

Percobaan

4

JUDUL :

PEMROGRAMAN LCD
Menggunakan DT-51 MinSys

TUJUAN :

- Mengakses built-in routine LCD pada DT-51 Minimum sistem.
- Membuat program untuk menampilkan karakter ASCII pada LCD.

TEORI :

Dalam DT-51 minimum sistem telah diberi built-in routine yang ditujukan untuk memudahkan pengguna LCD.

CBF (lokasi 0175h)

Fungsi CBF adalah untuk memeriksa Busy Flag LCD jika LCD akan diakses secara manual tanpa menggunakan built-in routine. Untuk memudahkan pemakaian LCD, kita akan mempelajari built-in routine sehingga tidak perlu mengakses CBF.

InitLCD (lokasi 0740h)

Routine ini digunakan untuk menginisialisasi LCD sebelum menggunakan LCD.

CommandLCD (lokasi 07B0h)

Routine ini digunakan untuk memberikan perintah untuk LCD. Routine ini dipanggil setelah kita mengisi register A dengan nilai command word. Ada beberapa perintah (command word) yang memiliki nilai tersendiri.

Command Word	Nilai	Fungsi
DisplayClear	01h	Menghapus semua tampilan di LCD
CursorHome	02h	Meletakkan kursor dan display pada posisi awal
DecCursor	04h	Kursor decrement setiap kali selesai menulis atau membaca LCD
CDDSR	05h	Kursor decrement dan display bergeser ke kanan setiap kali selesai menulis atau membaca LCD
IncCursor	06h	Kursor increment setiap kali selesai menulis atau membaca LCD
ICDSL	07h	Kursor increment dan display bergeser ke kiri setiap kali selesai menulis atau membaca LCD
DisplayOff	08h	Memadamkan display LCD, display bisa menampilkan kembali jika ada perintah untuk menyalakan display. Dari command word yang ada kita bisa mengakses kursor (memadamkan, menghidupkan atau blinking) untuk menyalakan display. Selama display padam LCD masih bisa ditulisi tidak menampilkan apa-apa.
CursorOff	0Ch	Memadamkan kursor
CursorOn	0Eh	Menghidupkan kursor
CursorBlink	0Fh	Kursor menjadi blinking (berkedip)
CurShLeft	10h	Menggeser kursor ke kiri
CurShRight	14h	Menggeser kursor ke kanan
DispShLeft	18h	Menggeser display ke kiri
DispShRight	1Ch	Menggeser display ke kanan

Jadi jika kita ingin menghapus semua tampilan LCD, kita isi A dengan 1h (DisplayClear) lalu panggil routine CommandLCD.

WriteLCD (lokasi 07D0h)

Routine ini berfungsi untuk menuliskan karakter ke LCD yang terdapat pada register A. Karakter pada LCD membutuhkan 8 bit data.

Kode DDRAM (Display Data RAM) dari tiap karakter bisa dilihat pada manual LCD. Sebagian dari karakter pada manual LCD sama dengan karakter ASCII, Jadi

jika kita ingin menampilkan huruf J, kita tinggal mengisi register A dengan 01001010h (4Ah) dan memanggil routine WriteLCD.

ReadLCD (lokasi 07F0h)

Routine ini berfungsi untuk membaca karakter dari LCD atau CGRAM (Character Generator RAM) kemudian memindahkannya ke register A. Untuk membaca karakter dari LCD, kita bisa memilih alamat dengan cara memanggil routine SetDDRAM untuk menentukan alamat yang kita inginkan. Sedangkan untuk membaca karakter dari CGRAM, kita panggil dulu routine SetCGRAM untuk menentukan alamatnya.

ReadAddrLCD (lokasi 0820h)

Routine ini berfungsi untuk membaca alamat LCD atau CGRAM kemudian memindahkannya ke register A. Untuk membaca alamat dari LCD, kita bisa menentukan alamat dengan cara memanggil routine SetDDRAM. Sedangkan untuk membaca alamat dari CGRAM, kita panggil dulu routine SetCGRAM untuk menentukan alamatnya.

SetDDRAM (lokasi 0850h)

Routine ini berfungsi untuk menentukan alamat DDRAM pada alamat tertentu yang terdapat pada register A sebelum memulai menulis atau membaca LCD. Alamat DDRAM adalah alamat pada tampilan LCD.

Konfigurasi alamat DDRAM untuk dua baris adalah sebagai berikut :

Display baris I : 00h – 27h

Display baris II : 40h – 67h

Jika LCD yang ada berformat 16 x 2, maka pada baris I adalah alamat DDRAM 00h – 0Fh dan baris II adalah alamat 40h – 4Fh. Jika display digeser ke kiri satu kali, maka yang tampil pada baris I adalah alamat 01h – 10h dan baris II adalah alamat 41h – 50h.

