

# SPC

SMART PERIPHERAL CONTROLLER

## SERIAL LCD

### Quick Start

#### Trademarks & Copyright

XT, AT, IBM, PC, and PC-DOS are trademarks of International Business Machines Corp.  
MS-DOS is a registered trademark of Microsoft Corporation.  
Pentium is a registered trademark of Intel Corporation.  
MetaLink ASM51 is copyright by MetaLink Corporation

#### 1. PENDAHULUAN

*Smart Peripheral Controller* / SPC SERIAL LCD merupakan Pengendali modul CHARACTER LCD yang mendukung 3 macam jalur penyampaian data yaitu I<sup>2</sup>C-bus, Synchronous Serial, atau Universal Asynchronous Receiver/Transmitter (UART). Ketiga protokol penyampaian data tersebut bekerja secara serial sehingga dapat lebih menghemat dan mempermudah pengkabelan. Contoh aplikasi dari SPC SERIAL LCD adalah untuk character monitor, display, penampil counter, dan lain-lain.

*Untuk manual dan source-source yang lebih lengkap dapat dilihat di dalam disket/CD.*

#### 2. SPESIFIKASI EKSTERNAL SPC SERIAL LCD

Spesifikasi Eksternal SPC SERIAL LCD sebagai berikut :

- Kompatibel penuh dengan DT-51 Minimum System Ver 3.0.
- Hanya perlu 2 jalur kabel untuk interface dengan mikroprosesor / mikrokontroler lain.
- Dapat digunakan pada I<sup>2</sup>C-bus , Synchronous Serial, maupun UART.
- Synchronous Serial bekerja dengan taraf logic TTL.
- UART bekerja dengan taraf logic TTL maupun RS-232.
- Dilengkapi dengan jumper untuk setting alamat, sehingga dapat di-ekspan sampai 8 board tanpa tambahan perangkat keras (baik secara I<sup>2</sup>C maupun Synchronous Serial).
- Kompatibel sampai dengan LCD Module 40 x 2 dengan driver yang kompatibel dengan HD44780.
- Tersedia prosedur siap pakai untuk aplikasi SPC SERIAL LCD.
- Varian SPC Serial LCD:
  - LCD 8 x 2 , tanpa backlight
  - LCD 16 x 2 , dengan backlight
  - LCD 20 x 2 , dengan backlight
  - LCD 20 x 2 Large , dengan backlight
  - LCD 20 x 4 , dengan backlight
  - LCD 24 x 2 , dengan backlight
  - LCD 40 x 2 , dengan backlight

#### 3. ALAMAT DDRAM VARIAN

8 x 2	16 x 2	20 x 2	24 x 2	40 x 2
00H.....07H	.....0FH	.....13H	.....17H	.....27H
40H.....47H	.....4FH	.....53H	.....57H	.....67H

20 x 4	
00H.....	13H
40H.....	53H
14H.....	27H
54H.....	67H

**4. SISTEM YANG DIANJURKAN**

Perangkat keras :

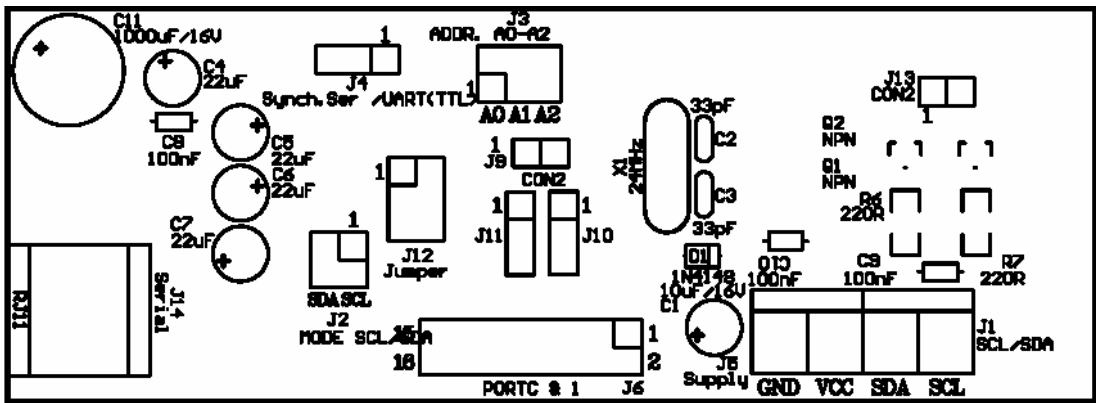
- PC XT / AT Pentium™ IBM Compatible dengan port serial (COM1 / COM2).
- Board DT-51 Minimum System.
- Floppy Disk 3.5", kapasitas 1,44 Mbytes atau CD-ROM Drive.
- Hard disk dengan kapasitas minimum 500 Kbytes.

Perangkat lunak :

- Sistem operasi MS-DOS™ atau PC-DOS™.
- Assembler ASM51©.
- File-file yang ada pada pada disket/CD program:  
EXPL\_I2C.ASM, EXPL\_I2C.HEX, EXPL\_SS.ASM, EXPL\_SS.HEX, LCDCHAR.INC,  
ENG\_I2C.INC, ENG\_SS.INC, MANUAL\_SPC\_SERIAL\_LCD.PDF, QUICK START SPC  
SERIAL\_LCD.PDF, DAN MANUAL\_LCD.

**5. PERANGKAT KERAS SPC SERIAL LCD**

**5.1. TATA LETAK KOMPONEN SPC SERIAL LCD**



**5.2. HUBUNGAN DT-51 MINIMUM SYSTEM DENGAN SPC SERIAL LCD**

SPC SERIAL LCD merupakan suatu sistem yang ‘Smart’. Selain dapat dihubungkan dengan DT-51 Minimum System atau dengan sistem mikroprosesor / mikrokontroler yang lain, SPC SERIAL LCD juga dapat dihubungkan dengan komputer. Apabila Anda ingin menghubungkan SPC SERIAL LCD dengan sistem yang lain kami sarankan untuk mempelajari skema SPC SERIAL LCD (lihat lampiran A pada Manual SPC Serial LCD). Untuk menghubungkan SPC SERIAL LCD dengan DT-51 Minimum System dianjurkan untuk menggunakan kabel pita (flat ribbon cable). Hubungan secara I<sup>2</sup>C-bus ditunjukkan pada tabel berikut :

I <sup>2</sup> C Bus	DT-51 Minimum System PORT C & PORT 1	SPC SERIAL LCD J6
SCL	Pin 15 (Port 1.6)	Pin 15
SDA	Pin 16 (Port 1.7)	Pin 16

Jumper J2 (SCL/SDA) digunakan untuk resistor pull up SDA (I<sup>2</sup>C-bus data input / output) dan SCL (I<sup>2</sup>C-bus clock input).

**Penting !**

Apabila lebih dari satu board SPC SERIAL LCD dihubungkan pada I<sup>2</sup>C-bus maka jumper J2 (SCL/SDA) salah satu board saja yang perlu dipasang.

Hubungan secara Synchronous Serial bus ditunjukkan pada tabel berikut :

Synch Serial Bus	DT-51 Minimum System PORT C & PORT 1	SPC SERIAL LCD J4
SerIn	Pin 13 (Port 1.4)	SerIn (Pin 3)
SerClk	Pin 14 (Port 1.5)	SerClk (Pin 2)

Jumper J12 digunakan untuk resistor pull up SerIn (Synchronous Serial bus data input / output), SerClk (Synchronous Serial bus clock input), dan SerClr (Synchronous Serial bus reset).

**Penting !**

Apabila lebih dari satu board SPC SERIAL LCD dihubungkan pada Synchronous Serial bus maka jumper J12 (SerIn, SerClk, dan SerClr) salah satu board saja yang perlu dipasang.

Catu daya 5V DC dihubungkan dengan konektor J5 (Power). Perhatikan polaritasnya jangan sampai terbalik, karena dapat mengakibatkan kerusakan.

