

# DT-51

## DT-51 *Application Note* AN72 – SPC Seven Segment Display dengan Bahasa C untuk MCS-51®

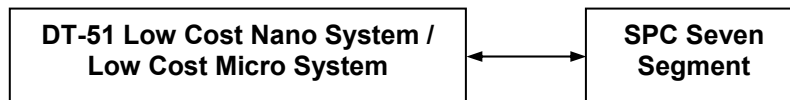
Oleh: Tim IE

Aplikasi kali ini masih memanjakan para pecinta bahasa C menggunakan  $\mu\text{C}/51^{\circ}$ . Pada AN ini bahasa C akan digunakan untuk menjembatani DT-51 Low Cost Series dengan SPC Seven Segment Display pada mode display dengan masing-masing antarmuka yang tersedia (serial UART RS232 / RS485, paralel 4-bit, dan SPI). Jika dikembangkan, AN ini dapat digabungkan dengan AN69 untuk membentuk display besar yang menampilkan suhu. Dengan seven segment besar yang dihubungkan ke SPC Seven Segment Display, tampilan tersebut dapat diletakkan di perkantoran (indoor) atau taman kota (outdoor).

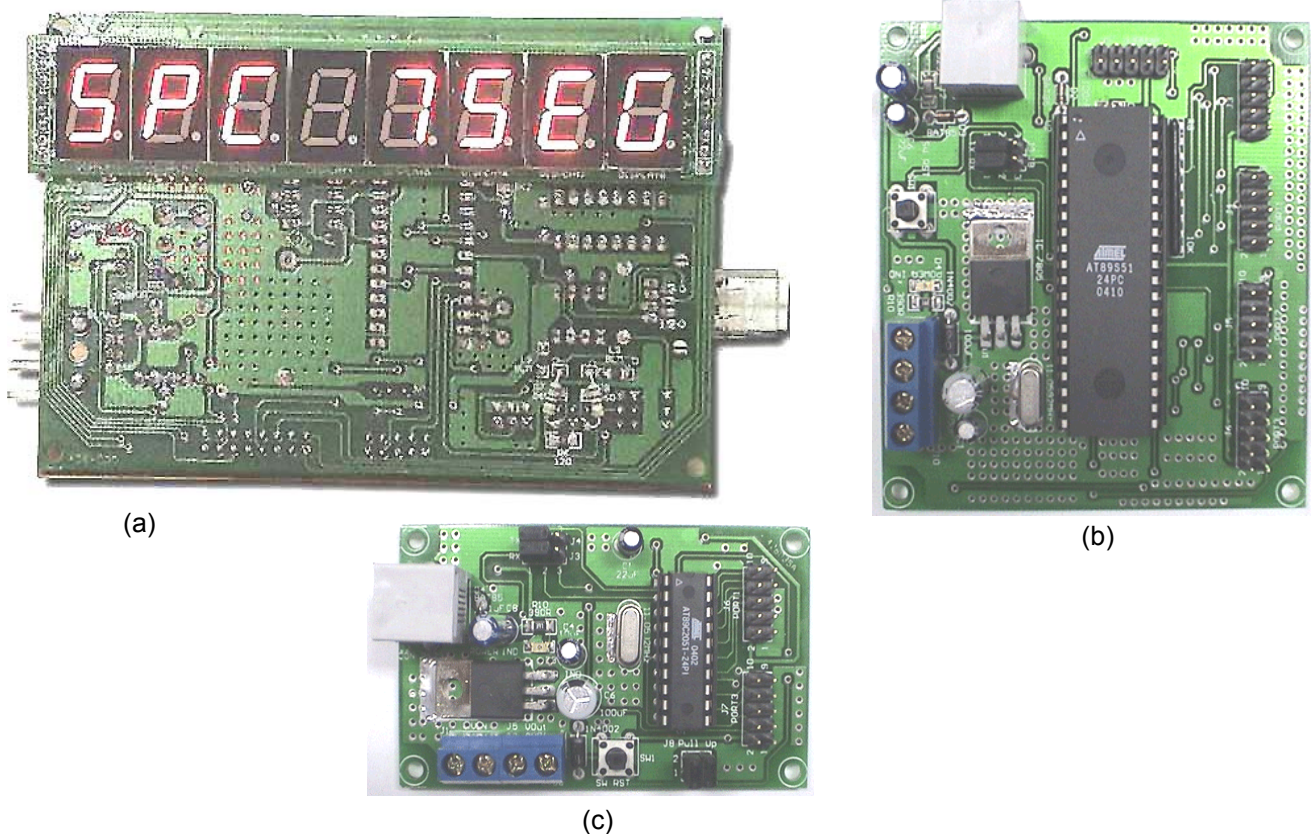
Komponen yang diperlukan:

- 1 DT-51 Low Cost Nano System / Low Cost Micro System
- 1 SPC Seven Segment Display

Adapun blok diagram sistem secara keseluruhan adalah sebagai berikut:



Gambar 1  
Blok Diagram AN72



Gambar 2  
SPC Seven Segment (a), DT-51 Low Cost Micro System (b), dan DT-51 Low Cost Nano System (c)

Hubungan antara modul-modul untuk antarmuka serial UART RS232 adalah sebagai berikut:

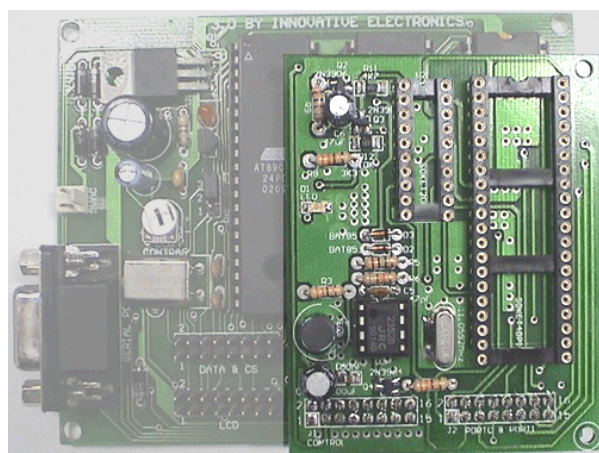
DT-51 Low Cost Nano System / Low Cost Micro System (RJ11)	SPC Seven Segment J10
RX	RX232RJ (pin 5)
TX	TX232RJ (pin 6)
GND	SGND (pin 7)

**Tabel 1**  
**Hubungan DT-51 Low Cost Nano System / Low Cost Micro System dengan SPC Seven Segment Menggunakan Antarmuka Serial UART RS232**

Pada SPC Seven Segment aturlah jumper S2, jumper J8, J9, J11, dan jumper J2, J3, J4 agar dapat bekerja pada mode display dengan antarmuka RS232 atau RS485. Konfigurasi jumper ini dapat dilihat pada Manual SPC Seven Segment bagian 2.2.

DT-51 Low Cost Series harus menggunakan level converter RS232-RS485 agar dapat bekerja dalam mode RS485. Level converter ini dihubungkan dengan SPC Seven Segment Display pada konektor RJ45 J10 pin 1 (B+), 2 (A-), dan 4 (SGND).

Apabila menggunakan DT-51 Low Cost Nano System, programlah terlebih dulu serial.hex ke chip AT89C2051 menggunakan **DT-HiQ Programmer** atau **DT-51 MinSys ver 3.0 + DT-51 ProgPAL** atau parallel programmer lain. Untuk DT-51 Low Cost Micro System, pemrograman serial.hex ke chip AT89S51 dapat dilakukan secara "in-system" menggunakan AT89S ISP Programmer Cable.



**Gambar 3**  
**DT-51 MinSys ver 3.0 + DT-51 ProgPAL**

Program serial.c (serial.hex) dapat digunakan untuk mode RS232 maupun RS485, tetapi untuk mode RS485 pengguna perlu mengatur terlebih dahulu alamat pada program agar sesuai dengan alamat awal pada SPC Seven Segment atau sebaliknya. Pada aplikasi ini SPC Seven Segment yang digunakan memiliki alamat awal 53.

