What is AY-3-8910/12 Emulator

"ZX Spectrum Sound Chip Emulator" is designed for playing music for AY-3-8912 sound chip (or its analogs are AY-3-8910 and YM2149F). Emulator emulates these sound chips and does not require real ones. Additionally emulator can play CD's audio tracks and MIDI-files. Optional BASS.DLL v2.2 by Ian Luck can be used for playing MP3 and MOD music.

Emulator can play files of next types.

1. Dumps of registers were recorded in many computers emulators: - OUT (recorded in ZX Spectrum emulator 'Z80' v3.xx by G.A. Lunter); - PSG (recorded in ZX Spectrum emulator 'Z80 Stealth' by Mr.Kirill, and in many oth - EPSG (recorded in ZX Spectrum emulator 'Z80 Stealth'); - YM ('StSound Project' by Leonard/Oxygen files, supported YM2, YM3, YM3b, YM5 and - VTX ('Vortex Project' by V_Soft files); - ZXAY (designed specially for Ay_Emul). 2. Popular ZX Spectrum musical editors' modules: - STC - Sound Tracker v1.xx; - PSC - Pro Sound Creator v1.xx; - ASC - ASC Sound Master v0.xx-2.xx; - PT1, PT2, PT3 - Pro Tracker v1.xx-3.xx, Vortex Tracker II v1.0; - STP - Sound Tracker Pro; - FTC - Fast Tracker v1.xx; - FLS - Flash Tracker; - SQT - SQ-Tracker; - GTR - Global Tracker v1.x; - FXM - Fuxoft AY Language; - AY of AMAD subtype - Amadeus modules, FXM analog; - PSM - Pro Sound Maker. 3. ZX Spectrum's or Amstrad CPC's memory dumps with player for Z80 processor: - AY of EMUL subtype (DeliAY and AYPlay projects files); - AYM (RDOSPLAY project files). 4. CD audio tracks (CDA). Emulator can work with several CD drives. 5. MIDI-files: - MID, MIDI - standard MIDI-files; - RMI - MID-files in RIFF-container; - KAR - standard MID-files with song texts for karaoke; - XMI - Miles Design XMIDI format. 6. Files are played by BASS library: - MP3, MP2, MP1 - MPEG 1 encoded sound record; - OGG -Vorbis encoded sound record; - WAV - sound record; for all that have codec installed in the system; - WMA - Windows Media encoded sound record (through basswma.dll); - MO3 - modules with MP3/OGG encoded samples; - IT - Impulse Tracker; - XM - Fast Tracker 2; - S3M - Scream Tracker 3; - MTM - MultiTracker;

- MOD generic module format;
- UMX Unreal Tournament music package.

Emulator supports playlist files of Winamp (M3U) and of own format (AYL).

AYL can contain full info about each item.

Emulator can be used as powerful tool for working with AY/YM data files. Built-in Speccy music ripper can to find modules of many ZX Spectrum musical editors in any source data. Emulator can convert modules to WAV, ZXAY, VTX, YM6 or PSG formats. Emulator can play digital sound is stored in OUT, ZXAY, EPSG, AY or AYM files. Beeper music in AY and AYM formats are supported too. Emulator can play all special effects are used by Atari ST musicians and stored in YM2, YM5 and YM6 formats.

Skin-files can change view of emulator main window.

BASS.DLL library is loaded only before playing corresponding file and unloaded immediately after stopping playing.

System requirements: Windows 95 or compatible; for work processor 386 or greater is required; for playing sound card with support of 8 or 16 bit Stereo or Mono digital sound at any sample frequency in range from 8000 to 300000 Hz and high performance processor (Pentium 133 is tested). Some beeper music in AY and AYM files require Pentium 166 MHz and higher. By default emulator uses special filter which requires more powerful processor, but it gives more quality sound chip and beeper emulation. For playing extra file types of BASS.DLL require DirectX 3 or above, for MOD music additionally processor with MMX is required.

What's new in this version

No reason to say about all changes due 3 years period between 2.7 and 2.8 versions (there was many fix- and beta-versions in that period). So, only main news are here.

- Added BASS.DLL and BASSWMA.DLL libraries support for playing digital tracker modules and sound records.
- Added FIR-filter for more quality downsampling from sound chip frequency to sound card sample rate. This removed noise of silent envelopes emulation (envelope generator mixed with ultrasound tone generator) and improved sound of some beeper music.
- Playlist window is improved: tuning all colors, added "Play order" and "Loop" button, current item number and total number of items indicators, several ways to sort playlist and find item function.
- Added many new icons.
- Main window and playlist accept drag'n'droping of whole folders.
- Added checking modules integrity option in finder and autodetection.
- All new features of ZX Pro Tracker v3.6x are implemented to PT3-player.
- Added support of Pro Tracker 3.x Utility modules, compiled Pro Sound Maker and Pro Sound Creator v1.00–1.03 modules.
- Improved PT2-files player: portamento parameters are calculated during playing now and also fixed error in duration calculator.
- FLS and FTC players are improved.
- Fixed error of working with GTR-files, including error in GTR v1.1

duration calculator.

- YM2-files samples are supported now.
- Added Audio-CD and MIDI-files (MID, MIDI, RMI, KAR and XMI) support.
- Added command line new abilities.
- Added directory/header analyzer in TRD, SCL and Hobeta formats.
- Max sound chip frequency is 3.5 MHz now.
- Seeking position feature doesn't use prescan of sound chip files now.

Main window elements

Emulator is controlled by left button mouse clicking on active elements of window (see the picture) or by pressing key combinations (see text in brackets). You can drug and drop one or more files or folders from some file shells to main window (would be made new list from them and start playing of first file) or to playlist window (simple adding to the end of list).



Header – window header. You can move window by left mouse clicking on header.

AY – AY-3-8910/12 emulation indicator.

YM – YM2149F emulation indicator.

Stereo – Stereo output indicator.

Volume (cursor keys 'Up' and 'Down' or numpad keys '8' and '2', mouse wheel) – global volume control.

Minimize – minimize Emulator button.

Close (ALT+F4) – close Emulator button.

Mixer (G) – "Mixer" window caller.

Tools (P) – "Tools" box caller.

Play List (E) – "Playlist" window caller.

About – "About program" box caller.

Amplitude (1) – amplitude analyzer, click for on or off.

Spectrum (2) – spectrum analyzer, click for on or off.

Author/Title – field for song author and title displaying. Double click for auto scrolling on or off. By left clicking and moving mouse to the right or left you can do manual scrolling of title.

Time (T) – field for playing time elapsed counter, playing time remained counter and module length displaying. Each of three modes is choosing by left mouse clicking.

Progress – progress slider. By left mouse button clicking on it or moving slider to desired position, you can restart playing from any point. You can also use cursor keys "Left" and "Right" to seek to 5 seconds backward or forward. Also you can press J to call "Jump To Time" dialog for more accurate seeking. Precision of seeking is about 1/50 of seconds (one interrupt).

Loop (R) – infinite looping song button.

Previous (Z or numpad 4) – go to and play previous item of playlist.

Play (X or numpad 5) – play current melody.

Pause (C) – pause/continue playing button.

Stop (V) – stop playing button.

Next (B or numpad 6) – go to and play next item of playlist.

Open (L or numpad 0) – open files box caller. Playlist is cleared before adding new file(s).

Hold SHIFT key during clicking on Open button to call "Open files from folder" dialog (you can use also SHIFT+L). This dialog allows you to choose folder to open. You can check "Recourse all subfolders" to analyze all subfolders contents. Also check "Search for tunes in files" if you want to search music in files of unknown for AY_Emul formats.

Hold CTRL key during clicking on Open button to call "Add CD(s)" dialog (you can use also CTRL+L). This dialog allows to add whole CD audio tracks (useful for mixed CDs).

All described key combinations also work in playlist window.

"Mixer" dialog box

Hot key is **G**. "Mixer" box give you full control under AY/YM sound emulation. All settings are used not only for playing, all of them used during generating WAV files too. The most parameters can be changed during playing. If "Get from list" option were checked then corresponding parameter would be got from playlist global or item parameters or format headers. In dark fields current playing parameters are shown.

Tab sheet AY Emulation.

