

DT-BASIC

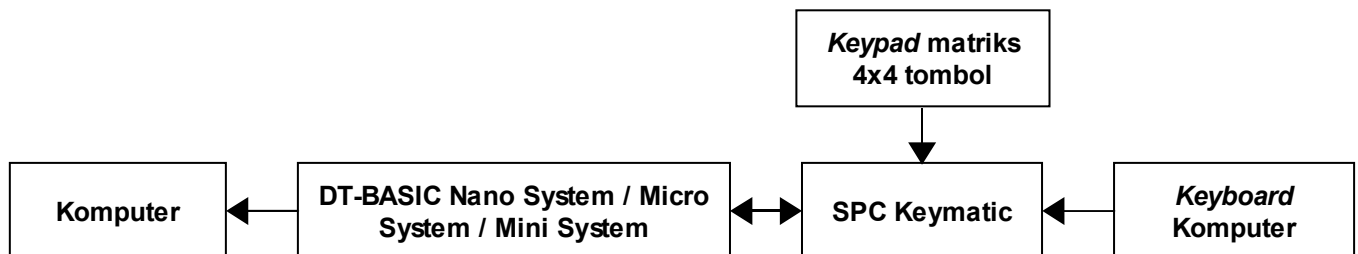
DT-BASIC *Application Note* AN124 – BASIC Keymatic

Oleh: Tim IE

Artikel ini membahas penggunaan SPC Keymatic bersama dengan modul DT-BASIC Nano System. Di sini SPC Keymatic akan bekerja pada mode komunikasi paralel maupun I²C. SPC Keymatic akan menerima *input* dari *keypad* matriks 4 x 4 tombol dan *keyboard* komputer, lalu DT-BASIC Nano System akan menampilkan data tombol pada jendela Debug Terminal BASIC Stamp® Editor®. AN ini dapat menjadi salah satu referensi tentang bagaimana cara membangun rutin komunikasi I²C menggunakan bahasa PBASIC™ untuk DT-BASIC Nano System / Micro System. Modul atau komponen yang diperlukan dalam aplikasi ini adalah:

- 1 DT-BASIC Nano System (dapat juga menggunakan DT-BASIC Micro System / Mini System),
- 1 SPC Keymatic,
- 1 *Keypad* matriks 4x4 tombol,
- 1 *Keyboard* komputer PS/2.

Adapun blok diagram sistem secara keseluruhan adalah sebagai berikut:



Gambar 1
Blok Diagram AN124

Hubungan antara modul-modul tersebut adalah sebagai berikut:

DT-BASIC Nano System	SPC Keymatic
GND (J4)	GND (J5)
5VDC (J4)	+5V (J5)
P0 (J2 Pin 3)*	Data bit 0 (J4 Pin 3)
P1 (J2 Pin 4)*	Data bit 1 (J4 Pin 4)
P2 (J2 Pin 5)*	Data bit 2 (J4 Pin 5)
P3 (J2 Pin 6)*	Data bit 3 (J4 Pin 6)
P4 (J2 Pin 7)*	RD (J4 Pin 1)
P5 (J2 Pin 8)*	INT (J4 Pin 2)

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

Tabel 1
Hubungan DT-BASIC Nano System dengan SPC Keymatic Secara Paralel

Catatan: Penekanan tombol pada *keyboard* komputer yang terhubung ke SPC Keymatic tidak dapat dibaca oleh DT-BASIC Nano System dalam konfigurasi hubungan paralel ini. Hal ini sesuai dengan spesifikasi SPC Keymatic. Pada mode komunikasi paralel ini hanya penekanan tombol *keypad* yang dapat terbaca oleh DT-BASIC Nano System dan diteruskan ke komputer.

