

# SPC

SMART PERIPHERAL CONTROLLER

---

## KEYMATIC

### **Trademarks & Copyright**

XT, AT dan PS/2 is a trademark of International Business Machines Corp.  
IBM, PC, and PC-DOS are trademarks of International Business Machines Corp.

MS-DOS and Windows are registered trademarks of Microsoft Corporation.

Pentium is a registered trademark of Intel Corporation.

MetaLink ASM51 is copyright by MetaLink Corporation

# Daftar Isi

---

---

<b>1</b>	<b>Pendahuluan</b> .....	3
1.1	Spesifikasi Eksternal SPC KEYMATIC .....	3
1.2	Spesifikasi Internal SPC KEYMATIC .....	3
1.3	Sistem yang Dianjurkan .....	4
<b>2</b>	<b>Perangkat Keras SPC KEYMATIC</b> .....	4
2.1	Tata Letak Komponen SPC KEYMATIC .....	4
2.2	Hubungan DT-51 Minimum System dengan SPC KEYMATIC ..	5
2.3	Penggunaan SPC KEYMATIC secara Paralel .....	5
2.4	Hubungan SPC KEYMATIC dengan Keyboard dan Keypad .....	5
2.5	Setting Jumper .....	6
2.6	Ekspansi SPC KEYMATIC .....	6
2.7	Mencoba SPC KEYMATIC dengan Example.Hex .....	6
<b>3</b>	<b>Perangkat Lunak SPC KEYMATIC</b> .....	7
3.1	Spesifikasi Paralel .....	7
3.2	Spesifikasi I <sup>2</sup> C Bus .....	8
3.2.1	Kondisi Start dan Stop .....	8
3.2.2	Transfer Data .....	8
3.2.3	Pengalamatan .....	9
3.2.4	Command .....	10
3.2.4.1	Command Input Mode .....	10
3.2.4.2	Command Data .....	11
3.2.4.3	Command Status .....	11
3.2.4.4	Driver dan Rutin .....	12
3.2.4.5	Contoh Aplikasi dan Program .....	13
3.2.4.6	Kerangka Program .....	13
<b>Lampiran</b>		
A.	Skematik SPC KEYMATIC .....	15
B.	Protokol SPC KEYMATIC .....	16
C.	Tabel Kode Data Output SPC KEYMATIC .....	17

## 1. PENDAHULUAN

*Smart Peripheral Controller* / SPC KEYMATIC merupakan kontroler penerima masukan data yang berasal dari keypad dan/atau keyboard dengan menggunakan I<sup>2</sup>C bus sebagai jalur penyampaian data sehingga dapat lebih menghemat dan mempermudah pengkabelan. Selain itu, untuk data yang berasal dari keypad, SPC KEYMATIC dapat digunakan secara paralel. Contoh aplikasi dari SPC KEYMATIC adalah sebagai input karakter pada LCD Display.

### 1.1 SPESIFIKASI EKSTERNAL SPC KEYMATIC

Spesifikasi Eksternal SPC KEYMATIC sebagai berikut:

- Kompatibel penuh dengan DT-51 Minimum System Ver 3.0.
- Kompatibel penuh dengan keypad 4x4 (16 tombol) dan keyboard PS/2.
- Hanya perlu 3 jalur kabel untuk interface dengan mikroprosesor / mikrokontroler lain.
- Dapat digunakan secara I<sup>2</sup>C bus maupun secara paralel.
- Penggunaan secara paralel diakses dengan taraf logic TTL.
- Dapat menyimpan data input ke dalam buffer sebanyak 8 data.
- Untuk angka dan alphabet, memiliki kode yang sesuai dengan standar kode ASCII.
- Dilengkapi dengan jumper untuk setting alamat, sehingga bila menggunakan I<sup>2</sup>C bus dapat di-ekspan sampai 4 board tanpa tambahan perangkat keras.
- Tersedia prosedur siap pakai untuk aplikasi SPC KEYMATIC.

### 1.2 SPESIFIKASI INTERNAL SPC KEYMATIC

Dalam penggunaan dari SPC KEYMATIC akan dikenal adanya tiga layer (lapisan) penggunaan:

Pertama	:	Engine Layer
Kedua	:	Protocol Layer
Ketiga	:	Application Layer

**Engine Layer** adalah lapisan yang mengurus semua kegiatan dari tiap bit yang akan diterima atau yang akan dikirim.

Bagian ini tidak boleh diubah kecuali untuk keperluan khusus. Bagi pengguna yang belum mahir dan berpengalaman tidak dianjurkan untuk mengubah bagian ini.

Bagian ini terdapat dalam *ENG\_I2C.INC*.

**Protocol Layer** adalah lapisan yang terletak satu lapis lebih tinggi dari *Engine Layer* dan dipergunakan untuk mengatur semua lalu lintas data dan sudah tersusun sesuai dengan kegunaan menjadi paket Sub-rutin.

Bagian ini tidak boleh diubah kecuali untuk keperluan khusus. Bagi pengguna yang belum mahir dan berpengalaman tidak dianjurkan untuk mengubah bagian ini.

Bagian ini terdapat dalam *KEYMATIC.INC*.

**Application Layer** adalah lapisan terluar yang dipergunakan untuk berinteraksi secara langsung dengan Users/pengguna.

Bagian ini tidak boleh diubah kecuali untuk keperluan khusus. Bagi pengguna yang belum mahir dan berpengalaman tidak dianjurkan untuk mengubah bagian ini.

