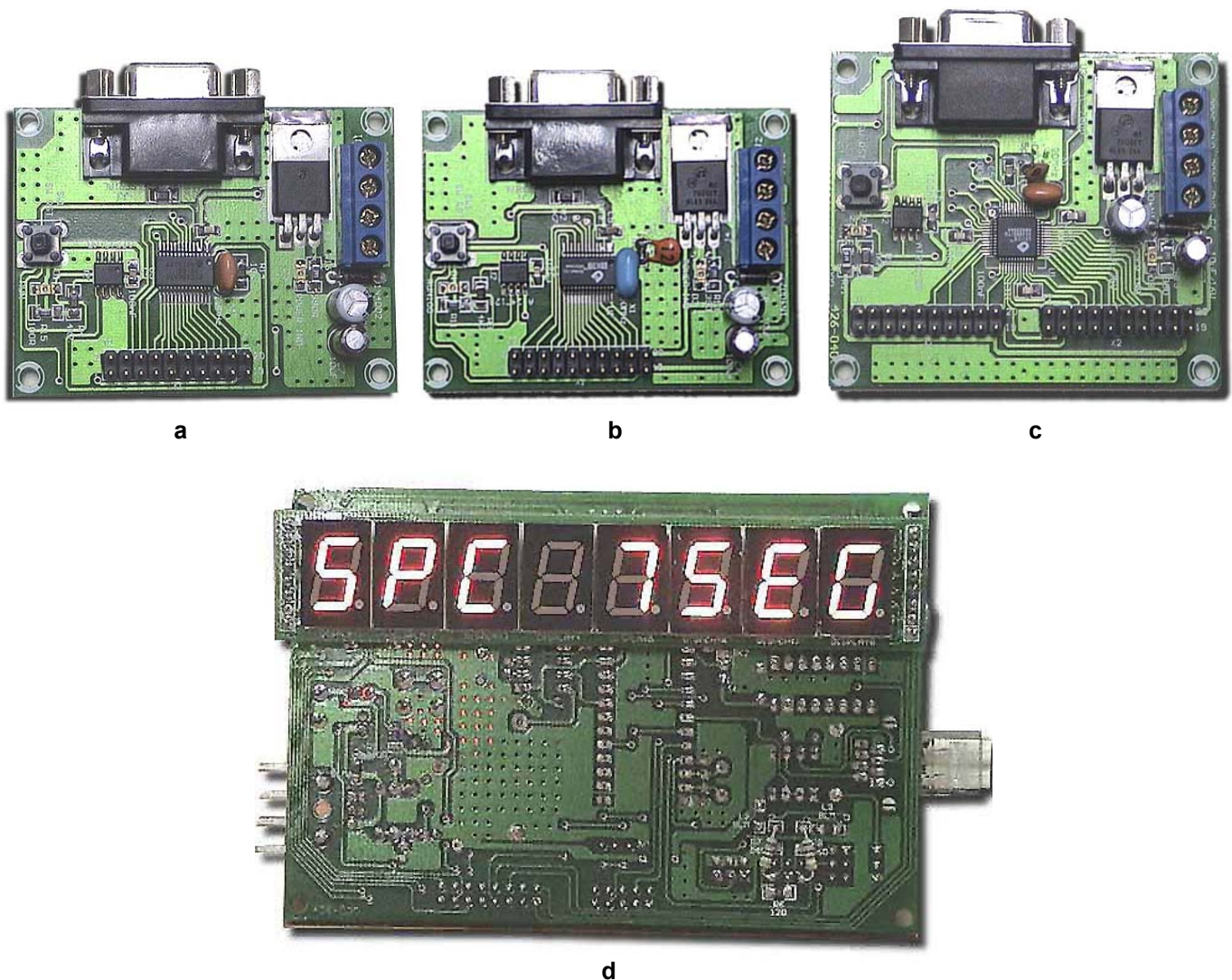


Tidak haram jika de KITS SPC Seven Segment Display akan dihubungkan dengan modul DT-BASIC meski de KITS SPC Seven Segment menggunakan AT89C51 yang berbeda keluarga dengan DT-BASIC. DT-BASIC dapat menampilkan karakter, tanggal, jam, atau counter, dengan menggunakan semua antarmuka yang disediakan SPC Seven Segment Display. Dengan cara seperti ini, DT-BASIC yang “imut” pun dapat menampilkan angka berukuran besar, misalnya 4 inci (± 10 cm). Bahkan jika dikembangkan lebih lanjut dan digabungkan dengan sensor suhu seperti SHT11, aplikasi ini dapat digunakan untuk menampilkan suhu dalam ukuran yang cukup besar untuk dalam ruangan maupun luar ruangan.

Komponen yang diperlukan:

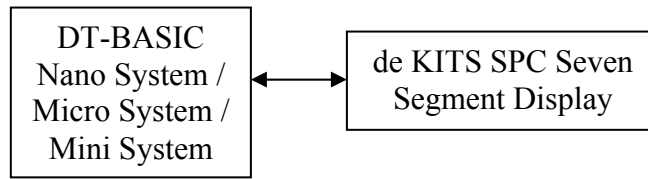
- 1 DT-BASIC Nano System / Micro System / Mini System
- 1 de KITS SPC Seven Segment Display



Gambar 1

DT-BASIC Nano System (a), DT-BASIC Micro System (b), DT-BASIC Mini System (c), de KITS SPC Seven Segment Display (d)

Adapun blok diagram sistem secara keseluruhan adalah sebagai berikut:



Gambar 2
Blok Diagram AN66

Hubungan antara modul-modul tersebut adalah sebagai berikut:

a. Secara UART-TTL

| DT-BASIC Nano System / Micro System / Mini System | De KITS SPC Seven Segment Display |
|---|-----------------------------------|
| GND | GND |
| P2* | TX (pin 2 J9) |
| P3* | RX (pin 2 J8) |

* Pin ini tidak mutlak dan dapat diganti pin lain dengan cara mengubah program

Tabel 1

Hubungan DT-BASIC Nano System / Micro System / Mini System dengan de KITS SPC Seven Segment Display Secara UART-TTL