Channels amplification. For each of chip channels (A, B and C) amplification to right and left channels can be set by input number (range 0–255, equivalent to k = 0.0-1.0) or by moving controls. Use predefined patterns by choosing item of list in lower side of Channels amplification control group. Beeper amplification is not attached to ABC-levels in this version. Preamp is a volume of AY and beeper emulation, use higher value for better emulation.

Optimization. Choose emulation algorithm. If *for quality* is checked then linear interpolation is added, but process is a little slower, than if *for performance* is checked. You can add FIR-filter in "*for quality*" mode for better quality of downsampling from sound chip frequency to sample rate of sound card. For AY frequency 1,7734 MHz and sample rate 44,1 KHz *Filter quality* option must be not lower than 32 to complete rejecting all ultrasound frequencies.

Sound chip frequency. Choose standard external clock frequency or input own value.

Chip type. Choose chip type (AY or YM).

Interrupt frequency. Choose or input player frequency (not used for playing non-synchronized files – OUT, ZXAY, EPSG, AY and AYM, they use *TStates per frame* instead).

OUT, ZXAY, AY, AYM. *Interrupt offset* parameter is used for converting from OUT and ZXAY into PSG, VTX and YM6. Its value is time of first outing to chip from beginning of OUT or ZXAY file (in Z80 tacts). The offset must be

smaller than *TStates per frame* parameter. *TStates per frame* parameter defines range in Z80 TStates (tacts) between interrupts. The parameter is used with *Z80 frequency* for playing AY and AYM files, and with *Interrupt offset* for converting from OUT and ZXAY. If OUT and ZXAY are created by using "Z80" Speccy emulator by G.A. Lunter then *TStates per frame* must be 69888.

Z80 frequency. Set Z80 clock frequency for playing OUT, ZXAY, EPSG, AY and AYM. If OUT files are created in "Z80" Speccy emulator by G.A. Lunter then Z80 frequency is 3494400 Hz. If EPSG files are created in "Pentagon 128K" mode of Z80 Stealth emulator then Z80 frequency is 3500000 Hz.

YM5, YM6 (MFP Timer). Timer frequency for playing special effects in YM5 and YM6 files. Option *AY/YM x 16/13* attaches MFP timer frequency to clock frequency of the sound chip with factor 16/13. Very conveniently for listening these files on frequency AY/YM other than 2 MHz. Option *Atari ST* allows to set documented frequency 2457600 Hz. Option *Another* allows to enter any other value.

Tab sheet WaveOut.

Sample rate, Bit rate, Channels. Parameters of your sound card for digital audio playing. For exact emulation use 16 bit Stereo sound at sample rate frequency is equal to AY counters frequency (press *AY* / *8* button and required frequency will be calculated automatically). In fact, this high sample rate can be used only for generating WAV files for resampling to some lower rate with external resampler. For listening sound in Ay_Emul select sample rate not lower than 44100 Hz. If your sound device does not support selected sample rate, error message will appear during trying start playing or maybe sound quality will be worse even than at lower but supported rate. Anyway, refer to your sound card documentation.

Buffers. Allows set sound *Buffer length* in ms and select *Number of buffers*. *Total length* is calculated automatically. It defines sound latency after changing AY emulation parameters on corresponding tabsheet. Small length can bring to unstable sound.

Device. Allows select sound device. *Wave Mapper* is recommended.

All setting of tab sheet *WaveOut* cannot be changed during playing. To stop

playing, press **Stop playing** button at lower side of tab sheet.

Tab sheet MIDIOut.

Device. Allows select MIDI-device (cannot be changed during playing).

Seek to first "Note On" MIDI-event. Allows to skip silence at begin of MIDIfiles. There are many ones in Internet MIDI-collections, that have 2–3 seconds delay before playing (some time about 10 seconds), so it is very useful feature. Option works only if pause is greater then 0.5 sec, then *Progess* thumb jumps to 20 ms before first note in MIDI-file.

Tab sheet Global Volume.

Select desired mixer to associate with global volume control on main window. By checking *Linear scale* you are disabling logarithmic one. *Save volume position* allows to store current volume before closing application.

Tab sheet BASS.DLL v2.2.

For the moment, you can adjust only visualization of playing BASS.DLL files. *FFT Quality* – higher value for better visualization in low frequency range, but it takes more CPU time. *Hanning window* – check for more accurate FFT algorithm. *AmpMin/AmpMax* – parameter for logarithmic amplitude scale. 0.003 corresponds to about 300 levels of sound (very close to reality).

Restore button is used for setting all "Mixer" box parameters to default values. **Close** button closes mixer window (use Esc from keyboard).

"Tools" dialog box

Hot key is **P**. This box contains emulator parameters and tools are not placed in other boxes and windows.

Tab sheet General.

Current Skin. Here current skin information is showing. By pushing *Choose...* button you can load new skin for main Emulator window (skin files has extension AYS). By pushing *Standard* button you can set original skin.

Language. Choose interface language: *Russian* or *English*.

Folder with music. You can to type path to default folder and save it by pressing *Save* button. Emulator automatically will switch to this folder after its each starting. If you want automatically save last used path, check *Automatically save current* option.

'Start' menu. Add or remove Windows Start menu item "AY Emulator".

Icon on system tray. You can choose *never* to remove icon from system tray and *always* to add. If you choose *minimize* option then icon would be placed on tray during minimizing Emulator window. Double click icon on tray to restore window.

Priority. Choose application priority. *High* priority is recommended for Windows 9x/ME.

Other. Check *Save windows coords* if you want to save windows positions during closing Ay_Emul.

Playlist colors. Click on described elements to define its colors.

'Start' menu icon. You can see icon picture and its author name. Selected icon is used during creating link at 'Start' menu. If link is already created, its icon will be changed automatically.

Tray icon. This icon would be shown at system tray area.

Application icon. This icon is shown on task bar and in some emulator windows title.

Tab sheet Files registration.

Allow you to associate supported files with Ay_Emul in system registry. Ay_Emul saves previous associations, so you can restore it when need. All file types is divided to groups (Music files, BASS files, Playlists and Skin files), each group can be represented by own icon picture. If information about association in registry was edited manually or by previous Ay_Emul version, you can reregister association by pressing *Reregister* button. To restore previous associations press *Restore* button.

Tab sheet Searching for tunes in files.

This is powerful tool for ripping music from Speccy data files and memory dumps. In field *Source files* enter names of files (one name on one line) or click on *Choose*... button. After using *Choose*... button selected files replace previously typed *Source files* list. In field *Work folder* enter name of temporary folder for storing found files or click on *Choose*... button. Name of new files is XXX_XXXXX.ext, where XXX is source file number (hex), XXXXXXXX is offset from beginning of source file to found data, ext is a standard Emulator file extension (SQT, PT2 and so on). Check desired modules for searching in field *Search for tunes*. Flash tracker searching algorithm is very slow and not reliable and corresponding option is unchecked by default. If searching will produce wrong (not music) file, then uncheck corresponding option in field *Search for tunes* and repeat process. To decrease number of mistakes, don't check "*Do not check modules integrity (faster)*" option (unchecked by default). Start searching process by clicking *Begin* button, to stop click it one more time.

Tab sheet Track descriptor.

You can adjust feature "Ay_Emul Track Descriptor" for saving current Ay_Emul status in text file. Good tradition exists among FIDO users. They like to design their letters and conference messages with some original texts. Many message editors allow adding text from file automatically. So, you can insert cool strings

of text into your letters (for example, as signature). One of these strings can be Ay_Emul status string. What you need that simply adjust Ay_Emul and your post program:

File Name – file name with path of text file for storing Ay_Emul current status.

Prefix – piece of text, which will be placed at the beginning of status string.

Suffix – text, which will be placed at the end of status string, in case of Ay_Emul is playing now.

Nothing – text, which will be placed after the *Prefix*, in case of Ay_Emul is silent now.

Enabled – turn on the feature.

Kill on exit – if checked, then file will be erased after closing Ay_Emul.

Kill on nothing – if checked, then file will be erased each time, when Ay_Emul is not playing.

Windows encoding – if checked, then text will be saved in Windows encoding, otherwise in MS-DOS one.

If you are using mail client The Bat!, then for adding string into your letter make next.

1. Open "Account Properties" dialog (Account->Properties...).

2. Click on "+" near of "Templates" option.

3. Chose needed template (for example, "New message").

4. Somewhere in template text (usually at the end) add next string:

%PUT="C:\Ay_Emul\AYSTATUS.TXT"

where in quotes type same string as in *File Name* field of "FIDO Tools" dialog.

