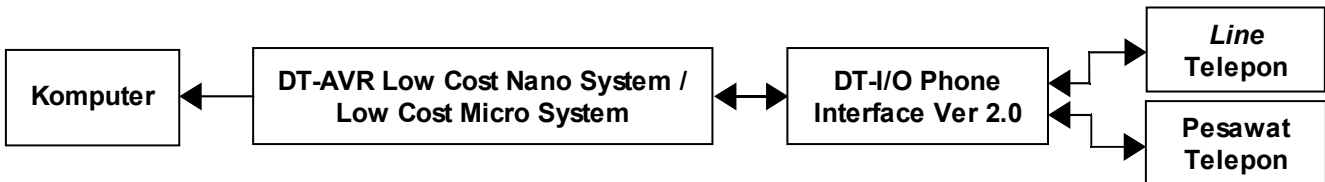


Artikel ini akan memberikan berbagai contoh penerapan DT-I/O Phone Interface Ver 2.0 yang dikendalikan oleh DT-AVR Low Cost Series menggunakan bahasa pemrograman C dengan bantuan *compiler* CodeVisionAVR®. Di sini DT-I/O Phone Interface Ver 2.0 akan digunakan untuk mendeteksi kondisi *hook*, mendeteksi sinyal dering, mendeteksi *dial tone*, membaca sinyal DTMF/*pulse*, melakukan *dialing* DTMF/*pulse*, dan mendeteksi *call progress*.

Aplikasi ini membutuhkan modul/komponen berikut:

- 1 unit DT-AVR Low Cost Nano System dengan AT90S2313 (dapat juga menggunakan ATtiny2313 ataupun DT-AVR Low Cost Micro System),
- 1 unit DT-I/O Phone Interface Ver 2.0 dan sebuah pesawat telepon.

Adapun blok diagram sistem secara keseluruhan adalah sebagai berikut:



Gambar 1
Blok Diagram AN123

Hubungan antara modul-modul tersebut adalah sebagai berikut:

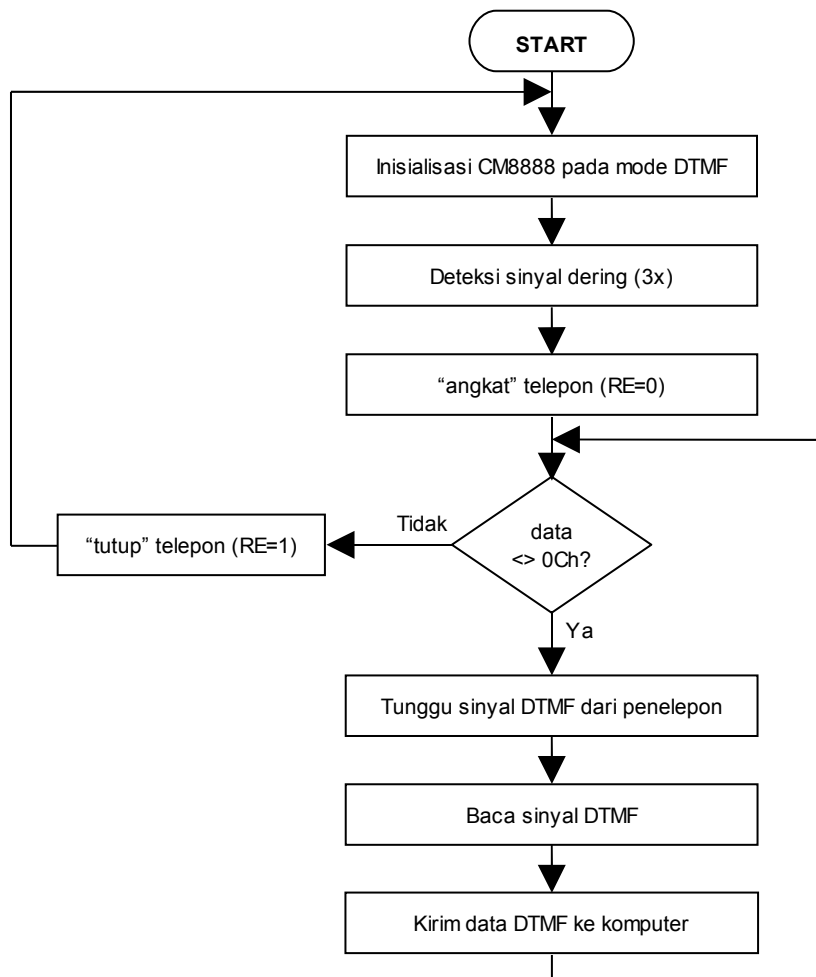
DT-AVR Low Cost Nano System	DT-I/O Phone Interface Ver 2.0
GND (J2)	DGND (J7 pin 1 ATAU J2 pin 2)
+5VDC (J2)	DVCC (J7 pin 2 ATAU J2 pin 1)
PB0 (J7 pin 3)*	RD (J2 pin 8)
PB1 (J7 pin 4)*	WR (J2 pin 7)
PB2 (J7 pin 5)*	RE (J3 pin 11)
PB3 (J7 pin 6)*	RI (J3 pin 9)
PB4 (J7 pin 7)*	D0 (J1 pin 1)
PB5 (J7 pin 8)*	D1 (J1 pin 2)
PB6 (J7 pin 9)*	D2 (J1 pin 3)
PB7 (J7 pin 10)*	D3 (J1 pin 4)
PD2 (J8 pin 5)*	IRQ (J2 pin 3)
PD3 (J8 pin 6)*	HK (J3 pin 10)
PD4 (J8 pin 7)*	T0 (J2 pin 5)
PD5 (J8 pin 8)*	RS0 (J3 pin 12)
PD6 (J8 pin 9)*	CS (J1 pin 9)