Hubungan antara modul-modul untuk antarmuka SPI adalah sebagai berikut:

SPI Bus	DT-51 Low Cost Nano System / Low Cost Micro System	SPC Seven Segment J12
SCK	Port 1.3*	Pin 7
MOSI	Port 1.2*	Pin 8
CLR	Port 1.1*	Pin 9
MISO	Port 1.0*	Pin 10
GND	GND	Pin 1

\* Pin ini tidak mutlak dan dapat diganti pin lain.

**Tabel 2**  
**Hubungan DT-51 Low Cost Nano System / Low Cost Micro System dengan SPC Seven Segment Menggunakan Antarmuka SPI**

Setelah hubungan tersebut pada tabel 2 sesuai, aturlah jumper S2 pada SPC Seven Segment agar dapat bekerja pada mode display dengan antarmuka SPI. Dan juga aturlah alamat pada program agar sesuai dengan alamat awal pada SPC Seven Segment atau sebaliknya. Pada aplikasi ini SPC Seven Segment yang digunakan memiliki alamat awal 53. Kemudian programlah spi.hex ke DT-51 Low Cost Series dengan cara yang sama seperti pada aplikasi yang menggunakan antarmuka Serial UART.

**H**ubungan antara modul-modul untuk antarmuka Paralel adalah sebagai berikut:

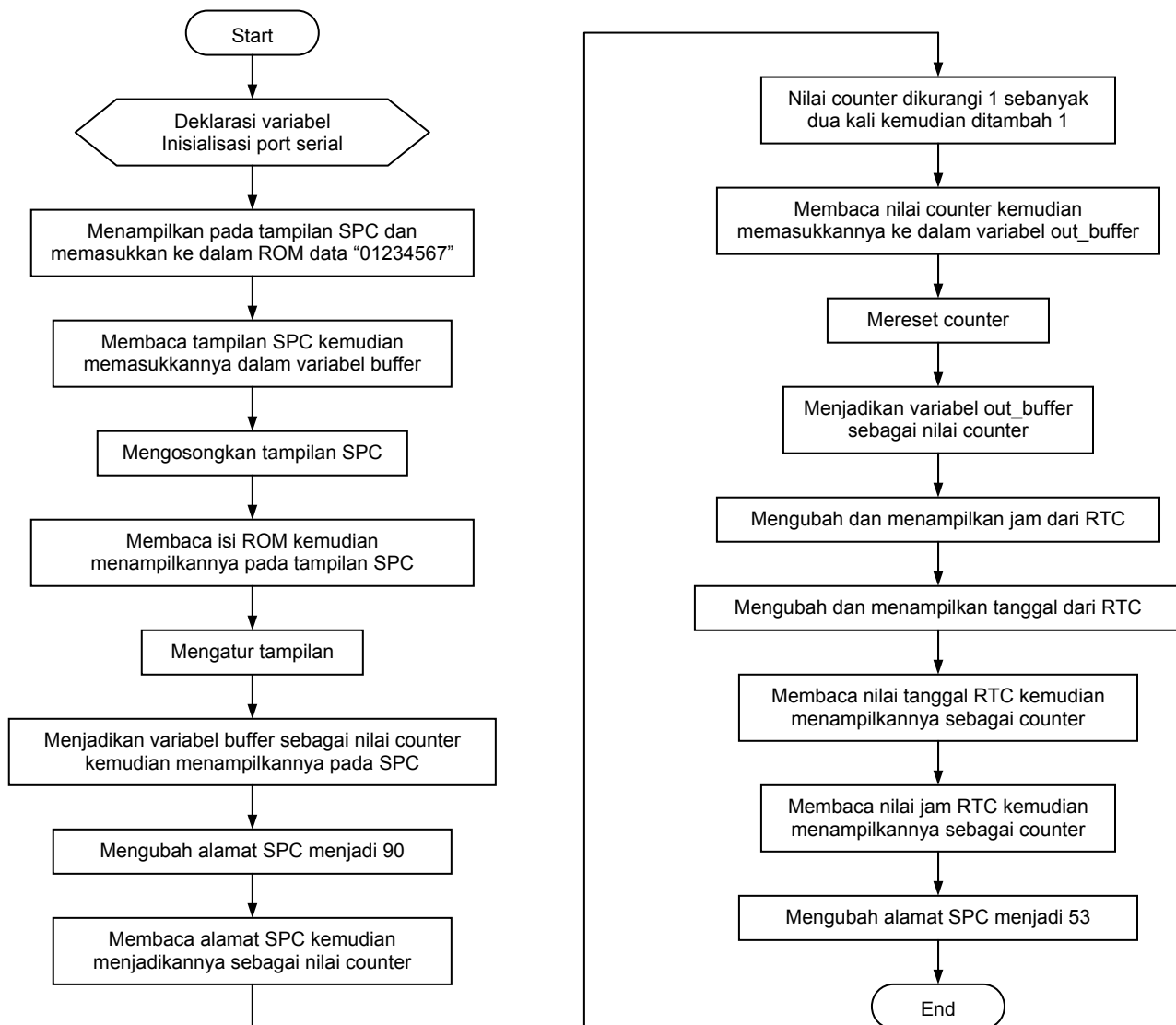
4-bit Paralel	DT-51 Low Cost Mini System / Low Cost Micro System	SPC 7 Segment J12
Data bit 0	Port 1.7*	Pin 3
Data bit 1	Port 1.6*	Pin 4
Data bit 2	Port 1.5*	Pin 5
Data bit 3	Port 1.4*	Pin 6
Clock	Port 1.3*	Pin 7
DOT	Port 1.2*	Pin 8
Clear	Port 1.1*	Pin 9
GND	GND	Pin 1

