

Cuprins

Prezentare Proiect	
Fișa de Asamblare	
1. Funcționare	2
2. Schema	2
3. Lista de componente	3
4. PCB	4
5. Tutorial – Porti Logice	5 - 9



DIAL ALARM - V2

ALARMĂ PRIN TELEFON

- Avantaj Pret/Calitate
- Livrare rapida
- Design Industrial
- Proiecte Modificabile
- Adaptabile cu alte module
- Module usor de asamblat
- Idei Interesante

Idei pentru afaceri

Hobby & Proiecte Educationale

O replică inteligentă dată alarmelor prin linia telefonică.

O nouă viziune asupra sistemelor de alarmă și modului de control. Un accesoriu de nelipsit.

Caracteristici:

- Sesizează efracția
- Formează numărul
- Redă convorbirea

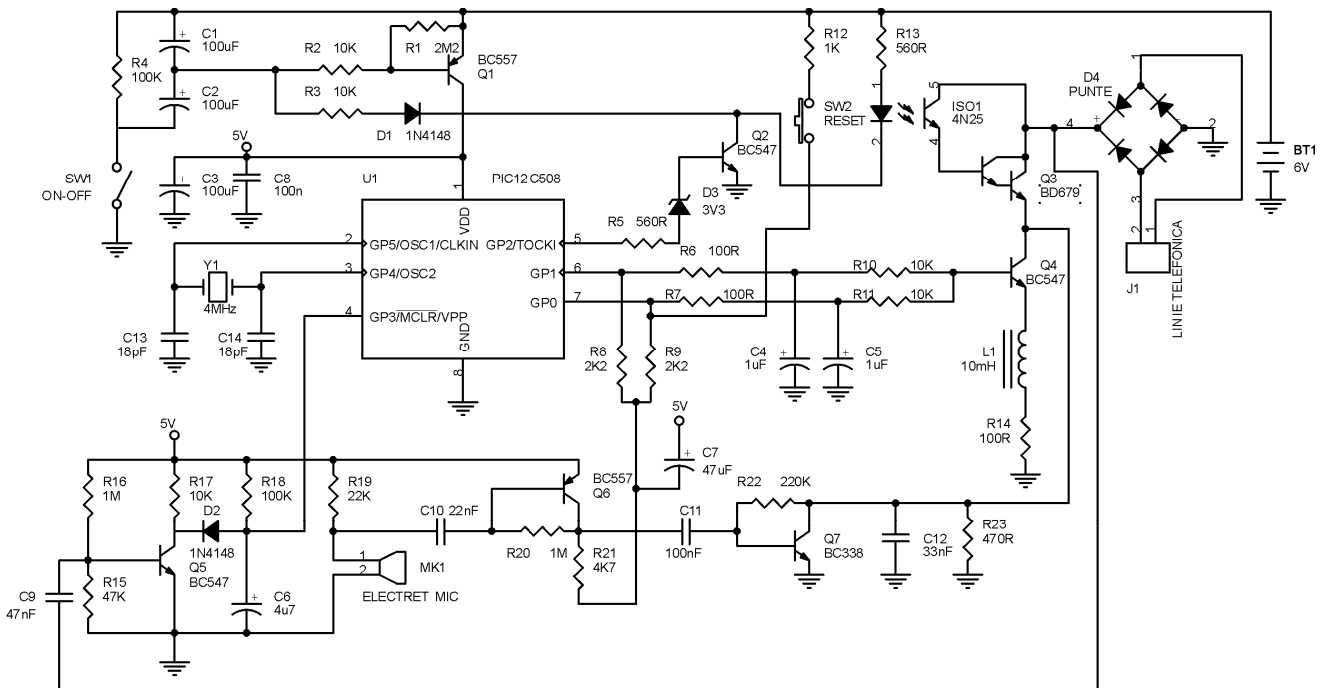
Funcționare

Prin închiderea contactului SW1 (sau contactele de la mai mulți senzori) condensatoarele C1 și C2 se încarcă la o valoare suficientă pentru deschiderea tranzistorului Q1 ce permite alimentarea controllerului PIC12C506. Se startează programul iar "prima grijă" a acestuia este de menținere a alimentării prin trimiterea unui semnal prin GP2 ce deschide tranzistorul Q2 iar prin D1 menține deschis pe Q1. Automat se apelează două numere de telefon pentru alertare. Acțiunea interesantă este cea de "identificare vocală" a intrușilor prin transmiterea sunetelor captate de microfonul

electret. După două cicluri de apelări ale celor două numere de telefon, circuitul se deconectează automat. Detecția tonului se face cu etajul format din Q5 iar tensiunea de pe C6 este aplicată pe GP3. Semnalele DTMF sunt generate prin GP1 și GP0 și transmise pe linie prin Q4, cu condiția de validare transmisă tranzistorului Darlington Q3 prin optocuplor.