5. Insert this string into other templates, if you want.

Fidolook Express 2001 (plug-in for Outlook Express) can be adjusted too. Press "Templates" button and choose page "New Post". In field "Signature" add string

%file(c:\Ay_Emul\AYSTATUS.TXT).

If you want, add the string into other templates. Both described programs are for Windows, so check option *Windows encoding* of "FIDO Tools" dialog. Fidolook SL can be adjusted in same way.

String format:

<*Prefix*><*String* "Author-*Title*" from main window><*Suffix*>

if Ay_Emul is playing something, otherwise

<*Prefix*><*Nothing*>

Uninstall button allows remove both emulators settings and files associations from system registry. Use it before deleting Ay_Emul from system.

Close button is used for closing "Tools" box.

"Playlist" window

Hot key is **E**. This window allows you manipulate play list items. You can move items from one position to another via mouse left button pressed. Type and time length of item are shown at the right side. If during loading item some error was occurred, corresponding item's file name and short error message are shown. Colors of playlist items in all states can be defined in "Tools" dialog.

All main window hot keys work in playlist window too.

Add items button allows you add item(s) with standard "Open Files" dialog box. You can use *Insert* key also. If you hold *Shift* during clicking on this button, you call "Open files from folder" dialog (also you may use *Shift+Insert*). If you hold *Ctrl* during clicking on this button, you call "Add CD(s)" dialog (also you may use *Ctrl+Insert*).

Clear list button removes all list items and set playlist global settings to *By default*.

Save list button allows to save current playlist into file (AYL or M3U). Remember, that in M3U-files only a file names are stored. Some AY-files can contain more than one tune, and this situation cannot be saved in M3U. So, use AYL if you want to save all parameters of current playlist and of each item.

List tools button allows you to sort playlist in different ways and also to find an item by string queries (F7 hot key).

Next button shows one of four icons and determines way of playing playlist items. These are backward, forward, random and only current selected cyclically.

Loop button can be in two states: when it is lowered, playlist is looped (after finishing playing last item, it starts playing first and so on).

Total number of playlist items and current item number both are shown in low-right corner.

Delete key deletes selected items from playlist.

Ctrl-A allows to select all items in playlist.

Use cursor keys for navigation.

Shift and Ctrl keys with cursor keys and mouse clicks can be used to selecting/unselecting items.

Use mouse double clicking or *Enter* key to play an item.

You can drag and drop files and folders to playlist window to add it to the end of list or at any desired point.

You may call popup menu by mouse right button clicking on an item. There are

- *Item adjusting...* calls "*List's item adjusting*" dialog box.
- *Convert to WAV...* allows you convert an item to WAV-file. It is recommended to create WAV-file at sample rate, equal to AY/YM counters frequency (AY frequency / 8, use corresponding button in mixer to set this rate), and then resample it to 44100 Hz with any quality resampler. This way can give you better result, than Ay_Emul built-in resampler.
- *Convert to ZXAY...* allows you convert an item to ZXAY-file (only from OUT, EPSG, AY and AYM files).
- *Convert to VTX*... allows you convert an item to VTX-file.
- *Convert to YM6...* allows you convert an item to YM6-file.
- *Convert to PSG*... allows you convert an item to PSG-file.
- *Save as...* calls standard dialog for saving selected item with new name (only for tracker modules). This feature is useful as alternative method of searching tunes in files. You can open TRD or SNA file and then save all found modules one by another.

There is total playlist time length is shown at the lower right corner. Usually, it includes time length of all visible items (in this case '+' sign added at the right). To include length of invisible for the moment items, click on this field. During

time calculation, Ay_Emul extracts additional information from MIDI-files and from files playing by BASS library.

Use **Esc, E** or **Alt+F4** to close playlist.

"List's item adjusting" dialog box

This box is called from "Playlist" window by right clicking on an item and choosing *Item adjusting*... from the popup menu.

In field "*Information*" input text information about item: song author, title, original program, musical editor, original computer, date of composing and any comments. Comments can be divided to strings, press *Enter* key to do it.

In field "*File*" input file name with path, module type, module offset from beginning of file, length of module in bytes, address of compilation (not needed for most types), time of playing in ms for OUT, ZXAY EPSG, MIDI and BASS.DLL files, in special units for CD-tracks or in VBLs for other types (0 to autodetect), loop time in VBL (–1 to autodetect), OUT, ZXAY, PSG and EPSG files loops to 0 in any cases, and for AY and AYM files term "loop point" is not defined at all. Into parameter *Time* you may include all reiterations. So, if song length is 1000 VBL, you may set *Time* parameter to 500 VBL for playing only half of song, or to 1500 for playing it one with half times, and so on. Other values changing is not recommended.

In field "*Playing*" mixer settings are stored for this item. Playlist consists of two types of settings: global (for all items), and individual. For looking global setting push **Load** button. After new playlist creating all global setting are *By default*. For saving global setting set all required elements (Chip, Number of channels, Chip frequency, Channels amplification and Player frequency) and push **Save** button. For setting selected item settings (not global) do all as global settings, but don't push **Save** button.

For more details about filling field *"Playing"* see <u>"Mixer" dialog box</u> description.

OK button closes the box and saves new settings. **Cancel** button closes box and does not save new settings. Corresponding key combinations are *Enter* and *Esc*.

"Information in the file header" dialog box

This box appears before converting to VTX or YM6.

Field *Year* must be an integer number or empty. All other fields like a fields of "Mixer" and "List's item adjusting" dialog boxes.

Apply button saves settings and close the box.

Restore button restores all settings to previous values.

Command line

Emulator can take parameters from command line. Parameters must be divided with spaces. If parameters are starts with '/' (slash line) then they are interprets as keys else as files names. If file name contains spaces, it must be quoted. If any other key contains spaces then it must be quoted too. You can quote only part of key containing spaces. Files in command line will be added to cleared playlist and will start playing. You can give in command line one or more file names (including playlist files: M3U and AYL). You can specify track/song number for multitrack/multisong files (AY, AYM, MID or XMI) after filename (using ":N" addition, where N is track number starting from 0). List of available keys here.

```
Sample rate
                                 sXXXXX
                                 XXXXX = 48000, 44100, 22050 or 11025.
Bit rate
                                 hxxxxx
                                 XXXXX = 16 \text{ or } 8.
Channels number for sound card dmono
                                 dstereo
                                 dlist - get from playlist
                                 dmixer - don't get from playlist.
Number of sound buffers
                                 wonXXXXX
                                XXXXX = from 2 to 10.
Length of sound buffer
                                 wolXXXXX
                                 XXXXX = from 5 to 2000.
Sound device number
                                 wodXXXXX
                                 XXXXX >= 0.
Z80 chip frequency
                                 ZXXXXX
                                 XXXXX = from 1000000 to 8000000
                AY-3-8910/12
Chip type
                                 eav
                YM2149(F)
                                 eym
                                 elist - get from playlist
                                 emixer - don't get from playlist.
Sound chip frequency
                                 YXXXXX
                                 XXXXX = from 1000000 to 3000000
                                 ylist - get from playlist
                                 ymixer - don't get from playlist.
First interrupt time
                                 † XXXXX
                                 XXXXX greater or equal to zero.
Number of tacts between
                interrupts
                                 iXXXXX
                                 XXXXX greater than zero.
Amplitude analyzer
                        on
                                 aon
                         off
                                 aoff
Spectrum analyzer
                         on
                                 fon
                         off
                                 foff
Optimization
                by performance op
                by quality
                                 рo
Interface language:
                        Russian lr
                        English le
Interrupt frequency
                                 nXXXXX
                                 XXXXX from 1000 to 2000000 (in mHz)
                                 nlist - get from playlist
```

nmixer - don't get from playlist. Loop playing on con off coff Emulator priority ri - idle, rn – normal, rh – high. Channels amplification hMONO hAYABC hAYACB hYMACB and so on for any standard combination. hlist - get from playlist hmixer - don't get from playlist. Manual channels amplification hXX,XX,XX,XX,XX,XX where after key h 3 pairs of amplificator for each chip cha Icon on system tray never дO always g1 minimize g2 Auto scroll name on kon koff off "elapsed" jΘ Timer mode "remained" i1 "length" j2 standard Load skin р user defined pAYS_File_Name MFP timer frequency attach to AY/YM q user defined qXXXXX XXXXX from 1000000 to 3000000 Beeper amplification uXXXXX XXXXX = from 0 to 255 Window minimize vhide vshow restore Auto save path to folders won on off woff Auto saving volume position !on on off !off Auto saving position of windows on xon off xoff FIDO Tools keys File Name fdfFile_Name Nothing fdnString Suffix fdsString Prefix fdpString Enabled yes fde1 fde0 no Kill On Nothing yes fdk1 no fdk0 Kill On Exit fdx1 yes no fdx0 Win Encoding yes fdw1 fdw⊙ no Adding files to the end of playlist add Adding files to the end of playlist and starting playing first of them adp