Jika ingin menggunakan UART RS-232 atau RS-485 (konektor J10 pada de KITS SPC Seven Segment Display), maka jalur komunikasi pada DT-BASIC (dalam AN ini yaitu P2 dan P3), harus diberi konverter tegangan agar logika komunikasi bersesuaian dan tidak merusak salah satu modul.

Aturlah jumper mode S2 pada SPC Seven Segment Display agar antarmuka yang digunakan adalah UART RS-232 atau RS-485.

Sesuaikan alamat SPC Seven Segment Display dengan alamat yang digunakan pada program Serial.bs2.

Setelah semua rangkaian dan sumber tegangan terhubung dengan benar, programlah Serial.bs2 ke dalam DT-BASIC Nano System dengan menggunakan program BASIC STAMP® Editor dan kabel serial DT-BASIC. Untuk menggunakan tipe DT-BASIC yang lain, ubah dulu konfigurasi awal pada program:

- Untuk DT-BASIC Nano System: '{\$STAMP BS2}
- Untuk DT-BASIC Micro System: '{\$STAMP BS2sx}
- Untuk DT-BASIC Mini System: '{\$STAMP BS2p}

b. Secara SPI

| DT-BASIC Nano System / Micro System / Mini System | De KITS SPC Seven Segment Display |
|---|-----------------------------------|
| GND | GND |
| P3* | SCK (pin 7 J12) |
| P2* | MOSI (pin 8 J12) |
| P1* | CLR (pin 9 J12) |
| P0* | MISO (pin 10 J12) |

* Pin ini tidak mutlak dan dapat diganti pin lain dengan cara mengubah program

Tabel 2

Hubungan DT-BASIC Nano System / Micro System / Mini System dengan de KITS SPC Seven Segment Display Secara SPI

Aturlah jumper mode S2 pada SPC Seven Segment Display agar antarmuka yang digunakan adalah SPI. Sesuaikan alamat SPC Seven Segment Display dengan alamat yang digunakan pada program SPI.bs2.