Fișierele se descarcă free de pe net

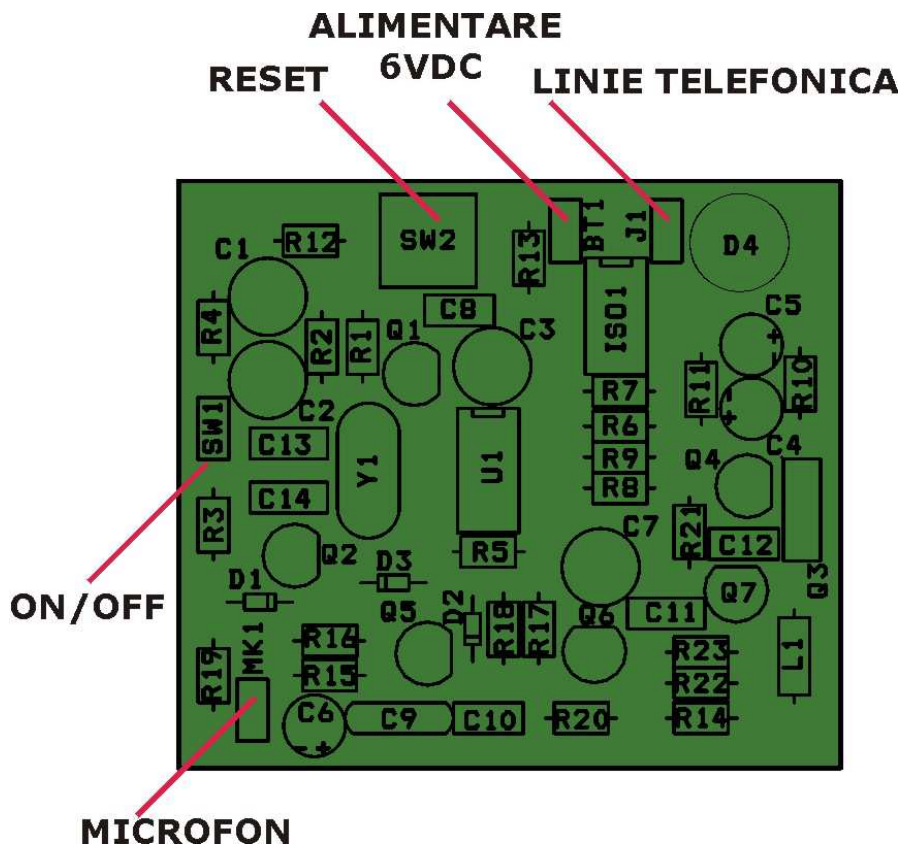
<http://www4.tpg.com.au/users/talking>



Schema electrică

Lista de componente

Nr.Crt.	Componenta	Denumire	Valoare	Cant
1	BT1	Baterie	6V	1
2	C1,C2,C3	Condensator POL	100 μ F	3
3	C4,C5	Condensator POL	1 μ F	2
4	C6	Condensator POL	4,7 μ F	1
5	C7	Condensator POL	47 μ F	1
6	C8, C12	Condensator NP	100nF	2
7	C9	Condensator NP	47nF	1
8	C10	Condensator NP	22nF	1
9	C11	Condensator NP	100nF	1
10	C13,C14	Condensator NP	18pF	2
11	D2,D1	Diodă	1N4148	2
12	D3	Diodă Zenner	3V3	1
13	D4	Punte redresoare	PUNTE	1
14	ISO1	C.I.	4N25	1
15	J1	Conector	LINIE TELEFONICA	1
16	L1	Bobină	10mH	1
17	MK1	Conector	ELECTRET MIC	1
18	Q6,Q1	Tranzistor	BC557	2
19	Q2,Q4,Q5	Tranzistor	BC547	3
20	Q3	Tranzistor	BD679	1
21	Q7	Tranzistor	BC338	1
22	R1	Rezistență	2,2M Ω	1
23	R2,R3,R10,R11,R17	Rezistență	10K Ω	5
24	R4,R18	Rezistență	100K Ω	2
25	R13,R5	Rezistență	560 Ω	2
26	R6,R7,R14	Rezistență	100 Ω	3
27	R8,R9	Rezistență	2,2K Ω	2
28	R12	Rezistență	1K Ω	1
29	R15	Rezistență	47K Ω	1
30	R16,R20	Rezistență	1M Ω	2
31	R19	Rezistență	22K Ω	1
32	R21	Rezistență	4,7K Ω	1
33	R22	Rezistență	220K Ω	1
34	R23	Rezistență	470 Ω	1
35	SW1	Switch	ON-OFF	1
36	SW2	Push buton	RESET	1
37	U1	C.I.	PIC12C508	1
38	Y1	Quartz	4MHz	1



Amplasarea componentelor

Acest produs se livrează în varianta circuit imprimat, circuit imprimat + componente sau în varianta asamblată în scopuri educaționale.

Dacă doriți să aflați mai multe despre produsele noastre, vizitați situl www.epsicom.com

Dacă ați întâmpinat probleme cu oricare dintre produsele noastre sau dacă doriți informații suplimentare, contactați-ne prin e-mail office@epsicom.com

Pentru orice întrebări, comentarii sau propuneri de afaceri nu ezitați să ne contactați pe adresa office@epsicom.com

31 Sararilor Street | 200570 Craiova, Dolj, Romania | 0723.377.426, 0743.377.426

Program

Una dintre cele mai mari sarcini este finalizarea unui proiect. Ca mai toate proiectele, programul este realizat dintr-un număr de subrutine testate din alte proiecte, pentru a funcționa corect, la fel ca și circuitele. Individual, totul funcționează însă provocarea este de a face funcționale toate blocurile, împreună.

