

Percobaan

6

JUDUL :

SERIAL INTERFACE
Menggunakan DT-51 MinSys

TUJUAN :

- Membuat aplikasi serial interface untuk komunikasi secara serial melalui pin RXD dan TXD pada MCS-51.
- Membuat program menggunakan serial port (DB9) dari DT-51 MinSys sebagai jalur komunikasi dalam berbagai kecepatan, baik sebagai penerima data maupun pengirim data.

TEORI :

MCS-51 memiliki kemampuan untuk berkomunikasi secara serial melalui pin RXD dan TXD. Satu hal yang perlu diingat adalah tingkat tegangan komunikasi kedua pin serial menggunakan tingkat tegangan TTL.

STANDART SERIAL INTERFACE

Pada prinsipnya, komunikasi serial adalah komunikasi dimana transmisi data dilakukan per bit. Interface serial hanya membutuhkan jalur yang sedikit (umumnya hanya 2 jalur) sehingga lebih menghemat pin jika dibandingkan dengan interface parallel.

Komunikasi serial ada dua macam, asynchronous serial dan synchronous serial. Synchronous serial adalah komunikasi dimana hanya ada satu pihak (pengirim atau

penerima) yang menghasilkan clock dan mengirimkan clock tersebut bersama-sama dengan data. Contoh penggunaan synchronous serial terdapat pada transmisi data keyboard.

Asynchronous serial adalah komunikasi dimana kedua pihak (pengirim dan penerima) masing-masing menghasilkan clock namun hanya data yang ditransmisikan, tanpa clock. Agar data yang dikirim sama dengan data yang diterima, maka kedua frekuensi clock harus sama dan harus terdapat sinkronisasi. Setelah adanya sinkronisasi, pengirim akan mengirimkan datanya sesuai dengan frekuensi clock pengirim dan penerima akan membaca data sesuai dengan frekuensi clock penerima. Contoh penggunaan asynchronous serial adalah pada Universal Asynchronous Receiver Transmitter (UART) yang digunakan pada serial port (COM) komputer.

PORT SERIAL DT-51 MinSys

Port serial pada DT-51 MinSys menggunakan konektor DB-9. Port serial bersifat full duplex yang berarti dapat digunakan sebagai input dan output secara simultan. Terdapat dua pin, yaitu TX untuk pengiriman data dan RX penerimaan data. Kedua pin ini, pada DB-9, terdapat pada kaki nomor 2 dan 3. Pada kabel serial DT-51 MinSys, selain kedua kabel tersebut terdapat dua kabel lagi yaitu kabel Ground di kaki 5 dan sinyal DTR di kaki 4.

Register untuk mengirim dan menerima terdapat pada Special Function Register (SFR) yang disebut SBUF. Menulis ke SBUF akan mengaktifkan register pengiriman dan membaca SBUF akan mengakses register penerimaan yang secara fisik terpisah. Dalam menggunakan port serial sebagai jalur komunikasi, maka ada SFR yang harus diatur untuk menentukan mode, yaitu SCON (Serial Control) merupakan register kontrol port serial.

SCON

MSB		LSB
SM0	SM1	SM2
REN	TB8	RB8
TI	RI	

Tabel 7-1. Serial Port Control

Simbol	Posisi	Fungsi
SM0	SCON.7	Menentukan mode komunikasi port serial

SM1	SCON.6	Menentukan mode komunikasi port serial
SM2	SCON.5	Memungkinkan komunikasi multiprosesor pada mode 2 dan 3. Pada mode 2 dan 3 jika diisi 1, maka RI tidak akan diaktifkan jika bit data ke-9 yang diterima (RB8) bernilai 0. Pada mode 1, jika SM2 =1 maka RI tidak akan diaktifkan jika bit stop yang valid tidak diterima. Pada mode 0 SM2 harus bernilai 0.
REN	SCON.4	Diisi 1 untuk mengaktifkan penerimaan dan 0 untuk mematikan penerimaan secara software.
TB8	SCON.3	Bit ke-9 yang akan dikirimkan pada mode 2 dan 3 secara software
RB8	SCON.2	Pada mode 2 dan 3, RB8 adalah bit data ke-9 yang diterima. Pada mode 1, jika SM2 =0, RB8 adalah bit stop yang diterima. Pada mode 0 RB8 tidak digunakan.
TI	SCON.1	Flag interrupt pengiriman (transmit). Di-set (diisi 1) oleh hardware pada akhir bit ke-8 pada mode 0, atau pada awal bit stop pada mode lain. Harus di-clear (diisi 0) secara software.
RI	SCON.0	Flag interrupt penerimaan (receive). Di-set secara hardware pada akhir bit ke-8 pada mode 0, atau ditengah-tengah bit stop pada mode lain (perkecualian pada SM2). Harus di-clear secara software.

