MP3 Trigger v2 User Guide
2012.02.01

Overview

The MP3 Trigger v2 is a versatile, low-cost, low-power embedded audio unit that plays MP3 tracks directly from a FAT32 or FAT16 formatted microSD flash card to a stereo 1/8" headphone output jack, supporting up to 192kbps stereo playback. The board has 18 external input pins that when pulled to ground, trigger pre-selected MP3 tracks, and a full-duplex serial control port that provides real-time volume control as well remote triggering for up to 255 tracks. There is also an on-board navigation switch for local access and playback of all MP3 tracks on the flash card.

Version 2.40 firmware (and above) supports the use of an initialization file on the microSD card that can be used to change the serial baud rate, as well as to repurpose any of the 18 trigger inputs to alternate functions, such as random and sequential triggers, navigation controls and even volume controls. In addition, a restart lockout option can be used to prevent any trigger from starting a track if audio is already playing. Using these features, custom applications can often be implemented without the use of a separate microcontroller.

The MP3 Trigger v2 has a resident, permanent boot-loader that enables firmware upgrades directly from a microSD card without the need for a hardware programming device.

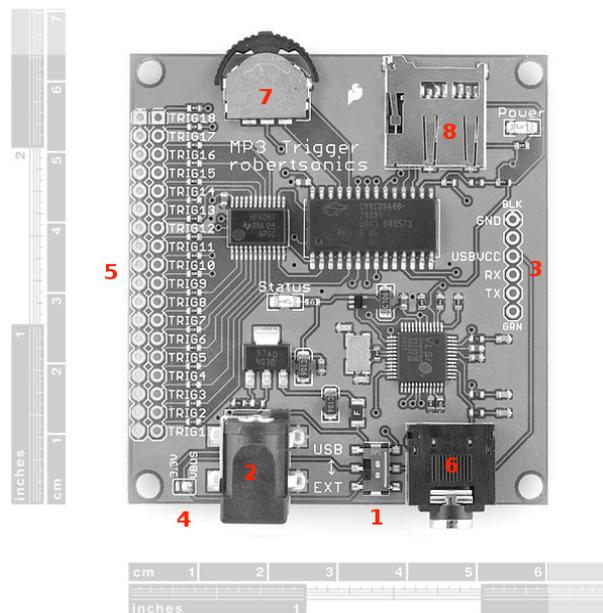The basic layout of the MP3 Trigger v2 can be seen in Figure 1 below.

Figure 1

1) USB or external power switch

2) External power connector (5.5x2.1mm center positive)

3) UART and USB power interface for the FTDI Basic (DEV-09115)

4) Power selection jumper

5) Trigger pins

MP3TRIGGER_UG_090917

6) 1/8" stereo headphone jack

7) Navigation switch

8) microSD socket

Specifications

- Input Voltage Range: 4.5V to 12.0V DC, or regulated 3.3V (jumper selectable)

- Current Consumption: Approximately 45mA idle, 85mA playing

- Media: SDSC and SDHC microSD cards

- File system: FAT32 and FAT16

- Audio output: Headphone stereo (1/8" stereo jack)

- Trigger inputs: Logic level 3.3V – 5.0V, active low inputs, w/ internal pull-ups. (Connector provides individual grounds, allowing switches or jumpers to be connected directly to each trigger input)

- Serial: Full duplex, 8-bit, 38.4Kbaud (default, other baud rates supported via initialization file)

Figure 2

Basic Operation

Simply drag and drop the desired MP3 files into the root directory of a FAT32 or FAT16 formatted microSD flash card using a PC.

The MP3 Trigger v2 does not support hot-swapping of the microSD Card. While this won't damage anything, the microSD media is only initialized during power up. So whenever the card is changed or updated, be sure to power cycle the MP3 Trigger v2 after installing the card.

When power is applied to the MP3 Trigger v2, the on-board (green) status LED indicates the state of the installed media as follows:

1 long blink - No formatted microSD media found.

MP3TRIGGER_UG_090917

1 long blink, followed by 1 short blink - microSD media found, no MP3 files located.

Constant short blinks - Hardware problem with MP3 Decoder.

3 short blinks - microSD media found, at least 1 MP3 file located.

As soon as the MP3 Trigger powers up with 3 short blinks, the on-board navigation switch can be used to play all of the tracks on the card, regardless of the filenames.