Pentru acest proiect, se verifică fiecare bloc în parte:

- Tonul DTMF - durata, amplitudinea, claritate, pentru a fi acceptat 100% pe linie,
- Optocuplorul, tranzistorul Darlington
- Circuitul pornire
- Amplificatorul audio, reducerea zgomotului de fond, brumul, îmbunătățirea amplitudinii de ieșire, suprimare a fascicului.

Modificarea programului

Pentru a prelucra programul, avem nevoie de un adaptor de la 18 pini la 4 pini, prezentat mai jos.

18 pini	4 pini
14 și 4	1
16	2
15	3
9	4
8	5
7	6
6	7
5	8

Acest lucru ne va permite ca procesorul PIC16F84 să fie conectat pe placă, astfel încât să putem modifica cu ușurință codul.

Următorul lucru de care vom avea nevoie este un program de asamblare (asamblor) pentru a converti fișierul .asm la Hex.

Putem folosi MPASM.

Programul este scris într-un editor de text, cum ar fi Notepad sau Textpad și are exact același aspect ca și programul de mai jos.

Trebuie să-l numim xxxxxxxx.asm (până la 8 caractere, apoi .asm). MPASM încarcă acest fișier și generează un fișier Hex.

El creează și un fișier. Ist care conține erorile. În cazul în care s-a făcut o greșeală, MPASM nu va produce un fișier hex, ci doar un fișier .Ist pentru a vedea unde se află eroarea.

De fiecare dată când salvăm fișierul, trebuie să îi dăm un nou nume, pur și simplu o altă literă a alfabetului, cum ar fi dial-A.asm, dial-B.asm etc. Astfel rulat programul va indica cea mai recentă versiune. În plus, MPASM nu va produce un fișier dacă fișierul hex este în prezent utilizat de PIP-02, de exemplu.

Atunci când programul este fără erori, MPASM va produce un fișier Hex.

Pentru a descărca cea mai recentă versiune de MPASM de la adresa:

<http://ww1.microchip.com/downloads/en/DeviceDoc/asm21500.zip>

Acum avem posibilitatea de a modifica programul, fără a pierde procesoare. Oricare dintre rutine pot fi modificate pentru a se potrivi opțiunilor noastre.