**Penting !**

Referensi ground (GND) antara modul SPC SERIAL LCD dengan DT-51 Minimum System harus sama.

**5.3. HUBUNGAN KOMPUTER DENGAN SPC SERIAL LCD**

SPC SERIAL LCD menyediakan interface yang dapat langsung berhubungan dengan port serial pada komputer.

Hubungannya ditunjukkan pada tabel berikut :

Serial/COM Port	SPC SERIAL LCD J14
Pin 5 (GND)	Pin 1 (GND)
Pin 3 (TX)	Pin 4 (TX)
Pin 2 (RX)	Pin 3 (RX)

J14 Tampak Depan



**5.4. HUBUNGAN BUZZER DENGAN SPC SERIAL LCD**

SPC SERIAL LCD menyediakan konektor untuk buzzer yang dapat berbunyi jika kode ASCII 7Fh dikirimkan ke SPC SERIAL LCD. Kondisi buzzer ini bersifat toggle. Pengiriman pertama akan membunyikan buzzer sedangkan pengiriman kedua akan mematikan buzzer.

Hubungannya ditunjukkan pada tabel berikut :

Buzzer	SPC SERIAL LCD J13
Pin +	Pin 2
Pin -	Pin 1

**5.5. SETTING JUMPER UART / SYNCHRONOUS SERIAL**

Penggunaan jalur komunikasi antara Synchronous Serial dan UART tidak bisa dilakukan bersamaan. Pemilihan antara jalur komunikasi Synchronous Serial atau UART diatur dengan cara mengganti setting jumper J10-J11 dan J9.

J10-J11	J9	Jalur Komunikasi	Konektor
		Synchronous Serial	J4
		UART TTL	J4
		UART RS-232	J14

Keterangan : => jumper tersambung (ON)  
 => jumper terlepas (OFF)  
Arah posisi jumper sesuai dengan posisi pada gambar bagian 5.1

Konektor J4 dapat digunakan sebagai interface UART TTL maupun Synchronous Serial.

SPC Serial LCD Konektor J4	UART TTL	Synchronous Serial
Pin 1	<i>Tidak Terpakai</i>	SerClr
Pin 2	TX (out)	SerClk
Pin 3	RX (in)	SerIn

#### 5.6. SETTING JUMPER ALAMAT TERPROGRAM

Alamat terprogram setiap board SPC SERIAL LCD ditentukan oleh setting jumper J3.

J3 (A2)	J3 (A1)	J3(A0)	Alamat Terprogram	
			0	000
			1	001
			2	010
			3	011
			4	100
			5	101
			6	110
			7 (default)	111

Keterangan :  
 : jumper tersambung (ON)

#### 5.7. EKSPANSI SPC SERIAL LCD

SPC SERIAL LCD dapat di-ekspan sampai 8 board. Beberapa hal yang perlu diperhatikan apabila menggunakan lebih dari satu board SPC SERIAL LCD :

- Setiap board harus mempunyai alamat terprogram yang berbeda, ditentukan oleh jumper J3 (A0/A1/A2).
- Jumper J2 pada salah satu board saja yang dipasang (jika menggunakan I<sup>2</sup>C-bus).
- Jumper J12 pada salah satu board saja yang dipasang (jika menggunakan Synchronous Serial bus).

#### 5.8. MENCoba SPC SERIAL LCD DENGAN EXPL\_I2C.HEX

##### Setting Hardware

- ◆ Hubungkan DT-51 Minimum System dengan SPC SERIAL LCD (lihat **bagian 5.2**) melalui I<sup>2</sup>C-bus.
- ◆ Hubungkan port serial DT-51 MinSys dengan COM1/COM2 dari komputer dengan menggunakan kabel serial.
- ◆ Setting alamat SPC SERIAL LCD pada alamat terprogram ke-7 (default, lihat **bagian 5.6**).
- ◆ Download Expl\_I2C.HEX yang terdapat pada disket/CD.

### Proses Program Expl\_I2C

- ◆ Setelah program selesai di-*download* maka akan tampak tulisan “INNOVATIVE” pada baris 1 LCD dan “ELECTRONICS” pada baris 2. Setelah delay sejenak, layar LCD akan dibersihkan dan tampil karakter-karakter yang terdapat pada tabel character generator ROM pattern LCD secara bergantian dimulai pada baris 1 dan kolom 1 LCD.

## 5.9. MENCoba SPC SERIAL LCD DENGAN EXPL\_SS.HEX

### Setting Hardware

- ◆ Hubungkan DT-51 Minimum System dengan SPC SERIAL LCD (lihat **bagian 5.2**) melalui Synchronous Serial bus.
- ◆ Atur setting jumper sehingga jalur komunikasi yang digunakan adalah Synchronous Serial. (lihat **bagian 5.5**)
- ◆ Hubungkan port serial DT-51 MinSys dengan COM1/COM2 dari komputer dengan menggunakan kabel serial.
- ◆ Setting alamat SPC SERIAL LCD pada alamat terprogram ke-7 (default, lihat **bagian 5.6**).
- ◆ Download Expl\_SS.HEX yang terdapat pada disket/CD.

### Proses Program Expl\_SS

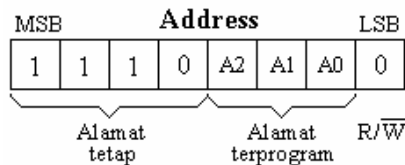
- ◆ Setelah program selesai di-*download* maka akan tampak tulisan “INNOVATIVE” pada baris 1 LCD dan “ELECTRONICS” pada baris 2. Setelah delay sejenak, layar LCD akan dibersihkan dan tampil karakter-karakter yang terdapat pada tabel character generator ROM pattern LCD secara bergantian dimulai pada baris 1 dan kolom 1 LCD.

## 6. PERANGKAT LUNAK SPC SERIAL LCD

Waktu yang dibutuhkan SPC Serial LCD mulai menyala hingga siap dioperasikan (Start-up Time) = 600 ms.

Bacalah Manual LCD yang terdapat dalam disket/CD sebagai referensi untuk tabel kode Character Generator ROM (untuk rutin WriteSingle dan WriteString) dan tabel instruksi LCD (untuk rutin CommandLCD).

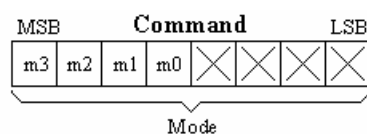
### 6.1. PENGALAMATAN UNTUK I<sup>2</sup>C-BUS



#### Pengalamatan memanfaatkan register : *AddressI2C*

Semua penggunaan dari I<sup>2</sup>C-bus selalu diawali dengan pengalamatan. Pada pengalamatan itu sendiri dibedakan menjadi tiga bagian : alamat tetap, alamat terprogram, dan Read/Write (R/ $\overline{W}$ ) pada SPC SERIAL LCD selalu menggunakan alamat tetap dengan nilai “1110”, Bagian Read/Write (R/ $\overline{W}$ ) bernilai “1” jika Master I<sup>2</sup>C (DT-51 MinSys / mikrokontroler lain) akan membaca data dari Slave I<sup>2</sup>C (SPC SERIAL LCD) dan bernilai “0”, jika Master I<sup>2</sup>C akan menulis data ke Slave I<sup>2</sup>C, sedangkan untuk alamat terprogram digunakan untuk memberikan alamat terhadap modul sesuai dengan kehendak pemakai. Alamat terprogram diatur dengan cara mengganti setting jumper (dapat dilihat pada bagian 5.6) sehingga pada jalur I<sup>2</sup>C yang sama dengan alamat tetap yang sama (“1110”) dapat digunakan 8 buah modul secara bersamaan dengan membedakan alamat terprogram.