Left - Plays the previous MP3 file in the directory

Right - Plays the next MP3 file in the directory

Center - Starts/Stops the current MP3 file

Using the Trigger Inputs

The MP3 Trigger v2 provides 18 input pins (TRIG01 – TRIG18) that can be used to trigger specific MP3 tracks on the microSD card. MP3 tracks are associated with triggers by placing a 3-digit number (using leading 0s) at the beginning of the filename; 001 for TRIG01, 002 for TRIG02 and so on. The rest of the filename can be anything. For example, the following are both valid names for TRIG14:

"014TRACK.MP3"

"014 Breaking Glass.mp3"

The trigger inputs are active low and pulled high internally. Therefore, they can be activated either by digital outputs from another microcontroller (such as an Arduino) or by a simple contact closure (switch) to ground. The inputs support voltage levels of either 5V or 3.3V.

The trigger inputs are made available on the even-numbered pins of a dual row connector, and all the opposing (odd-numbered) pins are ground, making it easy to wire individual switches or contact closures directly to the MP3 Trigger v2 board.

Installing a 36-pin dual-row header allows shunt jumpers to be installed on the trigger inputs to automatically sequence and loop tracks on power-up as follows.

When a triggered track reaches the end, the MP3 Trigger v2 looks to see if any trigger inputs are active, and will automatically start another track if so. If only the same trigger is active, then that track will restart (loop). If other triggers are active, the MP3 Trigger v2 will always start the next higher trigger track, wrapping back to 1 after 18.

This, combined with the fact that the MP3 Trigger v2 will automatically start the lowest numbered active trigger on power up, means that by installing shunt jumpers on the trigger inputs, the MP3 Trigger v2 can be set to automatically sequence and loop from 1 to 18 tracks on power up with no externally programming or control required. (Beginning with firmware version 2.40, installing a shunt jumper on a single sequential trigger will do the same thing.)

Using the initialization file describe later in this document, triggers can be reprogrammed to start sequential or random tracks. For example, if a trigger is reprogrammed to be a random trigger, and that one trigger is shunted closed, then the MP3 Trigger v2 will power up and continuously play tracks in a (pseudo) random order.

MP3TRIGGER_UG_090917

Quiet Mode

The MP3 Trigger v2 can be placed into Quiet Mode using the serial control port. In this mode, the trigger inputs will not start tracks but instead will cause serial messages to be sent upon activation. (See "MP3 Trigger v2 Outgoing Message Summary" below.) This allows the trigger inputs to be decoupled from specific tracks, so that a PC or microcontroller can monitor the trigger inputs and then start any track or sequence of tracks via the serial control port.

Quiet Mode is off by default and is not preserved through a power cycle.

Serial Control Protocol

The MP3 Trigger v2 comes with a full duplex 3.3-5V serial TTL interface that allows for control of all the MP3 tracks (up to 256) on the microSD card as well as volume, and for monitoring input trigger activity. You can use our FTDI Basic (DEV-09115) or connect to any serial interface that uses the format: 8-bits, 1-start, 1-stop, no parity, flow control = none. The serial port baud rate defaults to 38.4kbps, but can be changed using the initialization file. All commands to the MP3 Trigger v2 are 1 or 2 bytes in length.

1-byte commands are upper case ASCII characters.

2-byte commands start with an ASCII character. Those starting with an upper case character use an ASCII value ('0' – '9') as the second byte. (These commands can be typed on a keyboard.) 2-byte commands starting with a lower case character require a binary value (0 – 255) as the second byte.

Bytes sent to the MP3 Trigger v2 are not echoed. If echoing is required, set your terminal program to echo locally.

Command Summary

Command: **Navigation – Start/Stop**
Number of bytes: 1
Command byte: 'O'
Data byte: none
Comments: This command performs the same function as pushing the on-board nav switch center position. If the current track is playing, it stops. If the current track is stopped, it will restart from the beginning

Command: **Navigation – Forward**
Number of bytes: 1
Command byte: 'F'
Data byte: none
Comments: This command performs the same function as pushing the on-board nav switch right position. The next MP3 track in the directory will be started.

Command: **Navigation – Reverse**
Number of bytes: 1
Command byte: 'R'
Data byte: none
Comments: This command performs the same function as pushing the on-board nav switch left position. The previous MP3 track in the directory will be started.

MP3TRIGGER_UG_090917

Command: **Trigger (ASCII)**
Number of bytes: 2
Command byte: 'T'
Data byte: N = ASCII '1' through '9'
Comments: If it exists, the track with the filename "00Nxxxx.MP3" will be started, where N is the data byte. xxxx can be any valid filename characters of any length.