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

Tabel 1
Hubungan DT-AVR Low Cost Nano System dengan DT-I/O Phone Interface Ver 2.0

Gunakan kabel serial DT-AVR Low Cost Series untuk menghubungkan DT-AVR Low Cost Nano System ke *COM port* komputer. Kabel serial ini digunakan untuk mengirimkan data dari DT-AVR Low Cost Nano System ke komputer untuk ditampilkan pada program Terminal[®] atau HyperTerminal[®]. Sebelumnya atur dulu program Terminal[®] atau HyperTerminal[®] pada *baudrate* 9600bps, 8 bit data, tanpa bit *parity*, 1 bit stop, dan tanpa *flow control*. Setelah semua rangkaian dan sumber tegangan terhubung dengan benar, bukalah *project* "Phone Interface.prj" pada CodeVisionAVR[®] dan *copy* salah satu contoh program (TST1.c, TST2.c, TST3.c, TST4.c, TST5.c, TST5B.c, TST6.c, atau TST9.c) yang akan dicoba ke dalam Program.c. Kemudian lakukan *compile project* dan *download*-lah hasilnya ke DT-AVR Low Cost Nano System menggunakan bantuan DT-HiQ AVR In System Programmer.

Flowchart program TST1.c



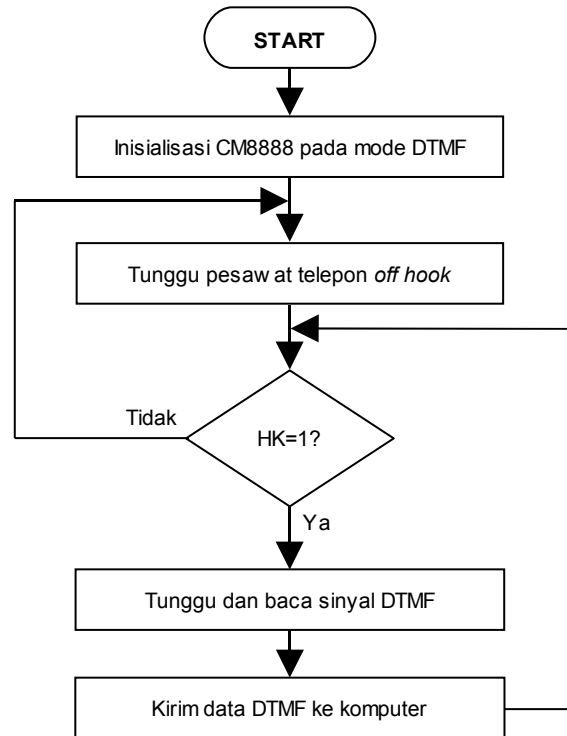
Gambar 2
Flowchart Program TST1.c

TST1.c bekerja dengan DT-I/O Phone Interface Ver 2.0 yang diatur pada mode 2 (*jumper mode* terpasang) dan tanpa bantuan pesawat telepon yang terhubung ke DT-I/O Phone Interface Ver 2.0. Cara kerja program ini adalah sebagai berikut:

1. Program melakukan inisialisasi CM8888 pada mode DTMF dan menunggu sinyal dering.
2. Jika sudah terdeteksi 3 kali sinyal dering maka program akan "mengangkat" (*pick up*, RE=0) telepon.
3. Program memeriksa apakah variabel data (yang menampung data DTMF) berisi 0Ch (tombol "#"). Jika data=0Ch maka program akan menutup telepon (*hang up*, RE=1) dan kembali ke langkah 1, tetapi jika data<>0Ch maka program akan lanjut ke langkah berikutnya.
4. Program menunggu sinyal DTMF dari penelepon (sisi *remote*). Sinyal DTMF ini muncul saat penelepon menekan tombol pada pesawat teleponnya.

