

DT-BASIC DT-BASIC *Application Note*

AN121 – BASIC Man Machine Interface

Oleh: Tim IE

Antarmuka antara manusia dengan peralatan atau *man machine interface* adalah suatu fitur penting yang harus dimiliki oleh sebagian besar peralatan elektronika. Dalam aplikasi berikut DT-BASIC Series akan menggunakan DT-I/O Low Cost Man Machine Interface sebagai sarana antarmuka dengan manusia sebagai operator. DT-BASIC Series akan menggunakan 2 tombol dan 8 digit tampilan seven segment pada DT-I/O Low Cost Man Machine Interface sebagai tampilan 8 digit *counter up* (pencacah naik). Aplikasi ini membutuhkan modul-modul berikut:

- 1 DT-BASIC Nano System / Micro System / Mini System.
- 1 DT-I/O Low Cost Man Machine Interface.

Adapun blok diagram sistem secara keseluruhan adalah sebagai berikut:



Gambar 1
Blok Diagram AN121

Hubungan antara modul-modul tersebut adalah sebagai berikut:

DT-BASIC Nano System (J2) / Micro System (J3) / Mini System (J1)	DT-I/O Low Cost Man Machine Interface
5VDC (pin 19)	VCC (J1 pin 2)
GND (pin 1 atau pin 2)	GND (J1 pin 1)
P0* (pin 3)	S1 (J2 pin 3)
P1* (pin 4)	S2 (J2 pin 4)
P2* (pin 5)	S3 (J2 pin 5)
P3* (pin 6)	S4 (J2 pin 6)
P4* (pin 7)	S5 (J2 pin 7)
P5* (pin 8)	S6 (J2 pin 8)
P6* (pin 9)	S7 (J2 pin 9)
P7* (pin 10)	S8 (J2 pin 10)
P8* (pin 11)	SI (J1 pin 3)
P9* (pin 12)	SCK (J1 pin 4)
P10* (pin 13)	RCK (J1 pin 5)
P11* (pin 14)	SCLR (J1 pin 6)
P12* (pin 15)	IN1 (J1 pin 7)
P13* (pin 16)	IN2 (J1 pin 8)

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

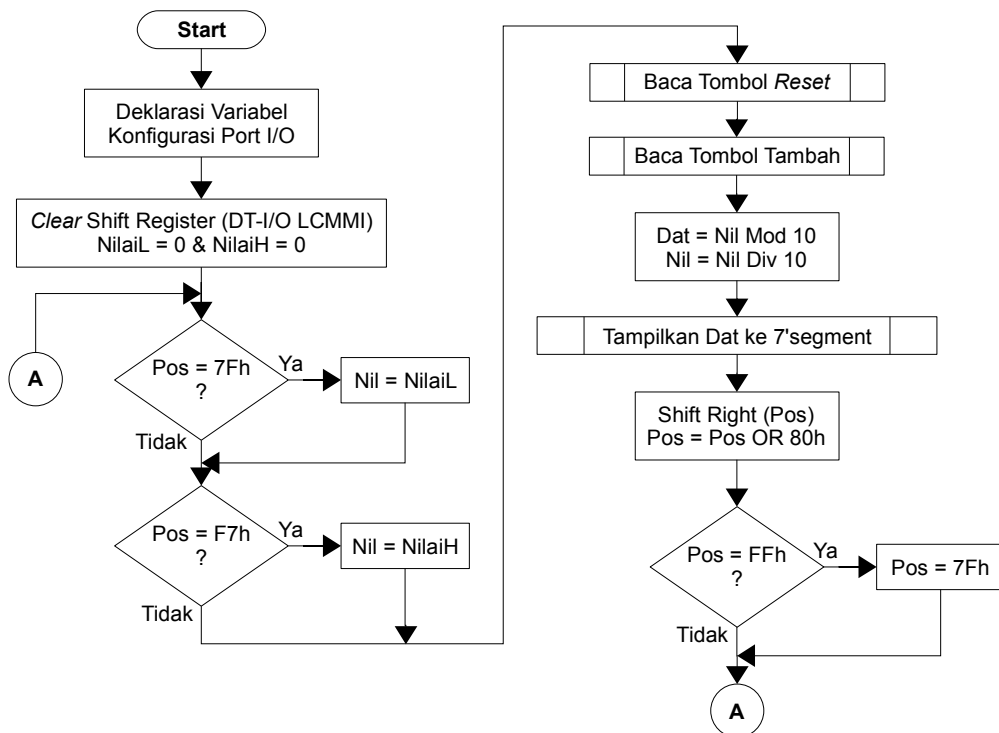
Tabel 1
Hubungan DT-BASIC Nano System / Micro System / Mini System
Dengan DT-I/O Low Cost Man Machine Interface