SetCGRAM (lokasi 0870h)

Routine ini berfungsi untuk menentukan alamat CGRAM pada alamat tertentu yang terdapat pada register A sebelum memulai menulis atau membaca CGRAM.

CGRAM adalah alamat per baris dari tiap kotak tampilan. Satu karakter memiliki 8 alamat CGRAM (7 baris untuk karakter dan 1 baris terakhir untuk kursor).

LCD Port

LCD (Liquid Crystal Display) Port ini disiapkan untuk men-drive LCD melalui Port 1 mikrokontroler 89C51. Rutin untuk keperluan LCD sudah tersedia pada PEROM 89C51 sehingga user tinggal memakainya.

PERALATAN :

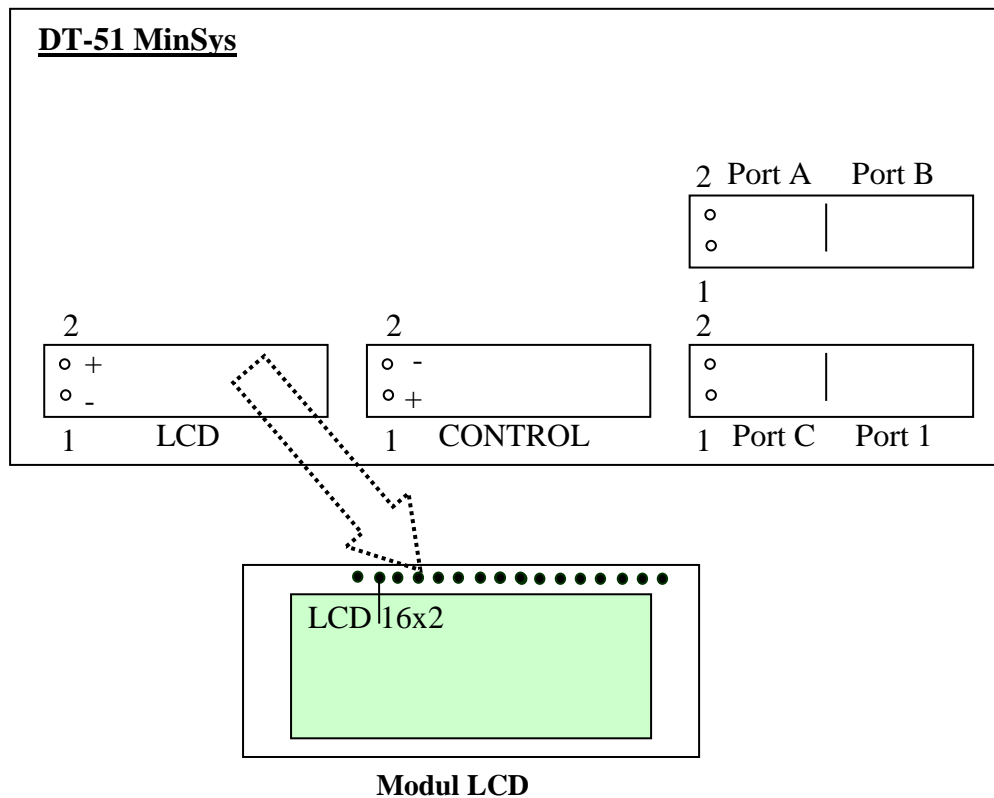
- Modul DT-51
- PC beserta software
 - Debugger
 - Downloader
 - Editor
- LCD 16 x 2 sebagai modul display

PROSEDUR :

Pemakaian Port LCD pada DT-51

1. Hardware

- Hubungkan konektor (kabel pita) Port LCD pada DT-51 ke Modul Display LCD seperti ditunjukkan pada gambar dibawah ini (gambar 4-1).
- Hubungkan kabel serial dari konektor DB-9 pada DT-51 ke PC pada Port serial.
- Hubungkan catu daya 9 Volt AC pada modul DT-51.
- Pin Konektor Select pada posisi Pin 1 dan 2 untuk mode *download program*, pada posisi Pin 2 dan 3 untuk mode *stand alone* , setelah download program berhasil.



Gambar 4-1. Koneksi DT-51 MinSys dengan Modul LCD

Selanjutnya

- Hubungkan kabel serial dari konektor DB-9 pada DT-51 ke PC pada Port serial.
- Hubungkan catu daya 9 Volt AC pada modul DT-51.
- Pin Konektor Select pada posisi Pin 1 dan 2 untuk mode **download program**, pada posisi Pin 2 dan 3 untuk mode **stand alone**, setelah download program berhasil.