\* Pin ini tidak mutlak dan dapat diganti pin lain

**Tabel 3**  
**Hubungan DT-51 Low Cost Nano System / Low Cost Micro System dengan SPC Seven Segment Menggunakan Antarmuka Paralel**

Untuk aplikasi yang menggunakan antarmuka paralel, sesuaikan hubungan antar modul dengan tabel 3. Dan aturlah jumper S2 pada SPC Seven Segment agar dapat bekerja pada mode display dengan antarmuka paralel. Kemudian programlah port.hex ke DT-51 Low Cost Series dengan cara yang sama seperti pada aplikasi yang menggunakan antarmuka Serial UART.

**F**lowchart program untuk aplikasi dengan antarmuka serial UART dan SPI adalah sebagai berikut:



**Gambar 4**  
**FlowChart Program untuk Aplikasi dengan Antarmuka Serial UART dan SPI**

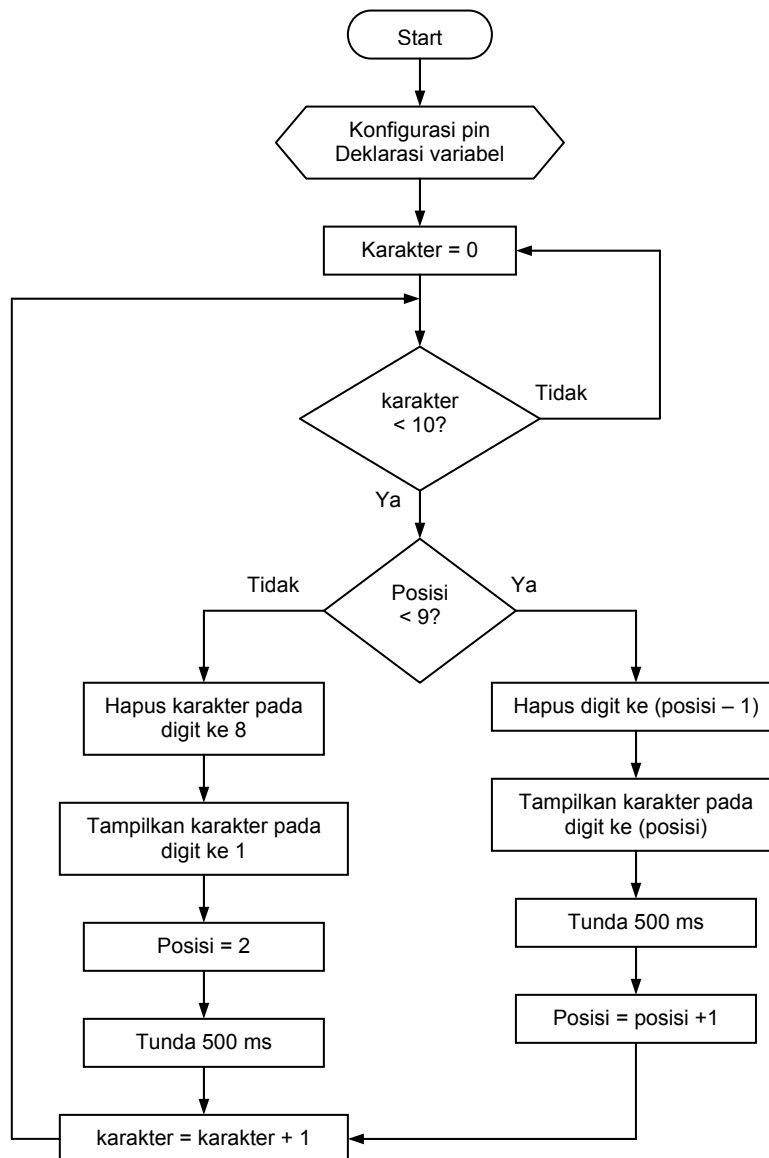
**P**rogram Utama untuk aplikasi dengan antarmuka serial UART dan SPI akan diproses sebagai berikut:

- Proses yang pertama dilakukan adalah deklarasi variabel. Pada bahasa pemrograman C setiap prosedur memiliki variabel lokal masing-masing. Sehingga fungsi tiap variabel pada suatu prosedur belum tentu sama dengan prosedur yang lain. Sedangkan variabel-variabel pada program utama dapat dijelaskan sebagai berikut:  
 out\_hour, out\_minute, out\_second, out\_mode12, out\_pm adalah variabel-variabel bertipe character untuk menyimpan data jam, menit, detik, mode, dan status am/pm dari RTC.  
 out\_date, out\_month, out\_year adalah variabel-variabel bertipe character untuk menyimpan data tanggal, bulan, dan tahun dari RTC.  
 buffer adalah variabel bertipe unsigned long integer sebagai tempat menyimpan data sementara.  
 out\_buffer adalah variabel bertipe unsigned long integer sebagai tempat menyimpan data sementara.  
 addr adalah variabel bertipe character untuk menyimpan data alamat SPC Seven Segment.  
 i, j adalah variabel bertipe character sebagai indeks untuk proses pengulangan.  
 puluhan adalah variabel bertipe unsigned long integer untuk data perkalian.
- Pada program untuk antarmuka serial UART (serial.c) dilakukan proses inisialisasi port serial mikrokontroler, yaitu menentukan baudrate 9600 bps, mode 2 (1 start bit, 8-bit data, tanpa parity, dan 1

stop bit). Sedangkan pada program untuk antarmuka SPI (spi.c) tidak menggunakan port serial mikrokontroler.

