

**A**lphanumeric display dapat menampilkan karakter-karakter alfabet relatif lebih jelas jika dibandingkan dengan *seven segment display*. Sayangnya penggunaan *alphanumeric display* ini relatif lebih rumit jika dibanding dengan *seven segment display*. SPC Alphanumeric Display akan membuat hal ini jauh lebih sederhana. Penggunaan DT-AVR Low Cost Nano System dan bahasa pemrograman BASIC dengan *compiler* BASCOM-AVR© akan membuat aplikasi berikut ini menjadi sangat mudah. Dalam aplikasi ini, masukan dari keyboard komputer akan ditampilkan ke SPC Alphanumeric Display melalui DT-AVR Low Cost Nano System. Dan komunikasi antara kedua modul ini dapat dilakukan secara I<sup>2</sup>C atau *synchronous serial*. AN ini dilengkapi dengan program aplikasi komputer (Project1.exe) yang memungkinkan masukan dari *keyboard* komputer ditampilkan ke modul secara langsung.

Komponen yang diperlukan:

- 1 DT-AVR Low Cost Nano System / Low Cost Micro System
- 1 SPC Alphanumeric Display

**A**dapun blok diagram sistem secara keseluruhan adalah sebagai berikut:



Gambar 1  
Blok Diagram AN83

**H**ubungan antara modul-modul tersebut adalah sebagai berikut:

DT-AVR Low Cost Nano System / Low Cost Micro System	SPC Alphanumeric Display
VCC	+5V (J5)
GND	GND (J5)
PD.4*	SCL (J1)
PD.5*	SDA (J1)
PB.0*	Clear (J4-S1)
PB.2*	Data Serial (J4-S2)
PB.4*	Clock (J4-S3)

\* Pin ini tidak mutlak dan dapat diganti pin lain, tetapi harus mengubah program.

Tabel 1

Hubungan DT-AVR Low Cost Nano System / Low Cost Micro System dengan SPC Alphanumeric Display secara I<sup>2</sup>C dan *Synchronous Serial*

Gunakan kabel serial DT-AVR Low Cost Nano System / Low Cost Micro System untuk menghubungkan modul dengan komputer. Dan aturlah *jumper* J4 & J5 DT-AVR Low Cost Nano System / Low Cost Micro System pada posisi 1-2 (untuk komunikasi serial UART RS-232). Pasanglah *jumper* J2 SPC Alphanumeric Display untuk memberi resistor *pull up* pada jalur komunikasi I<sup>2</sup>C. Aturlah *jumper* A2-A0 (J3) SPC Alphanumeric Display untuk alamat terprogram 111b (*default*).

Setelah semua rangkaian dan sumber tegangan terhubung dengan benar, programlah ALPHE.BIN / ALPHE.HEX ke dalam DT-AVR Low Cost Nano System (berbasis AT90S2313) dengan menggunakan DT-HiQ AVR In System Programmer atau divais AVR ISP programmer lain yang memiliki konektor 10-pin standar Atmel. Dan jalankan program Project1.exe pada komputer.

**Catatan:**

Dalam aplikasi ini menggunakan DT-AVR Low Cost Nano System (dengan mikrokontroler AT90S2313) sehingga ALPHE.BIN / ALPHE.HEX yang disertakan dalam **AN83.ZIP** hanya berlaku untuk AT90S2313.

Apabila menggunakan mikrokontroler tipe lain, bukalah Alphe.bas menggunakan BASCOM-AVR© dan ubahlah baris pertama dalam program tersebut:

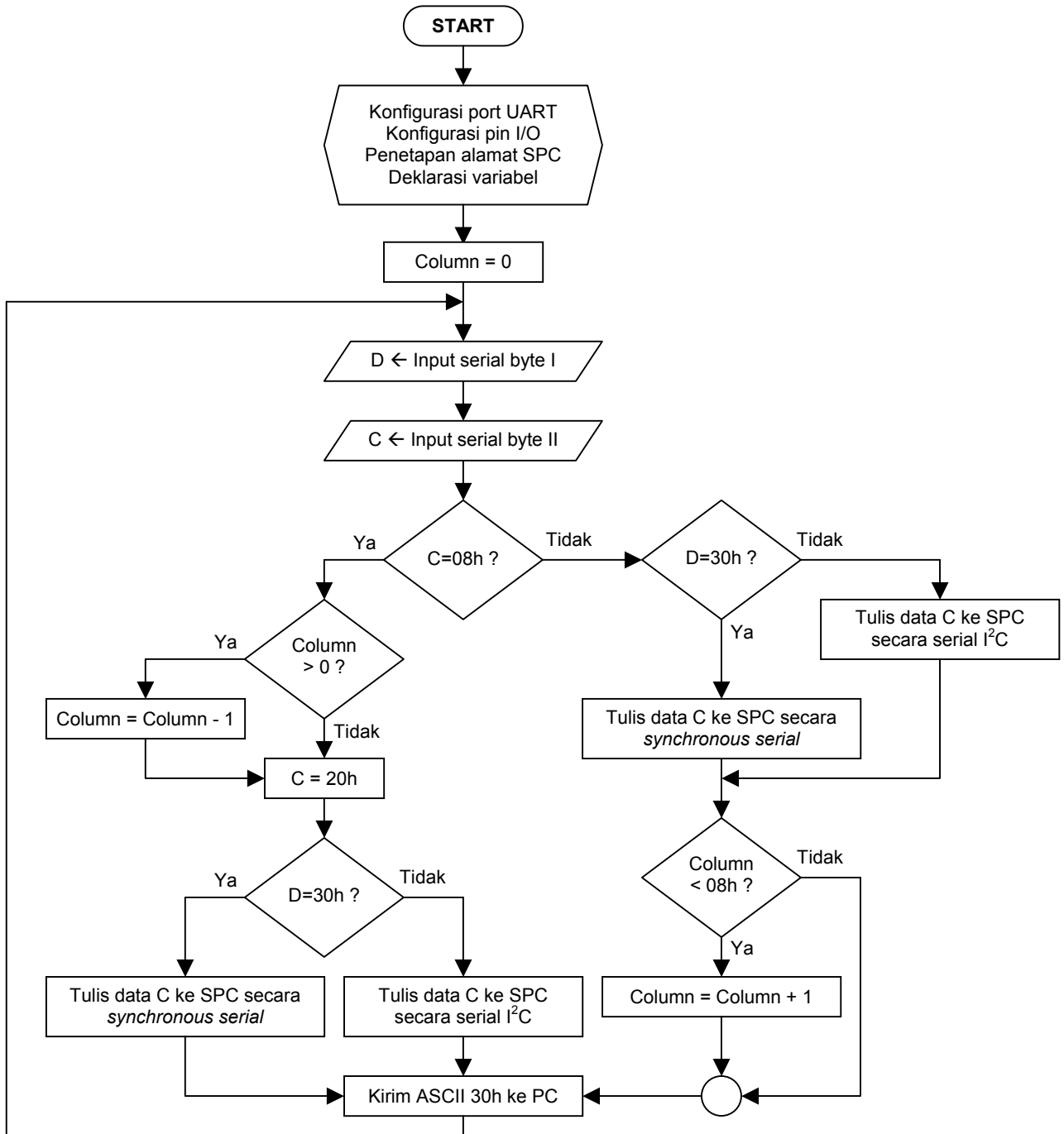
\$regfile = "2313def.dat" '→ untuk AT90S2313

\$regfile = "8535def.dat" '→ untuk AT90S8535

\$regfile = "m8535.dat" '→ untuk ATmega8535 (DT-AVR Low Cost Micro System)

agar sesuai dengan tipe mikrokontroler yang akan digunakan. Lalu *compile* ulang program sehingga menghasilkan file dengan ekstensi .bin atau .hex yang sesuai dan dapat diprogram ke dalam modul mikrokontroler.

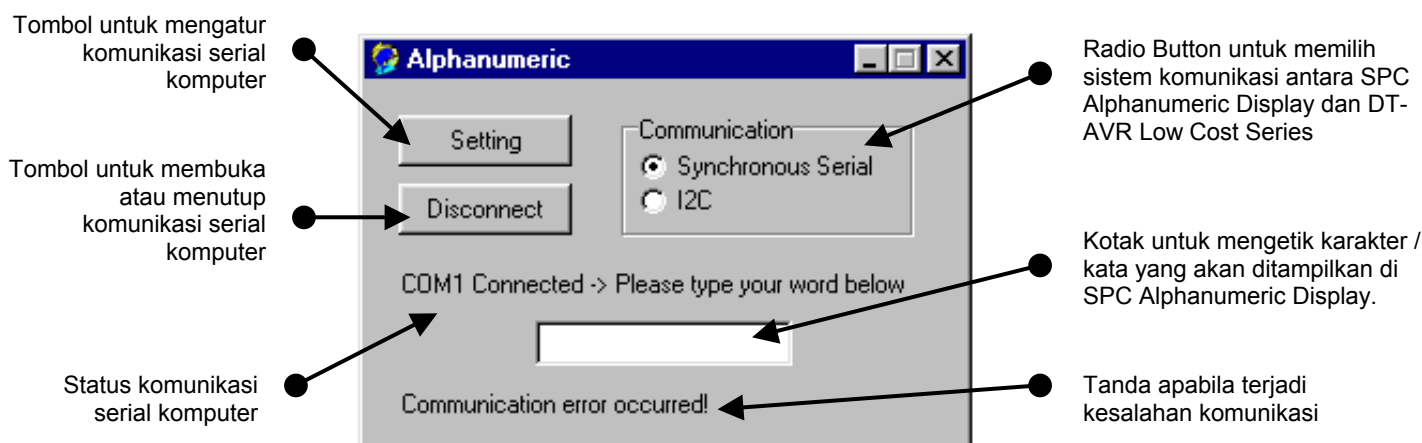
**F**lowchart program Alphe.bas adalah sebagai berikut:



**Gambar 2**  
**Flowchart Program Alphe.bas**

Program utama *Alphe.bas* akan diproses sebagai berikut:

1. Proses yang pertama dilakukan adalah konfigurasi port UART pada baudrate 9600 bps (8 bit data, 1 bit *stop*, dan tanpa bit *parity*).
2. Kemudian menentukan definisi pin I/O (SDA & SCL) untuk komunikasi I<sup>2</sup>C serta pin I/O (Serdata, Serclr, & Serclk) untuk komunikasi *synchronous serial*. Lalu penetapan alamat terprogram SPC yaitu 111b.
3. Kemudian deklarasi variabel Column, C, D, Dat, I, Sync\_addr, Sync\_column, I2c\_addr, dan I2c\_column. Column adalah variabel yang berisi posisi kolom dari tampilan *alphanumeric*. C adalah variabel yang berisi kode ASCII yang akan ditampilkan pada tampilan *alphanumeric*. D adalah variabel yang menentukan jenis komunikasi SPC yang akan digunakan. Dat adalah variabel I/O yang digunakan dalam pemanggilan prosedur *write\_sync\_byte*. I adalah variabel yang dipakai dalam proses pengulangan. Sync\_addr adalah variabel yang berisi alamat SPC untuk komunikasi *synchronous serial*. Sync\_column adalah variabel yang berisi posisi kolom dari tampilan *alphanumeric* yang telah disesuaikan dengan struktur komunikasi *synchronous serial*. I2c\_addr adalah variabel yang berisi alamat SPC untuk komunikasi I<sup>2</sup>C. I2c\_column adalah variabel yang berisi posisi kolom dari tampilan *alphanumeric* yang telah disesuaikan dengan struktur komunikasi I<sup>2</sup>C.
4. Lalu program menetapkan nilai awal untuk variabel Column adalah 0.
5. Pada tahap berikutnya program akan menunggu 2 byte data dari komputer. Byte pertama nantinya akan disimpan ke dalam variabel D dan akan menentukan jenis komunikasi SPC. Lalu byte kedua akan dimasukkan ke dalam variabel C dan merupakan data yang akan ditampilkan. Kedua byte data ini dapat dimasukkan melalui *keyboard* komputer menggunakan bantuan HyperTerminal© atau Terminal© (baudrate 9600 bps, 8 bit data, 1 bit *stop*, tanpa bit *parity*, dan tanpa *flow control*). Dalam **AN83.ZIP** telah disertakan pula program aplikasi komputer (Project1.exe) untuk mengirimkan urutan kedua byte data ini dari *keyboard* komputer ke rangkaian.



**Gambar 3**  
**Tampilan Program Project1.exe**

6. Lalu program akan memeriksa nilai data dalam variabel C, jika berisi 08h (*backspace*) maka program akan menuju ke langkah 7 tetapi jika tidak maka program akan melompat ke langkah 9.
7. Bila Column lebih besar dari nol maka Column akan dikurangi satu. Lalu program mengisi variabel C dengan data 20h (spasi).
8. Dan program akan mengirimkan data dari variabel C ke SPC secara *synchronous serial* bila variabel D berisi 30h, tetapi bila variabel D berisi nilai lain maka proses ini dilakukan secara I<sup>2</sup>C. Kemudian program menuju ke langkah 10.
9. Jika variabel D berisi 30h maka program akan mengirimkan data dari variabel C ke SPC secara *synchronous serial*, tetapi jika variabel D berisi nilai lain maka proses ini dilakukan secara I<sup>2</sup>C. Lalu, jika Column lebih kecil dari 08h maka program akan menambah variabel Column dengan 1.
10. Setelah itu program akan mengirim data berupa kode ASCII 30h ke komputer (bila menggunakan program Project1.exe maka hal ini berfungsi untuk mematikan timer dan menandakan tidak terjadi kesalahan / *time out*).
11. Kembali ke langkah 5.

Listing program terdapat pada **AN83.ZIP**.

**S**elamat berinovasi!

BASCOM-AVR is a copyright by MCS Electronics.  
HyperTerminal is a copyright by Hilgraeve Inc.  
Terminal is a copyright by Bray++.  
I<sup>2</sup>C is a registered trademark of Philips Semiconductors.