

The Tiger Sound

Gunther Zielosko

1. Basics

In the manual of the BASIC-Tiger™ you will find lots of hints that it can record and play back sounds. At the plug-and-play-lab there are an audio amplifier, a microphone and a speaker. In the following we will use normal sound files as they commonly appear in today's PC applications. So we make the tiger to a sound machine, that can speak, play music or other sounds. With a sound card in PC and some sound tools as a precondition for our experiments we can feed the BASIC-Tiger™ with all kinds of sounds.

1.1. The WAVE format, the sound standard for Windows

Files with the extension ".wav" are a must for the multimedia world of today. You will find barely a PC software, not using some noise while starting or finishing. Now it is time to test it with the BASIC-Tiger™ – apart from some fun there are serious applications for sounds like alarm signals, user announcements and hints, failure messages and others.

The wide distribution of sounds in the PC world has the consequence that the most PCs have a sound card and the most operating systems like Windows itself come with software to record and play sounds. To get sound data for the BASIC-Tiger™ at an easy manner, we will use these resources.

The standard of sound data in the Windows system are WAVE files. To be able to use them in the BASIC-Tiger™ system we have to know their structure, which we learn in the following.

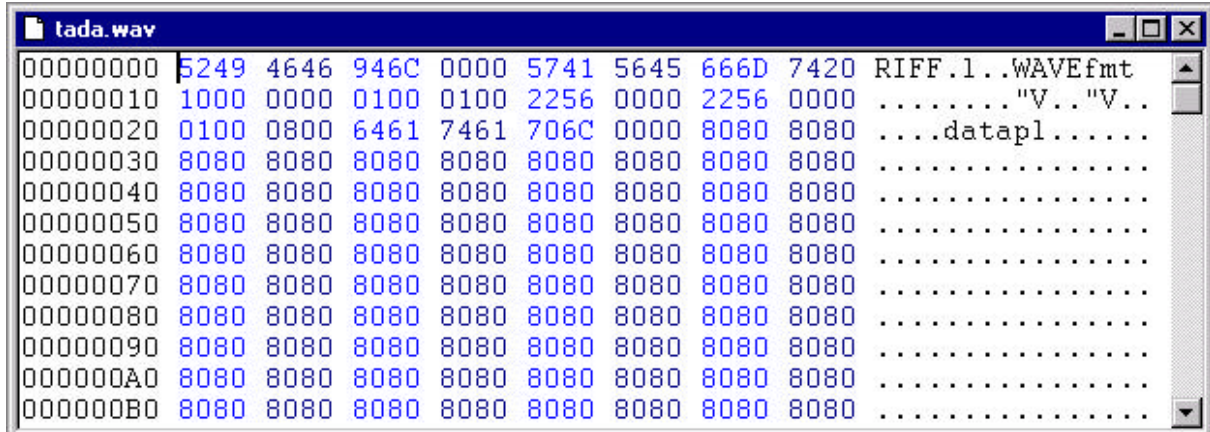
1.2. The structure of WAVE-files

The WAVE format is one of the so called RIFF formats (Resource-Interchange-File Format) which have their beginning on the Amiga system. Later they were used in Microsofts Windows system. You can find some WAVE files in each Windows system, so for instance "ding.wav", "tada.wav", the "microsoft sound.wav" and so on, search for this files in your Windows system if you want...

There are different kinds of wave files. These are discriminated by sampling rate (number of discrete values per second, typically 11,025 or 22,050 Hz), data width (number of bits per value, typically 8, 12 or 16 bits) and number of channels (mono, stereo). Many combinations and intermediate stages are existing. How such a wave file was recorded and how it is played is stored in a so called header at the beginning of the file. In the PC the software identifies the special kind of wave format and gives the data to the sound card.

With a hex editor like Hex Workshop (shareware), XTGOLD or Norton Commander you can see the structure details of a wave file. As a sample we see in the following screenshot (Pic. 1)

the first bytes with the main data of the wave file “tada.wav”, with its help we have a deeper look into the “secrets” of the wave format.



Pic. 1 The begin of the “tada.wav” file from the Windows system

In table 1 we see some explanations of the wave file header. With this information we can indentify the file “tada.wav” as an 8 bit mono record with a sampling rate of 22,050 kHz and a length of 27,804 bytes. This data are always at the same places of the header so we could read it in own projects with the BASIC-Tiger™ directly, but for the first time we will keep this data only in mind to create the BASIC-Tiger™ program with correct settings.