Berikut ini adalah penjelasan masing-masing bit SCON yang berkaitan dengan serial port :

Tabel 7-2. Mode Komunikasi Serial

SM0	SM1	Mode	Deskripsi	Baud Rate
0	0	0	8-bit Shift register	Frekuensi osilator/12
0	1	1	8-bit UART	Variabel
1	0	2	9-bit UART	FOsc/64 atau FOsc/32
1	1	3	9-bit UART	Variabel

➤ Mode Operasi

Ada 4 mode komunikasi serial. Mode 0 berupa synchronous serial (shift register), sedangkan tiga mode yang lain berupa asynchronous serial (UART). Pada semua mode pengiriman dilakukan jika ada instruksi yang mengisi nilai register SBUF. Sedangkan pada saat penerimaan, data yang diterima akan disimpan pada register SBUF.

Mode 0

Data serial masuk dan keluar melalui RXD. TXD menghasilkan shift clock. Ke-8 bit data dikirim/diterima dengan LSB (bit terendah) terlebih dahulu. Pada mode ini baud rate yang digunakan adalah sebesar 1/12 dari frekuensi osilator.

Mode 1

Pada mode ini jumlah data yang dikirimkan sebanyak 10 bit yang dikirim melalui TXD atau diterima melalui RXD yang terdiri dari 1 bit start (0), 8 bit data (LSB terlebih dahulu), dan 1 bit stop (1). Pada proses penerimaan, nilai stop bit akan dimasukkan ke RB8 pada SFR SCON. Pada saat proses pengiriman, stop bit akan bernilai '1' secara otomatis. Pada mode 1, baud rate yang digunakan dapat diatur melalui Timer 1 (baud rate variable).

Mode 2

Sebelas bit dikirim (melalui TXD) atau diterima (melalui RXD) yang terdiri dari 1 bit start (0), 8 bit data (LSB terlebih dahulu), 1 bit data ke-9 yang bisa deprogram, dan 1 bit stop (1). Pada proses pengiriman, nilai bit data ke-9 pada TB8 di SCON bisa diberi nilai 1 atau 0. Sebagai contoh, bit parity (P, pada SFR PSW) bisa dipindah ke TB8. Pada proses penerimaan, bit ke-9 akan dimasukkan ke RB8 pada SFR SCON, sedangkan bit stop tidak diperhatikan. Pada mode 2, baud rate yang digunakan adalah sebesar 1/64 frekuensi osilator atau 1/32 frekuensi osilator jika SMOD bernilai '1'.

Mode 3

Mode 3 hampir sama dengan mode 2. Perbedaannya terdapat pada baud rate yang digunakan. Jika mode 2 menggunakan baud rate yang pasti, mode 3 menggunakan baud rate yang dihasilkan oleh timer 1.

Pada semua mode, pengiriman diaktifkan oleh semua instruksi yang menggunakan SBUF sebagai register tujuan. Penerimaan diaktifkan pada mode 0 oleh kondisi RI = 0 dan REN = 1. Pada mode lain, penerimaan diaktifkan oleh masuknya bit start dengan syarat REN harus bernilai 1.

Mode 1 merupakan mode yang lebih banyak dan mudah digunakan. Pada saat pengiriman, sinyal 'write to SBUF' akan mengisi 1 ke dalam bit ke-9 dan mengirimkan permintaan pengiriman data ke TX. Jika semua data termasuk bit ke-9 sudah terkirim, maka TI akan di-set. Pada penerima, jika semua data sudah diterima, maka SBUF dan

RB8 akan di-load, dan RI di-set. Kondisi ini akan terjadi jika RI = 0 dan SM2 = 0 atau bit stop yang diterima = 1. Jika kedua kondisi ini terpenuhi, maka bit stop akan masuk ke RB8 dan 8 bit data masuk ke SBUF serta RI di-set. Jika SM2 =1, interrupt penerimaan tidak akan diaktifkan kecuali bit stop yang valid diterima. Pengertian bit stop yang valid adalah bit stop yang sama antara pengirim dan penerima.