Bagian ini terdapat dalam *KEYMATIC.INC*.

Protokol dari SPC KEYMATIC dapat dilihat pada **lampiran B**.

### 1.3 SISTEM YANG DIANJURKAN

Sistem yang dianjurkan untuk penggunaan SPC KEYMATIC adalah:

Perangkat keras :

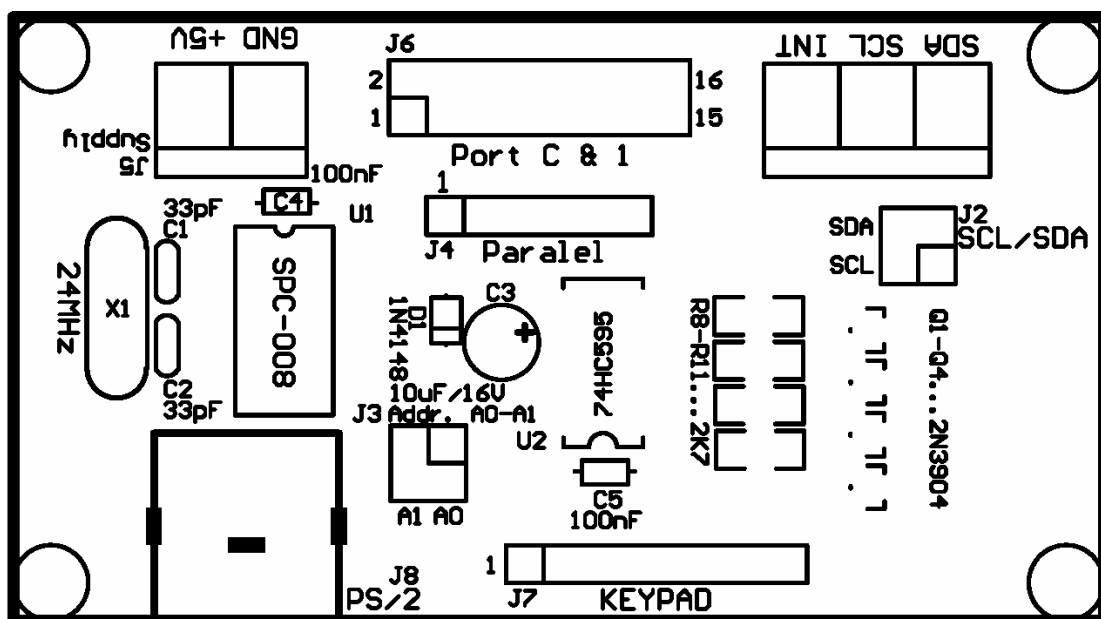
- PC XT / AT Pentium™ IBM Compatible dengan port serial (COM 1/ COM2).
- Board DT-51 Minimum System.
- Floppy Disk 3.5" , kapasitas 1,44 Mbytes atau CD-ROM Drive.
- Hard disk dengan kapasitas minimum 500 Kbytes.

Perangkat lunak :

- Sistem operasi MS-DOS™ atau PC-DOS™.
- Assembler ASM51©
- File-file yang ada pada pada disket/CD program :  
EXAMPLE.ASM, EXAMPLE.HEX, KEYMATIC.INC, ENG\_I2C.INC,  
MANUAL\_SPC\_KEYMATIC.PDF, QUICK\_START\_SPC\_KEYMATIC.PDF

## 2 PERANGKAT KERAS SPC KEYMATIC

### 2.1 TATA LETAK KOMPONEN SPC KEYMATIC



## 2.2 HUBUNGAN DT-51 MINIMUM SYSTEM DENGAN SPC KEYMATIC

Untuk menghubungkan SPC KEYMATIC dengan DT-51 Minimum System dianjurkan untuk menggunakan kabel pita (flat ribbon cable).

Hubungannya ditunjukkan pada tabel berikut :

I <sup>2</sup> C Bus	DT-51 Minimum System PORT C & PORT 1	SPC KEYMATIC J6
INT	Pin 11 (Port 1.2)	Pin 11
SCL	Pin 15 (Port 1.6)	Pin 15
SDA	Pin 16 (Port 1.7)	Pin 16

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

### **Penting !**

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

SPC KEYMATIC merupakan suatu sistem yang ‘**Smart**’. Selain dapat dihubungkan dengan DT-51 Minimum System atau dengan sistem mikroprosesor / mikrokontroler yang lain dengan menggunakan komunikasi I<sup>2</sup>C, SPC KEYMATIC untuk input yang berasal dari keypad dapat juga difungsikan secara paralel (lihat **bagian 2.3**). Apabila Anda ingin menghubungkan SPC KEYMATIC dengan sistem yang lain kami sarankan untuk mempelajari skema dari SPC KEYMATIC (lihat **lampiran A**).

## 2.3 PENGGUNAAN SPC KEYMATIC SECARA PARALEL

SPC KEYMATIC dapat pula berkomunikasi secara paralel (dengan input Keypad) melalui J4. Nama dan fungsi dari J4 dapat dilihat pada tabel berikut.

