **Mengapa LED yang dipasang pada *port* 1 tampak lebih redup bila dibandingkan bila dipasang pada *port* A, B, atau C? Apakah DT-51 MinSys rusak?**
- A: Tidak rusak. Kemampuan kedua jenis *port* dalam mengalirkan arus memang berbeda. Bacalah *data sheet* 89C51 dan 82C55 untuk lebih jelasnya.
- Q: **Bagaimana *interface* ke LCD?**
- A: Perhatikan alokasi kaki modul LCD yang akan dihubungkan ke DT-51 MinSys. Bacalah *data sheet* modul LCD untuk mengetahui apakah alokasi *pin*, *level* tegangan, dan *timing diagram* sinyalnya kompatibel dengan kontroler HD44780.
- Q: **Bagaimana caranya mematikan *back light* LCD?**
- A: *Back light* LCD pada *port* LCD tidak dapat dimatikan kecuali ada semacam rangkaian pemutus arus antara *port* LCD dengan modul LCD (lihatlah skema DT-51 MinSys ver 3.0 untuk jelasnya). Jika tidak ingin menyalakan *back light* sama sekali, *pin* 15 dan 16 pada *port* LCD tidak usah dihubungkan ke modul LCD.
- Q: **Bagaimana cara menjalankan dalam kondisi Stand Alone?**
- A: *Download*-lah program yang akan dijalankan. Pindah *jumper* RES SLCT ke posisi 2-3 sebelum menyalakan sumber tegangan berikutnya.
- Q: **Dapatkah menambah IC 232 tambahan secara paralel untuk melakukan komunikasi serial antara tiga pihak? Bagaimana?**
- A: Tidak disarankan. Perhatikan ilustrasi pada gambar 13 untuk lebih jelasnya.



**Gambar 13**  
**Ilustrasi Komunikasi Serial 3 Pihak**

Pada ilustrasi gambar 13 terdapat satu *master* (TX – RX) dengan dua *slave* (TX1 – RX1 dan TX2 – RX2). Tidak menjadi masalah jika TX *master* mengirimkan data dan diterima oleh kedua RX. Namun jika terdapat salah satu *slave* yang mengirimkan data (melalui TX1 atau TX2) ke RX *master*, data tersebut juga akan masuk ke TX *slave* yang lain. Kondisi ini dapat merusak IC 232 karena *pin* TX seharusnya mengeluarkan data, bukan menerima data.

**S**elamat berinovasi!

DT-51 merupakan merk dagang dari Innovative Electronics.  
MS-DOS dan Windows adalah merk dagang terdaftar dari Microsoft Corporation.