2. Software

Editor :

- Tulislah program pada editor DOS prompt, Notepad atau lainnya.
- Program / file baru tersebut, simpan dalam direktori(folder) DT51 dengan extension *[namafile].ASM*.
- Exit

Debugger :

- Compile file baru dengan perintah `C:\DT51\ASM51 [nama file].ASM` pada DOS Prompt, dengan terlebih dahulu pindah ke direktori DT51.

- Maka akan terbentuk file-file yang berekstension .HEX, .OBJ, dan .LST disamping itu juga muncul pesan, bahwa file yang telah dicompile tersebut terdapat kesalahan atau tidak.
- Bila ada kesalahan, bisa dilihat letak kesalahannya dengan membuka file yang berekstension .LST, melalui C:\DT51\edit.
- Untuk memperbaiki kesalahan (error), buka file yang berekstension .ASM. Jangan lupa disimpan kemudian dicompile kembali.
- Exit

Downloader :

- Untuk mendownload program yang telah sukses dicompile, maka gunakan perintah C:\DT51\DT51L [nama file].HEX
- Apabila proses download selesai, maka akan muncul pesan Download Succeeded.
- Bila gagal, maka akan muncul beberapa prosedur untuk mengecek kembali kesalahan tersebut.

PROGRAM PERCOBAAN :

- *Program 4.1. Program untuk menampilkan karakter "1ABC"*

```

$MOD51
;-----
;Procedure Address
;-----
InitLCD          EQU    0740H
CommandLCD      EQU    07B0H
WriteLCD        EQU    07D0H
;-----
;Command Word
;-----
CursorHome      EQU    02H
DisplayOff      EQU    08H
CursorOff       EQU    0CH
CursorOn        EQU    0EH
CursorBlink     EQU    0FH
CurShRight     EQU    14H
CurShLeft      EQU    10H
DispShRight     EQU    1CH
DispShLeft      EQU    18H

```

```

                                CSEG
                                ORG  4000H
                                LJMP  Start

                                ORG  4100H
LDelay:  PUSH  02H
                                PUSH  03H
                                MOV   R2,#0bH
Del1:    MOV   R3,#0FFH
Del2:    MOV   R4,#0FFH
                                DJNZ  R4,$
                                DJNZ  R3,Del2
                                DJNZ  R2,Del1
                                POP   03H
                                POP   02H
                                RET

Start:   MOV   SP,#40H
                                LCALL InitLCD

;-----
;Menampilkan "1ABC"
;-----
                                MOV   A,#31H
                                LCALL WriteLCD
                                LCALL LDELAY
                                MOV   A,#41H
                                LCALL WriteLCD
                                LCALL LDELAY
                                MOV   A,#'B'
                                LCALL WriteLCD
                                LCALL LDELAY
                                MOV   A,#43H
                                LCALL WriteLCD
                                LCALL LDELAY

;-----
;Menampilkan kursor blinking
;-----
                                MOV   A,#CursorBlink
                                LCALL CommandLCD
                                LCALL LDELAY

;-----
;Memadamkan kursor
;-----
                                MOV   A,#CursorOff
                                LCALL CommandLCD
                                LCALL LDELAY

;-----
;Menampilkan kursor
;-----
                                MOV   A,#CursorOn
                                LCALL CommandLCD
                                LCALL LDELAY

```

```

;-----
;Memadamkan DISPLAY
;-----
        MOV    A,#DisplayOff
        LCALL  CommandLCD
        LCALL  LDELAY
;-----
;Menyalakan tampilan dgn mengakses kursor
;-----
        MOV    A,#CursorOn
        LCALL  CommandLCD
        LCALL  LDELAY
;-----
;Menggeser kursor ke kanan
;-----
        MOV    A,#CurShRight
        LCALL  CommandLCD
        LCALL  LDELAY
;-----
;Menggeser kursor ke kiri
;-----
        MOV    A,#CurShLeft
        LCALL  CommandLCD
        LCALL  LDELAY
;-----
;Meletakkan kursor pada posisi awal
;-----
        MOV    A,#CursorHome
        LCALL  CommandLCD
        LCALL  LDELAY
;-----
;Menggeser display ke kanan 1 kali
;-----
        MOV    A,#DispShRight
        LCALL  CommandLCD
        LCALL  LDELAY
;-----
;Menggeser display ke kiri 2 kali
;-----
        MOV    A,#DispShLeft
        LCALL  CommandLCD
        LCALL  LDELAY
        MOV    A,#DispShLeft
        LCALL  CommandLCD
        SJMP  $
        END

