

This editor is used to compose music in a pattern manner. The music consists of individual patterns and these are then played in the desired order.

**Basic data:**

Minimum equipment required:

- \* Atari XE / XL Compatible (64KB RAM)
- \* MIDI interface (via serial input / output - standard serial connector)
- \* Keyboard Musical Instrument with MIDI (Standard MIDI Standard)

**Editor capacity:**

- \* Song has 256 steps
- \* 32 Samples (Tool and Dynamics Specifications, Chord Possibility)
- \* 128 patterns (pattern consists of 4 tracks and has 32 bars)

**More information:**

- \* 4 MIDI channels (0 to 3) are used for output
- \* For any channel input (Keypad dynamics not tracked)
- \* Control of the program is performed by a "hot" key system, ie by pressing the key directly you can call up the corresponding function.

**PATTERN:**

Use this mode to write notes to the individual patterns that make up the music. The lines represent bars, there are tracks in four columns. Each track has a fixed MIDI channel (0 to 3) and the notes placed in it will be sent just for that channel.

Each note starts to sound when the program hits it and continues to sound. It ends when the "(" or a new note is found.

**Example:**

```
00 | C-3 00 | E-3 01 | --- 00 | C-4 05 |  
01 | --- 00 | --- 00 |  
02 | --- 00 | ( ) 00 | --- 00 |  
03 | 00 | --- 00 | --- 00 | ( ) 00 |  
04 | 00 | --- 00 | --- 00 | --- 00 |
```

**Channel 0:**

The note C-3 starts to sound (instrument specification, etc., given by sample number 00) and sounds for 3 bars. Nothing sounds in the fourth bar (line 03). The "end" symbol on line 04 causes the entire pattern to have a total of 4 bars.

**Channel 1:**

The note E-3 starts to sound (tool specification, etc., given by sample number 01) and sounds for 2 bars. Nothing sounds from the third bar.

**Channel 2:**

One space gap - nothing sounds. In the second bar, the G-3 note (sample 07) starts to sound and continues to sound (even after this pattern is over). It only ends when another note or "()" symbol (in this track) interrupts it.

**Channel 3:**

The note C-4 (sample 05) starts to sound and 3 bars sound. The "spd 09" symbol in line 02 will cause the playback speed to change (all tracks!). From now on, one beat will be 9/50 seconds. "spd" can be placed in any track (where you have space) and the new speed is valid until the new "spd" changes.

**G**

Go to SONG mode.

**M**

Go to SAMPLE mode.

**Inverse**

Enable / Disable Data Receiving from the Keyboard Tool.

**ARROW-up**

Scroll up (clockwise).

**ARROW-down**

Move down (forward clock).

**ARROW-left**

Move to the left (also go to the left track).

**ARROW-right**

Move right (also go to the right track).

**Tab**

Go to the right track.

**Return**

Go to the beginning of the pattern.

**0..9, A..F**

Write a number at the current cursor position (HEX - hexadecimal system).

**Del**

Erasing both digits numbers on act. position and move down.

**Shift+ARROW-up**

**Shift+ARROW-down**

Set the current sample (change after 1).

**Shift+ARROW-left**

**Shift+ARROW-right**

Set the current pattern (change by 1).

**Control+ARROW-left**

**Control+ARROW-right**

Set the current pattern (change after 16).

**L**

Set the instrument tone (on the Ach channel) and play the current sample (while the key is held) in C-5 (according to the data in the current sample).

- AND -

Initialize bar clock (for play). Performs the setting from the 0th sample (see below).

**T**

Write "( )" to act. place. This symbol marks the end of the currently sounding tones in the track (ie, using the corresponding keys).

- WITH -

Writing "spd" to act. place. The number after this symbol indicates a new pause value for one beat (valid for the beat in which this symbol is).

**N**

Writing "end" to 0.-th track. Indicates the end of the pattern (this symbol is added to the first invalid bar).

Is it used for patterns that have less than standard 32 bars. (The "end" program only watches in 0.track, ignores it in others!).

**spacebar**

Write "--- 00" to current location and move down.

**Playing a song:**

**O**

Start playing the current pattern (from the current bar).

**P**

Starting a song play from the beginning of the pattern to the current line in the SONG mode (then continues with the next lines).

**R**

Playback of the entire song (from the beginning of the Song).

**Any key**

Interrupting play.

**Instrument Keyboard record:**

If data input from the instrument keyboard is turned on (see inversion), each time the key is pressed, the note is written to the current location (ie, the current pattern, track, and bar). The next line moves. This also applies to simultaneous playback of a song (or pattern) so that REAL TIME RECORDING is also enabled!

**Block operations:****Shift+T**

Save the current track to the buffer.

**Control+T**

Overwrite current track with track from buffer.

**Shift+P**

Save the current pattern to the buffer.

**Control+P**

Overwrite the current pattern with a pattern from the buffer.

**Shift+Control+del**

Delete the current track.

**Shift+Control+clr**

Delete the current pattern.

**Recording and other operations:****Shift+Control+W**

Writing an entire song to a disc (including song and samples).

The number of patterns will be derived from the highest number used in the SONG.

**Shift + Control + R**

Loading a song from a disc (overwrites the current song).

**Shift + Control + D**

Delete everything (song, all samples, and patterns).