Examples:

- 1) AY_Emul /h0,0,255,255,0,0 /hmixer My_Song.vtx
- AT_ELMAI / NO, 0, 233, 233, 0, 0 / NMILAET Ny_Song.vtx
 play My_Song.vtx, use only channel B in middle.
 AY_Emul.exe "My New Song.sqt" /eay
 play My New Song.sqt by emulating AY-3-8910/12.
 AY_Emul.exe /hYMABC /eym /y1750000 /n48828
- 2)
- 3)
- set in mixer next values: standard amplification YMABC, chip type YM2149, chip frec Ay_Emul.exe /p"C:\Sergey Bulba\Pascal\Skin Manager\"Example.ays 4)
- load skin Example.ays
- 5) Ay_Emul /add DESERT.XMI:0 "Dizzy 7.ay":2 "Dizzy 6.aym:1" add to playlist first track of DESERT.XMI, third melody from Dizzy 7.ay and second

Hot keys

Terminology.

Numpad

Numeric keypad

Standard key combinations.

Alt+F4 Tab Shift+Tab Cursor keys Shift+cursor keys Shift+Tasert	Close active window Cyclic jumps among window controls Same in reverse order Cursor moving or selecting group items Selecting text Conv. selected to the system cliphoard
Shift+Delete	Cut selected to the system clipboard
Enter	Pushes active button. If no, pushes button with bold frame
Esc	Cancel: escaping function or closing dialog without action

Hot keys of main window.

Playing controls

R	Looping melody on/off
Z	Jump and play previous item in selected playing order
Х	Play current item
С	Pause/continue playing
V	Stop playing
В	Jump and play next item in selected playing order
J	Call "Jump to time" dialog
Cursor left	Seek to 5 seconds backward
Cursor right	Seek to 5 seconds forward
Cursor up	Increase volume
Cursor down	Decrease volume
Next keys work if NumLoc	ck is on:
Numpad 2	Decrease volume
Numpad 4	Jump and play previous item in selected playing order
Numpad 5	Play current item
Numpad 6	Jump and play next item in selected playing order
Numpad 8	Increase volume

Opening files

L	Call	stand	dard	ope	n file	dialog
Numpad 0	Same					
Shift+L	Call	open	fold	der	dialog	

Visualization

2	Specter analyzer	on/off
Т	Toggle time mode	representation

Call other emulator windows

E	Playlist window on/off
Р	Tools window on/off
G	Mixer window on/off

Playlist hot keys.

Esc	Close playlist window
Home	Move cursor to begin of list
End	Move cursor to the end of list
Page Up	Move cursor to one page up
Page Down	Move cursor to one page down
Cursor Up	Move cursor to one item up (cyclic)
Cursor Down	Move cursor to one item down (cyclic)
Shift+Home	Select items from cursor to begin of list
Shift+End	Select items from cursor to end of list
Shift+Page Up	Select one page up
Shift+Page Down	Select one page down
Shift+cursor up	Add to selection one item upper cursor (cyclic)
Shift+cursor down	Add to selection one item under cursor (cyclic)
Ctrl	Invert selection with mouse click
Shift	Select range with mouse click
Ctrl+Shift	Add to selection range of items with mouse click
Delete	Delete selected items from playlist
Insert	Call open files dialog to add items to playlist
Shift+Insert	Call open folder dialog to add folder content to playlist
Enter	Play selected item, random play order will be recreated
F7	Call 'Find playlist item' dialog

All main window hot keys work in playlist too.

Playing order

Emulator plays modules from playlist either from first to last item, from last to first one, randomly and only current item. All modes can be selected by corresponding button in playlist window.

Looping melody

Number of reiterations and loop point is defined in "List's item adjusting" dialog box by *time* and *loop* members. Time include on itself all number of reiterations. In this version of Emulator OUT, ZXAY, PSG and EPSG files always loop to begin. AY and AYM files have no loop point anyway.

If **Loop** button of main window is pressed then melody will be play infinitely. After real end of melody Progress slider is stops, but playing and time counting are continues. If playing is simply repeats, then thumb of Progress bar jumps to corresponding position.

In current version looping melody can not be applied to CD audio-tracks.

Starting time of Emulator

During starting Emulator sets all of its parameters to values, was previously saved to system register.

Emulator does not allow starting second instance of it. It simply redirects all command line parameters to first instance Emulator. So, you can change emulator settings from command line during playing even.

And one more. After starting Emulator searches for file Ay_Emul.ayl in folder with Ay_Emul.exe and load it into "Playlist" window. After closing Emulator automatically saves current playlist to Ay_Emul.ayl.

So, if you have problems during starting Ay_Emul, you can remove from registry HKEY_CURRENT_USER\Software\Sergey Vladimirovich Bulba\ZX Spectrum Sound Chip Emulator or Ay_Emul.ayl before bug-reporting.

Files opening

Emulator can try to recognize file type during opening. If file has standard extension (VTX, STC, etc) then Emulator doesn't do additional checks (in the most cases). If file has non-standard extension (\$m, \$c, TRD, SNA, etc) then Emulator tries to recognize type by header analyzing. And if header analyzing cannot recognize file type, then Emulator searches for tunes in this file. If this process is long then you may push "Cancel" button of the popup dialog box. All found files could be saved from Playlist window by popup menu item *Save as...* Emulator tries to extract additional information about module from TRD, SCL and Hobeta headers.

Converting from OUT and ZXAY

All synchronized files (PSG, YM6, VTX) can be converted from OUT, EPSG, ZXAY, AY (EMUL) and AYM files in case of playing on real Speccy is really synchronized with interrupt frequency. Then can consider all registers of sound chip are loads instantly one time for each interrupt. EPSG file contains interrupt markers, so no additional parameters need for conversion. For correct conversion from OUT and ZXAY to other formats you must properly set two parameters of "**Mixer**" and "**Tools**" box. *TStates per frame* must be set accurately; by default it is 69888 tacts for compliance with 'Z80' Spectrum emulator. If after conversion music are plays little wrong, try change *First interrupt time* parameter by increasing or decreasing with half of first parameter (+35000 or – 35000). If music is not synchronized on real Speccy with interrupt frequency, conversion cannot be correctly.

Seeking tunes

From version 2.5, emulator allows you to restart playing from any point of tune during its listening. For restarting playing from desired point simply click on slider bar or move slider to desired position on main window. Also you can press cursor keys "Left" and "Right" to seek 5 seconds backward or forward. You can call "Jump To Time" dialog by pressing **J**. In appeared dialog type desired time to play from.

How to interrupt converting

During converting, main window is becoming active and progress slider shows current progress of converting process. In most cases you can simply wait for end of converting (when slider will come to max right position and stops). But if you want, you can stop process at any time by pressing *Escape* button or by clicking on main window by left mouse button. This behavior of emulator is coming from first version. You can use any partially converted file as ordinary one.

Windows managing

Four AY_Emul windows can be active at one time: main, playlist, mixer and tools. You can switch between these windows by mouse or from keyboard: ALT+F6 is switching between last two active windows; ALT+SHIFT+F6 is cyclical switching among all opened windows (only when icon on tray). By the way, you can play music and switch windows even during work of "Search for tools in files" tool.

OUT

Supported OUT files of ZX Spectrum emulator 'Z80' versions 3.02 and 3.03 by G.A. Lunter. OUT file has regular structure: sequence of five bytes blocks. First word in each block is time in Z80 processor tacts (range from 0 to 17471). Second word of block is ZX Spectrum port address. And last value of block (byte) is data is sent to this port during this tact. Each 17472nd tact in any case one block is wrote into OUT file, and if during this time no outing to any port of Speccy then first word of block is 65535. One tact of Z80 processor equal to 1/3494400 sec. OUT file can contain outing either to any Speccy ports or only to selected in 'Z80' emulator ports. To record outing only to AY-3-8910 ports (#BFFD and #FFFD), adjust 'Z80' emulator for storing only one port: #FD. And after creating OUT file convert it to other format (ZXAY file is optimized analog of OUT file).