5. Jika ada sinyal DTMF yang diterima maka data DTMF tersebut dikirimkan ke komputer, data DTMF yang diterima ini dapat dilihat melalui program Terminal[®] atau HyperTerminal[®].
6. Program kembali ke langkah 3.

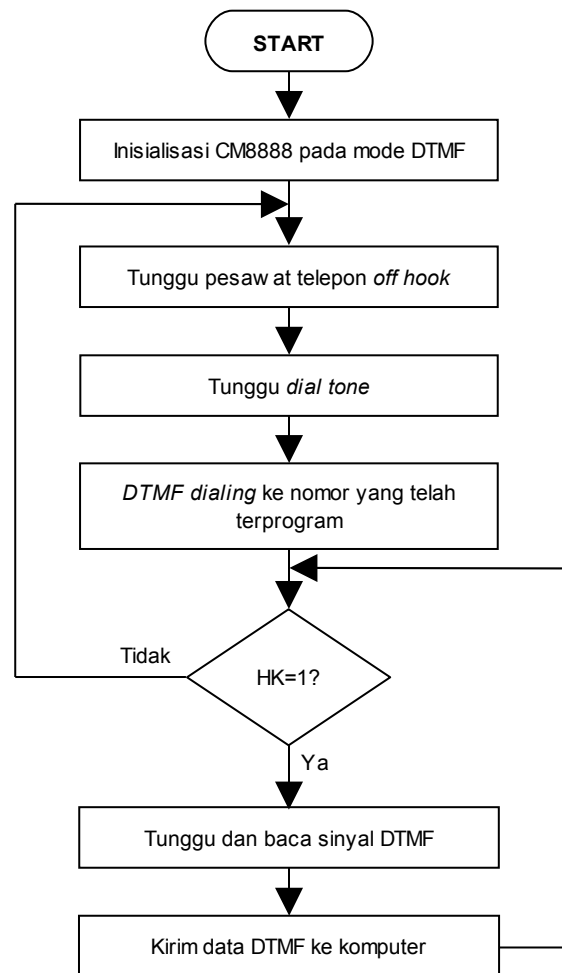
Flowchart program TST2.c



Gambar 3
Flowchart Program TST2.c

TST2.c bekerja dengan DT-I/O Phone Interface Ver 2.0 yang diatur pada mode 1 (*jumper mode tidak terpasang*) dan dengan bantuan pesawat telepon yang terhubung ke DT-I/O Phone Interface Ver 2.0. Cara kerja program ini adalah sebagai berikut:

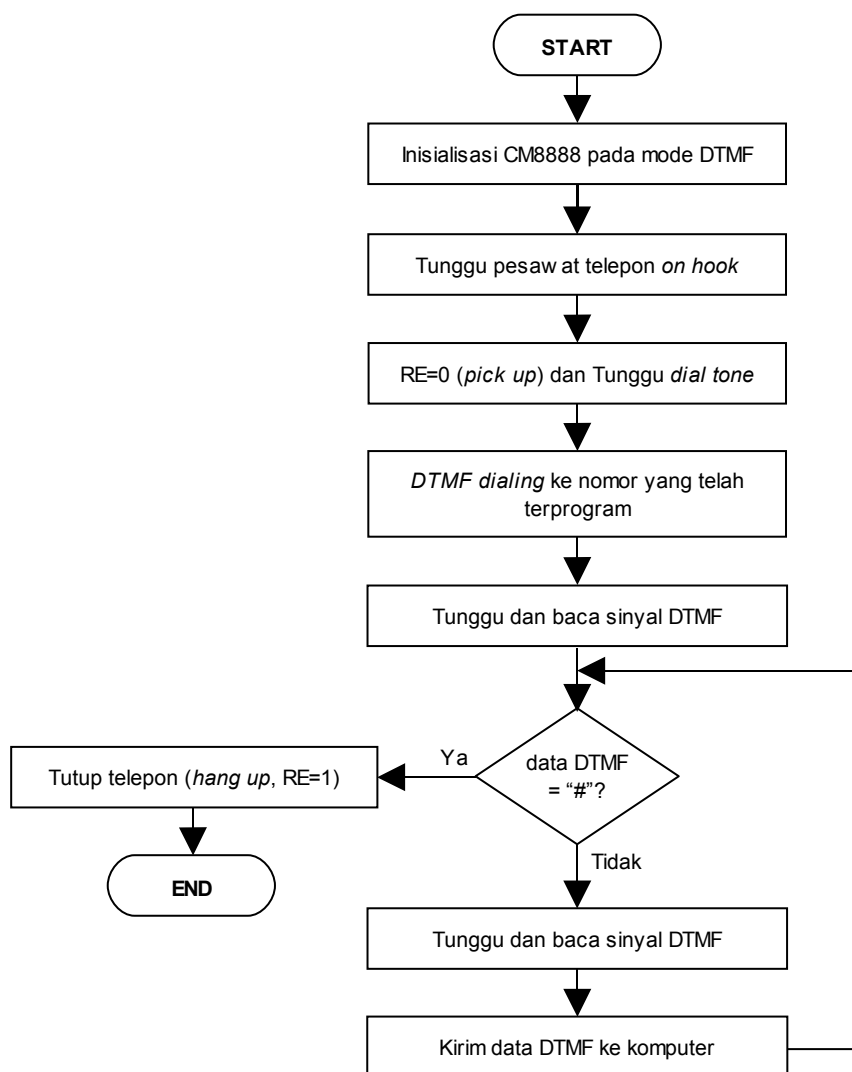
1. Program melakukan inisialisasi CM8888 pada mode DTMF (*Dual Tone Multi-Frequency*).
2. Kemudian program menunggu hingga gagang pesawat telepon diangkat (*off hook*).
3. Jika pesawat telepon telah *off hook* maka program akan menunggu sinyal DTMF dari tombol yang ditekan pada pesawat telepon.
4. Selanjutnya data DTMF akan dikirimkan ke komputer, data DTMF ini dapat dilihat melalui program Terminal[®] atau HyperTerminal[®].
5. Jika kondisi pesawat telepon masih *off hook* (HK=1) maka program akan kembali ke langkah 3, tetapi jika kondisi pesawat telepon adalah *on hook* maka program akan kembali ke langkah 2.