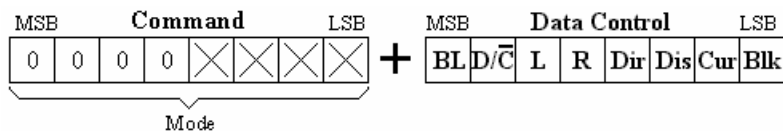
#### 6.1.1. Command



m3	m2	m1	m0	Mode
0	0	0	0	<b>Control</b>
0	0	0	1	<b>Write Single Character</b>
0	0	1	0	<b>Write Multiple Character</b>
0	0	1	1	<b>Locate DDRAM</b>
0	1	0	0	<b>Locate CGRAM</b>
0	1	0	1	<b>Set Contrast</b>
0	1	1	0	<b>Read Data DDRAM</b>
0	1	1	1	<b>Read Data CGRAM</b>
1	0	0	0	<b>Read Address DDRAM</b>
1	0	0	1	<b>Read Address CGRAM</b>
1	0	1	0	<b>Command LCD</b>
1	0	1	1	<i>Tidak Terpakai</i>
1	1	X	X	<i>Tidak Terpakai</i>

Pada perintah command terdapat bagian utama yaitu Mode. Mode digunakan untuk memilih perintah selanjutnya yang akan diberikan pada device sesuai dengan pilihan mode yang diberikan.

### 6.1.2. Control



Mode Control memanfaatkan register : **OutControl**

Control digunakan untuk mengatur tampilan LCD. Setelah mengirimkan command control diikuti dengan data control yang mengandung bit untuk mengatur Back Light (**BL**), Display/Cursor Shift (**D/C**), Shift Left (**L**), Shift Right (**R**), Arah gerak cursor (**Dir**), Display On/Off (**Dis**), Cursor On/Off (**Cur**), dan Cursor Blink (**Blk**).

**BL** = 1 : Back Light On  
**BL** = 0 : Back Light Off

**Dir** = 1 : Increment  
**Dir** = 0 : Decrement

**D/C** = 1 : Display Shift  
**D/C** = 0 : Cursor Shift

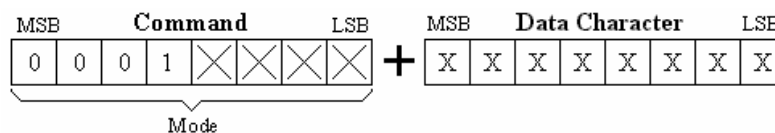
**Dis** = 1 : Display On  
**Dis** = 0 : Display Off

**L R** = 00 : No Shift  
**L R** = 01 : Shift Right  
**L R** = 10 : Shift Left  
**L R** = 11 : No Shift

**Cur** = 1 : Cursor On  
**Cur** = 0 : Cursor Off

**Blk** = 1 : Blink On  
**Blk** = 0 : Blink Off

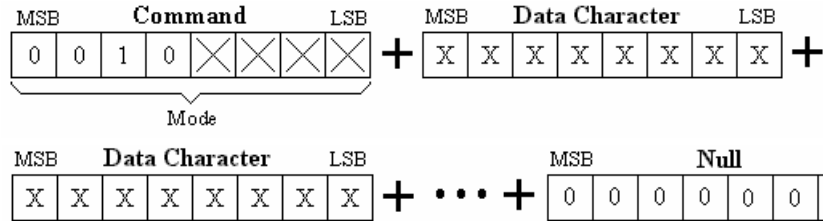
### 6.1.3. Write Single Character



Mode Write Single Character memanfaatkan register : **OutCharacter**

Write Single Character digunakan untuk menulis sebuah character ke LCD.

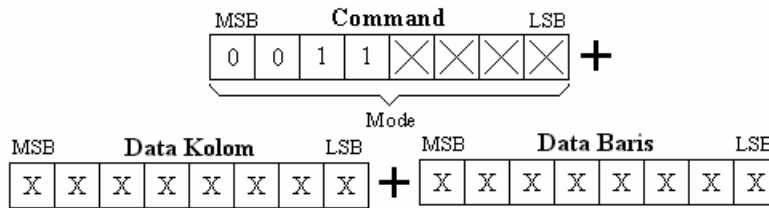
#### 6.1.4. Write Multiple Character



Mode Write Multiple Char. memanfaatkan register : **Data Pointer (DPTR)**

Write Multiple Character digunakan untuk menulis string atau rangkaian character ke LCD. Mode Multiple Character diakhiri dengan mengirimkan data Null (“00h”) ke slave.

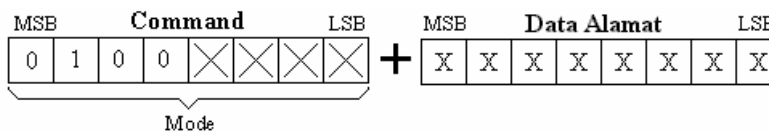
#### 6.1.5. Locate DDRAM



Mode Locate DDRAM memanfaatkan register : **OutKolom** dan **OutBaris**

Locate DDRAM digunakan untuk mengatur posisi cursor pada DDRAM atau layar LCD.

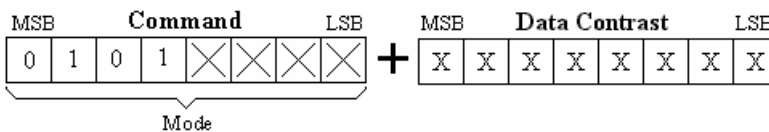
#### 6.1.6. Locate CGRAM



Mode Locate CGRAM memanfaatkan register : **OutCursorCGRAM**

Locate CGRAM digunakan untuk mengatur posisi cursor pada CGRAM.

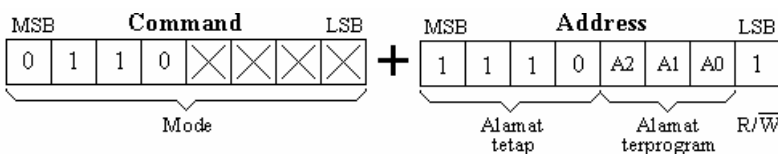
#### 6.1.7. Set Contrast



Mode Set Contrast memanfaatkan register : **OutContrast**

Set Contrast digunakan untuk mengatur kontras LCD. Kisaran data contrast bernilai 0 – 20 desimal (00H – 14H). Makin besar nilainya, warna karakter makin pudar.

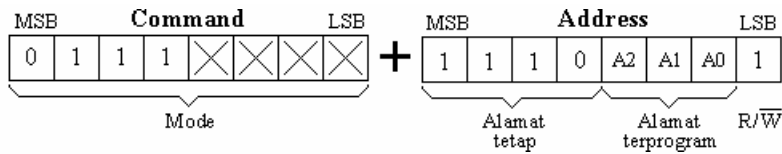
#### 6.1.8. Read Data DDRAM



Mode Read Data DDRAM memanfaatkan register : **InDataDDRAM**

Read Data DDRAM digunakan untuk membaca data DDRAM yang ditunjuk oleh cursor. Setelah command ini, posisi cursor akan berada pada DDRAM.

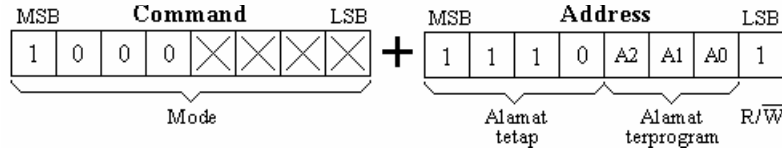
### 6.1.9. Read Data CGRAM



Mode Read Data CGRAM memanfaatkan register : *InDataCGRAM*

Read Data CGRAM digunakan untuk membaca data CGRAM yang ditunjuk oleh cursor. Setelah command ini, posisi cursor akan berada pada CGRAM.

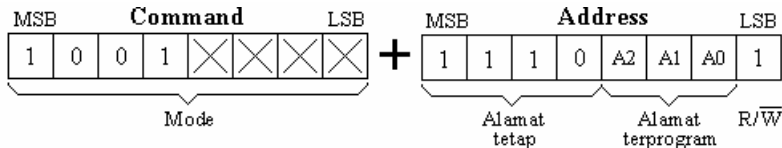
### 6.1.10. Read Address DDRAM



Mode Read Address DDRAM memanfaatkan register : *InAddrDDRAM*

Read Address DDRAM digunakan untuk membaca data alamat cursor DDRAM sekarang. Command ini tidak mempengaruhi posisi cursor.

