

```

VAR
  LONG clkfreqms,sign
  WORD freq,ditlength
  B-TE vari,varj,idx,cwpin,ledpin
OBJ Synth:"Synth"
PUB init(Sfreq,Scwpin,Sledpin,Sditlength)
  clkfreqms := clkfreq/1000
  freq := Sfreq
  cwpin := Scwpin
  ledpin := Sledpin
  ditlength := Sditlength
  dira[ledpin]~~
PUB send(letter)
  letter := ucase(letter)
  idx := l'kd'wn(letter : "A".."Z", "0".."9")
  if (idx>0)
    sign := byte[@A][idx-1]
    repeat vari fr'm 7 t' 0
      if sign & |< vari
        repeat varj fr'm vari-1 t' 0
          if sign & |< varj
            dah
          elseifn't sign & |< varj
            dit
            waitdit
        quit
PUB waitdit
  waitcnt(cnt + clkfreqms*ditlength)
PUB waitdah
  waitcnt(cnt + clkfreqms*ditlength*3)
PUB wait7dits
  waitcnt(cnt + clkfreqms*ditlength*7)
PUB ucase(ch)
'' C'nverts l'wercase ch t' uppercase
  if ((ch => "a") and (ch =< "z"))
    ch -= 32
  return ch
PRI dit
  Synth.Synth("A",cwpin,freq)
  'uta[ledpin]~~
  waitcnt(cnt + clkfreqms*ditlength)
  Synth.silence_a(cwpin)
  'uta[ledpin]~
PRI dah
  Synth.Synth("A",cwpin,freq)
  'uta[ledpin]~~
  waitcnt(cnt + clkfreqms*ditlength*3)
  Synth.silence_a(cwpin)
  'uta[ledpin]~

```

DAT

A	byte	%0000_0101
B	byte	%0001_1000
C	byte	%0001_1010
D	byte	%0000_1100
E	byte	%0000_0010
F	byte	%0001_0010
G	byte	%0000_1110
H	byte	%0001_0000
I	byte	%0000_0100
J	byte	%0001_0111
K	byte	%0000_1101
L	byte	%0001_0100
M	byte	%0000_0111
N	byte	%0000_0110
O	byte	%0000_1111
P	byte	%0001_0110
Q	byte	%0001_1101
R	byte	%0000_1010
S	byte	%0000_1000
T	byte	%0000_0011
U	byte	%0000_1001
V	byte	%0001_0001
W	byte	%0000_1011
X	byte	%0001_1001
Y	byte	%0001_1011
Z	byte	%0001_1100
Z0	byte	%0011_1111
Z1	byte	%0010_1111
Z2	byte	%0010_0111
Z3	byte	%0010_0011
Z4	byte	%0010_0001
Z5	byte	%0010_0000
Z6	byte	%0011_0000
Z7	byte	%0011_1000
Z8	byte	%0011_1100
Z9	byte	%0011_1110