Gambar 4
Flowchart Program TST3.c

TST3.c bekerja dengan DT-I/O Phone Interface Ver 2.0 yang diatur pada mode 1 (*jumper* mode tidak terpasang) dan dengan bantuan pesawat telepon yang terhubung ke DT-I/O Phone Interface Ver 2.0. Cara kerja program ini adalah sebagai berikut:

1. Program melakukan inisialisasi CM8888 pada mode DTMF (*control word* = 0Dh).
2. Program menunggu hingga gagang pesawat telepon diangkat (*off hook*).
3. Kemudian program menunggu adanya *dial tone*. Pada langkah ini CM8888 diatur pada mode *call progress* atau CP (*control word* = 0Eh) agar dapat mendeteksi *dial tone*. Setelah itu CM8888 diinisialisasi kembali ke mode DTMF.
4. Setelah *dial tone* terdeteksi, program akan melakukan *DTMF dialing* secara otomatis ke nomor yang telah ditentukan dalam program.
5. Program mendeteksi kondisi pesawat telepon apakah *off hook*. Jika *off hook* (HK=1) maka program lanjut ke langkah berikutnya, tetapi jika $HK \neq 1$ maka program akan kembali ke langkah 2.
6. Kemudian program akan menunggu datangnya sinyal DTMF dari penekanan tombol pada pesawat telepon. Sinyal DTMF ini dapat berasal dari pesawat telepon di sisi lokal ataupun sisi *remote*.
7. Lalu data DTMF ini dikirimkan ke komputer dan dapat dilihat melalui program Terminal® atau HyperTerminal®.
8. Program kembali ke langkah 5.

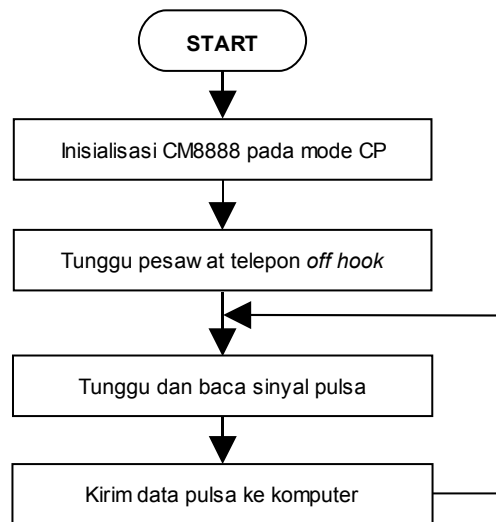


Gambar 5
Flowchart Program TST4.c

TST4.c bekerja dengan DT-I/O Phone Interface Ver 2.0 yang diatur pada mode 2 (*jumper* mode terpasang), tanpa bantuan pesawat telepon yang terhubung ke DT-I/O Phone Interface Ver 2.0. Cara kerja program ini adalah sebagai berikut:

1. Program melakukan inisialisasi CM8888 pada mode DTMF (*control word* = 0Dh), lalu program menunggu hingga kondisi pesawat telepon *on hook* (HK=0).
2. Program “mengangkat” telepon (*pick up*, RE=0) dan menunggu adanya *dial tone*. CM8888 diatur pada mode *call progress* atau CP (*control word* = 0Eh) untuk mendeteksi adanya *dial tone* ini. Setelah itu CM8888 diinisialisasi kembali ke mode DTMF.
3. Program melakukan *DTMF dialing* ke nomor yang telah ditentukan dalam program.
4. Kemudian program akan menunggu datangnya sinyal DTMF dari penekanan tombol pada pesawat telepon di sisi *remote*.
5. Program memeriksa apakah tombol yang ditekan adalah “#” (0Ch). Jika data DTMF = 0Ch maka program akan “menutup” telepon (*hang up*, RE=1) dan program berakhir, tetapi jika data DTMF <> 0Ch maka program akan lanjut ke langkah berikutnya.
6. Program menunggu datangnya sinyal DTMF dari penekanan tombol pada pesawat telepon di sisi *remote*.
7. Data DTMF ini dikirimkan ke komputer dan dapat dilihat dengan bantuan program Terminal® atau HyperTerminal®. Program kembali ke langkah 5.

Flowchart program TST5.c

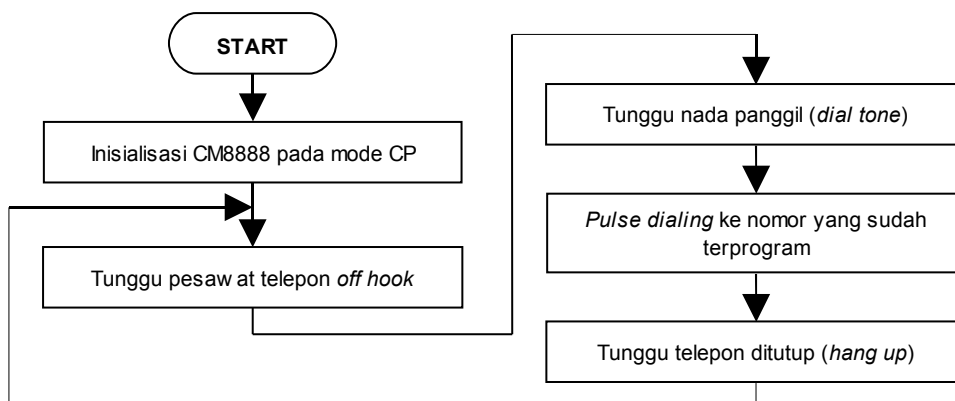


Gambar 6
Flowchart Program TST5.c

TST5.c bekerja dengan DT-I/O Phone Interface Ver 2.0 yang diatur pada mode 1 (*jumper* mode tidak terpasang) dan dengan bantuan pesawat telepon yang terhubung ke DT-I/O Phone Interface Ver 2.0. Cara kerja program ini adalah sebagai berikut:

1. Program menginisialisasi CM8888 pada mode CP (*control word* = 0Eh) dan menunggu kondisi pesawat telepon *off hook*.
2. Setelah pesawat telepon *off hook*, program menunggu datangnya sinyal pulsa (*dial pulse*) dari penekanan tombol pada pesawat telepon. Pesawat telepon harus diatur pada mode *pulse* agar dapat mengirimkan sinyal pulsa.
3. Data pulsa (tombol yang ditekan) akan dikirimkan ke komputer dan dapat ditampilkan melalui program Terminal[®] atau HyperTerminal[®]. Program kembali ke langkah 2.