Command: **Trigger (binary)**
Number of bytes: 2
Command byte: 't'
Data byte: n = 1 to 255
Comments: If it exists, the track with the filename "NNNxxxx.MP3" will be started, where NNN is the ASCII equivalent of the data byte n with leading 0s. xxxx can be any valid filename characters of any length.

Command: **Play (binary)**
Number of bytes: 2
Command byte: 'p'
Data byte: n = 0 to 255
Comments: If it exists, the nth track in the directory will be played. The total number of available tracks in the directory can be retrieved using Status Request command below.

Command: **Set Volume (binary)**
Number of bytes: 2
Command byte: 'v'
Data byte: n = 0 to 255
Comments: The VS1053 volume will be set to the value n. Per the VS1053 datasheet, maximum volume is 0x00, and values much above 0x40 are too low to be audible.

Command: **Status Request (ASCII)**
Number of bytes: 2
Command byte: 'S'
Data byte: N = ASCII '0' through '1'
Comments: If N = '0', the MP3 Trigger will respond with a version string. If N = '1', the MP3 Trigger v2 will respond with the total number of tracks on the installed microSD card, in ASCII. Both responses will be preceded by the '=' character.

Command: **Quiet Mode (ASCII)**
Number of bytes: 2
Command byte: 'Q'
Data byte: N = ASCII '0' or '1'
Comments: If N='1', Quiet mode is turned on. If N='0', Quiet mode is turned off. Default state is off.

MP3TRIGGER_UG_090917

MP3 Trigger v2 Outgoing Message Summary

The MP3 Trigger v2 sends the following ASCII messages:

      'X': When the currently playing track finishes.
      'x': When the currently playing track is cancelled by a new command.
      'E': When a requested track doesn't exist (error).

In response to a Status Request Command, data byte = '0', the MP3 Trigger v2 sends an 18-byte version string: e.g. "=MP3 Trigger v2.50" .

In response to a Status Request Command, data byte = '1', the MP3 Trigger v2 sends the number of MP3 tracks on the currently installed microSD card: e.g. "=14".

In Quiet Mode only, when one or more trigger inputs are activated, the MP3 Trigger v2 sends 'M' followed by a 3-byte bit mask indicating which triggers were activated:

      Data byte 0: TRIG01 through TRIG08
      Data byte 1: TRIG09 through TRIG16
      Data byte 2: TRIG17 and TRIG18

A value of 1 in a bit position indicates that the corresponding trigger input was activated.

Initialization File

Version 2.40 firmware (and above) supports the use of an initialization file to change some of the operation parameters of the MP3 Trigger v2 upon power up. This file is ASCII text only, and can be created and edited with any text editor such as Notepad.

The initialization file must be named "MP3TRIGR.INI" and must, like all the mp3 files, be in the root directory. The file is optional. If it does not exist, then the MP3 Trigger v2 defaults to normal operation at 38.4K baud and all triggers starting their respect tracks.

Initialization file commands must begin with the "#" character and be followed by a space. The initialization file example on the following two pages is self-documented and describes the commands currently supported:

MP3TRIGGER_UG_090917

Sample Initialization File

```
#BAUD 38400
#RAND 2
#TRIG 01, 0, 0
#TRIG 02, 0, 0
#TRIG 03, 0, 0
#TRIG 04, 0, 0
#TRIG 05, 0, 0
#TRIG 06, 0, 0
#TRIG 07, 0, 0
#TRIG 08, 0, 0
#TRIG 09, 0, 0
#TRIG 10, 0, 0
#TRIG 11, 0, 0
#TRIG 12, 0, 0
#TRIG 13, 0, 0
#TRIG 14, 0, 0
#TRIG 15, 0, 0
#TRIG 16, 0, 0
#TRIG 17, 0, 0
#TRIG 18, 0, 0

******************** ALL INIT COMMANDS ABOVE THIS LINE ********************
```

This is a sample init file for the MP3 Trigger v2, firmware version 2.40.

The init file is optional. If not present, the default parameters will be in effect: 38.4Kbaud, and all triggers will start their corresponding tracks with restart lockout disabled. If it is present, it must be named MP3TRIGR.INI and be located in the root directory.