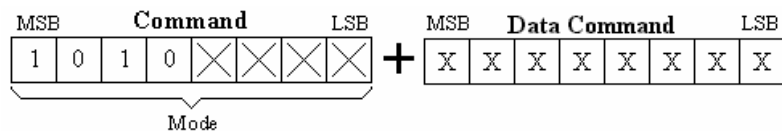
### 6.1.11. Read Address CGRAM



Mode Read Address CGRAM memanfaatkan register : *InAddrCGRAM*

Read Address CGRAM digunakan untuk membaca data alamat cursor CGRAM sekarang. Command ini tidak akan mempengaruhi posisi cursor.

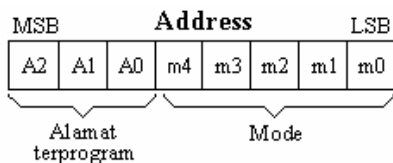
### 6.1.12. Command LCD



Mode Command LCD memanfaatkan register : *OutCommand*

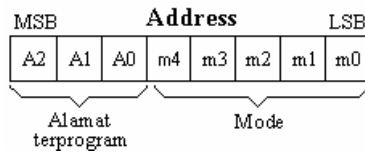
Command LCD digunakan mengirimkan instruksi ke modul LCD sesuai dengan tabel instruksi pada manual LCD.

## 6.2. PENGALAMATAN UNTUK SYNCHRONOUS SERIAL-BUS



Semua penggunaan dari Synchronous Serial selalu diawali dengan pengalamatan. Pada pengalamatan itu sendiri dibedakan menjadi dua bagian : alamat terprogram dan mode kerja atau command. Alamat terprogram digunakan untuk memberikan alamat terhadap modul sesuai dengan kehendak pemakai. Alamat terprogram diatur dengan cara mengganti setting jumper (dapat dilihat pada bagian 5.6) sehingga pada jalur Synchronous Serial yang sama dapat digunakan 8 buah modul secara bersamaan dengan membedakan alamat terprogram.

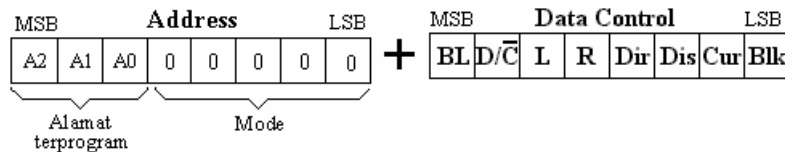
### 6.2.1. Command



m4	m3	m2	m1	m0	Mode
0	0	0	0	0	<b>Control</b>
0	0	0	0	1	<b>Write Single Character</b>
0	0	0	1	0	<b>Write Multiple Character</b>
0	0	0	1	1	<b>Locate DDRAM</b>
0	0	1	0	0	<b>Locate CGRAM</b>
0	0	1	0	1	<b>Set Contrast</b>
0	0	1	1	0	<b>Read Data DDRAM</b>
0	0	1	1	1	<b>Read Data CGRAM</b>
0	1	0	0	0	<b>Read Address DDRAM</b>
0	1	0	0	1	<b>Read Address CGRAM</b>
0	1	0	1	0	<b>Command LCD</b>
0	1	0	1	1	<i>Tidak Terpakai</i>
0	1	1	X	X	<i>Tidak Terpakai</i>
1	X	X	X	X	<i>Tidak Terpakai</i>

Pada pengalamatan terdapat bagian Mode. Mode digunakan untuk memilih perintah selanjutnya yang akan diberikan pada device sesuai dengan pilihan mode yang diberikan.

### 6.2.2. Control



Mode Control memanfaatkan register : *OutControl*

Control digunakan untuk mengatur tampilan LCD. Setelah mengirimkan command control diikuti dengan data control yang mengandung bit untuk mengatur Back Light (**BL**), Display/Cursor Shift (**D/C**), Shift Left (**L**), Shift Right (**R**), Arah gerak cursor (**Dir**), Display On/Off (**Dis**), Cursor On/Off (**Cur**), dan Cursor Blink (**Blk**).

**BL** = 1 : Back Light On  
**BL** = 0 : Back Light Off

**Dir** = 1 : Increment  
**Dir** = 0 : Decrement

**D/C** = 1 : Display Shift  
**D/C** = 0 : Cursor Shift

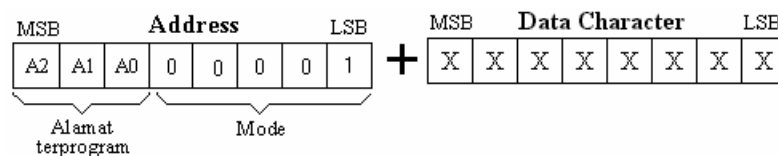
**Dis** = 1 : Display On  
**Dis** = 0 : Display Off

**L R** = 00 : No Shift  
**L R** = 01 : Shift Right  
**L R** = 10 : Shift Left  
**L R** = 11 : No Shift

**Cur** = 1 : Cursor On  
**Cur** = 0 : Cursor Off

**Blk** = 1 : Blink On  
**Blk** = 0 : Blink Off

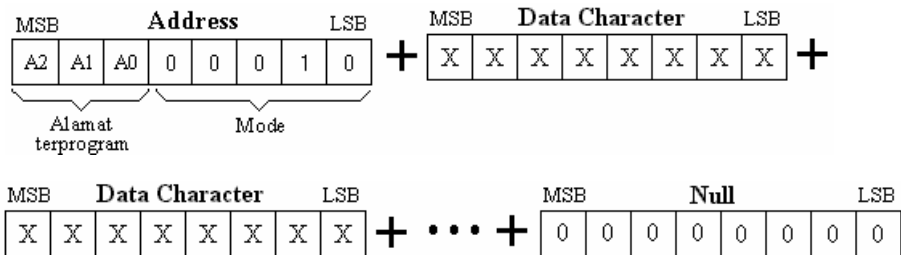
### 6.2.3. Write Single Character



Mode Write Single Character memanfaatkan register : *OutCharacter*

Write Single Character digunakan untuk menulis sebuah character ke LCD.

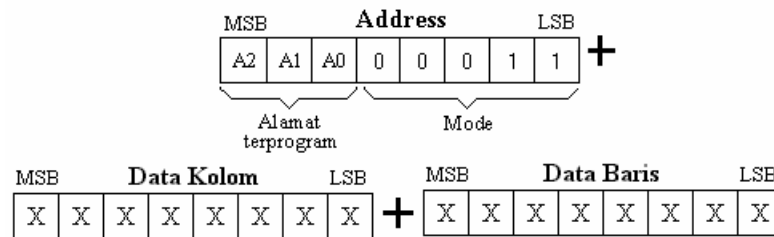
#### 6.2.4. Write Multiple Character



Mode Write Multiple Char. memanfaatkan register : *Data Pointer (DPTR)*

Write Multiple Character digunakan untuk menulis string atau rangkaian character ke LCD. Mode Multiple Character diakhiri dengan mengirimkan data Null (“00h”) ke slave.

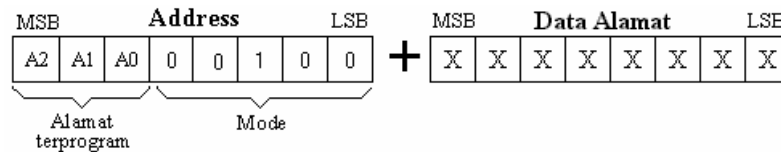
#### 6.2.5. Locate DDRAM



Mode Locate DDRAM memanfaatkan register : *OutKolom* dan *OutBaris*

Locate DDRAM digunakan untuk mengatur posisi cursor pada DDRAM atau layar LCD.