Flowchart program TST5B.c

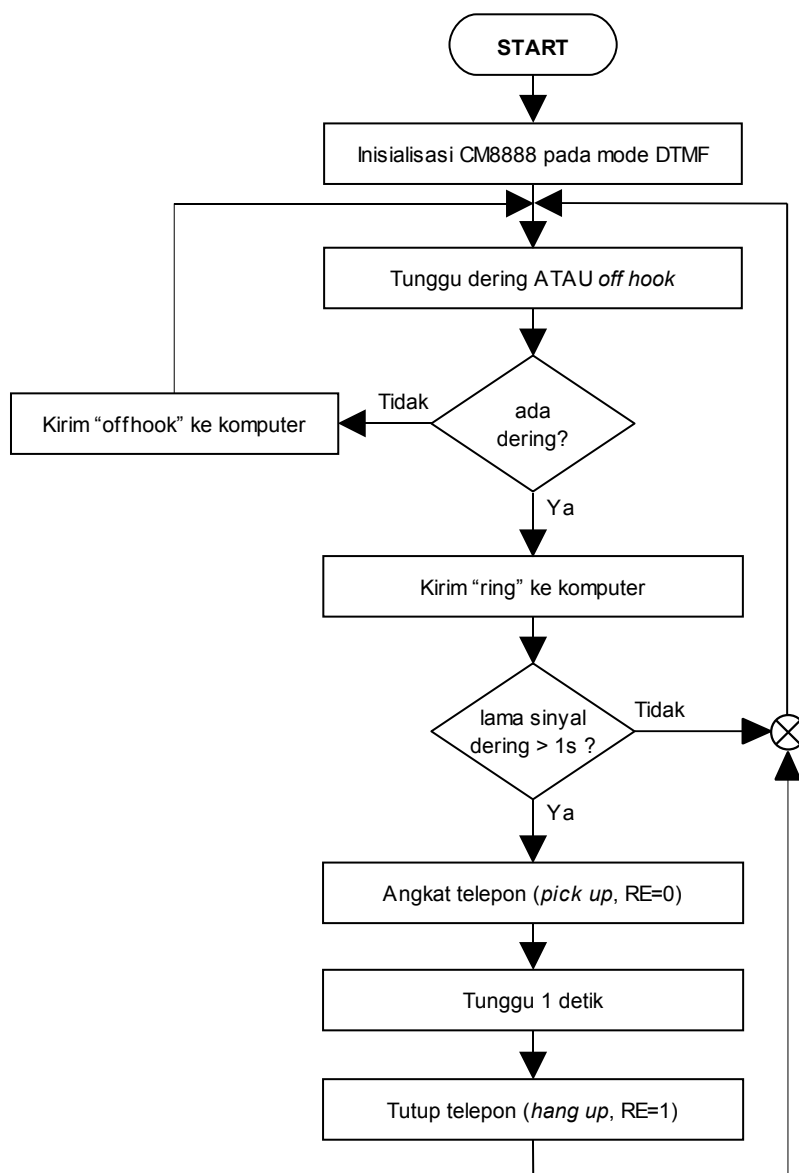


Gambar 7
Flowchart Program TST5B.c

TST5B.c bekerja dengan DT-I/O Phone Interface Ver 2.0 yang diatur pada mode 1 (*jumper* mode tidak terpasang) dan dengan bantuan pesawat telepon yang terhubung ke DT-I/O Phone Interface Ver 2.0. Cara kerja program ini adalah sebagai berikut:

1. Program menginisialisasi CM8888 pada mode CP (*control word* = 0Eh).
2. Program menunggu kondisi pesawat telepon *off hook*.
3. Setelah kondisi pesawat telepon *off hook*, maka program akan menunggu adanya *dial tone*.
4. Kemudian program akan melakukan *pulse dialing* secara otomatis ke nomor yang telah ditetapkan dalam program.
5. Program menunggu hingga telepon ditutup (*hang up*) secara manual, kemudian program kembali ke langkah 2.