ZXAY

ZXAY file can be converted from any OUT, EPSG, AY and AYM file. This is more optimal format than OUT for storing AY outing. ZXAY file structure is also regular: sequences of 4 bytes blocks. First block of ZXAY file is ASCII string 'ZXAY'. All other blocks have next structure.

Bit numbersName

Bit numbers	Name
0-19	Time
20-23	Register
24-31	Data

Time is time of outing in Z80 processor tacts in range 0–0FFFFFh during of which was outing *Data* to *Register*. Range of register field is 0–13. If during *Time* = 0 was no outing then block is writes into ZXAY file and field *Register* = 15. Available range of *Data* filed:

if Register = then available range of Data 1, 3, 5 or 13 0-15 6, 8, 9 or 10 0-31 7 0-63 other 0-255.

If previous value of registers from 0 to 12 is equal to value of current outing, then current block is not wrote to ZXAY file. Outing to register 13 is always written. As you can see, ZXAY structure is analog of OUT structure.

ZX50

This format is designed for storing synchronized music (50 Hz). This format is better than PSG, but is worse than YM and VTX. So, ZX50 support is removed from AY Emulator.

First four bytes of ZX50 file are ASCII string 'ZX50'. After it byte-strings follows. Each string start from word is called *Mask*. If bits from 0 to 13 of this mask is set (1), then after mask bytes follow. Each byte is value of AY register with number corresponding to mask bit number. For example, if *Mask* = 0000000010010001b then next three byte after this mask are data of register 0, 4 and 7. Each string of byte is data of current VBL (interrupt time). If during current VBL no any outing then *Mask* = 00000000000000000000 As ZXAY files, if previous value of registers 0–12 equal to current value then this value is not wrote to current string of bytes but outing to register 13 always is wrote. Ranges of data for each register number see in ZXAY description.

On RDOSPLAY site you can find PSG2 files description. ZX50 file ideology is very close to that PSG2.

STC

These files are created by ZX Spectrum program ST Song Compiler v1.2 or by built-in compiler of Sound Tracker (version from 1.1 to 1.3 of ST produces compatible modules). Simplified structure of STC module follows.

Offset +0 +1 +3 +5 +7 +25	Number of byte 1 2 2 2 18 2	Name Delay Positions Pointer Ornaments Pointer Patterns Pointer Identifier Size	Description Minimal number of interrupts between notes Pointer to Positions table Pointer to Ornaments table Pointer to Patterns table String 'SONG BY ST COMPILE' Length of all module
+27	Size-27	Data	Other data

Many STC files have Identifier 'SOUND TRACKER v1.3' and they can be played in AY Emulator too. What is more, Identifier can contain any data, and many authors store in it name of song and so on. Also field Size can contain wrong data (for example, zero) or last two chars of song title.

Files of Super Sonic by KLAV are fully compatible with Sound Tracker, so you can simply give it STC extension to play in Ay_Emul (emulator cannot detect them by structure).

ZXS-files of ZXMUS.EXE Version 1.02 Beta by Super Dima from Kaluga are full analog of STC.

ASC

Russian musical editor ASC Sound Master (ASM) versions from 0.xx to 2.xx produce these files. Compiler of ASM always adds play routine to data block. But ASC file must contain only data block. So, you must remove player from compiled module before giving extension ASC to your module. Unfortunately, author and name of song are stored in player routine (see comments below).

Simplified structure of ASC module follows.

Offset +0 +1 +2 +4	Number of byte 1 1 2 2	Name Delay Loop Position PatternPointers Pointer SamplePointers Pointer	Description Initial minimal number of interrupts betwe Position for looping song Pointer to PatternPointers table Pointer to SamplePointers table
+6	2	OrnamentPointers Pointer	Pointer to OrnamentPointers table
+8	1	Position's Number	Number of positions
+9	???	Data	Other data

Some files have no field LoopPosition (in this case loop position is zero). These are files of ASM version 0.xx. For playing it in AY Emulator, add manually this field (with zero value) and increase next three pointers by 1. Other way: give to this file another extension than ASC, and open it in AY Emulator. If extension is non-standard then Emulator tries to detect module type automatically.

Simplified structure of ASC module improved for storing song titles and author names by Ilya Kudryavcev (Himik's ZXZ) of Power of Sound group follows.

Offset +0 +1 +2 +4 +6 +8	Number of byte 1 2 2 2 1 2 2 2	Name Delay Loop Position PatternPointers Pointer SamplePointers Pointer OrnamentPointers Pointer Position's Number	Description Initial minimal number of Position for looping song Pointer to PatternPointers Pointer to SamplePointers Pointer to OrnamentPointen Number of positions
+9	Number	Position list	Position list in playing (
+9+Position's Number	19	ID	String 'ASM COMPILATION OF
+28+Position's Number	20	Title	Song title
+48+Position's Number	4	ID2	String ' BY '
+52+Position's Number	20	Author	Song author name
+72+Position's Number	???	Data	Other data

So, to add name and title to ASC module, extract whole string 'ASM COMPILATION OF <NAME> BY <AUTHOR>' from player manually and use ASCCONV utility (see official emulator site).

Version 2.8 of Ay_Emul gives you second way: if compiled ASM-file (as player+module) is stored in SCL, TRD or Hobeta formats, simply open it in Ay_Emul and save from playlist. In this case ASM and STP module-blocks are modified to store titles and names and saved separately from player.

FXM

These files store music is written on Fuxoft AY Language (terminology and file header was got from RDOSPLAY documentation).

Fuxoft (Frantisek Fuka) designed this language.

FXM file has next structure.

Offset	Size	Name	Description
+0	4	ID	Identifier 'FXSM'
+4	2	Addr	Data block allocation address in Z80 memory
+6	?	Data	Data block

Data block is original structure from software, in which this music is used.

Data block format

0ffset	Size	Name	Description
+0	2	AddrA	Fuxoft AY Language program first byte address for channel A
+2	2	AddrB	Fuxoft AY Language program first byte address for channel B
+4	2	AddrC	Fuxoft AY Language program first byte address for channel C
+6	?		From this point a programs, subprograms on Fuxoft AY Language, sam

Programs and subprograms (as in any other program language) consist of sequence of commands. In this sequence can to select phrases. Phrase is totality of commands is needed for setting one note to playing. Note command and Silence command define the end of phrase. 16-word stack is got to each channel for playing module. So, maximal number of subprogram calls from other subprograms is 16. Recursive calls are not allowed.

Farther description of **this language commands** follows (hex numbers).

00 XX - silence.

Sound off command. Byte parameter XX after this command code is number of interrupts, during which no sounding in corresponding channel.

01..54 XX – note number is increased by **1**.

Set for playing note (number from 00 to 53) during XX interrupts. AY tone

register value 0FBF corresponds to note number 00 is note LA of sub counter octave (if chip frequency is 1773400 MHz). Range between two close notes is halftone.

80 XX XX – jumping to address XXXX.

Simple jump to address XXXX (Z80 JP command analog). Usually this command ends main programs of corresponding channels (in this case it jump to infinite loop command address).

81 XX XX – subprogram at address XXXX calling.

Stores in stack address of next command and jumps to XXXX address (subprogram calling, Z80 CALL analog). Stack size is 16 words limited.

82 XX – setting loop point and number of reiterations.

Stores in stack address of next command (loop point) and XX parameter (number of reiterations increased by 1). Program part from this loop point to command 83 will be repeated (XX - 1) times (reprise).

83 – reprise.

Reiterates (XX - 1) times from loop point is set by 82 XX command. This command pops from stack XX parameter and loop point address, decreases XX by 1, and, if it not equal to zero, stores both parameter and loop point address in stack and jump to loop point.

84 XX – noise.

Parameter XX (00..1F) is AY noise register value.

85 XX- mixer.

Parameter XX (set/reset bit 0 for enable/disable ton, and same bit 3 for noise, all other bits are zeros) sets value of AY mixer register of corresponding channel.

86 XX XX – ornament.

Parameter XXXX defines address of ornament for using with this phrase note

(and by default will be used in next phrases).