Karena baud rate pada mode 1 variabel, maka kita harus menentukan baud rate yang akan digunakan. Baud rate disini memiliki peran yang sangat vital. Jika baud rate pengirim dan penerima tidak sesuai, maka data yang dikirim dan diterima menjadi tidak sesuai. Untuk menghasilkan baud rate, digunakan timer 1. Oleh karena itu baud rate tergantung dari rate overflow timer 1 dan nilai dari SMOD. Register SMOD pada SFR PCON akan menggandakan baud rate pada mode 1,2, dan 3 jika SMOD bernilai 1 dan timer 1 digunakan sebagai penghasil baud rate.

➤ **Baud Rate**

Baud rate adalah frekuensi clock yang digunakan dalam pengiriman dan penerimaan data. Satuan baut rate pada umumnya adalah bps (bit per second), yaitu jumlah bit yang dapat ditransmisikan per detik.

Baud rate untuk mode 0 bernilai tetap dengan rumus yang terdapat pada persamaan 1.

$$Baud\ Rate = \frac{\text{Frekuensi Osilator}}{12} \quad (1)$$

Sedangkan Baud rate untuk mode 2 memiliki dua variasi tergantung dari kondisi SMOD. Rumus baud rate untuk mode 2 terdapat pada persamaan 2.

$$Baud\ Rate = \frac{2^{SMOD}}{64} \times \text{Frekuensi Osilator} \quad (2)$$

Baud rate untuk mode 1 dan 3 dihasilkan oleh timer 1. Pengaturan baud rate untuk mode ini dapat dilakukan dengan cara mengubah nilai SMOD, TMOD, dan TH1. Nilai baud rate dapat diperoleh dengan menggunakan persamaan 3.

$$Baud\ Rate = \frac{2^{SMOD}}{32} \times \text{Timer 1 Overflow Rate} \quad (3)$$

Dalam aplikasi ini interrupt Timer 1 harus dimatikan. Timer 1 sendiri bisa dikonfigurasi untuk timer maupun counter yang beroperasi dalam salah satu dari ke-3 modusnya. Umumnya, Timer 1 digunakan sebagai timer pada mode auto reload atau 8-bit Auto Reload (TMOD upper untuk Timer 1 = 0010b). dalam hal ini, persamaan baud rate akan berubah menjadi seperti pada persamaan 4.

$$Baud\ Rate = \frac{2^{SMOD}}{32} \times \frac{Frekuensi\ Osilator}{12 \times (256 - TH1)} \quad (4)$$

Berdasarkan persamaan 4, dapat dihitung berapa nilai TH1 yang dibutuhkan jika diketahui baudrate yang diinginkan dengan menggunakan persamaan 5.

$$TH1 = 256 - \frac{2^{SMOD} \times Frekuensi\ Osilator}{384 \times Baud\ Rate} \quad (5)$$

Satu hal yang harus diperhatikan dalam pengaturan baud rate adalah nilai baud rate dan TH1 diusahakan harus tepat dan bukan merupakan pembulatan. Untuk komunikasi serial kecepatan tinggi, pembulatan terhadap nilai-nilai tersebut dapat mengakibatkan kekacauan dalam proses pengiriman atau penerimaan. Jika terdapat nilai pecahan, disarankan untuk mengganti osilatornya dengan frekuensi yang sesuai. Untuk komunikasi dengan kecepatan rendah, toleransi terhadap kesalahan cukup besar sehingga pembulatan masih boleh dilakukan.

Misalkan baud rate yang diinginkan adalah 19200 bps dengan frekuensi osilator 11,0592 MHz. Dengan memasukkan data ini ke dalam persamaan 5 maka akan didapat persamaan 6.

$$TH1 = 256 - \left(2^{SMOD} \times 1,5 \right) \quad (6)$$

Jika 2^{SMOD} bernilai '1', maka akan didapat TH1 sebesar 254,5. untuk menghindari TH1 berupa pecahan, 2^{SMOD} harus bernilai '2' (SMOD bernilai '1') sehingga didapat TH1 sebesar 253 atau FDh.

