

# MIDIbox VGM Player

This small synth is a demo of the [MBHP\\_Genesis](#) module ([available now from the MIDIbox Shop!](#)).

## Hardware Features

- Supports one to four [MBHP\\_Genesis](#) modules, allowing the user to play that number of VGM files simultaneously (one on each board)
- OPN2: supports YM2612 and YM3438
- PSG: supports SN76489 and SN76496 directly, SN76494 with external clock generator
- Simple front panel interface
- Entire synth can be powered from USB

## VGM Playback Engine Features

- Stream VGM data from SD card; supports Sega Genesis / Mega Drive, Sega Master System, Sega Game Gear, and any other VGM file that includes a YM2612 or SN76489
- Advanced multithreaded system for optimal performance
- Corrects OPN2 frequency commands in real time (hardware OPN2 clocked at 8 MHz rather than 7.67 MHz)
- Corrects PSG noise zero frequency issue to match Sega Genesis embedded PSG behavior
- Supports block and embedded samples
- Seamless VGM looping
- Simple API-project is contained in one short app.c file

## Construction

You need:

- STM32F4 core with SD card
- 2×20 or larger character LCD
- Small custom front panel board with 1 encoder and 6 buttons (see below)
- One to four [MBHP\\_Genesis](#) modules
- One [MBHP\\_Genesis\\_LS](#) level shifter board

## Front panel board

The front panel board has a single 74HC165 DIN shift register, with the following connections:

- Pin 0: Encoder A
- Pin 1: Encoder B
- Pin 2: Enter button (currently “Play”)

- Pin 3: Menu/Back button (currently “Stop”)
- Pin 4: Softkey 1
- Pin 5: Softkey 2
- Pin 6: Softkey 3
- Pin 7: Softkey 4

The four softkeys each correspond to five characters on the LCD display's bottom row.

## Software

An [official build](#) is available now on SVN! Report bugs to me on the forums.

From:

<http://midibox.org/dokuwiki/> - **MIDibox**

Permanent link:

[http://midibox.org/dokuwiki/doku.php?id=midibox\\_vgmplayer&rev=1459616409](http://midibox.org/dokuwiki/doku.php?id=midibox_vgmplayer&rev=1459616409)

Last update: **2016/04/02 17:00**