Setelah semua rangkaian dan sumber tegangan terhubung dengan benar, programlah SPI.bs2 ke dalam DT-BASIC Nano System dengan menggunakan program BASIC STAMP® Editor dan kabel serial DT-BASIC. Untuk menggunakan tipe DT-BASIC yang lain, ubah dulu konfigurasi awal pada program:

- Untuk DT-BASIC Nano System: '{ \$STAMP BS2 }
- Untuk DT-BASIC Micro System: '{ \$STAMP BS2sx }
- Untuk DT-BASIC Mini System: '{ \$STAMP BS2p }

c. Secara 4 Bit Paralel

| DT-BASIC Nano System / Micro System / Mini System | De KITS SPC Seven Segment Display |
|--|--------------------------------------|
| GND | GND |
| P7* | Data Bit 0 (pin 3 J12) |
| P6* | Data Bit 1 (pin 4 J12) |
| P5* | Data Bit 2 (pin 5 J12) |
| P4* | Data Bit 3 (pin 6 J12) |
| P3* | Clock (pin 7 J12) |
| P2* | DOT (pin 8 J12) |
| P1* | Clear (pin 9 J12) |

* Pin ini tidak mutlak dan dapat diganti pin lain dengan cara mengubah program

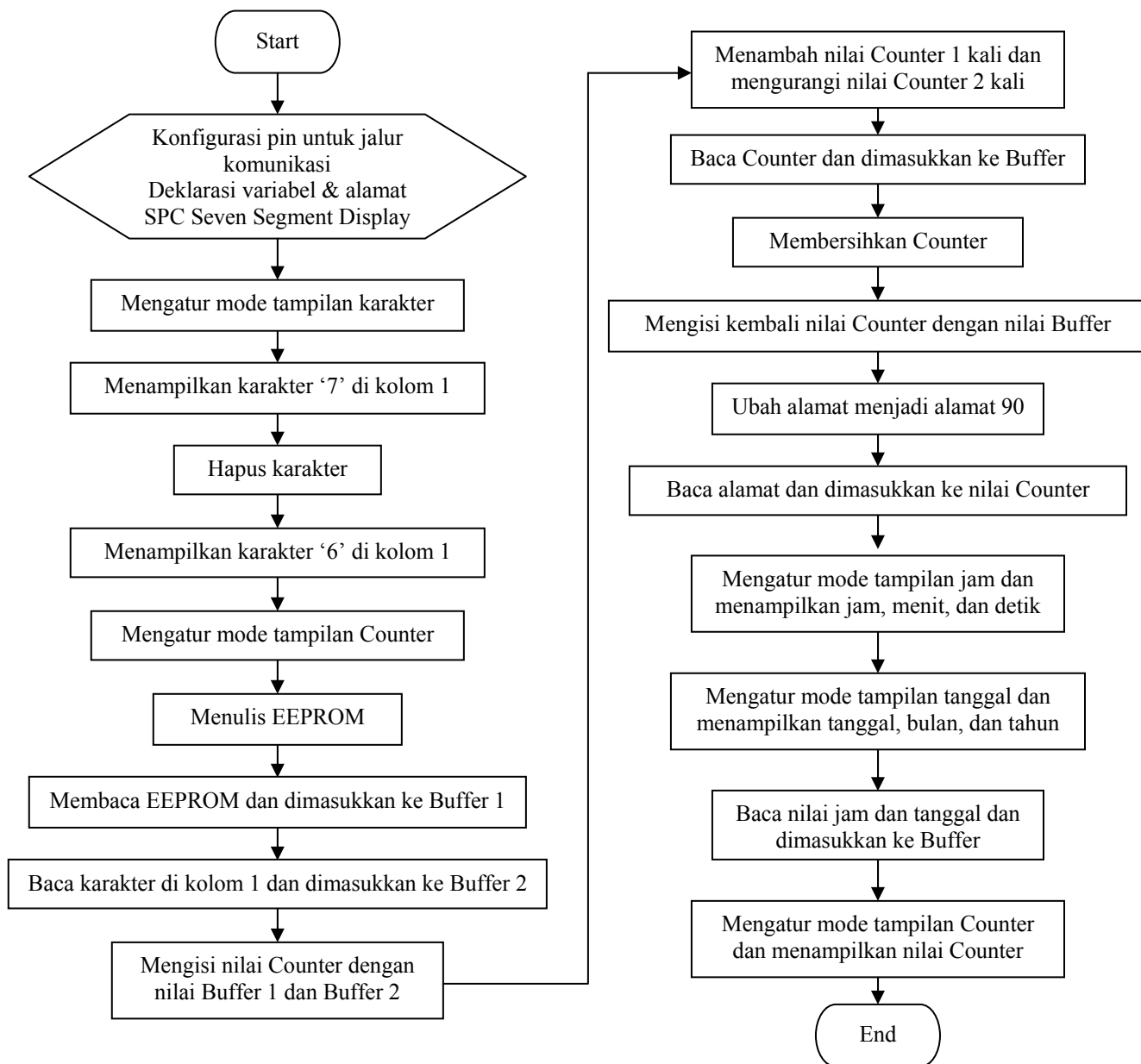
Tabel 3
Hubungan DT-BASIC Nano System / Micro System / Mini System dengan de KITS SPC Seven Segment Display Secara 4 Bit Paralel