DT-BASIC Nano System	SPCKeymatic
GND (J4)	GND (J5)
5VDC (J4)	+5V (J5)
P6 (J2 Pin 9)*	SDA (I ² C Bus)
P7 (J2 Pin 10)*	SCL (I ² C Bus)
P5 (J2 Pin 8)*	INT (J4 Pin 2)

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

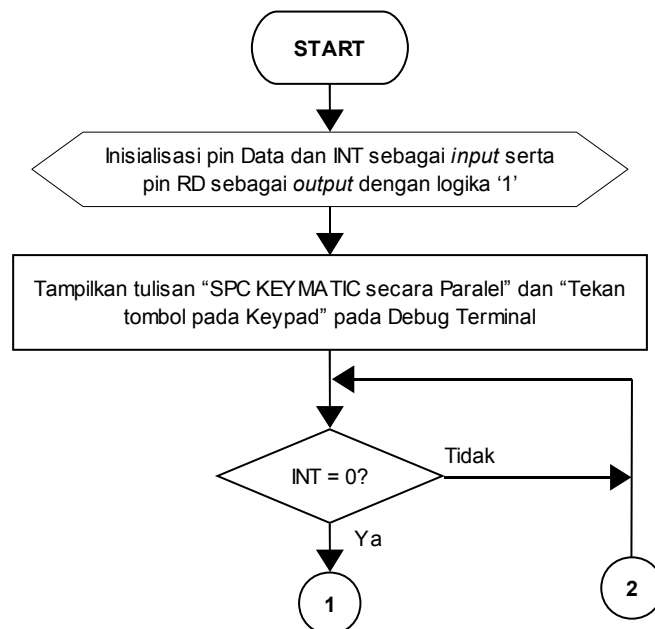
Tabel 2
Hubungan DT-BASIC Nano System dengan SPC Keymatic Secara Serial I²C

Hubungkan *keyboard* komputer ke konektor J8 (PS/2 *female*) pada SPC Keymatic, serta hubungkan *keypad* matriks 4x4 ke konektor J7 pada SPC Keymatic. Keterangan yang lebih detail mengenai susunan tombol *keypad* dan cara menghubungkannya ke SPC Keymatic terdapat dalam manual SPC Keymatic. Pasang semua *jumper* pada J3 (Address A1-A0), serta semua *jumper* pada SDA dan SCL (J2). Alamat I²C modul SPC Keymatic yang digunakan dalam aplikasi ini adalah E8H.

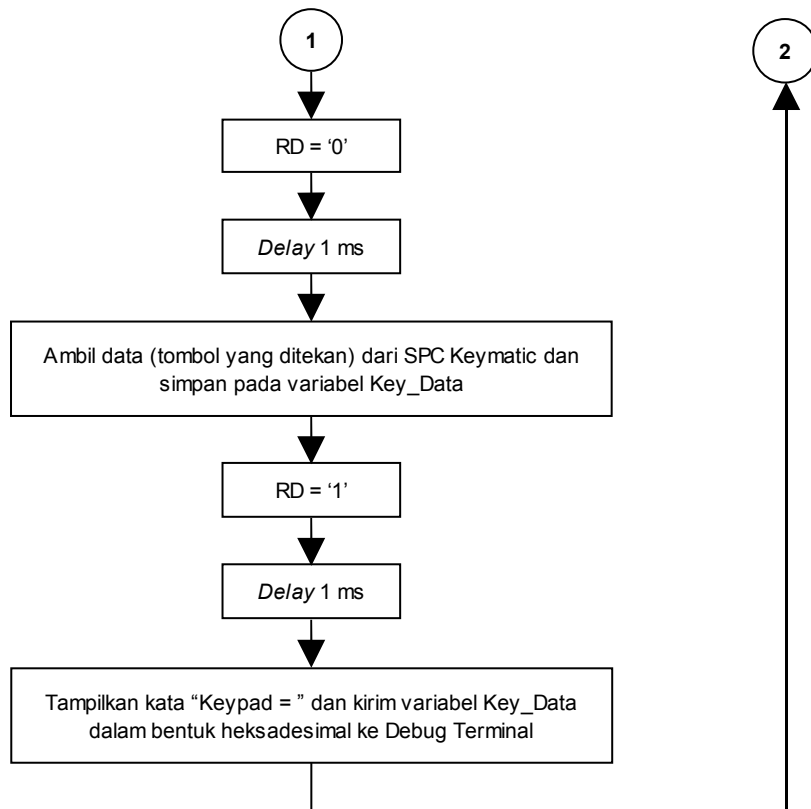
Gunakan kabel serial DT-BASIC Series untuk menghubungkan modul DT-BASIC Nano System / Micro System / Mini System ke *COM port* komputer. Kabel ini nantinya digunakan untuk melakukan *download* program dan untuk mengirimkan data penekanan tombol *keyboard/keypad* ke komputer. Setelah semua rangkaian dan sumber tegangan terhubung dengan benar, programlah KEY_PARALEL.BS2 (untuk komunikasi secara paralel) atau KEY_I2C.BS2 (untuk komunikasi secara I²C) ke DT-BASIC Nano System dengan menekan tombol **F9** atau **CTRL+R** pada BASIC Stamp[®] Editor[®], jendela Debug Terminal akan muncul secara otomatis setelah *download* program selesai. *STAMP directive* pada baris pertama *listing* program KEY_PARALEL.BS2 (atau KEY_I2C.BS2) harus disesuaikan jika menggunakan modul DT-BASIC Series lainnya, yaitu menjadi:

- '\$STAMP BS2sx' untuk DT-BASIC Micro System, atau
- '\$STAMP BS2p' untuk DT-BASIC Mini System.

Flowchart dari program KEY_PARALEL.BS2 adalah sebagai berikut:



Gambar 2
Flowchart Program KEY_PARALEL.BS2 untuk Komunikasi Paralel (Bagian 1)

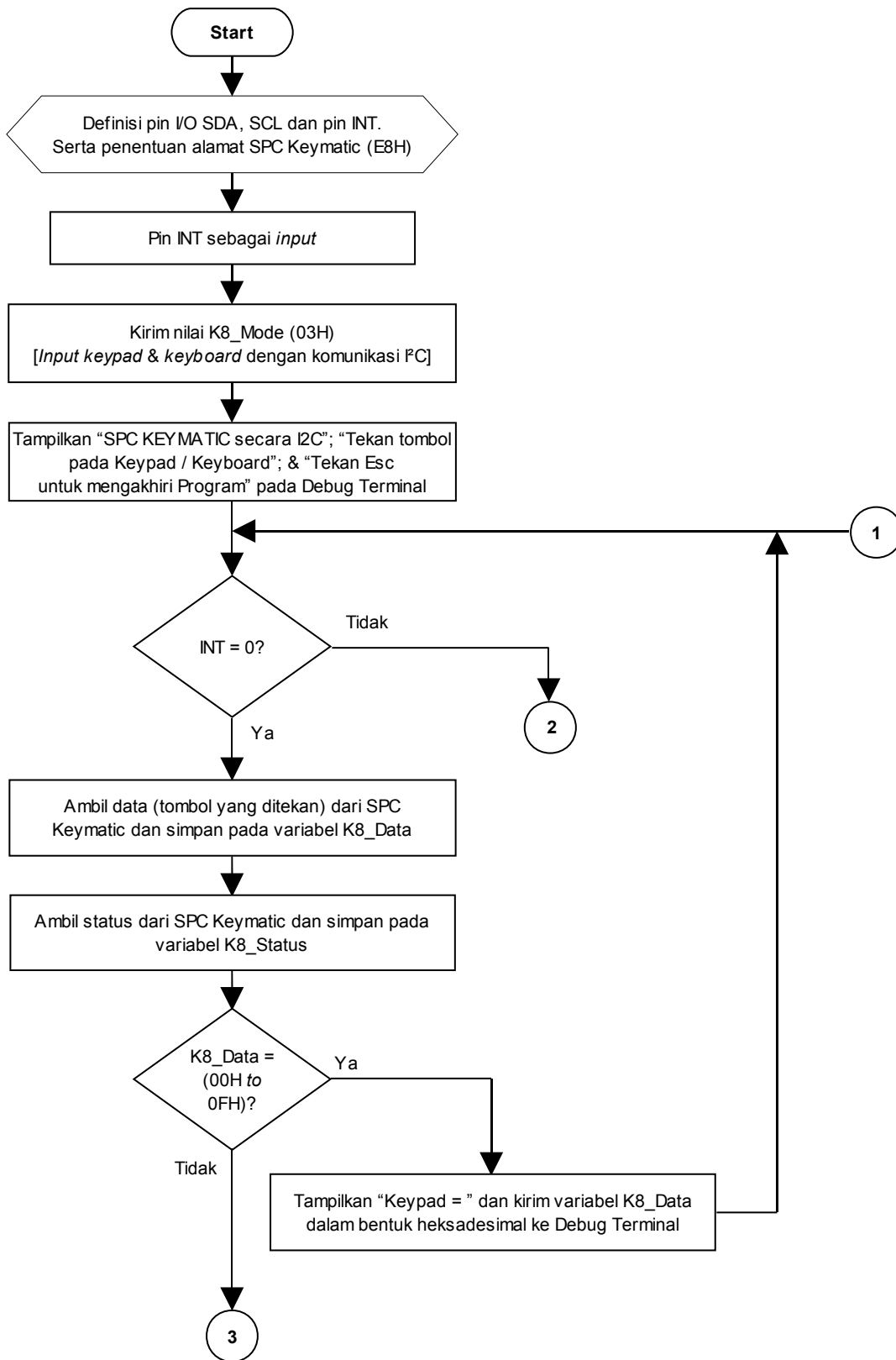


Gambar 3
Flowchart Program KEY_PARALEL.BS2 untuk Komunikasi Paralel (Bagian 2)

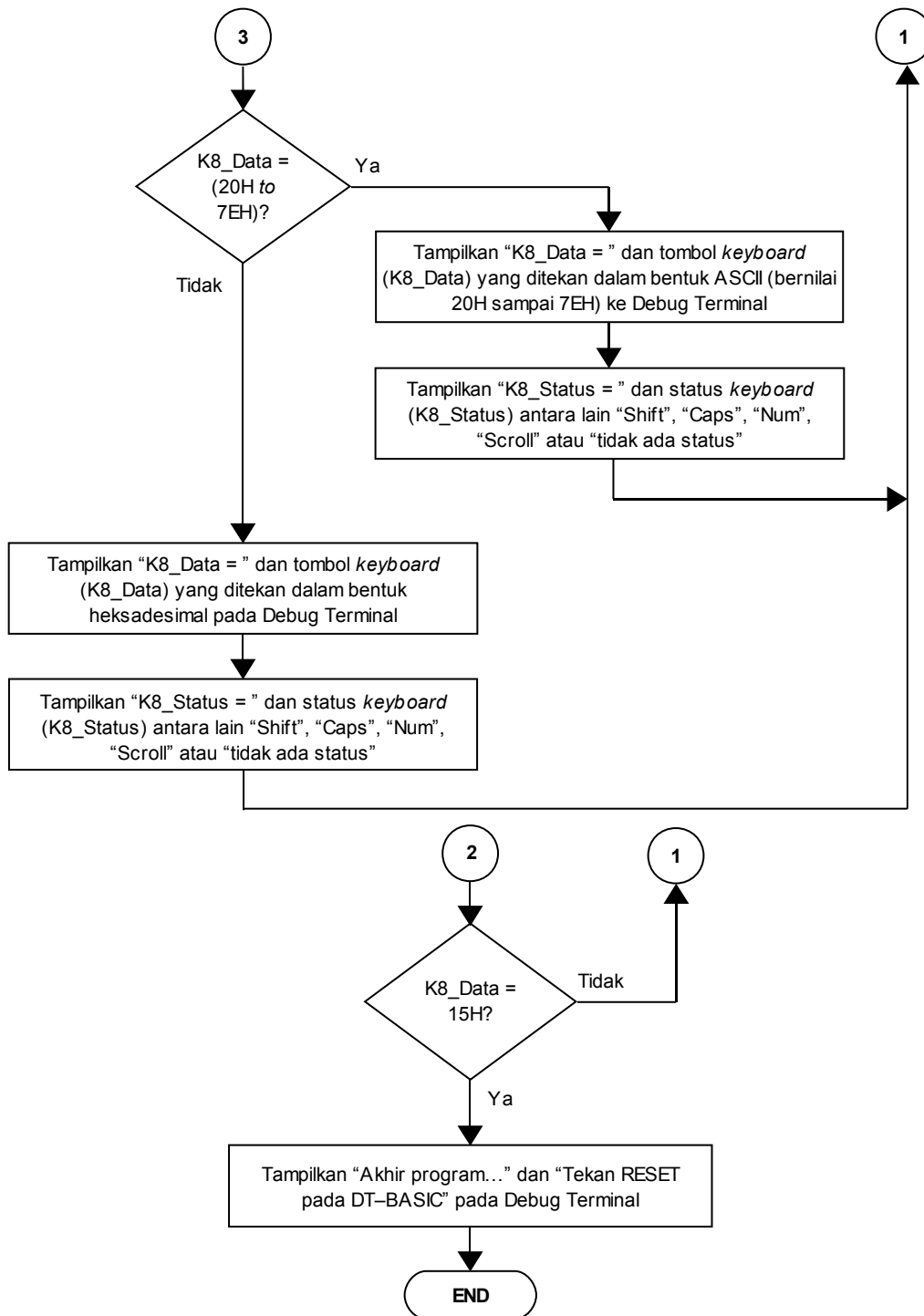
Program utama (KEY_PARALEL.BS2) akan diproses sebagai berikut:

1. Proses yang pertama kali dilakukan adalah konfigurasi pin Data dan INT sebagai *input* serta pin RD sebagai *output*. Pada pin RD diberi logika '1'.
2. Setelah proses konfigurasi pin selesai, berikutnya program akan mengirimkan tulisan "SPC KEYMATIC secara Paralel" dan tulisan "Tekan tombol pada Keypad" untuk ditampilkan pada Debug Terminal.
3. Kemudian program akan membaca kondisi logika pada pin INT. Jika pin INT berlogika '1' maka program akan kembali memeriksa logika pada pin INT. Jika pin INT berlogika '0' maka pin RD akan diberi logika '0' sebagai sinyal untuk membaca data pada SPC Keymatic.
4. Setelah langkah pada nomor 3 selesai, program akan berhenti selama 1 ms (*delay*) sebelum data dari SPC Keymatic siap untuk dibaca.
5. Setelah itu, program akan membaca Data dari SPC Keymatic dan menyimpan hasil pembacaan ke variabel Key_Data. Program akan berhenti lagi selama 1 ms untuk memberi kesempatan pada pin INT kembali ke logika '1'. Kondisi logika '1' pada INT menunjukkan bahwa data pada SPC Keymatic telah dibaca semua (*buffer* SPC Keymatic sudah tidak berisi data lagi).
6. Proses berikutnya adalah mengirimkan kata "Keypad =" dan variabel Key_Data dalam bentuk heksadesimal untuk ditampilkan pada jendela Debug Terminal.
7. Setelah pengiriman hasil penekanan tombol *keypad* (Key_Data) ke Debug Terminal selesai maka program akan kembali lagi ke langkah pada nomor 3.