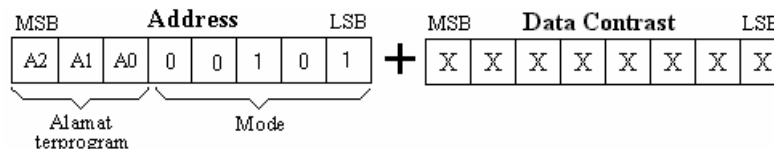
#### 6.2.6. Locate CGRAM



Mode Locate CGRAM memanfaatkan register : *OutCursorCGRAM*

Locate CGRAM digunakan untuk mengatur posisi cursor pada CGRAM.

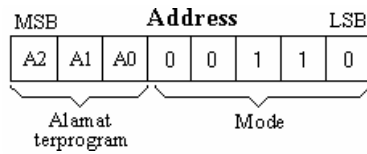
#### 6.2.7. Set Contrast



Mode Set Contrast memanfaatkan register : *OutContrast*

Set Contrast digunakan untuk mengatur kontras LCD. Kisaran data contrast bernilai 0 – 20 desimal (00H – 14H). Makin besar nilainya, warna karakter makin pudar.

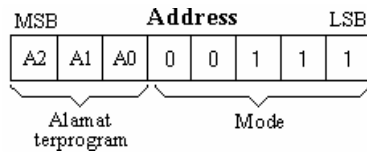
### 6.2.8. Read Data DDRAM



Mode Read Data DDRAM memanfaatkan register : *InDataDDRAM*

Read Data DDRAM digunakan untuk membaca data DDRAM yang ditunjuk oleh cursor. Setelah command ini, posisi cursor akan berada pada DDRAM.

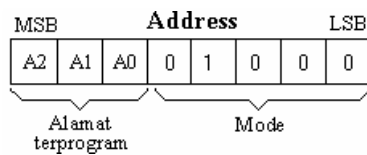
### 6.2.9. Read Data CGRAM



Mode Read Data CGRAM memanfaatkan register : *InDataCGRAM*

Read Data CGRAM digunakan untuk membaca data CGRAM yang ditunjuk oleh cursor. Setelah command ini, posisi cursor akan berada pada CGRAM.

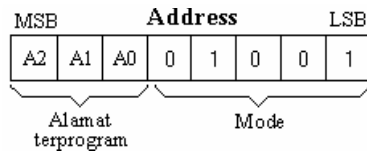
### 6.2.10. Read Address DDRAM



Mode Read Address DDRAM memanfaatkan register : *InAddrDDRAM*

Read Address DDRAM digunakan untuk membaca data alamat cursor DDRAM sekarang. Command ini tidak mempengaruhi posisi cursor.

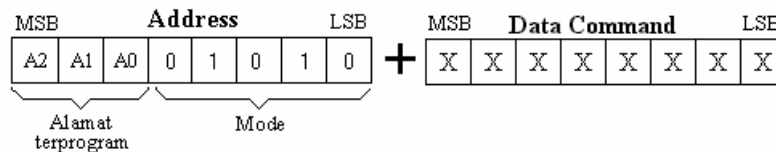
### 6.2.11. Read Address CGRAM



Mode Read Address CGRAM memanfaatkan register : *InAddrCGRAM*

Read Address CGRAM digunakan untuk membaca data alamat cursor CGRAM sekarang. Command ini tidak akan mempengaruhi posisi cursor.

### 6.2.12. Command LCD



Mode Command LCD memanfaatkan register : *OutCommand*

Command LCD digunakan mengirimkan instruksi ke modul LCD sesuai dengan tabel instruksi pada manual LCD.

### 6.3. SPESIFIKASI UART

Jalur komunikasi UART bekerja pada **Baud Rate 9600 bps, 8 Data Bit, No Parity Bit, 1 Stop Bit, No Flow Control**. Pada saat jalur komunikasi UART yang aktif maka SPC SERIAL LCD berfungsi sebagai monitor data yang masuk melalui jalur UART. Setiap data yang masuk akan ditampilkan pada layar LCD sesuai dengan tabel tabel character generator ROM pattern pada manual LCD, kecuali character pembuka command (dapat dilihat pada bagian 6.3.1) dan command.

SPC SERIAL LCD dengan jalur komunikasi UART juga mendukung perintah-perintah khusus misalnya: clear screen (Ctrl-L) atau "0Ch", pindah baris (Enter) atau "0Dh".

#### 6.3.1. Command

Semua Command pada komunikasi UART selalu diawali dengan mengirimkan data byte bernilai "FEh" yang kemudian diikuti dengan mengirimkan mode command yang diinginkan. Waktu yang dibutuhkan SPC SERIAL LCD setelah proses WriteSingle & Read hingga siap menerima perintah berikutnya = 250  $\mu$ s.

Waktu yang dibutuhkan SPC SERIAL LCD setelah perintah CommandLCD hingga siap menerima perintah berikutnya = 500  $\mu$ s.

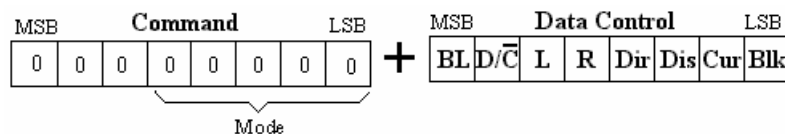
#### 6.3.2. Mode Command

MSEB	<b>Command</b>					LSB	
0	0	0	m4	m3	m2	m1	m0
<div style="border-top: 1px solid black; width: 100%; margin: 0 auto; padding-top: 5px;"> <span style="float: left;">Mode</span> </div>							

m4	m3	m2	m1	m0	Mode
0	0	0	0	0	<b>Control</b>
0	0	0	0	1	<i>Tidak Terpakai</i>
0	0	0	1	0	<b>Write Multiple Character</b>
0	0	0	1	1	<b>Locate DDRAM</b>
0	0	1	0	0	<b>Locate CGRAM</b>
0	0	1	0	1	<b>Set Contrast</b>
0	0	1	1	0	<b>Read Data DDRAM</b>
0	0	1	1	1	<b>Read Data CGRAM</b>
0	1	0	0	0	<b>Read Address DDRAM</b>
0	1	0	0	1	<b>Read Address CGRAM</b>
0	1	0	1	0	<b>Command LCD</b>
0	1	0	1	1	<i>Tidak Terpakai</i>
0	1	1	X	X	<i>Tidak Terpakai</i>
1	X	X	X	X	<i>Tidak Terpakai</i>

Pada pengalamatan terdapat bagian Mode. Mode digunakan untuk memilih perintah selanjutnya yang akan diberikan pada device sesuai dengan pilihan mode yang diberikan.

#### 6.3.3. Control



Control digunakan untuk mengatur tampilan LCD. Setelah mengirimkan command control diikuti dengan data control yang mengandung bit untuk mengatur Back Light (BL), Display/Cursor Shift ( $\overline{D/C}$ ), Shift Left (L), Shift Right (R), Arah gerak cursor (Dir), Display On/Off (Dis), Cursor On/Off (Cur), dan Cursor Blink (Blk).