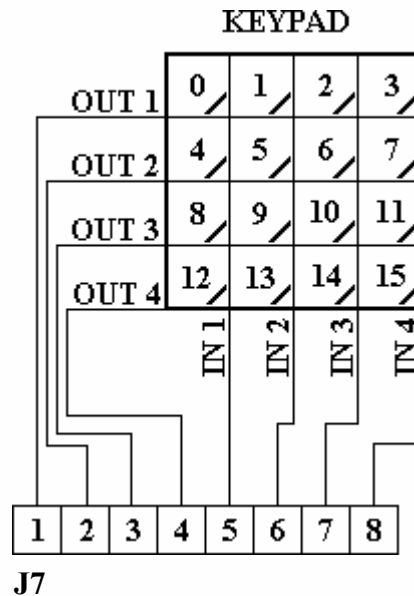
Pin J4	I/O	Nama dan Fungsi
1	Input	RD (Read)
2	Output	INT (Interrupt)
3	Output	Data bit 0
4	Output	Data bit 1
5	Output	Data bit 2
6	Output	Data bit 3

## 2.4 HUBUNGAN SPC KEYMATIC DENGAN KEYBOARD DAN KEYPAD

SPC KEYMATIC memiliki interface keyboard IBM PS/2 mini-DIN 6 pin tetapi tidak menutup kemungkinan untuk menghubungkannya dengan keyboard ber-interface DIN 5 pin yang digunakan pada komputer AT (dengan tambahan adapter). Umumnya keyboard AT memiliki data yang sama dengan keyboard PS/2 tetapi mungkin ada beberapa keyboard yang memiliki data yang berbeda.

Interface keypad yang disediakan memiliki 4 pin IN dan 4 pin OUT sehingga mampu mengakomodasi hingga keypad 4 x 4.

Berikut ini adalah alokasi pin dan konfigurasi hubungan SPC KEYMATIC dengan keypad 4 x 4.



## 2.5 SETTING JUMPER

SPC KEYMATIC memiliki alamat terprogram pada setiap board yang ditentukan oleh setting jumper J3.

J3 (A1)	J3 (A0)	Alamat Terprogram	
<input type="checkbox"/>	<input type="checkbox"/>	0	00
<input type="checkbox"/>		1	01
	<input type="checkbox"/>	2	10
		3 (default)	11
Keterangan :			
<input type="checkbox"/> : jumper tersambung (ON)			

Jumper J2 (Pull up SCL/SDA) digunakan untuk resistor pull up SDA ( $I^2C$  bus data input / output) dan SCL ( $I^2C$  bus clock input). Apabila lebih dari satu board SPC KEYMATIC dihubungkan pada  $I^2C$  bus maka hanya perlu memasang jumper J2 pada salah satu board saja.

## 2.6 EKSPANSI SPC KEYMATIC

SPC KEYMATIC dapat di-ekspan sampai 4 board. Beberapa hal yang perlu diperhatikan apabila menggunakan lebih dari satu board SPC KEYMATIC :

- Setiap board harus mempunyai alamat terprogram yang berbeda, ditentukan oleh jumper J3 (A0/A1).
- Jumper J2 pada salah satu board saja yang dipasang.

## 2.7 MENCOBA SPC KEYMATIC DENGAN EXAMPLE.HEX

### Setting Hardware

- ◆ Hubungkan DT-51 Minimum System dengan SPC KEYMATIC (lihat

bagian 2.2).

- ◆ Hubungkan input Keypad dan Keyboard dengan SPC KEYMATIC (lihat bagian 2.4).
- ◆ Hubungkan port serial DT-51 MinSys dengan COM1/COM2 dari PC dengan menggunakan kabel serial.
- ◆ Setting alamat SPC KEYMATIC pada alamat terprogram ke-3 (default).
- ◆ Download EXAMPLE.HEX yang terdapat pada disket/CD.

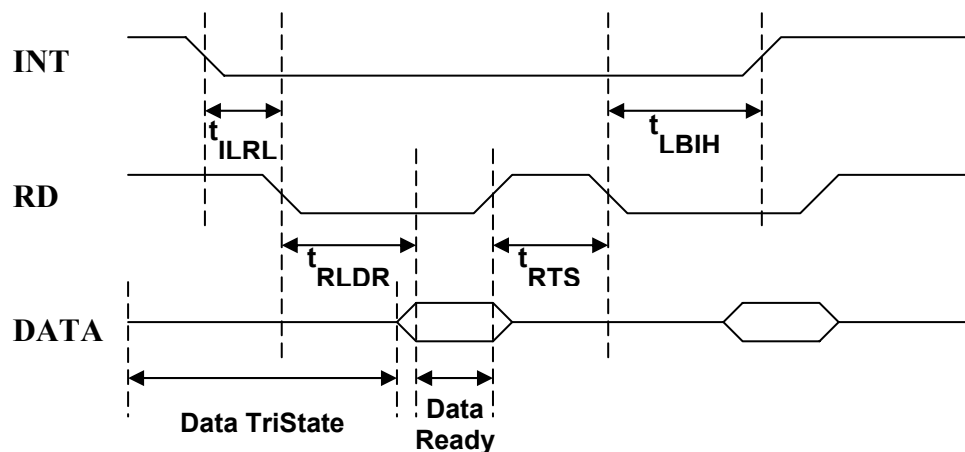
Proses Program EXAMPLE

- ◆ Saat ada penekanan input keypad atau keyboard, maka bit K8\_Int berada pada kondisi '0'. Kode data dari penekanan tombol tersebut dapat dilihat pada Port A dari DT-51 MinSys dan juga dapat dilihat dari HyperTerminal dengan baud rate = 9600 bps, 8 data bit, parity = none, 1 stop bit, dan flow control = none.