Acest adaptor pentru GP0, GP1, GP2, GP3 și Xtal pe pinii 2 și 3

Tipul B are 4K7 și 22p la bord.

Înainte de programare

Înainte ca procesorul 508A să fie programat, trebuie să ne asigurăm că programul este corect, deoarece procesorul nu se mai poate reprograma.

Singurul lucru pe care îl putem face este să programăm în jos instrucțiunile nedorite la 00. S-a folosit acest "truc" pentru a reprograma numerele de telefon (a se vedea mai jos), astfel încât cipul poate fi reprogramat cu un număr nou.

Pentru a ne asigura că programul funcționează corect, vom testa cu un PIC16F84.

Dacă suntem mulțumiți, putem programa procesorul 508A.

În continuare, tabelul trebuie să fie mărit la cel puțin 100 RETLW 0FFh pentru a permite reprogramarea numerelor de telefon. Asigurați-vă că nici unul din programe nu trece în a doua pagină a lui 508A căci instrucțiunile CALL nu vor funcționa. Pentru a fi absolut siguri, verificăm ca rutina apelată să fie în primii 256 de octeți de memorie. Programul DIAL-08.asm de mai jos, a fost realizat pentru a lucra atât cu PIC16F84 cât și cu PIC12C508A întrucât au fost utilizate numai fișierele și liniile de port comune ambelor chip-uri. În plus, rutina de pornire conține instrucțiuni care vor fi asamblate corect pentru ambele procesoare.

Mergem la rutina principală și stabilim durata de timp pentru fiecare activare, așa cu dorim. În program apar cca 15 secunde de ascultare în zona țintă, apoi producând un semnal de alarmă în jos linia, încă 15 secunde de ascultare, apoi formează al doilea număr de telefon. Acest ciclu se repetă de mai multe ori după care alarma se oprește.

Se introduc apoi cifrele numărului de telefon pe care dorim să îl apelăm, în locul celor 10 cifre aleatorii (Tabel1). Numărul de telefon pe care poate avea oricâte cifre. Încheie numărul cu RETLW 0E.

Valorile retlw pentru cifre coincid cu numerele 1-9, cu excepția 0 = 0A (0 reprezintă zece impulsuri iar zece în hex este 0A!)

1 = RETLW 01h

2 = RETLW 02h

3 = RETLW 03h

4 = RETLW 04h

5 = RETLW 05h

6 = RETLW 06h

7 = RETLW 07h

8 = RETLW 08h

9 = RETLW 09h

0 = RETLW 0Ah

E = RETLW 0Eh - Sfârșitul numărului de telefon

Nu protejați cu cod (nu loockati) cipul pentru ca valorile din tabel să poată fi șterse, realocând numerele de telefon prin reprogramare, prin arderea în jos a cifrelor vechi cu 00 00, următoarele locații RETLW fiind convertite în digiți 0FFh ca numere de apelare.

Reprogramarea lui '508A

Dacă aveți nevoie să schimbați numărul de telefon, vechile cifre sunt transformate în 00 00 (inclusiv comanda RETLW 0E), noii digiți înlocuind RETLW 0FFh.

Putem face acest lucru până când se utilizează toate locațiile RETLW 0FFh. De aceea, ar trebui să avem până la 100 de RETLW 0FFh atunci când facem prima programare întrucât avem nevoie de un RETLW 0FFh pentru fiecare nou digit. De exemplu, pentru un număr de 10 digiti, plus RETLW 0E, se folosesc 11 RETLW 0FFh.

Programul

Programul complet pentru alarmă este prezentat mai jos. Fișierul Hex nu a fost furnizat, deoarece există o serie de elemente pe care trebuie să se schimbe înainte de programarea procesorului. Programul inițial a fost testat cu numerele de telefon ale autorului, iar acestea trebuie să fie înlocuite cu noile numere. De aceea, programul a fost lăsat deschisă. Secvența de temporizare este prezentată pe diagrama de flux pe o pagină anterioară și acest lucru constă în apelarea primului număr, apoi a asculta zona țintă timp de 15 secunde, trimiterea Hee Haw în jos linia, apoi ascultare timp de încă 15 secunde înainte de a apela al doilea număr. Această secvență se repetă din nou, iar alarma se oprește.

Orice durată poate fi mărită sau micșorată, conform opțiunilor noastre.