### **Shift + Control + Q**

Leaving the program - going to DOS.

### **SONG Mode:**

Use this mode to build a track from each pattern. Numbers on individual lines mean:

\* 00 to 7f .. the pattern number to be played.

\* fe .. end of song (+ end of all sounding tones).

\* ff .. transition to the 0 th line of the SONG and resume playback.

### **Esc**

Return to PATTERN mode.

### **ARROW-Up, Down, Left, Right**

Scroll in the appropriate directions.

### **Bk Sp**

Move everything under this line one position up.

>

Move everything below this line (including this line) one position down (insertion point).

### **0..9, A..F**

Write the number at the current position.

### **Return**

Go to zero line (ie, at the beginning of the song).

### **Tab**

Move to next next eighth row.

### **SAMPLE mode:**

Use this mode to define the parameters for the compressed note. Allows you to specify the instrument to which the note is played, its volume, and possibly the chord derived from that note.

Each sample contains **8 parameters** as follows:

---

#### **Parm 1: 00**

Leave = 00 (reserved for eventual extension)

#### **Parm 2: To 00**

Tool Number 00 to 7f indicates which instrument the note should be played on.

**Parm 3: Ba 00**

Bank number (If the tool only has multiple banks. Otherwise = 0). Pozn. Because bank switching varies from tool to tool, this feature does not work in the basic version.

**Parm 4: V1 00**

Stroke dynamics (volume) of the 1<sup>st</sup> note (ie, notes recorded in the pattern). Allowed range 01 to 7f (00 means silence).

**Parm 5: 2+ 00**

Transposition of the 2<sup>nd</sup> note from 1. Expresses the shift (in semitones).

Additional number: ..., fd = -3, fe = -2, ff = -1, 00 = + 0, 01 = + 1, 02 = + 2, ...

**Parm 6: V2 00**

Stroke Dynamics (Volume) of the 2<sup>nd</sup> note. The allowable range of 01 to 7f (00 means that only the 1st moment will sound).

**Parm 7: 3+ 00**

Transposition of the 3rd note from 1. Expresses the shift (in semitones).

**Parm 8: V3 00**

Throw Dynamics (Volume) 3.Notes. The allowable range of 01 to 7f (00 means that only the 1<sup>st</sup> and 2<sup>nd</sup> days will sound).

*Note: A special exception is the number 00 sample, which differs by the 00 parameter. When you start playing the entire song (with the 'R' key), the current time for one beat is set with this value. This time will only change if a pattern in which the "spd" symbol is of a different value is played.*

*You can also use the 'I' key to initialize the value for the cycle length.*

---

**Esc**

Return to PATTERN mode.

**ARROW-Up, Down**

Scroll up and down. Moving to another sample area will make this sample the current sample.

**ARROW-Left, Right**

Scroll left and right.

**0..9, A..F**

Write the number at the current position.

**Return**

Transition to zero line (ie, zero zero).

## **Tab**

Move to next next eighth row (ie next sample).

## **L**

Play the current sample. It works the same as 'L' in PATTERN mode

## **Notes and important alerts:**

### **1.) Never press RESET**

In principle, use the Shift + Control + Q function and never the RESET button to exit the program.

Communication with a MIDI tool works on a messaging basis - if you press RESET, you will forcibly interrupt the program and some of these messages may not be delivered properly. This can result in so-called "hanging" tones (some tones remain on your instrument). The starting point of this state can then be the RESET button on the musical instrument (if equipped with this button) or the MIDI mode off and on. For some tools, the only way out of this situation is to turn the tool off and on again.

### **2.) Play a song**

Always use the 'R' function to play the entire song regularly. Only this function performs the initialization of the cycle length before the actual start (the 0th parameter in the sample number 00 - see SAMPLE mode). When triggered by the 'P' or 'O' functions, only the respective part is played without this setting and the measure length is left as it was from the previous playback.

### **3.) Starting MPE**

The initialization cleanup of memory is not performed after MPE is started. Therefore, perform the "Delete all" function (Control + Shift + D) before starting work. This tiny fact makes it possible to reload it without loss (or just a small loss of some part) of the compiled song when the program crashes.

### **4.) Module Portability (.MPE)**

The music you create is saved in the .MPE format, containing the song, all samples, and the necessary (in the song used) patterns. However, it is not possible to ensure that this module is played on another musical instrument when playing a module created on a given type of instrument. In particular, the problem lies in the difference between assigning sound samples to individual numbers on different MIDI instruments. Another difference can be the differently interpreted keystroke dynamics, caused by the use of another volume intensity curve for given dynamics values. This can be corrected with a little skill by changing the appropriate values in the samples used (that is, changing the tool number to match the desired; or changing the value for the dynamics so that the voice is the appropriate volume).

The biggest difficulty, however, is with the drum samples used, because the individual notes then mean different percussion instruments. Modifying a module for another MIDI instrument is very complicated in this respect and requires a lot of effort (it is necessary to change individual notes in all patterns).

## **5.) Program location in memory**

The MPE music editor is located in memory from \$ 1F00 and should work under most commonly used DOSs. The actual work area used is the rest of the conventional memory, that is, up to \$ BFFF (that is, including memory under BASIC ROM). The area under the ROM (\$ C000 to \$ FFFF) is no longer in use and can be occupied by RAM disk.