### 3 PERANGKAT LUNAK SPC KEYMATIC

#### 3.1 SPESIFIKASI PARALEL

Berikut ini adalah timing diagram dari penggunaan SPC KEYMATIC secara paralel.



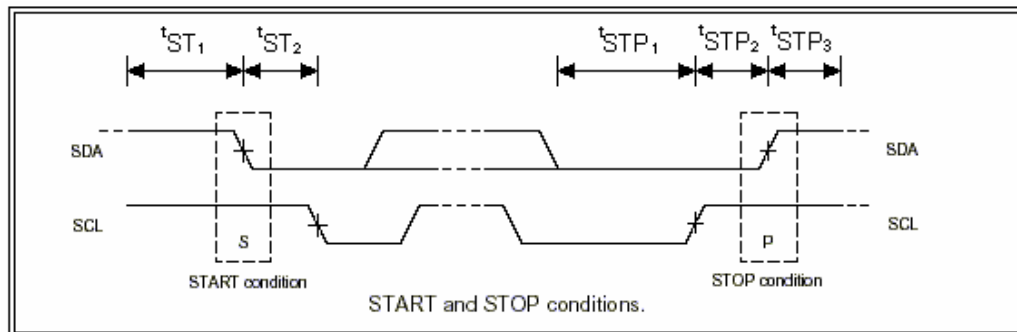
Symbol	Parameter	Min	Max	Units
$t^{ILRL}$	Time INT low to RD low	1	-	$\mu S$
$t^{RLDR}$	Time RD low to Data ready	150	-	$\mu S$
$t^{RTS}$	Time RD Tristate	120	-	$\mu S$
$t^{LBIH}$	Time Last Buffer to INT high	-	100	$\mu S$

- Saat terjadi penekanan tombol pada keypad, maka data dari input keypad akan masuk ke buffer dan pin INT akan diberi logika 'low' oleh SPC KEYMATIC. Pin INT baru akan kembali menjadi high jika buffer sudah tidak berisi data lagi.
- Untuk mengambil data dari buffer, Master harus mentransmisikan pin RD dari high ke low (falling edge). Saat pin RD berada pada kondisi high maka Data akan bersifat TriState (high impedance).

### 3.2 SPESIFIKASI I<sup>2</sup>C BUS

Berikut akan dijelaskan mengenai cara kerja dari komunikasi I<sup>2</sup>C Master (DT-51 Minimum System) – Slave (SPC KEYMATIC). Tugas dari master adalah mengontrol semua komunikasi yang dilakukan, seperti mengatur pulsa clock pada jalur SCL, bit data pada SDA, dan memeriksa atau mengirimkan acknowledge.

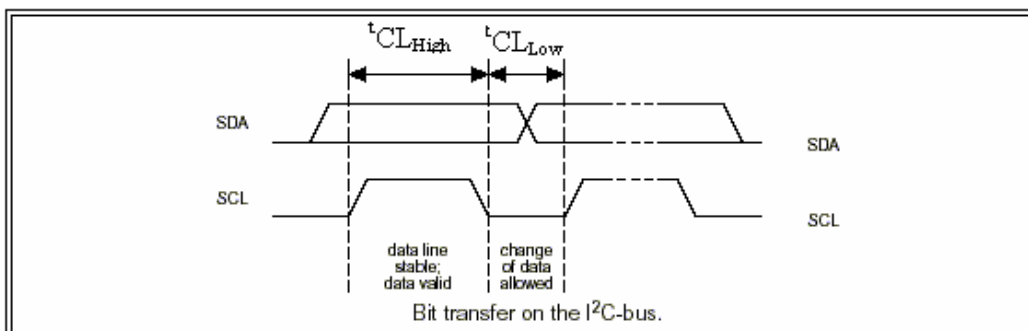
#### 3.2.1 Kondisi Start dan Stop



**Kondisi Start** dalam I<sup>2</sup>C selalu dilakukan dengan cara memberikan level High (“1”) pada jalur SCL kemudian pada jalur SDA terjadi transisi turun (High ke Low).

**Kondisi Stop** dalam I<sup>2</sup>C selalu dilakukan dengan cara memberikan level High (“1”) pada jalur SCL kemudian pada jalur SDA terjadi transisi naik (Low ke High).

#### 3.2.2 Transfer Data



**Transfer data** dapat dilakukan setelah memberikan kondisi **start** dan perubahan bit data (SDA) hanya boleh terjadi / diijinkan saat jalur SCL pada kondisi Low (“0”). Transfer data ini biasanya sebesar 8 bit yang dilakukan secara streaming (Bit Streaming), kemudian dilengkapi dengan pengiriman bit acknowledge.