Only the first few bytes of together 27,804 bytes contain information about the file, the rest are sound data which can be heard directly. Before studying this sound data we have to understand the header (Table 1):

Byte No.	hex-code	ASCII-Code	Meaning
0-3	52 49 46 46	R I F F	Marking "RIFF"
4-7	94 6C 00 00		Bytes without leading 8 bytes = 006C94h (27796)
8-11	57 41 56 45	W A V E	Marking „WAVE“
12-15	66 6D 74 20	f m t	Marking „fmt“ and 1 space code
16-19	10 00 00 00		Length of the following under block = 00000010h (16)
20-21	01 00		Code for the Windows pcm technique 01h (1)
22-23	01 00		Channels 0001h = 1 Mono=1 Stereo=2
24-27	22 56 00 00		Samplingrate = 00005622h (22,050 Hz) 2B11h = 11,025 Hz 5622h = 22,050 Hz AC44h = 44,100 Hz
28-31	22 56 00 00		Bytes per sec = 00005622h For 8 bit mono like above, for stereo double number, for 16 bit mono double number and so on

32-33	01 00		Bytes per sample = 0001h	8 bit mono = 1 8 bit stereo = 2 16 bit stereo = 4
34-35	08 00		Bits per sample = 0008h	8 bit data wide = 8h 12 bit data wide = 12h 16 bit data wide = 16h
36-39	64 61 74 61	data	Marking „data“	
40-43	70 6C 00 00		Data block length = 00006C70h (27760)	
44-	80 80		Here begins the real sound data	

Tab. 1 The structure of a WAVE file

It's not our intention to repeat the Windows sounds on the BASIC-Tiger™ system. But it seems to be easier to select a wave file like "tada.wav" which the most of us have on their PC. A little more problematic is the adaptation of a given wave file to the BASIC-Tiger™. The device driver TIMERA.TDD is not able to accept sampling rates over 13 kHz. So some wave files have to be converted first to a sampling rate which can be handled by the BASIC-Tiger™. We have converted the original sampling rate of 22,050 Hz to the lower standard sampling rate of 11,025 Hz, the new file is called "tada11.wav" (while 11 stands for 11 kHz). The next limitation of using wave files in the BASIC-Tiger™ system is the file length. The sample program makes a string from the sound data and a string length over 32767 bytes is not allowed. We have to change the sampling rate or to make longer files shorter – no problem if we have suitable tools for processing sound (wave) files. A very useful tool is the program "COOLEEDIT" (even in its demo version).

Link to sound tool "COOLEEDIT": <http://www.syntrillium.com>

Sometimes soundcards come with software tools for handling sound files, too.

To investigate and manipulate all kinds of files we additionally need a hex editor, for instance "Hex Workshop".

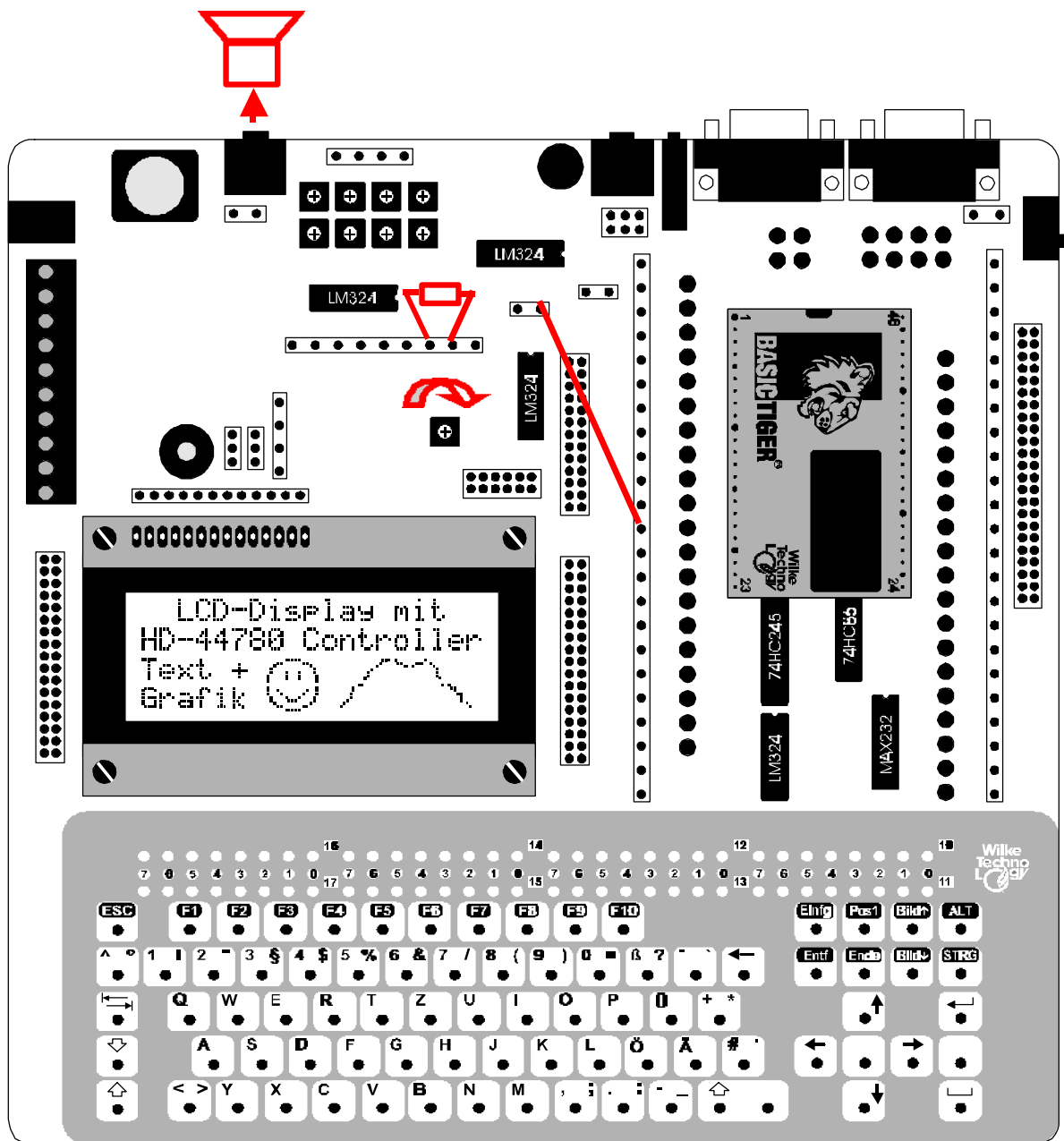
Link to "Hex Workshop": <http://www.bpssoft.com/downloads/hw32v25.exe>