Gunakan kabel serial DT-BASIC Series untuk melakukan *download* kode program (MMI.bsp) ke DT-BASIC Nano System / Micro System / Mini System dengan bantuan *software* BASIC Stamp® Editor®. Bila menggunakan DT-BASIC Nano System atau DT-BASIC Micro System maka STAMP *directive* pada baris awal program MMI.bsp harus disesuaikan sebagai berikut:

- '{\$STAMP BS2p} diubah menjadi '{\$STAMP BS2} untuk DT-BASIC Nano System
- '{\$STAMP BS2p} diubah menjadi '{\$STAMP BS2sx} untuk DT-BASIC Nano System

Setelah STAMP *directive* pada program sesuai dengan DT-BASIC yang digunakan serta rangkaian dan sumber tegangan terhubung dengan benar maka kode program MMI.bsp (atau MMI.bs2 atau MMI.bsx jika telah dimodifikasi) dapat di-*download* ke DT-BASIC Nano System / Micro System / Mini System dengan menekan tombol Ctrl+R atau memilih *menu bar* Run → Run pada BASIC Stamp® Editor© IDE.

Flowchart program MMI.bsp adalah sebagai berikut:

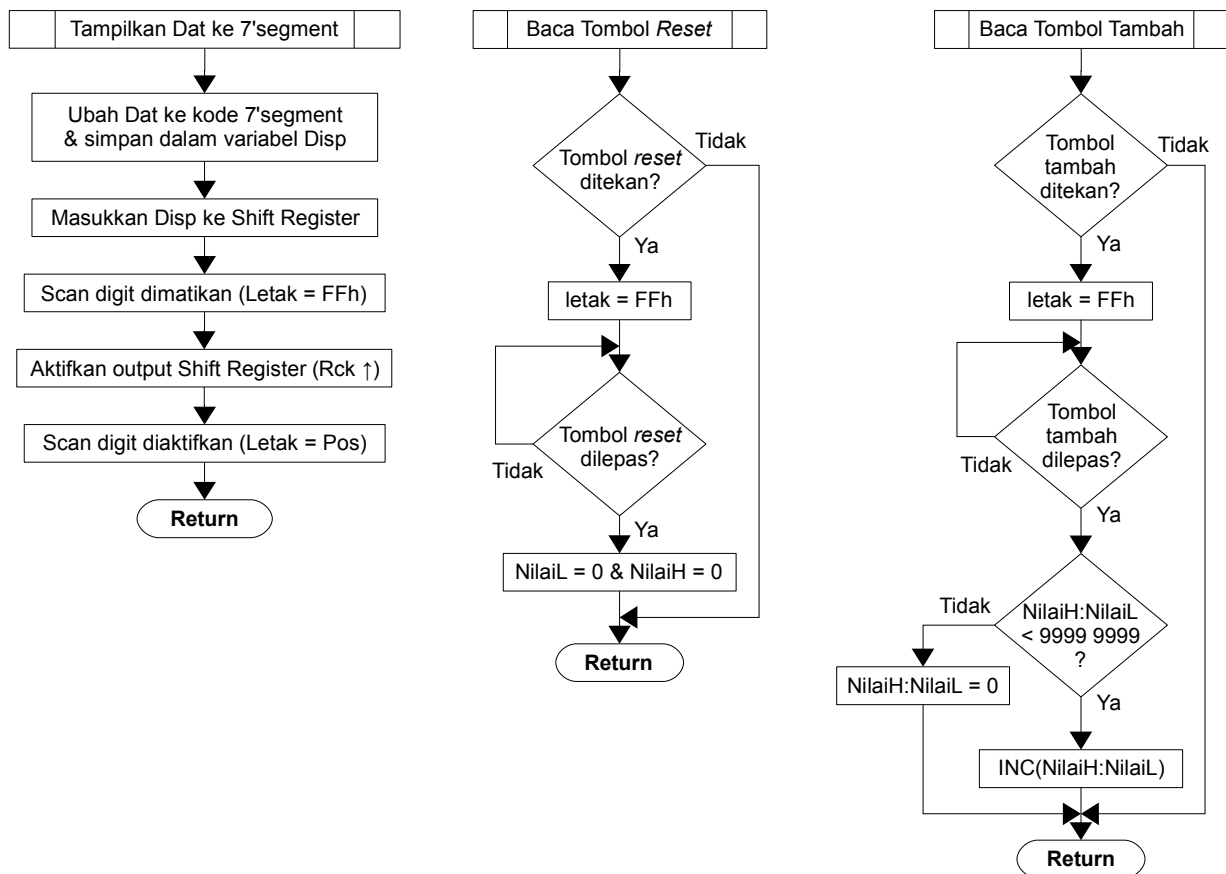


Gambar 2
Flowchart Program MMI.bsp Secara Garis Besar

Cara kerja program MMI.bsp secara garis besar adalah sebagai berikut:

1. Program melakukan deklarasi variabel:
 - Dat (bertipe Byte) untuk menampung kode BCD yang akan ditampilkan ke 7'segment.
 - Disp (bertipe Byte) untuk menampung kode 7'segment yang akan mengaktifkan 7'segment sesuai dengan data BCD dari variabel Dat.
 - Pos (bertipe Byte) untuk menampung kode *scanning* dari tampilan 8 digit 7'segment.
 - Nil (bertipe Word) untuk menampung 4 digit (LSB atau MSB) dari nilai *counter* yang akan ditampilkan ke 7'segment.
 - NilaiL (bertipe Word) untuk menampung 4 digit LSB dari nilai *counter*.
 - NilaiH (bertipe Word) untuk menampung 4 digit MSB dari nilai *counter*.
2. Program melakukan konfigurasi port I/O DT-BASIC dan menetapkan fungsi masing-masing pin (sebagai *output* atau *input*).

3. Melakukan inialisasi nilai awal *counter*, variabel NilaiH dan NilaiL diisi dengan nilai nol. Nilai *counter* dipisah menjadi dua variabel (NilaiH dan NilaiL) karena PBASIC tidak memiliki tipe variabel yang lebih panjang dari tipe Word (16-bit).
4. Bila posisi digit 7'segment yang akan di-*scanning* adalah digit 1 (Pos=7Fh) maka program mengkopi variabel NilaiL ke variabel Nil. Sedangkan bila posisi digit 7'segment yang akan di-*scanning* adalah digit 5 (Pos=F7h) maka program mengkopi variabel NilaiH ke variabel Nil.
5. Program membaca kondisi tombol *reset* (KEY2). Bila ada penekanan tombol maka program akan mengisi variabel NilaiH dan NilaiL dengan nilai nol.
6. Program membaca kondisi tombol tambah (KEY1). Bila ada penekanan tombol maka program akan memeriksa variabel NilaiH:NilaiL. Bila variabel NilaiH:NilaiL lebih kecil dari 9999 9999 maka program akan menambah variabel NilaiH:NilaiL dengan 1. Sedangkan bila variabel NilaiH:NilaiL tidak lebih kecil dari 9999 9999 maka variabel NilaiH dan NilaiL akan diisi dengan nilai nol.
7. Variabel Dat diisi dengan nilai satuan desimal dari variabel Nil (Nil MOD 10). Lalu variabel Nil dibagi 10.
8. Program mengisi variabel Disp dengan kode 7'segment yang akan digunakan untuk mengaktifkan tampilan 7'segment sesuai dengan data BCD dari variabel Dat.
9. Program mengaktifkan digit 7'segment yang sedang di-*scan* sesuai dengan data dari variabel Disp.
10. Menggeser kanan variabel Pos untuk *scan* digit berikutnya. Bila hasil dari operasi geser kanan ini adalah FFh maka variabel Pos akan diisi dengan 7Fh (kembali *scan* digit 1).
11. Program kembali ke langkah 4.



Gambar 3
Flowchart Program MMI.bsp Bagian Scan 7'Segment dan Baca Tombol

Program MMI.bsp dalam aplikasi AN121 ini memberikan contoh cara mengakses port *output* DT-BASIC secara paralel 8-bit. Penulisan port *output* 8-bit dapat dilakukan dengan menulis RAM bernama OUTL (untuk pin P0 – P7) atau OUTH (untuk pin P8 – P15). Sedangkan pembacaan port *input* 8-bit dapat dilakukan dengan membaca RAM bernama INL (untuk pin P0 – P7) atau INH (untuk pin P8 – P15). Akses port secara paralel ini sangat dibutuhkan

dalam aplikasi-aplikasi yang menggunakan modul / komponen yang bekerja dalam sistem bus mikrokontroler / mikroprosesor, seperti: LCD, ADC, dan sebagainya.

Listing program terdapat pada **AN121.ZIP**.

Selamat berinovasi!

All trademarks, trade names, company names, and product names are the property of their respective owners.
All softwares are copyright by their respective software publishers and/or creators.