#### Mode Write

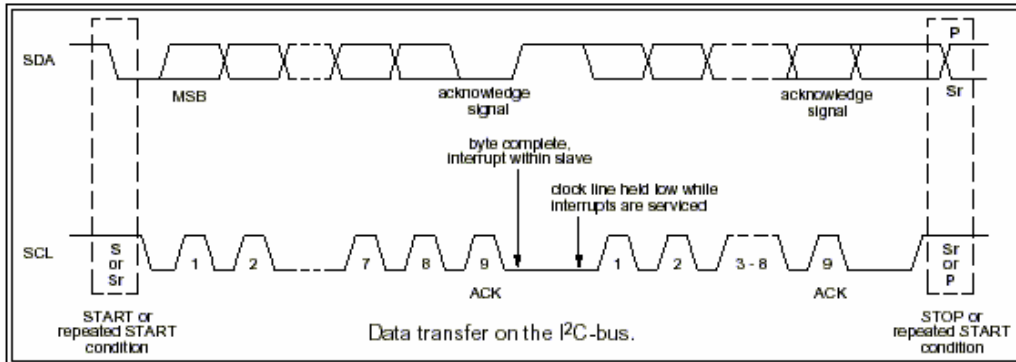
Saat master I<sup>2</sup>C berada pada kondisi **Write** (bit  $\overline{R/\overline{W}}$  pada **AddressI2C** = “0”), maka master I<sup>2</sup>C akan mengirimkan 8 bit data ke slave I<sup>2</sup>C, kemudian master akan menunggu adanya bit acknowledge dari slave, apabila tidak



diterima bit acknowledge maka slave tidak menerima data (bit streaming) secara lengkap (miss/hilang) atau terjadi salah alamat (alamat tetap atau terprogram).

### **Mode Read**

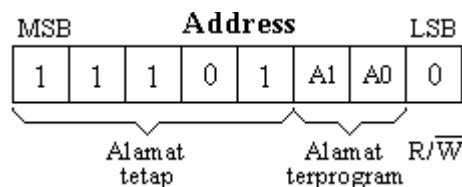
Saat master I<sup>2</sup>C berada pada kondisi **Read** (bit R/W pada **AddressI2C** = “1”), maka master I<sup>2</sup>C akan menerima 8 bit data dari slave I<sup>2</sup>C yang alamatnya sesuai dengan alamat master I<sup>2</sup>C, kemudian master I<sup>2</sup>C akan mengirimkan bit acknowledge untuk memberitahukan kepada slave apakah ingin mengambil data berikutnya atau tidak.



**Diagram** di atas adalah diagram lengkap tentang transfer data melalui I<sup>2</sup>C.

Symbol	Parameter	Min	Units
<sup>t</sup> ST <sub>1</sub>	Time before START action	15	μs
<sup>t</sup> ST <sub>2</sub>	Time after START action	7	μs
<sup>t</sup> STP <sub>1</sub>	Hold time for prepare STOP action	14	μs
<sup>t</sup> STP <sub>2</sub>	Time before STOP action	7	μs
<sup>t</sup> STP <sub>3</sub>	Time after STOP action	7	μs
<sup>t</sup> CL <sub>High</sub>	Time High for data hold	10	μs
<sup>t</sup> CL <sub>Low</sub>	Time Low for change data	12	μs

### **3.2.3 Pengalamatan**



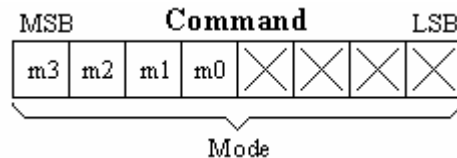
Pengalamatan memanfaatkan register : **AddressI2C**

Memanfaatkan alamat memory 2Fh

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/W). SPC KEYMATIC selalu menggunakan alamat tetap dengan nilai “11101”, 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 2.5**) sehingga pada jalur I<sup>2</sup>C yang sama dengan alamat tetap yang sama (“11101”) dapat digunakan 4 buah modul secara bersamaan dengan membedakan alamat terprogram. Bagian Read/Write (R/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 KEYMATIC) dan bernilai “0”, jika Master I<sup>2</sup>C akan menulis data ke Slave I<sup>2</sup>C.

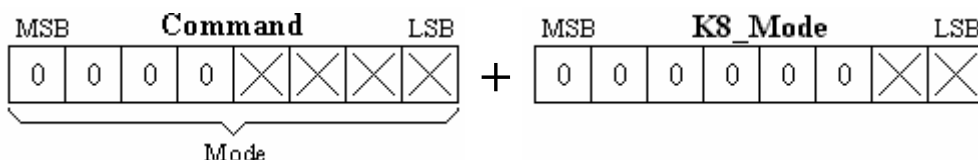
### 3.2.4 Command



m3	m2	m1	m0	x	x	x	x	Mode
0	0	0	0	X	X	X	X	Input Mode
0	0	0	1	X	X	X	X	Data
0	0	1	0	X	X	X	X	Status
0	0	1	1	X	X	X	X	Tidak Terpakai
.	.	.	.	.	.	.	.	.....
1	1	1	1	X	X	X	X	Tidak Terpakai

Command digunakan untuk memilih perintah selanjutnya yang akan diberikan pada device sesuai dengan pilihan mode yang diberikan. Command memiliki 16 kemungkinan mode, namun pada SPC KEYMATIC ini hanya digunakan 3 mode yang pertama saja.