PROGRAM Apelare Alarm-1

; DIAL-08.asm pentru ardere "508a chip-uri

NOTĂ: Acest program nu formeaza un număr de telefon recunoscut.

; PIC12c508A Fișiere

; Acestea sunt fisierele comune atât pentru F84 cat si "508A:

; 0Ch

; 0Dh

; Fișier buclă 0Eh

; 0Fh
 ; Fișier salt 10h pentru tabele
 ; 11h
 ; Fișier Count 12h
 ; DTMF 13h Carrier
 ; 14h ton scăzut
 ; 15h ton jos decrementable
 ; 16h ton ridicat
 ; 17h ton ridicat decrementable
 ; 18h
 ; 19h rutină de întârziere
 ; 1AH rutine de întârziere
 ; 1BH rutine de întârziere
 ; 1Ch rutine de întârziere
 ; 1DH fișier suna al doilea număr
 ; Numerele 1EH Suna a doua oară
 ; 1Fh

SetUp MOVLW 08 Put 0000 1000 în W

TRIS 06 ; Marcă de intrare GP3
 CLRF 06 ; Sterge Portul 6 de date ramase
 BSF 01h,0 ; prescalar bit0
 BSF 01h,1 ; prescalar bit1
 BSF 01h,2 ; prescalar bit2 = WDT x 128
 BSF 01h,3 ; prescalar atribuit WDT
 BSF 06,2 ; Activeaza circuitul TURN-ON
 NOP
 NOP
 BSF 06,0; Stop audio
 GOTO Main1

; Tabelul 1 pentru digiti celor două numere de telefon

Tabell ADDWF 02,1 ; Fișier 02 = biți scăzut al contorului programului

RETLW 0Ah; 0
 RETLW 03h
 RETLW 01h
 RETLW 02h
 RETLW 03h
 RETLW 02h
 RETLW 03h
 RETLW 07h
 RETLW 05h
 RETLW 08h
 Retlw 0Eh; E = Sfârșitul numărului

 RETLW 09h
 RETLW 05h
 RETLW 05h
 RETLW 08h
 RETLW 08h
 RETLW 08h
 08h RETLW;
 RETLW 08h
 Retlw 0Eh; E = Sfârșitul numărului

; O sută de valori RETLW 0FFh sunt plasate aici
; Pentru numere de telefon viitoare:

RETLW 0FFh
RETLW 0FFh
RETLW 0FFh
RETLW 0FFh
RETLW 0FFh
RETLW 0FFh
RETLW 0FFh
RETLW 0FFh
RETLW 0FFh

RETLW 0FFh
RETLW 0FFh
RETLW 0FFh
RETLW 0FFh
RETLW 0FFh
RETLW 0FFh
RETLW 0FFh
RETLW 0FFh
RETLW 0FFh

RETLW 0FFh
RETLW 0FFh
RETLW 0FFh
RETLW 0FFh
RETLW 0FFh
RETLW 0FFh
RETLW 0FFh
RETLW 0FFh
RETLW 0FFh

RETLW 0FFh
RETLW 0FFh
RETLW 0FFh
RETLW 0FFh
RETLW 0FFh
RETLW 0FFh
RETLW 0FFh
RETLW 0FFh
RETLW 0FFh

; tonuri DTMF Tabelul2 Low

Tabel2 ADDWF 02,1; Fișier 02 = scade biți din contorul programului

NOP
075h RETLW; 1
075h RETLW; 2
075h RETLW; 3

RETLW 06Bh; 4
RETLW 06Bh; 5
RETLW 06Bh; 6
060h RETLW; 7
060h RETLW; 8
060h RETLW; 9
057h RETLW; 0

; tonuri DTMF HIGH Tabelul 3

Tabelul 3 ADDWF 02,1; Fișier 02 = scade biți din contorul programului

NOP
044h RETLW; 1
RETLW 03Dh; 2
037h RETLW; 3
044h RETLW; 4
RETLW 03Dh; 5
037h RETLW; 6
044h RETLW; 7
RETLW 03Dh; 8
037h RETLW; 9
RETLW 03Dh; 0

; Întârziere scurtă de aprox 50ms între tonurile

Del1 CLRWDT

40h MOVLW ;64 bucle
MOVWF 1B;