Flowchart program TST6.c



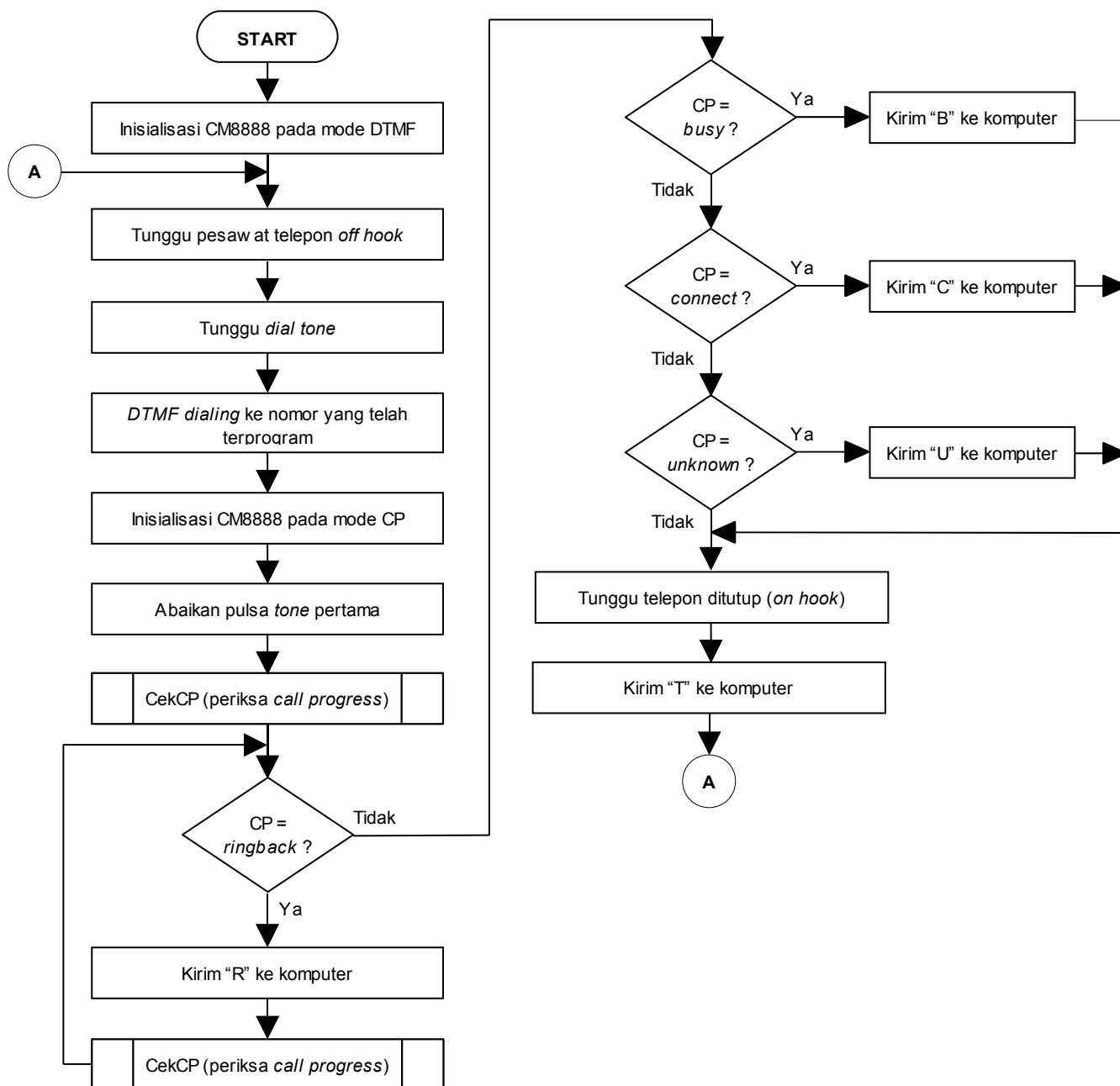
Gambar 8
Flowchart Program TST6.c

TST6.c bekerja dengan DT-I/O Phone Interface Ver 2.0 yang diatur pada mode 2 (*jumper* mode terpasang), tanpa bantuan pesawat telepon yang terhubung ke DT-I/O Phone Interface Ver 2.0. Cara kerja program ini:

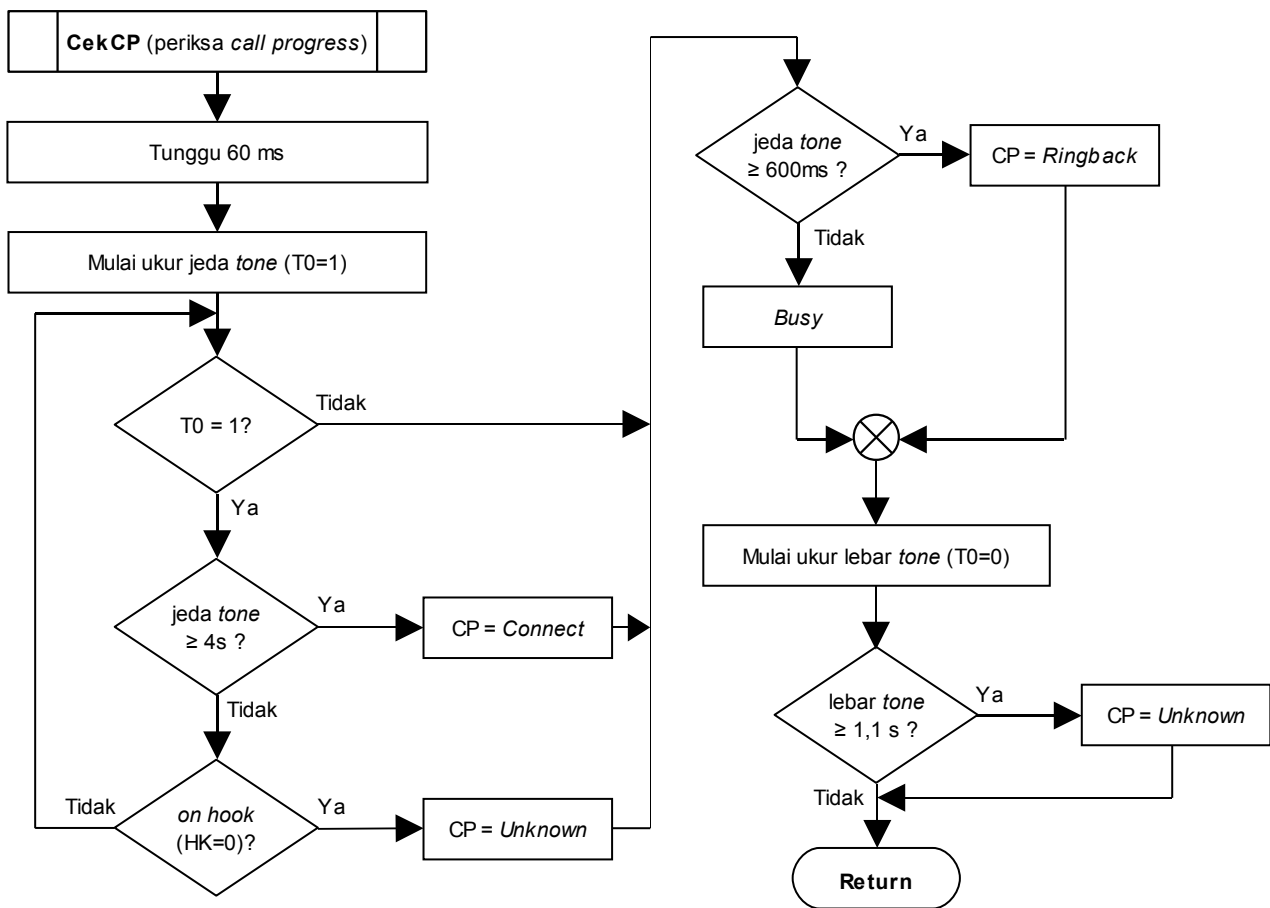
1. Program melakukan inisialisasi CM8888 pada mode DTMF (*control word* = 0Dh).