Flowchart dari program KEY_I2C.BS2 adalah sebagai berikut:



Gambar 4
Flowchart Program KEY_I2C.BS2 untuk Komunikasi I²C (Bagian 1)



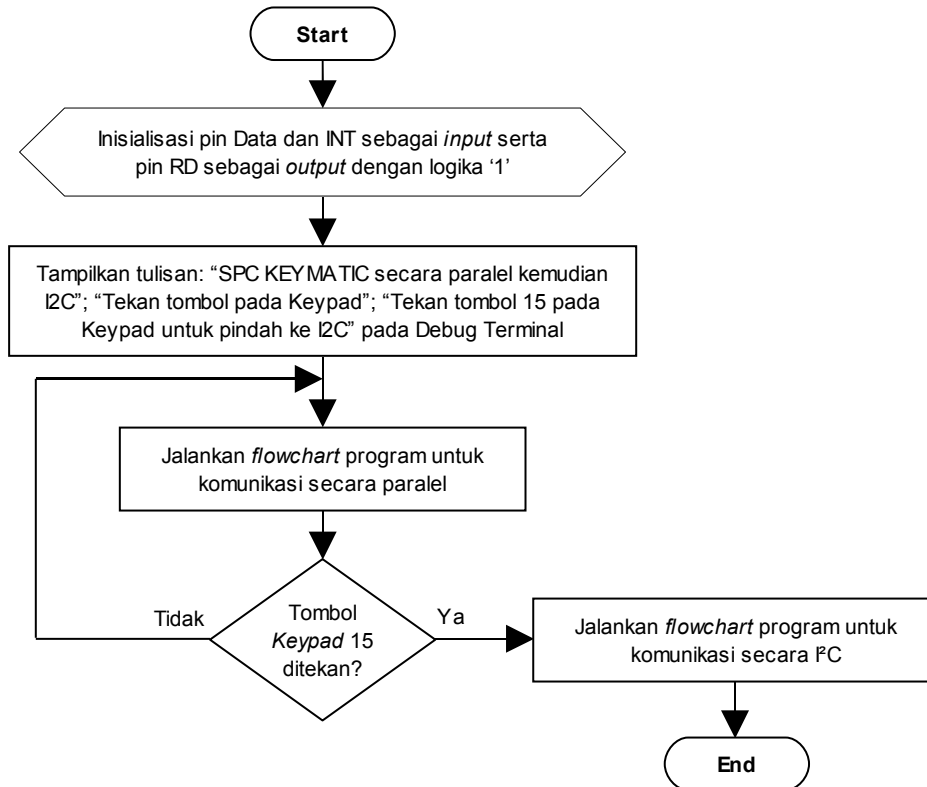
Gambar 5
Flowchart Program KEY_I2C.BS2 untuk Komunikasi I²C (Bagian 2)

Program utama (KEY_I2C.BS2) akan diproses sebagai berikut:

1. Proses yang pertama kali dilakukan adalah menentukan definisi pin I/O SDA dan SCL untuk komunikasi serial I²C serta pin INT sebagai sinyal *input* dari SPC Keymatic (akan berlogika '0' jika ada tombol yang ditekan). Dan membuat konstanta untuk alamat SPC Keymatic yaitu pada alamat E8H.
2. Proses selanjutnya yaitu mengirimkan nilai K8_Mode sebesar 03H yang mempunyai arti *input* SPC Keymatic berasal dari *keypad* dan *keyboard*. Jika pada saat pengiriman nilai K8_Mode tidak mendapat tanggapan dari SPC Keymatic (sinyal *no acknowledge*) maka program akan mengirimkan pesan kesalahan ke jendela Debug Terminal.