```

- Program 4.2. Program untuk menampilkan tulisan berjalan

```

$MOD51
;-----
;Procedure Address
;-----

```



```

Write                EQU    0700H
InitLCD              EQU    0740H
CommandLCD           EQU    07B0H
WriteLCD             EQU    07D0H
ReadLCD              EQU    07F0H
ReadAddrLCD         EQU    0820H
SetDDRAM             EQU    0850H
SetCGRAM            EQU    0870H
;-----
;LCD Command
;-----
DisplayClear        EQU    01H
CursorHome          EQU    02H
ShiftCursorRight   EQU    14H
ShiftCursorLeft    EQU    10H
ShiftDisplayRight   EQU    1CH
ShiftDisplayLeft    EQU    18H
DShiftCursorInc    EQU    07H
DShiftCursorDec    EQU    05H

                CSEG
                ORG    4000H
                LJMP  Start

                ORG    4100H
LDelay:          PUSH  02H
                 PUSH  03H
                 MOV   R2,#03H
Del1:            MOV   R3,#0FFH
Del2:            MOV   R4,#0FFH
                 DJNZ  R4,$
                 DJNZ  R3,Del2
                 DJNZ  R2,Del1
                 POP   03H
                 POP   02H
                 RET
Start:           MOV   SP,#40H
                 LCALL InitLCD
;-----
;Display at Line1
;-----
                MOV   A,#' '
                LCALL WriteLCD
                MOV   A,#' '
                LCALL WriteLCD
                MOV   A,#'L'
                LCALL WriteLCD
                MOV   A,#'A'
                LCALL WriteLCD
                MOV   A,#'B'
                LCALL WriteLCD
                MOV   A,#'O'
                LCALL WriteLCD
                MOV   A,#'R'

```

```

        LCALL WriteLCD
        MOV   A,#'A'
        LCALL WriteLCD
        MOV   A,#'T'
        LCALL WriteLCD
        MOV   A,#'O'
        LCALL WriteLCD
        MOV   A,#'R'
        LCALL WriteLCD
        MOV   A,#'I'
        LCALL WriteLCD
        MOV   A,#'U'
        LCALL WriteLCD
        MOV   A,#'M'
        LCALL WriteLCD
;-----
;Display at Line2
;-----
        MOV   A,#41H
        LCALL SetDDRAM
        MOV   A,#' '
        LCALL WriteLCD
        MOV   A,#'R'
        LCALL WriteLCD
        MOV   A,#'W'
        LCALL WriteLCD
        MOV   A,#'-'
        LCALL WriteLCD
        MOV   A,#'P'
        LCALL WriteLCD
        MOV   A,#'R'
        LCALL WriteLCD
        MOV   A,#'O'
        LCALL WriteLCD
        MOV   A,#'P'
        LCALL WriteLCD
        MOV   A,#'A'
        LCALL WriteLCD
        MOV   A,#'G'
        LCALL WriteLCD
        MOV   A,#'A'
        LCALL WriteLCD
        MOV   A,#'T'
        LCALL WriteLCD
        MOV   A,#'I'
        LCALL WriteLCD
        MOV   A,#'O'
        LCALL WriteLCD
        MOV   A,#'N'
        LCALL WriteLCD
;-----
;Shift display right and left
;-----
        MOV   R5,#10

```

```

EE:      MOV    R6,#16
BB:      MOV    A,#ShiftDisplayRight
          LCALL CommandLCD
          LCALL LDelay
          DJNZ  R6,BB
          MOV   R6,#16
CC:      MOV    A,#ShiftDisplayLeft
          LCALL CommandLCD
          LCALL LDelay
          DJNZ  R6,CC
          DJNZ  R5,EE

          AJMP  $

          END

```

TUGAS :

1. Buat program untuk menampilkan tulisan kelas anda di tengah-tengah baris pertama, dan jurusan anda di tengah-tengah baris kedua. Padamkan cursor, kemudian geser display ke kiri 5 kali, ke kanan 5 kali. Lalu buatlah agar tampilan berkedip.
2. Buat program untuk menampilkan tulisan nama anda di tengah-tengah baris pertama, dan alamat anda di tengah-tengah baris kedua. Geser display ke kiri 3 kali, ke kanan 5 kali dan ke kiri lagi 2 kali. Lalu buatlah agar tampilan berkedip, program akan menampilkan tulisan dengan respon seperti diatas ketika saklar input Port1 bernilai 01h.