Del1A DECFSZ 1A,1 ;va produce 256 descreșteri

GOTO Del1A
DECFSZ 1B,1 ;64 decrements fișier 1B
GOTO Del1A
RETLW 00

; Scop general intârziere aprox 0.7sec

Del2 MOVLW 03

MOVWF 19h

Del2A CLRWDT

DECFSZ 1A,1 ;după o trecere, fișierele 1A, 1B va fi
GOTO Del2A ;256 și va produce cea mai lungă
DECFSZ 1B,1 ;întârziere de timp.
GOTO Del2A
DECFSZ 19,1
GOTO Del2A
RETLW 00

Del3 MOVLW 0C; intârziere Aprox 5sec

MOVWF 19h; fișier intârziere

Del3A CLRWDT

DECFSZ 1A, 1
GOTO Del3A
DECFSZ 1B, 1

```
GOTO Del3A
DECFSZ 19,1
GOTO Del3A
RETLW 00
```

; DETECT detectează orice ton DTMF sau semnal

Detect1 BSF 06h,0; Turn off audio

```
CALL Del2 ;Se lasă să se descarce 1U electro-0.7sec
CALL Del2 ;Se lasă să se descarce 1U electro-0.7sec
CLRf 12h ;fișier detectare Clear
BTFSS 06,3 ;intrarea va fi scăzută atunci când se detectează tonul
INCF 12h, 1
BCF 06,0 ;Pornește audio
RETLW 00
```

; Dial1 apelează numărul de telefon DTMF
; 13h = fișier de transport
; 14h = ton scăzut
; 15h = ton scăzut decrementabil
; 16h = ton ridicat
; 17h = ton ridicat decrementabil

Dial1 MOVF 10h, 0; Pune 10h fișier în W

```
APEL Table1
MOVWF 13h ;Pune-W în 13h - purtător
MOVLW 0E ;Căutare pentru E - sfârșitul numărului
XORWF 13h,0 ; este 13h = E?
BTFSC 03,2
RETLW 00
MOVF 13h,0 ;fișierele 13h vor fi 1,2,3 ..0A
CALL Tabelul2 ;Ia-valoare-ton scăzut
MOVWF 14h ;Pune-ton scăzut în 14h
15h MOVWF ;Decrementabil ton low
MOVF 13h, 0
CALL Tabelul 3
MOVWF 16h ;Pune ton înalt în 16h
17h MOVWF ;Decrementabil ton înalt
APEL DTMF1
APEL Del1
APEL Del1 ;întârziere 100ms între tonuri
INCF 10h, 1
GOTO Dial1
```

```
80h DTMF1 MOVLW ;80 bucle de ton
11h MOVWF ;pentru a produce 1 / A 10 sec
DTMF2 DECFSZ 17h,1 ;decrementează fișier cu ton înalt
GOTO DTMF3
01h MOVLW ;pentru a comuta GP0
XORWF 06,1 ;Toggle GP0
MOVF 16h, 0 ;reîncărcare fișier cu ton înalt
17h MOVWF ;reîncărcare fișier de ton mare
DTMF3 DECFSZ 15h,1 ;decrementează fișier cu ton scăzut
```

```

GOTO DTMF2
02h MOVLW      ;pentru a comuta GP1
XORWF 06,1    ;Toggle GP1
MOVF 14h,0    ;reîncărcare fișier ton scăzut
15h MOVWF     ;reîncărcare fișier ton scăzut
DECFSZ 11h,1  ;decrementare bucle fișier
GOTO DTMF2
BSF 06,0      ;porneste ieșirea de pe audio
BCF 06,1      ;dezactivare
RETLW 00

```

; Hee Haw produce alarma sonora

hee CLRWDT

```
MOVLW 0FFh
```

```
MOVWF 0Ch
```

Hee1 MOVLW 0C0h

```
MOVWF 0Dh
```

```
BSF 06,2
```

Hee2 NOP

```
DECFSZ 0Dh, 1
```

```
GOTO Hee2
```

```
MOVLW 0C0h
```

```
MOVWF 0Dh
```

```
BCF 06,2
```

Hee3 NOP

```
DECFSZ 0Dh,1
```

```
GOTO Hee3
```

```
DECFSZ 0Ch, 1
```

```
GOTO Hee1
```

haw CLRWDT

```
MOVLW 0C0h
```

```
MOVWF 0Ch
```

Haw1 MOVLW 0FFh

```
MOVWF 0Dh
```

```
BSF 06,2
```

Haw2 NOP

```
DECFSZ 0Dh,1
```

```
GOTO Haw2
```

```
MOVLW 0FFh
```

```
MOVWF 0Dh
```

```
BCF 06,2
```

Haw3 NOP

```
DECFSZ 0Dh,1
```

```
GOTO Haw3
```

```
DECFSZ 0Ch,1
```

```
GOTO Haw1
```

```
BSF 06,2
```