Only the first 512 bytes of the file are examined for commands, and the first occurrence of the '*' character is treated as the end of file by the parser. Comments are not allowed in the command section, but there is no restriction on the length of the comments that follow the first '*'.

All commands must begin with the '#' character and be followed by a space, then the command parameters separated by commas. White space is ignored. All parameters are decimal numbers. Leading zeros are acceptable. See the above examples – which are redundant since they are all default values.

The following commands are supported in firmware version 2.40:

#BAUD N

    where N is one of the following: 2400, 9600, 19200, 31250 or 38400

#RAND N

    where N is from 1 to 255

    The default behavior of the random trigger function is to play a random track from all the MP3 files on the flash card. The #RAND function will

MP3TRIGGER_UG_090917

exclude the first N tracks (in the directory) from the random trigger
function. So if there are 18 MP3 files on the card and N=4, then the
first 4 MP3 files will be excluded from the random trigger function.

#TRIG N, F, L

    where: N is the trigger number (1 - 18)
    F is the trigger function type (see below)
    L is the restart lockout enable

    The defined trigger function types (F) are as follows:

    F = 0: Normal operation
    F = 1: Next (same as the forward Nav switch)
    F = 2: Random
    F = 3: Previous (same as the back Nav switch)
    F = 4: Start (restarts the current track)
    F = 5: Stop
    F = 6: Volume Up
    F = 7: Volume Down

    The restart lockout feature, if enabled, will prevent that trigger
    from working if audio is currently playing. Use this if you want
    to prevent restarts before the track has reached the end. This
    feature does not apply to function types 5 - 7.

    L = 0: Restart lockout disabled (default)
    L = 1: Restart lockout enabled

You only need to include entries for triggers that are to be non-default.
As an example, I use the following single-line init file to make trigger
18 be a "Next" function, then hard-wire the trigger so that my MP3 Trigger
powers up and loops continuously through all the tracks on the card.

#TRIG 18, 1, 0

8

MP3TRIGGER_UG_090917

Bootloader

The MP3 Trigger v2 has a resident boot-loader that allows updating the firmware directly from the microSD card, alleviating the need for a hardware programmer. Because this boot-loader is in located in protected sectors of the PSoC's flash memory, it cannot overwrite itself. The boot-loader can always be run on power up, thus making it possible to recover from a bad firmware load.

IMPORTANT NOTE: Use of a hardware programmer, such as the Cypress MiniProg, to program the MP3 Trigger v2 with anything other than the boot-loader image will erase the boot-loader. Don't do it!

Using the boot-loader

To update the MP3 Trigger v2 firmware, copy the new firmware hex file to a FAT16 or FAT32 formatted micro-SD card and rename the file to "MP3TRIGR.HEX". It doesn't matter if it's the only file on the microSD card or not - the boot-loader will find it as long as it has this exact filename. Insert the microSD card into the MP3 Trigger v2 with the power off. Hold down the center nav switch while turning on the power. Wait for the status LED to go solid, then power cycle the MP3 Trigger v2 to run the new firmware.

Here's how the boot-loader works in more detail:

The boot-loader is always entered whenever the board powers up. The first thing it does is look to see if the center nav switch is being held down. If not, it immediately vectors to the start of the firmware. Note that if you have previously loaded bad firmware, the board will simply halt or do whatever your bad code tells it to do – possibly with no activity other than the power LED. This is normal if there's no good firmware loaded.

If the center nav switch is being held on power up, the boot-loader searches the microSD card directory for a file named "MP3TRIGR.HEX". If there's no card installed, or the file doesn't exist on the card, it will blink the status LED very rapidly forever. If it finds and is able to open the file, it begins to program the PSoC Flash with the contents of the firmware file. The status LED will illuminate for each hex record programmed. If it successfully programs the entire file, the status LED will turn solid upon completion. You can then power cycle the MP3 Trigger v2 and you will be running the new firmware (don't hold the nav switch down again or you will simply re-enter the boot-loader.

If there's a flash memory programming error (or you pull the microSD card out before it finishes, for example) the status LED will indicate a bad programming cycle by blinking briefly at about 1 Hz (This looks very different than the blinking for programming records). You need to cycle the power again holding the nav switch to re-enter the boot-loader.

The key is that the boot-loader cannot over-write any part of itself, no matter what's in the firmware image file. The worst thing that can happen is you load bad firmware and the board won't run. But you can always hold the nav switch down on power up and get into the boot-loader to load new firmware.

MP3TRIGGER_UG_090917