Untuk mendapatkan baud rate yang lambat, didapat dengan mengoperasikan Timer 1 pada mode 1 dengan rumus pada persamaan 7.

$$Baud\ Rate = \frac{2^{SMOD}}{32} \times \frac{Frekuensi\ Osilator}{12 \times (65536 - Timer\ 1)} \quad (7)$$

Beberapa konfigurasi baud rate yang umum digunakan terdapat dalam tabel 6-3.

Tabel 7-3. Nilai dan Konfigurasi baud rate

Serial		Timer 1				
Mode	Baud Rate	Frekuensi Osilator	SMOD	C/T	Mode	Reload
0	1,6667 Mbps (max)	20 MHz	X	X	X	X
2	625 Kbps (max)	20 MHz	1	X	X	X
1,3	104,1667 Kbps (max)	20 MHz	1	0	2	FFh
1,3	19,2 Kbps	11,0592 MHz	1	0	2	FDh
1,3	9,6 Kbps	11,0592 MHz	0	0	2	FDh
1,3	4,8 Kbps	11,0592 MHz	0	0	2	FAh
1,3	2,4 Kbps	11,0592 MHz	0	0	2	F4h
1,3	1,2 Kbps	11,0592 MHz	0	0	2	E8h
1,3	137,5 bps	11,9856 MHz	0	0	2	1Dh
1,3	110 bps	6 MHz	0	0	2	72h
1,3	110 bps	12 MHz	0	0	1	FEEBh

➤ Inisialisasi Serial

Proses inisialisasi bertujuan untuk menentukan mode komunikasi serial baud rate yang digunakan.

Register yang harus diatur terlebih dahulu meliputi :

1. SCON

Langkah pertama adalah menentukan mode yang akan digunakan (mode 0, 1, 2, atau 3), kemampuan menerima data, dan nilai bit ke-9.

Misalnya mode yang digunakan adalah mode 1 dengan kemampuan menerima data tanpa komunikasi multiprosesor, maka instruksinya adalah sebagai berikut :

```
MOV SCON, #01010000b
```

atau

```
MOV  SCON, #50h
```

atau

```
SETB SM1
```

```
SETB REN
```

2. TMOD, TH1 dan/atau TL1, PCON, dan TCON

Jika komunikasi serial digunakan pada mode 1 atau 3, maka langkah berikutnya adalah menentukan baud rate.

Misalnya Timer/Counter 1 digunakan sebagai timer dalam mode 2 untuk membangkitkan baud rate 19200 bps, maka instruksinya adalah sebagai berikut :

```
MOV  TMOD, #00100000b
```

```
MOV  TH1, #0FDh
```

```
MOV  PCON, #10000000b
```

```
MOV  TCON, #01000000b
```

atau

```
MOV  TMOD, #20h
```

```
MOV  TH1, #0FDh
```

```
MOV  PCON, #80h
```

```
MOV  TCON, #40h
```

atau

```
MOV  TMOD, #20h
```

```
MOV  TH1, #0FDh
```

```
MOV  PCON, #80h
```

```
SETB TR1
```

3. IE dan/atau IP

Jika komunikasi serial yang deprogram akan digunakan sebagai sumber interrupt, maka IE dan/atau IP juga harus diatur.

Misalnya komunikasi serial digunakan sebagai sumber interrupt dengan prioritas tinggi, maka instruksinya adalah sebagai berikut :

```
MOV  IP, #00010000b
```

```
MOV  IE, #10010000b
```

atau

```
MOV IP, #10h
```

```
MOV IE, #90h
```

atau

```
SETB PS
```

```
SETB ES
```

```
SETB EA
```

PERALATAN :

- Modul DT-51
- PC beserta software
 - Debugger
 - Downloader
 - Editor
 - Hyper Terminal
- LED sebagai piranti output

PROSEDUR :

Pemakaian Port LCD pada DT-51

1. Hardware

- Hubungkan konektor (kabel pita) Port 1, Port C pada DT-51 ke *Modul Display LED* jika output yang digunakan adalah Port 1 atau Port C. Tetapi bila outputnya adalah Port A atau Port B maka hubungkan konektor (kabel pita) Port A atau Port B pada DT-51 ke *Modul Display LED* (seperti pada percobaan 1, gambar 1-3.)

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.
- Setelah download sukses, tutuplah window downloader dan bukalah program *Hyper terminal* dan aturlah port serial yang digunakan (COM1) dan kecepatan baudrate-nya.