;păstreaza alarma ON după Hee Haw

```
BCF 06,0
```

;porneste audio

```
RETLW 00
```

;Principal

Main1 MOVLW 02

```

MOVWF 1DH      ;al doilea număr
MOVWF 1EH      ;la numerele de apel a doua oară
BSF 06,2       ;pentru a păstra circuitul deschis
APEL Del2      ; 0.7sec întârziere
APEL Del2      ; 0.7sec întârziere
Main1A CLRf 10h ; sare la valoarea pentru Tael1
Main2 MOVf 10h,0 ; Căutare pentru PON în Table1
      CALL Table1 ; astfel încât cipul poate fi reprogramat
      XORLW 00h
      BTFSS 03,2
      GOTO Main3
      INCF 10h,1
      GOTO Main2
Main3 CALL Dial1
      CALL Del2      ;Silence după apelare
      BCF 06,0       ;Pornește audio
      04h MOVLW      ;Pune 4 bucle în W
      MOVWF 1C
Main4 APEL Del3, 5 secunde întârziere
      APEL Detect1
      BTFSC 12H, 0   ;A fost detectat ton
      GOTO Main8     ;Oprește alarma
      DECFSZ 1C,1
      GOTO Main4
      CALL Hee        ;sunet Hee Haw
      04h MOVLW      ;Pune 4 bucle în W
      MOVWF 1C
Main5 CALL Del3
      APEL Detect1
      BTFSC 12H,0   ;A fost detectat ton
      GOTO Main8     ; Oprește alarma
      DECFSZ 1C, 1
      GOTO Main5
      CALL Hee        ;sunet Hee Haw
      04h MOVLW      ;Pune 4 bucle în W
      MOVWF 1C
Main6 CALL Del3
      APEL Detect1
      BTFSC 12H,0   ; A fost detectat ton
      GOTO Main8     ; Oprește alarma
      DECFSZ 1C, 1
      GOTO Main6
      CALL Hee        ;sunet Hee Haw
      BCF 06,2       ; pentru a închide telefonul
      MOVLW 05
      MOVWF 1C
Main7 CALL Del2; 5xDel2 înainte de a suna
      DECFSZ 1C,1   ; Numărul doilea
      GOTO Main7
      BSF 06,2       ; Pentru a ridica linia telefonică
      DECFSZ 1DH,1   ; fisier de apelare al număr doilea
      GOTO Main3
      DECFSZ 1E,1    ;apeleaza a doua oară
      GOTO Main1A
BCF Main8 06,2; Pentru a activa proiectul off
CLRWDt

```

GOTO Main8; Micro va reseta când 0V detectează

END

Vom avea nevoie de un program editor de text cum ar fi Note Pad

Nu vom putea utiliza fișierul .asm de mai sus ca un .asm pentru Note Pad așa cum s-au adăugat spațiile. Aceste spații vor deranja liniile de program (din MPASM); încercați să retastați linia (e), deoarece poate conține spații nevăzute! Pentru a obține fișierul Dial-08asm ca .zip, descărcați-l de pe site-ul autorului.

Vom avea nevoie de: programator - pentru programarea procesorului '508 și de programul programul aferent.

Dacă doriți să aflați mai multe despre produsele noastre, vizitați situl www.epsicom.com

Dacă ați întâmpinat probleme cu oricare dintre produsele noastre sau dacă doriți informații suplimentare, contactați-ne prin e-mail office@epsicom.com

Pentru orice întrebări, comentarii sau propuneri de afaceri nu ezitați să ne contactați pe adresa office@epsicom.com

31 Sararilor Street | 200570 Craiova, Dolj, Romania | 0723.377.426, 0743.377.426