BL = 1 : Back Light On  
BL = 0 : Back Light Off

Dir = 1 : Increment  
Dir = 0 : Decrement

$\overline{D/C}$  = 1 : Display Shift

Dis = 1 : Display On

$\overline{D/C} = 0$  : Cursor Shift

**Dis** = 0 : Display Off

**L R** = 00 : No Shift

**Cur** = 1 : Cursor On

**L R** = 01 : Shift Right

**Cur** = 0 : Cursor Off

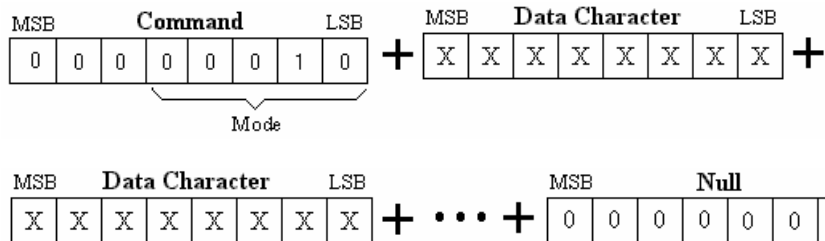
**L R** = 10 : Shift Left

**L R** = 11 : No Shift

**Blk** = 1 : Blink On

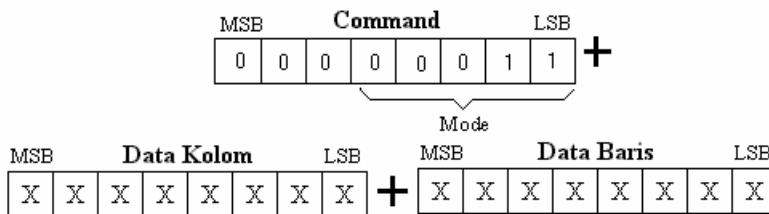
**Blk** = 0 : Blink Off

#### 6.3.4. Write Multiple Character



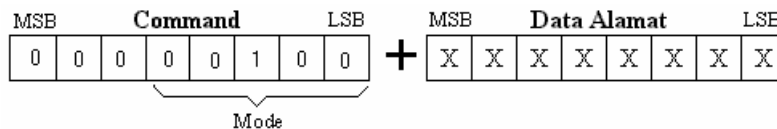
Write Multiple Character digunakan untuk menulis string atau rangkaian character ke LCD. Mode Multiple Character diakhiri dengan mengirimkan data Null ("00h") ke slave.

#### 6.3.5. Locate DDRAM



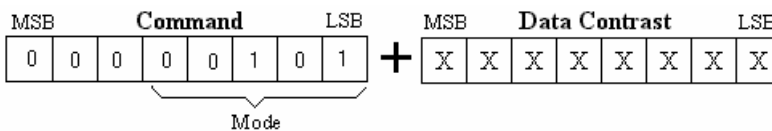
Locate DDRAM digunakan untuk mengatur posisi cursor pada DDRAM atau layar LCD.

#### 6.3.6. Locate CGRAM



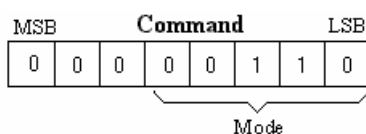
Locate CGRAM digunakan untuk mengatur posisi cursor pada CGRAM.

#### 6.3.7. Set Contrast



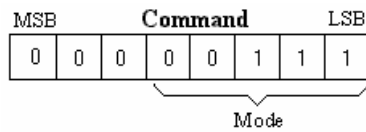
Set Contrast digunakan untuk mengatur kontras LCD. Kisaran data contrast bernilai 0 – 20 desimal (00H – 14H). Makin besar nilainya, warna karakter makin pudar.

#### 6.3.8. Read Data DDRAM



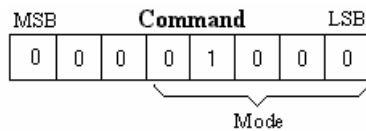
Read Data DDRAM digunakan untuk membaca data DDRAM yang ditunjuk oleh cursor. Setelah command ini, posisi cursor akan berada pada DDRAM.

### 6.3.9. Read Data CGRAM



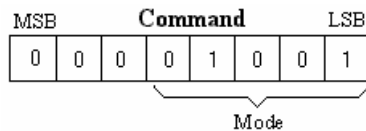
Read Data CGRAM digunakan untuk membaca data CGRAM yang ditunjuk oleh cursor. Setelah command ini, posisi cursor akan berada pada CGRAM.

### 6.3.10. Read Address DDRAM



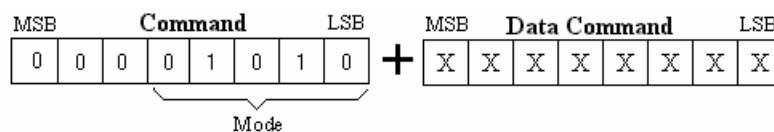
Read Address DDRAM digunakan untuk membaca data alamat cursor DDRAM sekarang. Command ini tidak mempengaruhi posisi cursor.

### 6.3.11. Read Address CGRAM



Read Address CGRAM digunakan untuk membaca data alamat cursor CGRAM sekarang. Command ini tidak mempengaruhi posisi cursor.

### 6.3.12. Command LCD



Command LCD digunakan mengirimkan instruksi ke modul LCD sesuai dengan tabel instruksi pada manual LCD.

## 6.4. DRIVER DAN RUTIN

SPC SERIAL LCD dilengkapi dengan modul **ENG\_I2C.INC**, **ENG\_SS.INC**, dan **LCDChar.INC** yang akan mempermudah user dalam pemrograman. SPC SERIAL LCD menggunakan resource dari mikrokontroler 89C51 sebagai berikut :

- Internal RAM dengan alamat 2Eh – 3Fh
- P1.6 dan P1.7
- P1.4 dan P1.5

Sehingga tidak boleh dipakai oleh user untuk keperluan lain, kecuali user mampu melakukan modifikasi pengaturan memori dengan benar.

Berikut adalah rutin-rutin yang digunakan :

#### **K10 Control**

- Fungsi : mengirimkan data control untuk mengatur tampilan LCD pada SPC SERIAL LCD melalui I<sup>2</sup>C-bus.
- Input : AddressI2C, OutControl (berisi data control)
- Output : -
- Keterangan :
- ❖ Pada saat instruksi I<sup>2</sup>C sedang dieksekusi maka semua instruksi lainnya tidak dapat dilaksanakan.
- Metode : Berikan nilai data control pada Register OutControl sesuai dengan kebutuhan, kemudian panggil sub rutin K10\_Control untuk mengirimkan data secara I<sup>2</sup>C.

#### **K10\_WriteSingle**

Fungsi : menulis sebuah character pada posisi cursor sekarang pada SPC SERIAL LCD melalui I<sup>2</sup>C-bus.

Input : AddressI2C, OutCharacter (berisi data character)

Output : -

Keterangan :  
❖ Pada saat instruksi I<sup>2</sup>C sedang dieksekusi maka semua instruksi lainnya tidak dapat dilaksanakan.

Metode : Pindahkan cursor pada lokasi yang diinginkan dengan command LocateDDRAM atau LocateCGRAM.  
Berikan nilai data character pada Register OutCharacter sesuai dengan sesuai tabel character generator ROM pattern pada manual LCD, kemudian panggil sub rutin K10\_WriteSingle untuk mengirimkan data secara I<sup>2</sup>C.

#### **K10\_WriteString**

Fungsi : menulis suatu rangkaian character / string pada SPC SERIAL LCD melalui I<sup>2</sup>C-bus.

Input : AddressI2C, DPTR (berisi pointer pada awal data string)

Output : -

Keterangan :  
❖ Data String harus diakhiri dengan data Null (00H).  
❖ Pada saat instruksi I<sup>2</sup>C sedang dieksekusi maka semua instruksi lainnya tidak dapat dilaksanakan.

Metode : Pindahkan cursor pada lokasi yang diinginkan dengan command LocateDDRAM atau LocateCGRAM.  
Berikan nilai pointer awal data string pada Data Pointer sesuai dengan kebutuhan, kemudian panggil sub rutin K10\_WriteString untuk mengirimkan data secara I<sup>2</sup>C.

#### **K10\_LocateDDRAM**

Fungsi : digunakan untuk mengatur posisi cursor pada DDRAM atau layar display LCD pada SPC SERIAL LCD melalui I<sup>2</sup>C-bus.

Input : AddressI2C, OutKolom (berisi data kolom) & OutBaris (berisi data baris)

Output : -

Keterangan :  
❖ Pada saat instruksi I<sup>2</sup>C sedang dieksekusi maka semua instruksi lainnya tidak dapat dilaksanakan.  
❖ Nilai OutBaris yang valid = 01H atau 02H.  
❖ Nilai OutKolom yang valid = 00H hingga 27H.

Metode : Berikan nilai kolom pada Register OutKolom dan baris pada Register OutBaris sesuai dengan kebutuhan, kemudian panggil sub rutin K10\_LocateDDRAM untuk mengirimkan data secara I<sup>2</sup>C.