PROGRAM PERCOBAAN :

- *Program 6.1. Program penerimaan data dari PC (Data dari keyboard ditampilkan pada LED di Port 1)*

```

$MOD51
                                CSEG
                                ORG  4000H
                                LJMP  Start

                                ;interrupt penerimaan

```

```

                ORG 4023H
                CLR RI
                MOV A, SBUF
                MOV P1, A
                RETI
;interrupt diaktifkan dan serial berada pada mode 1
;dengan REN = 1
START:         MOV SP, #30H
                MOV SCON, #50H
;inisialisasi baud rate (9600 bps)
                MOV TMOD, #20H
                MOV TL1, #0FDH
                MOV TH1, #0FDH
                MOV PCON, #00H
                SETB TR1
                SETB ES
                SETB EA
                SJMP $
                END

```

- **Program 6.2. Program pengiriman data ke PC (Data dari reg. A pada DT-51 ditampilkan ke hypper terminal)**

```

$MOD51
                CSEG
                ORG 4000H
                LJMP Start
;interrupt pengiriman
                ORG 4023H
                CLR TI
                CJNE A, #39H, PLUS
                MOV A, #30H
                SJMP EXIT
PLUS:          INC A
EXIT:          RETI

DELAY:        MOV R7, #0FFH
LUP:          MOV R6, #0FFH
                DJNZ R6, $
                DJNZ R7, LUP
                RET
START:        MOV SP, #30H
                CLR ET1
;serial berada pada mode 1 dengan REN=0
                MOV SCON, #40H
;inisialisasi baud rate (19200 bps)
                MOV TMOD, #20H
                MOV TL1, #0FDH
                MOV TH1, #0FDH
                MOV PCON, #80H
                SETB TR1
                SETB ES
                SETB EA
                MOV A, #30H

```

```

        ULANG:      MOV   SBUF,A
                   LCALL  DELAY
                   SJMP  ULANG
                   END

```

- Program 6.3. Program penerimaan dan pengiriman data (Gabungan prog. 6.1 dan 6.2)

```

        $MOD51
                   CSEG
                   ORG   4000H
                   LJMP  Start

;interrupt penerimaan
                   ORG   4023H
                   CLR   RI
                   MOV   A,SBUF
                   MOV   P1,A

;mengirimkan data ke PC
                   MOV   SBUF,A
                   JNB   TI,$
                   CLR   TI
                   RETI

;interrupt diaktifkan dan serial pada mode 1 REN = 1
START:      MOV   SP,#30H
                   MOV   SCON,#50H
;inisialisasi baud rate (9600 bps)
                   MOV   TMOD,#20H
                   MOV   TL1,#0FDH
                   MOV   TH1,#0FDH
                   MOV   PCON,#00H
                   SETB  TR1
                   SETB  ES
                   SETB  EA
                   SJMP  $
                   END

```

TUGAS :

1. Buat program untuk menampilkan LED di port A dengan syarat :
 - Semua LED dimulai dari keadaan padam.
 - Atur Komunikasi serial dalam mode 1 dengan kecepatan 19200 bps
 - Aktifkan serial interrupt.
 - DT-51 MinSys akan menerima data dari PC (dari penekanan keyboard) dan menampilkan data tersebut ke LED bergantian dengan komplemen data tersebut

hingga ada data lain yang masuk. Misalkan data yang diterima bernilai '35h' atau '00110101b', maka LED akan menampilkan data '00110101b' dan '11001010b' secara bergantian.

2. Buat program untuk menampilkan huruf "a" sampai "z" kemudian "A" sampai "Z" secara berulang-ulang pada Hyper Terminal dengan syarat :
 - Atur komunikasi serial dalam mode 1 dengan kecepatan 9600 bps.
 - Aktifkan Serial interrupt.
3. Buat program untuk komunikasi serial dengan syarat :
 - Semua LED (penggunaan port bebas) dimulai dari keadaan padam.
 - Atur komunikasi serial dalam mode 1 dengan kecepatan 19200 bps.
 - Aktifkan serial interrupt.
 - DT-51 MinSys akan menerima 2 set data dari PC, menampilkannya di LED, dan mengirimnya kembali secara terbalik. Misalkan data yang diterima berturut-turut adalah '30h' kemudian '31h'. LED akan menampilkan data '30h' kemudian '31h'. Data yang dikirim dan ditampilkan PC secara berturut-turut adalah '31h' dan '30h'.