87 XX XX – sample.

Parameter XXXX defines address of sample for using with this phrase note (and by default will be used in next phrases; command 8A defines to continue play this sample from current position in it with next notes and command 8B defines to restart playing of this sample with next notes; by default second described mode is on).

88 XX – transposition.

Set transposition equal to XX halftones. XX is signed byte. By default transposition is zero.

89 – return from subprogram.

Pops from stack address and jump to it (Z80 RET analog).

8A – do not initialize sample.

Switch to mode during which setting of new note do not allows to restart current sample playing (it simply continue to play from current position in this sample).

8B – initialize sample.

Switch to mode during which setting of new note restarts sample playing from its beginning (default mode).

8C XX XX – gives control to Z80 subprogram.

Gives control to subprogram in Z80 machine codes. I met only one FXM, which uses this command, and its Z80 subprogram does nothing for sounding. In this Emulator command 8C is not supported. I think Fuxoft designed this command to synchronize some visual effects with playing melody.

8D XX – addition to noise.

Addition to AY noise register current value by module 20 (or 10 in some versions, see Andsix parameter in AY-files description).

8E XX – addition to transposition.

Addition to current transposition value in halftones.

8F – push transposition.

Pushes current transposition value into channel stack.

90 – pop transposition.

Pops transposition value from channel stack.

Sample in this language has simple structure and control only amplitude of output signal. As programs, sample is commands sequence.

00..0F XX – amplitude.

Defines amplitude value in this and next interrupts (XX defines number of interrupts).

32..41 – amplitude.

Defines amplitude value in only this interrupt (increased by 32).

80 XX XX - jump to loop point.

Jumps to XXXX loop point of this sample.

Ornament is a bit compound structure. It changes current frequency of tone register in current interrupt by given value.

80 XX XX – jump to loop point.

Jumps to XXXX loop point of this ornament.

82 – halftones.

Directs to consider all values in ornaments are in halftones.

83 – tone register units.

Directs to consider all values in ornaments are in AY tone register units.

84 – invert mixer bits.

Inverts both AY mixer bits for corresponding channel.

Other values – addition.

Changes value of AY tone register by adding this addition (signed byte) to it. Can be in halftones or in tone register units (set by commands 82 and 83).

Seeing at this module structure, you can to understand, why searching for FXM tunes of this Emulator not so reliable, as other modules searching. So, there are many tunes, which you must to rip manually from your favorite soft. But you can to try use auto searching first, why not? During manual ripping FXM, you can try to use data in its player (same for all FXM).

YM

Arnaud Carre (Leonard/OXYGENE) designs this format for his emulator ST-Sound. First this emulator was an Atari ST sound chip emulator, but now this is a Multi-Computer Sound Emulator.

Now this project consists of many programs (players, converters) and archives of music. But Speccy's archives are in very bad condition.

Don't forget to set chip frequency to 2 000 000 Hz for Atari and to 1 000 000 Hz for Amstrad while listening YM2, YM3 and YM3b files with Atari and Amstrad music (this subtypes not consists data about frequency).

So, for AY Emulator any YM file is LHA archive (type 5), created by LHA.EXE version 2.00 or older with -h0 switch (header type 0) by Haruyasu Yoshizaki. This archive consists of one file with any name. Its available formats are next.

YM2

First four bytes is ASCII string 'YM2!'. Its structure like YM3, but information about samples can be stored additionally. All YM2-samples are fixed and stored in player (these are samples from Mad Max's music).

YM3

First four bytes are ASCII string 'YM3!'. Next bytes are data block of AY chip registers values. Registers are updates one time in one interrupt (VBL). If music length is N interrupts, then block consist first N bytes for register 0, further N bytes for register 1 and so on, in total N*14 bytes. If in current interrupt was no outing to register 13 then byte of data block for this interrupt and for this register has value 255.

YM3b

First four bytes are ASCII string 'YM3b'. Next bytes are data block (see YM3 description). And last four bytes are DWORD data: number of VBL for looping melody.

YM5

This format consists of additional information: chip frequency, player frequency, title, author name, comment and specific Atari ST data (Digi-Drum and SID-Sound effects).

From YMFORMAT.TXT:

	cut here					
Offset 0 4 12 16 20 22	Size 4 8 4 4 2 4	Type DWORD string[8] DWORD DWORD WORD DWORD	Comment ID of YM5 format. ('YM5!') Check String ('LeOnArD!') Nb of valid VBL of the file. Song attributes (see bellow) Nb of digi-drum sample (can be 0) YM2149 External frequency in Hz (ex:2000000 for ATARI-ST version, 1000000 for AMSTRAD CPC)			
26	2	WORD	Player frequency in Hz (Ex: 50Hz for almost player)			
28	4	DWORD	Vbl number to loop the song. (0 is default)			
32	2	WORD	Size (in bytes) of future additional data. (must be 0 for the moment)			
For each	n digidru	um sample:				
· · · · }	4 ?	DWORD BYTES	sample size sample data (8bits per sample)			
 	? ? ? 4	NT-String NT-String NT-String DWORD	Name of the song. Name of the author. Comments (YM file converter ?!) All YM2149 registers. End-File check. ('End!')			

Remarks.

1) VBL – Vertical Blank Time (interrupt).

2) Numeric data (WORD and DWORD) stored in MOTOROLA order (from hi byte to lo).

3) Data block contents now values for 16 registers (14 AY registers plus 2 virtual registers for Atari special effects).

4) If bit 0 of field Song Attributes is set, data block are stored in YM3-style order (interleaved). If this bit is reset, then data block consists first 16 bytes of first VBL, then next 16 bytes for second VBL and so on. In second case YM5 file is compressed more badly.

YM6

This format is equivalent of YM5, but can content yet two another special Atari ST effects is Sinus-SID and Sync Buzzer and can to play any two of special effects at one time.

AY Emulator supports all special effects in YM5 and YM6 files.

More info about YM files can be found in official ST-Sound package or on ST-Sound Project Homepage (<u>http://leonard.oxg.free.fr</u>). Some time ago author's STSoundLibrary with sources was released.

VTX

Roman Scherbakov (V_Soft) designed this type. This is the best format for storing interrupted music from different AY-3-8910/12 and YM2149 based platforms.

From README.TXT (translation from Russian):

			-cut here
Offset	Size	Туре	Description
00 dec	2 bytes	Word	VTX file identifier (ay or ym)
			Its chip type by default:
			this is 7961h or 31073 dec (ay)
			this is 6D79h or 28025 dec (ym)
02	1		Byte bit Playing mode by default:
			Bits 0-2 determine stereo mode:
			0 - MONO, 1 - ABC, 2 - ACB, 3 - BAC,
			4 – BCA, 5 – CAB, 6 – CBA (dec)
03	2	Word	Loop VBL number (from zero =
			beginning of melody).
05	4	Dword	AY chip frequency for this melody
			(for compatibility with non-Spectrum platforms)
			Standard values:
			ZX Spectrum: 1773400 (or 001B0F58h)
			Atari: 2000000 (or 001E8480h)
			Amstrad CPC: 1000000 (or 000F4240h)
09	1	Bvte	Player frequency (VBL per sec).
		y	Standard value 50 =
			1/50 sec (20ms).
10	2 bytes	Word	Year of composition creating.
12	4	Dword	Size of not packed data.
	·	2.1.0.1 4	Needed for unnacking.
16	1-255	NTString	String with melody name
			(1-254 chars), must be ended with zero even
			if no title at all!
22	1-255	NTString	String with author name (see \wedge)
22	1-255	NTString	Melody source program (see ^)
22	1-255	NTString	Musical editor name (see Λ)
22	1-255	NTString	Comment string (see Λ)
22	22	Array	Compressed data nacket
• •		, iii uy	
			-cut nere

Remarks.

1) Data compressed with LHA type 5. May to pack with LHA.EXE (as YM files), but needed to cut header and last zero byte of compressed file. Header of LHA-archive file is found at start of file. Its length is value of first byte of

header plus 2.

2) After unpacking the data block is equivalent to YM3 data block.

Also can be found old version of VTX files (of VTX player older than 2.00 version). AY Emulator can to play it too.

VTX files can be converted from emulators sound dump files (for ZX Spectrum and MSX this is usually PSG-files). But last time you can find emulators, which can to dump music in VTX format too. One of them is Unreal Speccy by SMT.