2. How to bring the sound to the BASIC-Tiger™

2.1. Hardware

There are two ways to get a BASIC-Tiger™ with sound either we use the Plug & Play Lab or we use a stand alone unit, then we have to integrate some "sound" parts of the Plug & Play Lab into this new system. For first experiments the Plug & Play Lab is the best choice, only some connections are needed to generate sounds. First we connect pin L72 of the BASIC-

Tiger™ with the input of the PWM preamplifier (with its filter circuit). Then we have to add a resistor of 470k between the output of the PWM preamplifier and the input of the audio amplifier. The best way to do this is to solder two sockets at the wires of the resistor to plug it on the pins of the Plug & Play Lab. The last step is to plug in a speaker into the speaker connector and to turn on the volume control potentiometer. Picture 2 shows the connections which are necessary:



Pic. 2 Connections at Plug & Play Lab

As an alternative you have to add a PWM preamplifier and an audio amplifier like those on the Plug & Play Lab and realize the connections like above.

2.2. Software

The program WAVE01e.TIG demonstrates how the BASIC-Tiger™ uses a normal WAVE file. The WAVE file will be played on the speaker of the Plug & Play Lab using the connections described in chapter 2.1. The WAVE file **must** meet the following demands:

File length shorter than 32765 bytes (maximal string length in Tiger-BASIC™ language)

Sampling rate = 11 kHz

Mono

The WAVE File should be in a place where the program WAVE01e.TIG searches for it (given in line 60, there you can enter the correct file name and path).

2.3. How to make it working - practical hints

While experimenting with sound files we should notice some things discussed in the following:

In the program WAVE01e.TIG the sound data will be placed into a string. In Tiger-Basic™ strings have a maximum length of 32765 bytes. This means the wave file must be shorter than 32765 or a longer file has to be shortened to 32765 bytes. This we can realize with a tool like COOLEEDIT. One way is to cut sound data from the file with this tool. An other way is to change the sampling rate from 22,050 Hz to 11,025 Hz for instance. This reduces the file length automatically. By the way a higher sampling rate is not realistic for the BASIC-Tiger™ and “much to good” for it because the PWM filter is designed for only 400 Hz!

Stereo files can not played correctly with the BASIC-Tiger™. Wave stereo files contain the sound data alternately for the left and right channel. To play stereo wave files we have to convert it into mono files with COOLEEDIT or other tools.

The file name and its path have to be entered correctly in the program WAVE01e.TIG. Before compiling or loading the program you have to guarantee that the wave file exists on your PC, name and path correspond with the data in the program.

The program WAVE01e.TIG only demonstrates the use of wave files in the BASIC-Tiger™ system. The sound begins automatically after a reset and the program ends. In a real-life environment you may want to play a sound after a failure of the user, as an alarm, for a talking measuring device, for a speaking clock and so on. There is much to do, let us begin...