#### 3.2.4.1 Command Input Mode



Command Input Mode diikuti dengan register : **K8\_Mode**

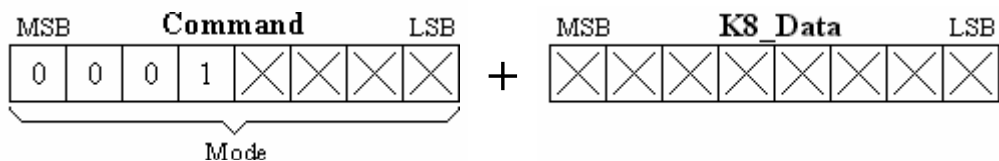
Register K8\_Mode memanfaatkan alamat memory 39h atau dengan nama lain *BufferOut1*

Command Input Mode ini digunakan untuk memilih perangkat masukan (input device) beserta sistem komunikasi yang akan digunakan SPC KEYMATIC untuk berkomunikasi dengan Master.

Ada 4 jenis pilihan Input Mode yang dapat digunakan pada SPC KEYMATIC seperti yang diperlihatkan pada tabel di bawah ini.

K8_Mode	Input Device	Sistem Komunikasi
00H	Keypad	Paralel
01H	Keypad	I <sup>2</sup> C
02H	Keyboard	I <sup>2</sup> C
03H	Keypad & Keyboard	I <sup>2</sup> C

### 3.2.4.2 Command Data



Command Data diikuti dengan register : **K8\_Data**

Register K8\_Data memanfaatkan alamat memory 30h atau dengan nama lain *BufferIn0*

Command Data ini digunakan untuk mengambil kode data output yang terdapat pada buffer SPC KEYMATIC dan disimpan ke dalam register **K8\_Data**. Jika buffer dalam keadaan kosong maka register **K8\_Data** akan berisi '**FFh**'.

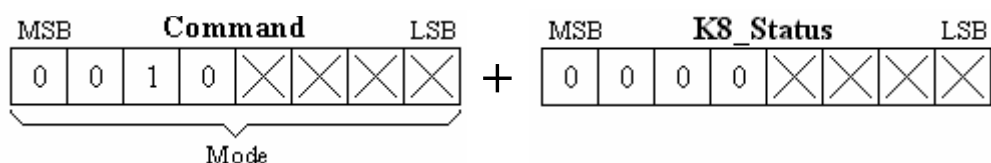
Bit **K8\_Int** digunakan untuk mengetahui apakah buffer SPC KEYMATIC berisi kode data output atau tidak. **K8\_Int** akan bernilai '**1**' jika buffer dalam keadaan kosong dan **K8\_Int** akan bernilai '**0**' jika buffer berisi kode data output.

SPC KEYMATIC mampu menampung buffer sebanyak 8 buffer dan bersifat FIFO (First In First Out) yang berarti data yang pertama kali masuk buffer akan dikeluarkan terlebih dahulu. Jika ada data yang masuk pada saat buffer masih penuh, maka data tersebut akan dibuang.

Untuk perangkat masukan (input device) dari keypad mempunyai kode data output mulai dari '**00h**' sampai '**0Fh**'.

Input device dari keyboard mempunyai kode data output mulai dari '**10h**' sampai '**A3h**'. Untuk abjad, angka dan tanda baca mempunyai kode yang sesuai dengan kode ASCII. Sedangkan kode untuk tombol yang lainnya seperti: F1, Ctrl, Alt dapat dilihat pada **lampiran C**.

### 3.2.4.3 Command Status



Command Status diikuti dengan register : **K8\_Status**

Register K8\_Status memanfaatkan alamat memory 31h atau dengan nama lain *BufferIn1*

Command Status digunakan untuk mengetahui status dari penekanan tombol **Shift**, **NumLock**, **CapsLock**, dan **ScrollLock**.

K8_Status							
7	6	5	4	3	2	1	0
0	0	0	0	Shift	Caps	Num	Scroll

<b>K8_Status</b>	<b>Status (H/L)</b>
Shift	Press/ <u>Un</u> pressed
Caps	ON/OFF
Num	ON/OFF
Scroll	ON/OFF

#### 3.2.4.4 Driver dan Rutin

SPC KEYMATIC dilengkapi dengan driver **KEYMATIC.INC** yang akan mempermudah user dalam pemrograman. **KEYMATIC.INC** menggunakan resource dari mikrokontroler 89C51 sebagai berikut :

- Internal RAM alamat 21h bit 0 dan 1
- Internal RAM dengan alamat 2Fh – 3Fh
- P1.2, P1.6, dan P1.7

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

Driver ini menggunakan 4 buah register dan 1 buah flag yang terdiri dari:

AddressI2C  
K8\_Mode  
K8\_Data  
K8\_Int (flag)  
K8\_Status

Kegunaan dari register dan flag tersebut dapat dilihat pada **bagian 3.2.4.1. - 3.2.4.3.**

Register dan flag tersebut digunakan dalam 3 rutin penting sebagai berikut :

##### **K8\_SetMode**

Fungsi : Memilih input device yang akan digunakan  
Input : AddressI2C, K8\_Mode  
Output : Flag Ack  
Keterangan : Tabel pengaturan register K8\_Mode ini dapat dilihat pada **bagian 3.2.4.1**  
Metode : Isi register AddressI2C dan K8\_Mode sesuai kebutuhan kemudian panggil rutin K8\_SetMode

##### **K8\_GetData**

Fungsi : Mengambil kode data output yang terdapat pada buffer SPC KEYMATIC  
Input : AddressI2C, K8\_Mode, K8\_Int  
Output : K8\_Data  
Keterangan :  
❖ Pada saat buffer berisi kode data output maka bit K8\_Int akan memberikan nilai '0'  
❖ Tabel hasil konversi dari kode data output ini dapat dilihat pada **Lampiran C**  
❖ Pemanggilan rutin K8\_GetData secara berurutan harus mempunyai selang interval minimal 1 ms.