PSG

Two types of PSG are exist – PSG and EPSG. On RDOSPLAY site you can find PSG2 files description also, but this type is not supported in Emulator. And also there are many PSGs from BK (Russian computers) emulators in the Internet.

PSG

These files are produced by 'x128' Speccy Emulator by James McKey and 'fMSX' emulator (I never saw this one). x128 creates PSG with errors. Therefore more better to use 'Z80 Stealth' Speccy Emulator by Kirill Kolpakov (Mr.Kirill). This emulator contains special features for creating PSG files and also has good debugger which very simplifies PSG creating process. Z80 Stealth Home Page is <u>http://z80.da.ru</u>.

This is all data about PSG found in Internet:

Offset +0	Number of byte	Description
+0	3	
+3	1	Marker "End of Text" (1Ah)
+4	1	Version number
+5	1	Player frequency (for versions 10+)
+6	10	Unknown

Further byte strings follows. Each string begins from byte 0FFh or 0FEh. Byte 0FFh is marker of interrupt beginning. If after it byte exists (in range 0–15) then it is number of AY register, and next byte is value of this register. Further next pair of byte follows, first byte of which is register number, and second byte is register value. And so on, until end of file, or byte 0FFh (next interrupt), or byte 0FEh will be meet. Byte after 0FEh marker is multiplied by four is number of interrupts without outing to AY. For example sequence "FE 01 FF" is equivalent to sequence "FF FF FF FF FF". If in PSG you will find register number in range 16–252, then you can ignore this and next byte (this is outing to other MSX devices).

RDOSPLAY documentation describes yet another marker is 253 called as 'End Of Music', but this marker is not supported in Emulator.

Also, RDOSPLAY documentation and some my researching of existing PSG-

files talk about is more simple header of PSG. This header consists of only first 4 bytes of described header and outing log starts at +4 file offset (instead of +16). So, for correct playing this PSG, need to add zero to expanse header size to 16 bytes length. If you are trying to play such PSG-files without header correcting, you don't hear differences in the most cases.

EPSG

Z80 Stealth emulator creates these files. EPSG additionally contents information about time of outing to AY registers.

Next text from z80s.faq file.

Q: What is it - this EPSG format? A: It's PSG format improved just a bit to handle output of digitized samples Here's the description: Offset Length Value _____ Header _____

 0
 4
 'EPSG'

 4
 1
 0x1A marker

 5
 1
 Machine type: 0x00 - ZX Spectrum 128 0x01 - Pentagon 0xFF - Other machines

 6
 4
 Zero for machine type 0x00 and 0x01 or Number of Z80 tacts between interrupt markers

 for other machines 6 zeroes 10 _____ AY(YM) log during 1 frame _____ 161AY(YM) register number171value written to this183T-state value written to this register 5 0xFFFFFFFFF – interrupt marker ??

AY

AY-file format is designed by Patrik Rak for his DeliAY plug-in to DeliTracker (Amiga computer). James McKay gave second life to this format (AYPlay and AYMake utilities) on PC. Next description I wrote after year of working with AY-files, but I could understand all only after discussion with Patrik Rak.

So, format is designed on Amiga with MC68000 processor. Due to this fact, all word values are in Motorola order (first byte is high) and must be aligned to two byte offsets from AY-file beginning. Second important thing is all pointers in the format are signed and relative (i.e. are offsets from current position in file to pointed data). Range of the offsets is from -32768 to +32767.

AY-file is sequence of records. They order have no limitation, but first record must be header. Header record structure follows.

Offset +0	Length 4	in bytes	Name FileID	Description File identifier 'ZXAY'
+4	4		TypeID	File type. Emulator supports 'EMUL' type, which rec
+8	1		FileVersion	File version, you can use this field as free as you
+9	1		PlayerVersion	Required player version for playing. Only three ver
				0 Use zero if you do not know what player ver
				1 Initial player version.
				2 First 256 byte of Z80 memory is fill with (
				3 Last version for now. Full Z80 emulation pl
+10	2		PSpecialPlayer	This is for AY-files which contain player in MC680(
+12	2		PAuthor	Pointer to null-terminated string with author name,
+14	2		PMisc	Same, but to string with miscellaneous information.
+16	1		NumOfSongs	Number of tunes in file decreased by 1.
+17	1		FirstSong	Tune number, which must be played first, decreased
+18	2		PSongsStructure	Relative pointer to "Song structure" record.

So, size of header is 20 bytes. Last offset of header points to records "Song structure" (one record per tune, the records follows one after another). Any "Song structure" record structure follows.

+0	2	PSongName	Relative offset to null-terminated string with na	ne
+2	2	PSongData	Offset to record "Song data"	

All described above is rightly for any AY-files. Next data only for 'EMUL' AY-file type.

Record "Song data" for 'EMUL' type has next structure.

+0	1	AChan	Amiga's channel number for emulating A AY channel.
+1	1	BChan	Amiga's channel number for emulating B AY channel.
+2	1	CChan	Amiga's channel number for emulating C AY channel.
+3	1	Noise	Amiga's channel number for emulating AY noise.
			Typically these four bytes are 0, 1, 2, 3. You also
+4	2	SongLength	Song duration in 1/50 of second. If zero, than leng
+6	2	FadeLength	Duration of fade after end of song in 1/50 of secor
+8	1	HiReg	Values of high halves of all Z80 common registers (
+9	1	LoReg	Values of low halves of all Z80 common registers in
+10	2	PPoints	Pointer to "Pointers" record.
+12	2	PAddresses	Pointer to "Data blocks" record.

Record "Pointers" is simple. There are stack, initialization and playing routines addresses for the tune.

+0	2	Stack	Value of SP register before starting emulation.
+2	2	INIT	Initialization subprogram address in Z80 memory.
+4	2	INTERRUPT	Playing subprogram address (calls 50 times per sec

Record "Data blocks" is simple too. It is sequences of groups by three word values. End of sequence is first zero address (i.e. Address = 0).

+0	2	Address1	First block address in Z80 memory.
+2	2	Length1	Length of first block in bytes.
+6	2	Offset1	Relative offset to begin of 1st block in the AY-fil
+8	2	Address2	Second block address in Z80 memory.
+10	2	Length2	Length of 2nd block in bytes.
+12	2	Offset2	Relative offset to begin of 2nd block in the AY-fil
and	so on wh	hile Address = 0 will H	pe met.

Follow next rules, if you want that your own AY-files player will be playing most now existing AY-files (this rules enable to play corrupted AY-files even) during loading blocks:

```
    If Address + Length > 65536 then decrease Length to make it 65536.
    if CurrentPositionInFile + Offset + Length > FileSize then decrease Length too.
```

And now about that how player of version 3 must to play AY-files of 'EMUL' type.

1. Fill range #0000-#00FF with #C9 byte value. 2. Fill range #0100-#3FFF with #FF byte value. 3. Fill range #4000-#FFFF with #00 byte value 4. Place at #0038 address #FB byte. 5. If INIT for this song is equal to zero, then place first block loading address at first 6. If INTERRUPT for the song is equal to zero, then place at zero address next player: DI CALL INIT LOOP: IM 2 EI HALT JR LOOP

7. If INTERRUPT is not equal to zero, then use another player:

DT CALL INIT L00P: IM 1 ΕI HAI T CALL INTERRUPT JR LOOP 8. Load all blocks for the song. 9. Load to low halves of Z80 common registers (including flag registers) LoReg value. 10. Load to high halves of Z80 common registers HiReg value. 11. Load into I register value 3 (this player version). 12. Load to SP register Stack value. 13. Load to PC zero value. 14. Disable interrupts and set IMO mode. 15. Reset AY sound chip. 16. Run Z80 emulation.

You can see that blocks can overwrite standard player even. You can use this possibility if you need to run non-standard player routine. In this case you can place it at INIT address or at #0001 address even. In common data blocks can be loaded at any Z80 memory address, except 0.

And now some words about 'AMAD' type.

Record "Song data" for 'AMAD' type has next structure.