3. Setelah pengiriman nilai K8_Mode selesai, program mengirimkan tulisan "SPC KEYMATIC secara I2C" (diikuti pengiriman CR / ENTER), tulisan "Tekan tombol pada Keypad / Keyboard" (diikuti pengiriman CR) dan tulisan "Tekan Esc untuk mengakhiri Program" (diikuti pengiriman CR / ENTER) ke jendela Debug Terminal. Pengiriman tulisan ini hanya digunakan sebagai informasi untuk pemakai.
 4. Proses selanjutnya yaitu membaca kondisi logika pada pin INT. Jika pin INT berlogika '1' maka program akan membandingkan variabel K8_Data dengan nilai 15H (tombol "Esc" pada *keyboard*), jika nilainya sama maka program akan berhenti (menampilkan tulisan "Akhir program..." dan "Tekan RESET pada DT-BASIC" ke Debug Terminal) sebaliknya jika nilainya tidak sama maka program akan kembali memeriksa logika pada pin INT. Jika pin INT berlogika '0' maka program akan mengerjakan langkah nomor 5.
 5. Kemudian program akan membaca data dari SPC Keymatic dan menyimpan hasil pembacaan ke variabel K8_Data serta membaca status dari SPC Keymatic dan menyimpan hasil pembacaan ke variabel K8_Status.
 6. Proses berikutnya adalah membandingkan nilai K8_Data. Ada 3 kategori perbandingan nilai yang digunakan dalam contoh ini, masing-masing akan menampilkan hasil yang berbeda pada jendela Debug Terminal. Tiga kategori tersebut yaitu:
 - ◆ Jika nilai K8_Data antara 00H sampai 0FH maka program akan menampilkan "Keypad = " dan mengirimkan variabel K8_Data dalam bentuk heksadesimal ke jendela Debug Terminal.
 - ◆ Jika nilai K8_Data antara 20H sampai 7EH maka program akan menampilkan "K8_Data = " dan mengirimkan variabel K8_Data dalam bentuk ASCII (sehingga tampil karakter sesuai tombol yang ditekan pada *keyboard*) ke jendela Debug Terminal. Kemudian juga menampilkan "K8_Status = " dan mengirimkan status *keyboard* sesuai dengan status tombol-tombol khusus pada *keyboard* antara lain "Shift", "Caps", "Num", "Scroll" atau "tidak ada status".
 - ◆ Jika nilai K8_Data di luar nilai di atas (tidak termasuk dalam nilai 00H sampai 0FH maupun nilai 20H sampai 7EH) maka program akan menampilkan kata "K8_Data = " dan mengirimkan variabel K8_Data dalam bentuk heksadesimal ke Debug Terminal. Kemudian juga menampilkan kata "K8_Status = " dan mengirimkan status *keyboard* sesuai dengan status tombol-tombol khusus pada *keyboard*.
- Setelah mengirimkan hasil ke jendela Debug Terminal sesuai dengan kategori nilai K8_Data maka program akan kembali ke langkah nomor 4.

Flowchart dari program BASIC_KEY.BS2 adalah sebagai berikut:



Gambar 6
Flowchart Program BASIC_KEY.BS2 untuk Komunikasi Paralel dan I2C

Program utama (BASIC_KEY.BS2) akan diproses sebagai berikut:

1. Proses yang pertama kali dilakukan adalah konfigurasi pin Data dan INT sebagai *input* serta pin RD sebagai *output*. Pada pin RD diberi logika '1'.
2. Setelah itu, program mengirimkan tulisan "SPC KEYMATIC secara Paralel kemudian I2C" (diikuti pengiriman CR / ENTER), tulisan "Tekan tombol pada Keypad" (diikuti pengiriman CR) dan tulisan "Tekan tombol 15 pada Keypad untuk pindah ke I2C" (diikuti pengiriman CR / ENTER) ke Debug Terminal. Pengiriman tulisan ini hanya digunakan sebagai informasi untuk pemakai.
3. Program melakukan pengambilan data penekanan tombol *keypad* sesuai dengan alur program untuk komunikasi paralel mulai langkah no 3.
4. Program memeriksa apakah tombol *keypad* yang ditekan adalah tombol 15. Jika tombol 15 yang ditekan maka program akan lanjut ke langkah 5, tetapi jika tidak maka program kembali ke langkah 3.
5. Program melakukan proses seperti alur program untuk komunikasi I²C. SPC Keymatic perlu di-*reset* untuk kembali ke komunikasi paralel (langkah 1). *Reset* ini dapat dilakukan dengan cara mematikan seluruh rangkaian kemudian menghidupkannya kembali.

Listing program KEY_PARALEL.BS2, KEY_I2C.BS2, dan BASIC_KEY.BS2 terdapat pada **AN124.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.