Metode : Isi register AddressI2C sesuai dengan kebutuhan kemudian panggil rutin K8\_GetData

#### **K8\_GetStatus**

Fungsi : Mengetahui status tombol Shift, NumLock, CapsLock, dan ScrollLock dari keyboard

Input : AddressI2C

Output : K8\_Status

Keterangan : Tabel pengaturan register K8\_Status ini dapat dilihat pada **bagian 3.2.4.3**

Metode : Isi register AddressI2C sesuai dengan kebutuhan kemudian panggil rutin K8\_GetStatus

### **3.2.4.5 Contoh Aplikasi dan Program**

Modul SPC KEYMATIC dengan alamat terprogram ke-2 menggunakan keyboard sebagai inputnya. Setiap ada penekanan tombol dari keyboard, SPC KEYMATIC akan langsung mengambil data dan status dari keyboard.

#### **Listing program untuk kasus diatas:**

```
$MOD51
    CSEG
    ORG 4000H
    LJMP Start

    ORG 4100H
    $INCLUDE(ENG_I2C.INC)           ;Driver untuk semua produk
                                   ;SPC I2C
    $INCLUDE(KEYMATIC.INC)       ;Driver SPC KEYMATIC

START:
    MOV SP,#50H
    MOV ADDRESSI2C,#11101100B    ;Memasukkan alamat I2C
    MOV K8_MODE,#02H
    ACALL K8_SETMODE             ;Input SPC = Keyboard

LOOP:
    JB K8_INT,$                 ;Tunggu sampai penekanan
                                   ;tombol keyboard
    ACALL K8_GETDATA             ;Ambil data keyboard dari
                                   ;buffer ke register K8_Data
    ACALL K8_GETSTATUS          ;Ambil data status keyboard
                                   ;ke register K8_Status
    AJMP LOOP
END
```

### **3.2.4.6 Kerangka Program**

Bagi user yang ingin membuat program aplikasi SPC KEYMATIC dengan menggunakan rutin yang sudah ada maka 2 driver berikut harus dimasukkan (include) : *ENG\_I2C.INC* dan *KEYMATIC.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.

*KEYMATIC.INC* merupakan driver yang khusus digunakan untuk SPC KEYMATIC.

#### **Penting !**

ENG\_I2C.INC harus dimasukkan terlebih dahulu sebelum KEYMATIC.INC

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

```
-----  
;FILE TEMPLATE UNTUK SPC I2C BUS  
;DENGAN DT-51 MINSYS  
-----  
$MOD51  
  
      CSEG  
      ORG   4000H  
      LJMP  START  
  
      ORG   4100H  
      $INCLUDE (ENG_I2C.INC)      ;DRIVER UNTUK SEMUA PRODUK  
                                  I2C BUS (harus dituliskan  
                                  sebelum KEYMATIC.INC)  
      $INCLUDE (KEYMATIC.INC)    ;DRIVER SPC KEYMATIC  
  
START:  MOV   SP, #40H  
        .  
        .  
        .  
        .  
      END
```