+0	2	AllocAddress	Allocation address of data block in Spectrum memory
+2	1	Andsix	The parameter must either 31 or 15. It is used for
+3	1	Loops	Number of loops.
+4	2	LoopLen	Length of one loop in interrupts (VBI).
+6	2	FadeOffset	Precise fade specification (unused in Ay_Emul)
+8	2	FadeLen	How long to fade (unused in Ay_Emul)
+10	1	AChan	Amiga's channel number for emulating A AY channel.
+11	1	BChan	Amiga's channel number for emulating B AY channel.
+12	1	CChan	Amiga's channel number for emulating C AY channel.
+13	1	Noise	Amiga's channel number for emulating AY noise.
			Typically these four bytes are 0, 1, 2, 3. You also
+14	???	ZXData	Original data block from ZX Spectrum program. See 1

As you can see, both EMUL and AMAD subtypes can contain more than one tune in one AY-file.

And now about insufficient flaws of AY-format and about AY-files supporting realization in AY_Emul.

First flaw is specification allows to set SongLength field to "infinite playing" value (zero). The AY emulator has no this term, so files with zero SongLength would be played 15000 interrupts instead of infinite.

Second flaw is vagueness with term "1/50 of second". If it is interrupt period, than it must to depend of two values: Z80 frequency and number of TStates between interrupts (maybe only for Russian ZX Spectrum clones). Anyway, interrupt period is different in different Spectrum models. So, specification standardizes no one of described parameters. In this version of AY_Emul is used 3494400 Hz for Z80 frequency and 69888 for TStates between interrupts (i.e. 69888/3494400 = 1/50 of second).

Third flaw is in that no able to load blocks at #0000 address. This problem can be decided if first block (and only first block) can to have zero address. But this decision is not compatible with other players and therefore not supported in AY_Emul for now.

I can to add that AY-format means only 48K-memory model. But the most part of Spectrum music can be adapted to 48K limitations in any way.

AY Emulator works as player of version 3.

AY-format developing on PC occurs through efforts of other programmers. For example, firstly James McKay made beeper supporting and more realistic Z80 emulation. But his AYMake remakes many AY-files with big error: information about time length was lost. Now bcass (Project AY site owner) is adding time information into existing AY-files. Patrik Rak want to make PC DeliPlayer plug-in for AY-files, maybe soon we'll can to listen AY-format author's PC realization of AY-player.

Well, and with other things in AY-format you can to meet by yourself, if you'll want rip AY-music from Speccy programs. You can find my utilities (aysplitr and AYMakeR) with source codes for splitting and creating new AY-files on official emulator site.

AYM

The AYM format stores same music as AY-format. Original description of format describes CPC and ZX AY ports emulation only. But AY Emulator additionally is emulating ZX beeper port. Before AY_Emul, AYM format was supporting only by one program is RDOSPLAY (OPL2/OPL3 chip music player).

Roman Dolejsi (roman@sorry.vse.cz) designed the AYM format. Next is slightly edited text from RDOSPLAY.TXT.

-----Begin------

name	size	pos	info
'AYM0'	dword	00	identifier (AY Music, format revision 0)
name	db 28	04	music name
author	db 16	20	music author name
init	word	30	music init entry (A = music number)
play	word	32	interrupt player entry
musmin	byte	34	min music number
musmax	byte	35	max music number (musmax - musmin = number of musics)
muspos	byte	36	starting music number
regpos	byte	37	reg in which musnum is stored when entering init (00-0B)
AFreg	word	38	AF register before entering init routine
BCreg	word	ЗA	BC register
DEreg	word	3C	DE register
HLreg	word	3E	HL register
IXreg	word	40	IX register
IYreg	word	42	IY register
blocks	byte	44	number of memory blocks
from po	s 45h, re	est of tl	he file contains memory blocks each carrying this:
begin	word	b+00	where in the memory this block begins (64k addr space)
size	word	b+02	how long is this block
bdata	[size]	b+04	block of Z80 music code data for AY-3-8910 (currently for ZX128 env

-----End-----

Distribution

Sound chip emulator is distributed in two variants:

1) archive with binary (executable) file and documentation;

2) archive with Ay_Emul source files, files, required for compilation executable file and help system files.

If you are using Ay_Emul sources, algorithms and ideas, don't forget to credit author.

Music archives

Emulator author collected big music archive in different ZX Spectrum musical editor formats. Now version 3.8 of the archive exists. There are about 18100 files in this archive. Archive file name is *Tr_Songs.7z*. Send to Emulator author all your modules and any info for this archive, please.

This and other archives can be found on official Emulator web page.

And many songs for Emulator can be found on Internet yet.

- <u>http://leonard.oxg.free.fr/</u> ST-Sound Project page. Many archives in YM-format.
- <u>http://www.brainbug.ch/stsound/</u> Very big archive of music in YM-format.
- <u>http://mfx.da.ru</u> Hacker KAY homepage. Little archive of tracker music.
- <u>http://sorry.vse.cz/dimension/rdos/rdosplay/</u> site of RDOSPLAY OPL2/3 player. There are AYM and FXM-files archive can be found on this site.
- <u>http://www.worldofspectrum.org/projectay</u> Big archive of AY-files.

All other tunes, which I found in Internet, I was including into Tr_Songs archive.

Skin files

You can download skins for Emulator on official Emulator home page. Also you can to make own skins. On Emulator home page you can find program Skin Manager II with instruction for creating skin files. Please, send to me all skins you'll make for my collection. These skins would be placed on official home page. For the moment more than 20 skins are made. Thanks to all skins' authors!

Official Emulator Page

Official Emulator Homepage is found on <u>http://bulba.untergrund.net</u>. There are new versions of Emulator, skins and music archives and also technical information about AY-3-8910/12 and YM2149 chips, music file formats, sources, tools and other can be found on the page. You can download YM-Engine package by Sledge Hammer for using YM sound in your own programs. Also, you can download source package squaretone-0.1.tgz by Martin Andersson to compile player on any non-Microsoft platforms. Vortex Tracker II project is powerful editor for editing AY/YM-music on PC, you can write own tunes in PT3-format in Windows. Download last version from this page.

Contact with author

You can send letters to svbulba@gmail.com, Sergey Bulba.

Message of thanks

This Emulator was born in March 1999. From this time many peoples had help me in my work. Thank to all by especially: Lee Dianhua; Ivan Reshetnikov; Roman Scherbakov aka V Soft; Kirill Kolpakov aka Mr.Kirill; Konstantin Yeliseev aka Hacker KAY; Tomas Hauerland aka TDM; Alexander Romanenko; Ilya Kudryavcev aka Himik's ZXZ; Matthew Westcott aka Gasman/Raww Arse; Ironfist; Martijn van der Heide; Andrey Gayvoronsky; Ivan Yuskin; Milan Matousek; Patrik Rak; Michal Seliga; bcass; Stefano Pucino; Roman Tutkevich aka Midi^EM; Daddy Karlo; Peter Cunderlik; Alex Astafiev; Alex Nikiforov; Alexey Kravchenko; Georgy Terentiev aka George; Kristoph; Maxim Krassa; Mikhail Sudakov; Mike aka Solo; Moroz; Relict/phantasy/k3l; Sledge Hammer of Grinder; SMT; Tanaka Haruhisa; Ulrich Huhn; Vyacheslav Pisarevsky; Ilya Vinogradov aka Siril; Dima Bystrov aka Alone Coder; Demiurge Ash; Polaris; Vitalic Svetashov aka Tash; Eugene Programmer; BlastOff; Macros; Nikolai Tikhonov; Lt. Columbo; Vadim Chumachenko; Vit Hotarek aka X-agon; Nuts; Shiru Otaku; Martin Andersson; Bekes Branislav aka z00m; Adam Klobukowski; Graham Goring; T.A.D.; Alexey Alexeenko aka AAA; Karbofos; Cardinal/AC; Key-Jee; Yerzmyey; Dexus; Mast; Bugsy; Pyza/Illusion; Aectaein aka Skull Chaser; KYV; Max Piwamoto; Andrey Bogdanovich aka Spectre; Alexander Groholsky; Asi; Dioniso Hernandez; Firestarter; Frantisek Fuka; Ivan Roshin; Kevin; Pavel A. Sukhodolsky; Ilya Abrosimov aka EA; Vladimir Anshukov.

Authors

Two men from Russia are making AY emulator from first versions.

Author Sergey Bulba (<u>svbulba@gmail.com</u>).

Graphics Ivan Reshetnikov.

Noise emulation algorithm of AY-3-8910 and YM2149F is written by Sergey Bulba and Hacker KAY.

Two AY-3-8910 and YM2149F output level tables' owner is Hacker KAY.

Different formats players are written by Sergey Bulba and based on original players and (or) descriptions.

©1999–2006 S.V. Bulba.