#### **K10\_LocateCGRAM**

Fungsi : digunakan untuk mengatur posisi cursor pada CGRAM pada SPC SERIAL LCD melalui I<sup>2</sup>C-bus.

Input : AddressI2C, OutCursorCGRAM (berisi data posisi CGRAM)

Output : -

Keterangan :  
❖ Pada saat instruksi I<sup>2</sup>C sedang dieksekusi maka semua instruksi lainnya tidak dapat dilaksanakan.

Metode : Berikan nilai posisi pada Register OutCursorCGRAM sesuai dengan kebutuhan, kemudian panggil sub rutin K10\_LocateCGRAM untuk mengirimkan data secara I<sup>2</sup>C.

#### **K10\_SetContrast**

Fungsi : mengirimkan data contrast untuk mengatur ketajaman tampilan LCD pada SPC SERIAL LCD melalui I<sup>2</sup>C-bus.

Input : AddressI2C, OutContrast (berisi data contrast)

Output : -  
 Keterangan :  
 ❖ Pada saat instruksi I<sup>2</sup>C sedang dieksekusi maka semua instruksi lainnya tidak dapat dilaksanakan.  
 ❖ Nilai OutContrast yang valid = 00H – 14H.  
 Metode : Berikan nilai data contrast pada Register OutContrast sesuai dengan kebutuhan, kemudian panggil sub rutin K10\_SetContrast untuk mengirimkan data secara I<sup>2</sup>C.

#### **K10\_ReadDDRAMData**

Fungsi : digunakan untuk membaca data DDRAM pada posisi cursor terakhir pada SPC SERIAL LCD melalui I<sup>2</sup>C-bus.  
 Input : AddressI2C  
 Output : InDataDDRAM (berisi data DDRAM)  
 Keterangan :  
 ❖ Pada saat instruksi I<sup>2</sup>C sedang dieksekusi maka semua instruksi lainnya tidak dapat dilaksanakan.  
 Metode : Panggil sub rutin K10\_ReadDDRAMData, data DDRAM pada posisi cursor terakhir akan tersimpan pada Register InDataDDRAM. Setelah command ini, posisi cursor akan berada pada DDRAM.

#### **K10\_ReadCGRAMData**

Fungsi : digunakan untuk membaca data CGRAM pada posisi cursor terakhir pada SPC SERIAL LCD melalui I<sup>2</sup>C-bus.  
 Input : AddressI2C  
 Output : InDataCGRAM (berisi data CGRAM)  
 Keterangan :  
 ❖ Pada saat instruksi I<sup>2</sup>C sedang dieksekusi maka semua instruksi lainnya tidak dapat dilaksanakan.  
 Metode : Panggil sub rutin K10\_ReadCGRAMData, data CGRAM pada posisi cursor terakhir akan tersimpan pada Register InDataCGRAM. Setelah command ini, posisi cursor akan berada pada CGRAM.

#### **K10\_ReadDDRAMAddr**

Fungsi : digunakan untuk membaca posisi cursor terakhir di DDRAM pada SPC SERIAL LCD melalui I<sup>2</sup>C-bus.  
 Input : AddressI2C  
 Output : InAddrDDRAM (berisi data alamat DDRAM)  
 Keterangan :  
 ❖ Pada saat instruksi I<sup>2</sup>C sedang dieksekusi maka semua instruksi lainnya tidak dapat dilaksanakan.  
 Metode : Panggil sub rutin K10\_ReadDDRAMAddr, data posisi cursor DDRAM terakhir akan tersimpan pada Register InAddrDDRAM. Command ini tidak akan mempengaruhi posisi cursor.

#### **K10\_ReadCGRAMAddr**

Fungsi : digunakan untuk membaca posisi cursor sekarang di CGRAM pada SPC SERIAL LCD melalui I<sup>2</sup>C-bus.  
 Input : AddressI2C  
 Output : InAddrCGRAM (berisi data alamat CGRAM)  
 Keterangan :  
 ❖ Pada saat instruksi I<sup>2</sup>C sedang dieksekusi maka semua instruksi lainnya tidak dapat dilaksanakan.  
 Metode : Panggil sub rutin K10\_ReadCGRAMAddr, data posisi cursor CGRAM sekarang akan tersimpan pada Register InAddrCGRAM. Command ini tidak akan mempengaruhi posisi cursor.

#### **K10\_CommandLCD**

Fungsi : mengirimkan instruksi ke modul LCD sesuai dengan tabel instruksi manual LCD pada SPC SERIAL LCD melalui I<sup>2</sup>C-bus.

Input : AddressI2C, OutCommand (berisi data command LCD)  
 Output : -  
 Keterangan :  
 ❖ Pada saat instruksi I<sup>2</sup>C sedang dieksekusi maka semua instruksi lainnya tidak dapat dilaksanakan.  
 Metode : Berikan nilai data command sesuai dengan tabel instruksi manual LCD sesuai dengan kebutuhan pada Register OutCommand kemudian panggil sub rutin K10\_CommandLCD untuk mengirimkan data secara I<sup>2</sup>C.

#### **K10\_Control\_SS**

Fungsi : mengirimkan data control untuk mengatur tampilan LCD pada SPC SERIAL LCD melalui SYNCHRONOUS SERIAL-bus.  
 Input : AddrSS, OutControl (berisi data control)  
 Output : -  
 Keterangan : -  
 Metode : Berikan nilai data control pada Register OutControl sesuai dengan kebutuhan, kemudian panggil sub rutin K10\_Control\_SS untuk mengirimkan data secara SYNCHRONOUS SERIAL.

#### **K10\_WriteSingle\_SS**

Fungsi : menulis sebuah character pada posisi cursor sekarang pada SPC SERIAL LCD melalui SYNCHRONOUS SERIAL-bus.  
 Input : AddrSS, OutCharacter (berisi data character)  
 Output : -  
 Keterangan : -  
 Metode : Pindahkan cursor pada lokasi yang diinginkan dengan command LocateDDRAM atau LocateCGRAM.  
 Berikan nilai data character pada Register OutCharacter sesuai dengan sesuai tabel character generator ROM pattern pada manual LCD, kemudian panggil sub rutin K10\_WriteSingle\_SS untuk mengirimkan data secara SYNCHRONOUS SERIAL.

#### **K10\_WriteString\_SS**

Fungsi : menulis suatu rangkaian character / string pada SPC SERIAL LCD melalui SYNCHRONOUS SERIAL-bus.  
 Input : AddrSS, DPTR (berisi pointer pada awal data string)  
 Output : -  
 Keterangan :  
 ❖ Data String harus diakhiri dengan data Null (00H).  
 Metode : Pindahkan cursor pada lokasi yang diinginkan dengan command LocateDDRAM atau LocateCGRAM.  
 Berikan nilai pointer awal data string pada Data Pointer sesuai dengan kebutuhan, kemudian panggil sub rutin K10\_WriteString\_SS untuk mengirimkan data secara SYNCHRONOUS SERIAL.

#### **K10\_LocateDDRAM\_SS**

Fungsi : digunakan untuk mengatur posisi cursor pada DDRAM atau layar display LCD pada SPC SERIAL LCD melalui SYNCHRONOUS SERIAL-bus.  
 Input : AddrSS, OutKolom (berisi data kolom) & OutBaris (berisi data baris)  
 Output : -  
 Keterangan :  
 ❖ Nilai OutBaris yang valid = 01H atau 02H.  
 ❖ Nilai OutKolom yang valid = 00H hingga 27H.  
 Metode : Berikan nilai kolom pada Register OutKolom dan baris pada Register OutBaris sesuai dengan kebutuhan, kemudian panggil sub rutin K10\_LocateDDRAM\_SS untuk mengirimkan data secara SYNCHRONOUS SERIAL.