3. Menampilkan huruf 0 hingga 7 ke display SPC Seven Segment pada kolom yang sesuai dengan nilai huruf yang akan ditampilkan. Dan juga memasukkan nilai tersebut ke dalam ROM pada alamat ROM yang sesuai dengan nilai huruf yang akan dimasukkan. (contoh: huruf 0 akan ditampilkan ke kolom 0 dan dimasukkan ke dalam ROM alamat 0) Dan mengatur tampilan agar dapat menampilkan karakter tersebut.
4. Membaca tampilan kolom 0 hingga 6 dari SPC Seven Segment, kemudian memasukkannya ke dalam variabel buffer. Dengan konversi nilai untuk kolom 0 adalah satuan dan seterusnya sampai dengan kolom 6 bernilai jutaan.
5. Menghapus tampilan dengan menggunakan prosedur `ss_cleardisplay`.
6. Membaca nilai ROM pada alamat memori 0 hingga 7 dan menampilkannya satu per satu ke kolom 5 pada display SPC Seven Segment dengan jeda waktu 500 ms.
7. Mengatur tampilan agar berfungsi sebagai tampilan counter. Lalu menetapkan nilai awal counter sesuai dengan nilai data pada variabel buffer dan menampilkannya pada SPC Seven Segment selama 1 detik.
8. Mengubah alamat SPC Seven Segment menjadi 90 lalu membacanya kembali untuk dijadikan sebagai nilai awal counter dan menampilkannya selama 1 detik.
9. Nilai counter dikurangi 1 sebanyak dua kali menggunakan prosedur `ss_countdown`. Kemudian ditambah 1 sebanyak satu kali menggunakan prosedur `ss_countup`. Masing-masing dengan jeda waktu 1 detik.
10. Membaca nilai counter dari SPC Seven Segment kemudian menyimpan nilainya pada variabel `out_buffer`.
11. Mengubah nilai counter kembali menjadi nol.
12. Menetapkan nilai counter sesuai dengan nilai data pada variabel `out_buffer` dan menampilkannya dalam satu detik.
13. Mengubah waktu (jam, menit, dan detik) RTC dan mengatur tampilan untuk menampilkan waktu RTC selama 1 detik.
14. Mengubah tanggal (termasuk bulan dan tahun) RTC dan mengatur tampilan SPC Seven Segment untuk menampilkan tanggal RTC selama 1 detik.
15. Membaca nilai tanggal RTC kemudian menampilkannya sebagai nilai counter selama 1 detik.
16. Membaca nilai waktu RTC kemudian menampilkannya sebagai nilai counter.
17. Mengubah alamat SPC Seven Segment kembali ke alamat 53.

**F**lowchart program untuk aplikasi dengan antarmuka paralel adalah sebagai berikut:



**Gambar 5**  
**FlowChart Program untuk Aplikasi dengan Antarmuka Paralel**

**P**rogram Utama untuk port.hex akan diproses sebagai berikut:

1. Proses yang pertama dilakukan adalah menentukan konfigurasi / definisi pin I/O untuk komunikasi paralel dan deklarasi variabel. Variabel-variabel yang digunakan pada program utama yaitu sebagai berikut:  
karakter adalah variabel bertipe character digunakan untuk menyimpan data yang akan ditampilkan.  
posisi adalah variabel bertipe character digunakan untuk menyimpan indeks posisi tampilan.
2. Program mengisi variabel karakter dengan nol. Sedangkan variabel posisi tidak ditentukan.
3. Melakukan pengecekan: apakah variabel karakter berisi data dengan nilai kurang dari 10? Kalau tidak maka program kembali ke tahap 2.
4. Dilakukan pengecekan: apakah variabel posisi berisi data dengan nilai kurang dari 9?  
Kalau ya maka program akan menghapus tampilan SPC Seven Segment pada digit ke (posisi - 1), menampilkan data dari variabel karakter pada digit ke (posisi), dan variabel posisi ditambah dengan 1.  
Kalau tidak maka program akan menghapus tampilan SPC Seven Segment pada digit ke 8, menampilkan data dari variabel karakter pada digit ke 1, dan mengisi variabel posisi dengan data bernilai 2.
5. Apapun hasil pengecekan variabel posisi, program akan melakukan penundaan selama 500 mili detik. Lalu program kembali ke tahap 3.

Bila program untuk antarmuka paralel ini dijalankan maka SPC Seven Segment akan menampilkan karakter 0 hingga 9 secara bergantian pada posisi digit yang terus berputar ke kanan. Aplikasi untuk antarmuka paralel ini jauh lebih sederhana dibandingkan dengan aplikasi sebelumnya karena SPC Seven Segment dengan antarmuka 4-bit paralel ini hanya dapat berfungsi sebagai display numerik 8 digit dan tidak dapat di-cascade.

*L*isting program terdapat pada **AN72.ZIP**.

*S*elamat berinovasi!

$\mu$ C/51 is copyright by Wickenhäuser Elektrotechnik.  
MCS-51 is a registered trademark of Intel Corporation.