2. Program menunggu adanya sinyal dering ATAU kondisi pesawat telepon *off hook*.
3. Jika terdeteksi sinyal dering maka program akan mengirimkan tulisan "ring" ke komputer. Lalu program akan memeriksa lamanya sinyal dering ini. Jika lama sinyal dering lebih dari 1 detik (dering valid) maka program akan mengangkat telepon (*pick up*, RE=0) dan menunggu 1 detik lalu menutup kembali telepon (*hang up*, RE=1). Program kembali ke langkah 2.
4. Jika yang terdeteksi bukan sinyal dering maka program akan mengirimkan tulisan "offhook" ke komputer lalu program kembali ke langkah 2.

Flowchart program TST9.c



Gambar 9
Flowchart Program TST9.c



Gambar 10
Flowchart Rutin CekCP dalam Program TST9.c

TST9.c bekerja dengan DT-I/O Phone Interface Ver 2.0 yang diatur pada mode 1 (*jumper* mode tidak terpasang) dan dengan bantuan pesawat telepon yang terhubung ke DT-I/O Phone Interface Ver 2.0. Cara kerja program ini:

1. Program melakukan inialisasi CM8888 pada mode DTMF (*control word* = 0Dh).
2. Program menunggu kondisi pesawat telepon *off hook* (HK=1), lalu program menunggu adanya *dial tone*.
3. Kemudian program akan melakukan *DTMF dialing* ke nomor yang telah ditentukan dalam program.
4. Program melakukan inialisasi CM8888 pada mode CP (*call progress*) yaitu dengan *control word* 0Eh.
5. Kemudian program akan memeriksa kondisi telepon (*call progress*) untuk pemanggilan tersebut.
6. Jika kondisi panggilan adalah *ringback* (*jeda tone* antara 0,6 detik hingga 4 detik) maka program akan mengirim karakter "R" ke komputer. Lalu program memeriksa lagi kondisi telepon dan mengulang langkah ini. Jika kondisi panggilan sudah bukan *ringback* maka program lanjut ke langkah berikutnya.
7. Jika kondisi panggilan adalah *busy* (*jeda tone* kurang dari 0,6 detik) maka program akan mengirim karakter "B" ke komputer.
8. Jika kondisi panggilan tersambung atau *connect* (*jeda tone* lebih dari 4 detik) maka program akan mengirim karakter "C" ke komputer.
9. Jika kondisi panggilan tidak diketahui atau *unknown* (*lama tone* lebih dari 1,1 detik atau pesawat telepon *on hook*) maka program akan mengirim karakter "U" ke komputer.
10. Setelah mengirimkan kondisi panggilan, program menunggu telepon ditutup (*hang up*) secara manual.
11. Program mengirimkan karakter "T" ke komputer dan kembali langkah 2.

Nama dan kegunaan dari fungsi-fungsi / prosedur-prosedur dalam contoh program ini sama dengan rutin-rutin siap pakai untuk mikrokontroler MCS-51[®] yang tersedia dalam CD produk DT-I/O Phone Interface Ver 2.0. Keterangan yang lebih jelas mengenai kegunaan rutin-rutin ini terdapat pada manual DT-I/O Phone Interface Ver 2.0.

Listing program terdapat pada **AN123.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.