Aturlah jumper mode S2 pada SPC Seven Segment Display agar antarmuka yang digunakan adalah 4 Bit Paralel. Sesuaikan alamat SPC Seven Segment Display dengan alamat yang digunakan pada program Port.bs2.

Setelah semua rangkaian dan sumber tegangan terhubung dengan benar, programlah Port.bs2 ke dalam DT-BASIC Nano System dengan menggunakan program BASIC STAMP® Editor dan kabel serial DT-BASIC. Untuk menggunakan tipe DT-BASIC yang lain, ubah dulu konfigurasi awal pada program:

- Untuk DT-BASIC Nano System: '{ \$STAMP BS2 }
- Untuk DT-BASIC Micro System: '{ \$STAMP BS2sx }
- Untuk DT-BASIC Mini System: '{ \$STAMP BS2p }

Pada dasarnya, *flowchart* untuk Serial.bs2 dan SPI.bs2 sama saja, yaitu sebagai berikut:



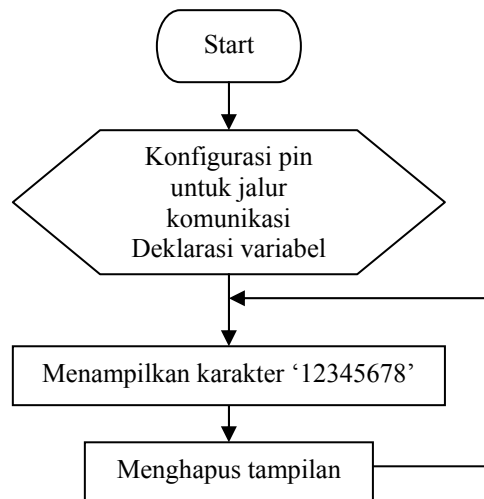
Gambar 3
Flowchart Program Serial.bs2 dan SPI.bs2

Program akan diproses sebagai berikut:

1. Proses yang pertama dilakukan adalah menentukan definisi pin untuk komunikasi serial dan konfigurasi *baud rate* (untuk UART).
2. Deklarasi variabel *addr*, *i*, *j*, *in_var*, *out_var*, *kolom*, *char*, *buffer*, *minus*, *stat*, *blink*, *kontras*, *display*, *hour*, *minute*, *second*, *pm*, *mode12*, *year*, *month*, *date*, *rom_addr*, *rom_data*, dan *dev_addr*.
Addr adalah variabel alamat dari SPC Seven Segment Display. Variabel ini harus sama dengan alamat sebenarnya dari SPC Seven Segment Display. Variabel ini harus sama dengan alamat sebenarnya dari SPC Seven Segment Display.
i, j adalah variabel untuk proses *looping*.
in_var, *out_var* adalah variabel pada prosedur *low level* (mis: *ss_write_byte*).
kolom, *char* adalah variabel yang dipakai dalam prosedur *ss_writetchar* dan *ss_readchar*.
buffer, *minus* adalah variabel yang dipakai dalam prosedur *ss_presetcounter* dan *ss_readcounter*.
stat, *blink*, *kontras*, *display* adalah variabel yang dipakai dalam prosedur *ss_setdisplay*.
hour, *minute*, *second*, *pm*, *mode12* adalah variabel yang dipakai dalam prosedur *ss_setrtcdate* dan *ss_readrtcdate*.
year, *month*, *date* adalah variabel yang dipakai dalam prosedur *ss_setrtcdate* dan *ss_readrtcdate*.

- rom_addr, rom_data adalah variabel yang dipakai dalam prosedur ss_writerom dan ss_readrom.
 dev_addr adalah variabel yang dipakai dalam prosedur ss_setaddress.
3. Mengisi Addr dengan alamat SPC Seven Segment (53), mengaktifkan RTC, mengatur kontras, dan mengaktifkan *blink*.
 4. Menampilkan karakter '7' pada kolom 1, lalu dihapus dan diganti dengan karakter '6'.
 5. Mengisi EEPROM alamat 5 dengan nilai 6.
 6. Membaca karakter kolom 1 dan nilai EEPROM alamat 5.
 7. Hasil pembacaan dimasukkan ke nilai Counter dan ditampilkan.
 8. Nilai Counter ditambah 1 kali dan dikurangi 2 kali.
 9. Membaca nilai Counter dan dimasukkan ke variabel buffer lalu menghapus nilai Counter.
 10. Variabel buffer dimasukkan kembali ke nilai Counter dan ditampilkan.
 11. Mengubah alamat SPC Seven Segment menjadi 90. Sehingga sejak bagian ini, alamat akan berubah menjadi 90.
 12. Membaca alamat SC Seven Segment dan ditampilkan.
 13. Menampilkan jam dan diubah menjadi '01 01 01'.
 14. Menampilkan tanggal dan diubah menjadi '02-02-02'.
 15. Membaca jam dan tanggal dan dimasukkan ke variabel buffer.
 16. Variabel buffer dimasukkan ke nilai Counter dan ditampilkan.
 17. Mengembalikan alamat SPC Seven Segment Display dari 90 menjadi 53.

Flowchart untuk Port.bs2 adalah sebagai berikut:



Gambar 4
Flowchart Program Port.bs2

Program akan diproses sebagai berikut:

1. Proses yang pertama dilakukan adalah menentukan definisi pin untuk antarmuka.
2. Deklarasi variabel char, kolom, datain.
 char dan kolom adalah variabel yang dipakai untuk menampilkan suatu karakter pada display SPC Seven Segment Display.
 datain adalah variabel yang digunakan dalam prosedur *low level* (mis: ss_write4bit).
3. Menampilkan '12345678'.
4. Menghapus tampilan dan kembali ke langkah 3.

Listing program terdapat pada **AN66.ZIP**.

Selamat berinovasi!

BASIC Stamp is a registered trademark of Parallax, Inc.