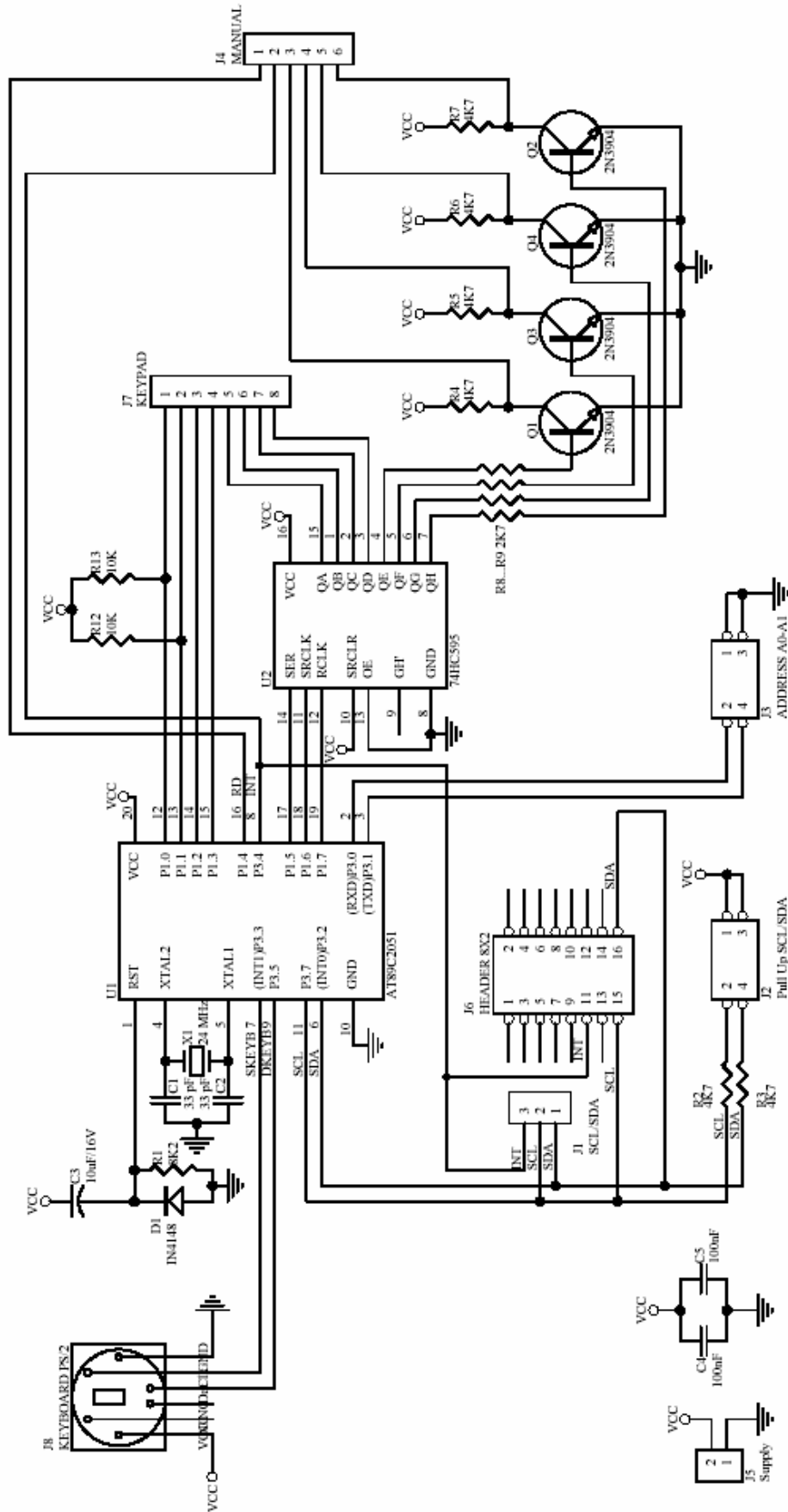
- ◆ *Terima Kasih atas kepercayaan Anda menggunakan produk kami, bila ada kesulitan, pertanyaan atau saran mengenai produk ini silahkan menghubungi technical support kami :*

***support@innovativeelectronics.com***

---

# LAMPIRAN

## A. Skematik SPC KEYMATIC







### C. Tabel Kode Data Output SPC KEYMATIC

Input	K8 Data	Input	K8 Data	Input	K8 Data
Pad 0	00h	0	30h	`	60h
Pad 1	01h	1	31h	a	61h
Pad 2	02h	2	32h	b	62h
Pad 3	03h	3	33h	c	63h
Pad 4	04h	4	34h	d	64h
Pad 5	05h	5	35h	e	65h
Pad 6	06h	6	36h	f	66h
Pad 7	07h	7	37h	g	67h
Pad 8	08h	8	38h	h	68h
Pad 9	09h	9	39h	i	69h
Pad 10	0Ah	:	3Ah	j	6Ah
Pad 11	0Bh	;	3Bh	k	6Bh
Pad 12	0Ch	<	3Ch	l	6Ch
Pad 13	0Dh	=	3Dh	m	6Dh
Pad 14	0Eh	>	3Eh	n	6Eh
Pad 15	0Fh	?	3Fh	o	6Fh
Insert	10h	@	40h	p	70h
End	11h	A	41h	q	71h
Arrow ↓	12h	B	42h	r	72h
Page Down	13h	C	43h	s	73h
Arrow ←	14h	D	44h	t	74h
Esc	15h	E	45h	u	75h
Arrow →	16h	F	46h	v	76h
Home	17h	G	47h	w	77h
Arrow ↑	18h	H	48h	x	78h
Page Up	19h	I	49h	y	79h
Ctrl	1Ah	J	4Ah	z	7Ah
Alt	1Bh	K	4Bh	{	7Bh
Enter	1Ch	L	4Ch		7Ch
Tab	1Dh	M	4Dh	}	7Dh
Delete	1Eh	N	4Eh	~	7Eh
Backspace	1Fh	O	4Fh	Prt Scr	7Fh
Space	20h	P	50h	Pause	80h
!	21h	Q	51h	F1	81h
“	22h	R	52h	F2	82h
#	23h	S	53h	F3	83h
\$	24h	T	54h	F4	84h
%	25h	U	55h	F5	85h
&	26h	V	56h	F6	86h
‘	27h	W	57h	F7	87h
(	28h	X	58h	F8	88h
)	29h	Y	59h	F9	89h
*	2Ah	Z	5Ah	F10	8Ah
+	2Bh	[	5Bh	F11	8Bh
,	2Ch	\	5Ch	F12	8Ch
-	2Dh	]	5Dh	Sleep	8Dh
.	2Eh	^	5Eh	Wake	8Eh
/	2Fh	_	5Fh	Power	8Fh
				GUI	90h
				APPS	91h
				(Application)	

## Windows Multimedia Data Output

<b>Input</b>	<b>K8 Data</b>
Next Track	92h
Prev Track	93h
Stop	94h
Play / Pause	95h
Mute	96h
Volume Up	97h
Volume Down	98h
Media Select	99h
E-mail	9Ah
Calculator	9Bh
My Computer	9Ch
WWW Search	9Dh
WWW Home	9Eh
WWW Back	9Fh
WWW Forward	A0h
WWW Stop	A1h
WWW Refresh	A2h
WWW Favorites	A3h