

Editor's note: Based on original code by Thomas Watson, which can be found at:

<http://alycesrestaurant.com/Media.htm>

Wav file format can be found here:

<http://www.wotsit.org/>

```
'for original code see tomas watson
'vesion 22-apr-2010
'this wil make muzik later
'and somthing verry strange
global samplerate , channels , bitpersample
global pi
pi = atn( 1 ) * 4
samplerate = 11025
channels = 1
bitpersample = 16
file$ = DefaultDir$ _
+ "\WAV\sin" + nr$( 440 ) _
+ "_" + nr$( 250 ) + ".wav"
call openwav file$
call drawwave 440 , 250
call closewav
playwave file$
end
function nr$( no )
nr$ = right$( "00000" + str$( no ) , 5 )
end function
sub openwav filename$
if ( samplerate <> 11025 ) _
and ( samplerate <> 22050 ) _
and ( samplerate <> 44100 ) then
samplerate = 11025
end if
if ( bitpersample <> 8 ) _
and ( bitpersample <> 16 ) then
bitpersample = 8
end if
if ( channels <> 1 ) _
and ( channels <> 2 ) then
channels = 1
end if
result = mkdir( "WAV" )
if result <> 0 then
print "MAKE DIR ERROR !!"
exit sub
end if
```

```
OPEN filename$ FOR OUTPUT AS #1
CLOSE #1
OPEN filename$ FOR BINARY AS #1
dat$ = "RIFF" + littleendian$( ascii$( length * samplerate * channels
* bitspersample / 8 + 36, 4))
PRINT #1, dat$
dat$ = "WAVEfmt " + littleendian$(ascii$(16, 4)) + littleendian$(ascii
$(1, 2)) + littleendian$(ascii$(channels, 2))
PRINT #1, dat$
dat$ = littleendian$(ascii$(samplerate, 4)) + littleendian$(ascii$(sam
plerate * bitspersample / 8, 4))
PRINT #1, dat$
dat$ = littleendian$(ascii$(channels * bitspersample / 8, 2)) + little
endian$(ascii$(bitspersample, 2))
PRINT #1, dat$
dat$ = "data" + littleendian$(ascii$(length * samplerate * channels *
bitspersample / 8, 4))
PRINT #1, dat$
end sub
sub drawwav Hz , ms
FOR i = 1 TO ms * 1000 * samplerate _
* bitspersample * channels / 8
num = SIN( i * Hz * pi * 2 / samplerate / 1000 )
dat$ = CHR$( 127 * num + 127 )
dat$ = CHR$( 120 * num + 7 * num2 + 128 )
PRINT #1, dat$
NEXT i
end sub
sub closewav
close #1
end sub
FUNCTION ascii$( num , leng )
remain = num
b$ = ""
pow = 0
ascii$ = ""
WHILE pow <> 1
a = remain
pow = 1
WHILE a > 255
a = INT( remain / ( 256 ^ pow ) )
pow = pow + 1
WEND
b$ = b$ + CHR$(a)
remain = remain - ( a * ( 256 ^ ( pow - 1 ) ) )
WEND
```

```
WHILE LEN( b$ ) <> leng
b$ = CHR$(0) + b$
WEND
ascii$ = b$
END FUNCTION
FUNCTION littleendian$( st$ )
'function to reverse byte order
res$ = ""
FOR i = LEN( st$ ) TO 1 STEP -1
res$ = res$ + MID$(st$, i, 1)
NEXT i
littleendian$ = res$
END FUNCTION
FUNCTION soundbyte( frequency , byref n )
o = ( n / 2 / pi _
* samplerate * bitspersample * channels / 8 / frequency + 1 ) _
/ samplerate / bitspersample / channels * 8 * frequency _
* 2 * pi
soundbyte = o
n = o
end function
```