#### **K10 LocateCGRAM SS**

Fungsi : digunakan untuk mengatur posisi cursor pada CGRAM pada SPC SERIAL LCD melalui SYNCHRONOUS SERIAL-bus.  
Input : AddrSS, OutCursorCGRAM (berisi data posisi CGRAM)  
Output : -  
Keterangan : -  
Metode : Berikan nilai posisi pada Register OutCursorCGRAM sesuai dengan kebutuhan, kemudian panggil sub rutin K10\_LocateCGRAM\_SS untuk mengirimkan data secara SYNCHRONOUS SERIAL.

#### **K10 SetContrast SS**

Fungsi : mengirimkan data contrast untuk mengatur ketajaman tampilan LCD pada SPC SERIAL LCD melalui SYNCHRONOUS SERIAL-bus.  
Input : AddrSS, OutContrast (berisi data contrast)  
Output : -  
Keterangan :  
❖ Nilai OutContrast yang valid = 00H – 14H.  
Metode : Berikan nilai data contrast pada Register OutContrast sesuai dengan kebutuhan, kemudian panggil sub rutin K10\_SetContrast\_SS untuk mengirimkan data secara SYNCHRONOUS SERIAL.

#### **K10 ReadDDRAMData SS**

Fungsi : digunakan untuk membaca data DDRAM pada posisi cursor terakhir pada SPC SERIAL LCD melalui SYNCHRONOUS SERIAL-bus.  
Input : AddrSS  
Output : InDataDDRAM (berisi data DDRAM)  
Keterangan : -  
Metode : Panggil sub rutin K10\_ReadDDRAMData\_SS, data DDRAM pada posisi cursor terakhir akan tersimpan pada Register InDataDDRAM. Setelah command ini, posisi cursor akan berada pada CGRAM.

#### **K10 ReadCGRAMData SS**

Fungsi : digunakan untuk membaca data CGRAM pada posisi cursor terakhir pada SPC SERIAL LCD melalui SYNCHRONOUS SERIAL-bus.  
Input : AddrSS  
Output : InDataCGRAM (berisi data CGRAM)  
Keterangan : -  
Metode : Panggil sub rutin K10\_ReadCGRAMData\_SS, data CGRAM pada posisi cursor terakhir akan tersimpan pada Register InDataCGRAM. Setelah command ini, posisi cursor akan berada pada CGRAM.

#### **K10 ReadDDRAMAddr SS**

Fungsi : digunakan untuk membaca posisi cursor terakhir di DDRAM pada SPC SERIAL LCD melalui SYNCHRONOUS SERIAL-bus.  
Input : AddrSS  
Output : InAddrDDRAM (berisi data alamat DDRAM)  
Keterangan : -  
Metode : Panggil sub rutin K10\_ReadDDRAMAddr\_SS, data posisi cursor DDRAM terakhir akan tersimpan pada Register InAddrDDRAM. Command ini tidak akan mempengaruhi posisi cursor.

#### **K10 ReadCGRAMAddr SS**

Fungsi : digunakan untuk membaca posisi cursor terakhir di CGRAM pada SPC SERIAL LCD melalui SYNCHRONOUS SERIAL-bus.  
Input : AddrSS  
Output : InAddrCGRAM (berisi data alamat CGRAM)  
Keterangan : -  
Metode : Panggil sub rutin K10\_ReadCGRAMAddr\_SS, data posisi cursor CGRAM terakhir akan tersimpan pada Register InAddrCGRAM. Command ini tidak akan mempengaruhi posisi cursor.

### **K10\_CommandLCD\_SS**

Fungsi : mengirimkan instruksi ke modul LCD sesuai dengan tabel instruksi manual LCD pada SPC SERIAL LCD melalui SYNCHRONOUS SERIAL-bus.  
Input : AddrSS, OutCommand (berisi data command LCD)  
Output : -  
Keterangan : -  
Metode : Berikan nilai data command sesuai dengan tabel instruksi manual LCD sesuai dengan kebutuhan pada Register OutCommand kemudian panggil sub rutin K10\_CommandLCD\_SS untuk mengirimkan data secara SYNCHRONOUS SERIAL.

## **6.5. CONTOH APLIKASI DAN PROGRAM**

Apabila ingin menampilkan tulisan “HALLO” yang memiliki label “String1” mulai pada posisi layar LCD baris ke-2 kolom ke-5 dengan I<sup>2</sup>C-bus atau Synchronous Serial-bus, maka register-register yang harus diisi dan listing programnya adalah:

### **Cuplikan Listing program dengan I<sup>2</sup>C-bus:**

```
MOV   AddresssI2C,#11101110B ;untuk nilai alamat
                                ;terprogram '111'
MOV   A,#2                      ;memasukan nilai baris
MOV   B,#4                      ;memasukan nilai kolom
ACALL K10_LocateDDRAM           ;memanggil rutin LocateDDRAM
MOV   DPTR,#String1            ;memasukan pointer label String1
                                ;ke Data Pointer
ACALL K10_WriteString          ;memanggil rutin WriteString
.
.

String1 db 'HALLO',0
```

### **Cuplikan Listing program dengan Synchronous Serial-bus:**

```
MOV   AddrSS,#11100000B        ;untuk nilai alamat
                                ;terprogram'111'
MOV   A,#2                      ;memasukan nilai baris
MOV   B,#4                      ;memasukan nilai kolom
ACALL K10_LocateDDRAM_SS       ;memanggil rutin LocateDDRAM
MOV   DPTR,#String1            ;memasukan pointer label String1
                                ;ke Data Pointer
ACALL K10_WriteString_SS       ;memanggil rutin WriteString
.
.

String1 db 'HALLO',0
```

## **6.6. KERANGKA PROGRAM**

Bagi user yang ingin membuat program aplikasi SPC SERIAL LCD dengan menggunakan rutin yang sudah ada maka 3 driver berikut harus dimasukkan (include) : *ENG\_I2C.INC*, *ENG\_SS.INC* dan *LCDChar.INC*

*ENG\_I2C.INC* merupakan driver yang akan selalu digunakan untuk setiap aplikasi Smart Peripheral Controller (SPC) yang menggunakan I<sup>2</sup>C-bus.

*ENG\_SS.INC* merupakan driver yang akan selalu digunakan untuk setiap aplikasi Smart Peripheral Controller (SPC) yang menggunakan Synchronous Serial Bus.

*LCDChar.INC* merupakan driver yang akan selalu digunakan untuk setiap aplikasi Smart Peripheral Controller (SPC) SERIAL LCD.

Kerangka pemrograman SPC SERIAL LCD menggunakan Assembler MetaLink ASM51<sup>®</sup> sebagai berikut :

```
-----  
;FILE TEMPLATE UNTUK SPC Serial LCD  
;DENGAN DT51-MINSYS  
-----  
$MOD51  
  CSEG  
  ORG   4000H  
  LJMP  START  
  
  ORG   4100H  
  $INCLUDE(ENG_I2C.inc) ;DRIVER UTK SEMUA PRODUK SPC I2C  
                        ;BUS  
  $INCLUDE(ENG_SS.inc) ;DRIVER UTK SEMUA PRODUK SPC  
                        ;Sync. Serial BUS  
  $INCLUDE(LCDChar.inc) ;DRIVER UTK SEMUA PRODUK SPC  
                        ;Serial LCD  
  
START:  MOV    SP, #40H  
        .      ;USER MAIN PROGRAM  
        .  
        .  
  
        END
```

### **Penting!**

Beberapa Beberapa merk / tipe LCD Karakter tertentu tidak sepenuhnya kompatibel dengan SPC SERIAL LCD. Beberapa command akan menghasilkan data yang tidak sesuai.

Misalnya:

Pembacaan karakter “A” dengan command ReadDDRAMData pada LCD 16 x 2 (yang dipaketkan dengan SPC SERIAL LCD) akan menghasilkan data “61H”. Namun pada LCD 24 x 2 (yang dipaketkan dengan SPC SERIAL LCD), data yang dihasilkan adalah “16H”.

- ◆ *Terima Kasih atas kepercayaan Anda menggunakan produk kami, bila ada kesulitan, pertanyaan atau saran mengenai produk ini silahkan menghubungi technical support kami :*  
***support@innovativeelectronics.com***
- 
-