PICAXE COMPATIBLE MP3 PLAYER

Overview:

The PICAXE compatible MP3 player allows playback of MP3 sound files stored on a compact flash card. The controlling PICAXE circuit has full control of the player and can easily select, start, stop and pause playback of the files.

Also required (not included):

Compact Flash Memory Card
PICAXE controller circuit
Amplifier & Speakers

Specification:

- Compatible with any PICAXE microcontroller.
- Small size, only 50x60 mm. Card protruding 14 mm on long edge.
- High quality 4-layer PCB for low EMI.
- Compact Flash card based. Using standard PC formatted FAT16 cards. Card can be up to 4GB, allowing more than 70 hours of continuous, unrepeated near CD quality music (at 128 kbps).
- Easy and fast firmware upgrade through a file on the CF card.
- Serial interface for remote control of the player (like RS-232, but at 3.3V levels).
- BUSY status output line (high signal when file is playing).
- Flexible 4.5V-15V DC supply voltage.
- Line level output for connection to an external amplifier / speakers.
- Low power consumption. Approx. 40 mA (operating, with Sandisk 128 MB card).
- Plays all MP3 bitrates up to and including 320kbps.

General Operation:

1. Download the MP3 files onto the compact flash memory card. Ensure the files are correctly named with the file number. Don't steal copyrighted music.
2. Insert the compact flash card into the MP3 player.
3. Connect the amplifier/speakers to the MP3 player line output.
4. Connect the MP3 player to the PICAXE system.
5. Connect the power supply.
6. Develop and download a PICAXE program to play the MP3 files.

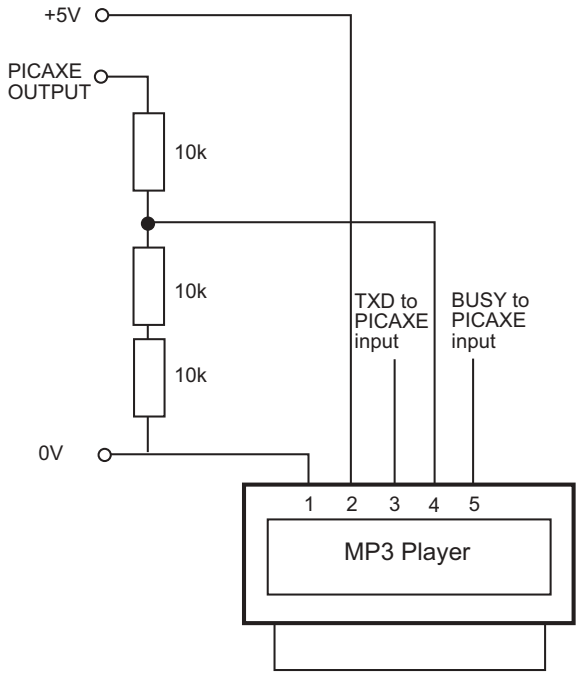
MP3 Player Pin Connections:

1.	GND	0V
2.	VIN	4.5V to 15V DC
3.	TXD	Serial signal from MP3 player to PICAXE (optional)
4.	RXD	Serial signal from PICAXE to MP3 player
5.	BUSY	Busy signal from MP3 player to PICAXE (optional)
6.	RIGHT OUT	Right audio channel
7.	GNDA	Audio ground
8.	LEFT OUT	Left audio channel

Power Supply and PICAXE Connection:

The MP3 player will operate with a supply voltage of 4.5V to 15V DC. However the player internally operates at 3.3V as it contains an on-board voltage regulator. Therefore all signals to, and from, the MP3 player **must** be at 3.3V.

Most PICAXE users will therefore power the PICAXE circuit and MP3 player from a single 4.5V or 5V DC supply. It is therefore essentialto **step down the voltage from the PICAXE output pin** before connection to the serial input (RXD) pin of the MP3 player. This is achieved by use of a simple potential divider circuit as shown in the diagram.



The signal from the MP3 players BUSY and TXD outputs will be at 3.3V. However in most cases this will be sufficient to trigger the PICAXE input and so no voltage amplification is generally required.

In the simplest form of connection the MP3 player can be controlled by a single PICAXE output pin, connected (via the potential divider arrangement) to the MP3 players RXD pin. If PICAXE feedback that the file is still playing is required, it is recommended that the MP3 player's BUSY signal is used, via connection to a PICAXE input. This can then be used, for example, to trigger a PICAXE interrupt when the playing finishes.

For advanced use the MP3 players TXD pin can be connected to a second PICAXE input pin to receive and interpret the serial data responses from the MP3 player (via the 'serin' command).

Initialisation and power-on delay:

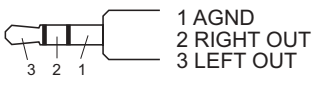
The MP3 player requires approximately 1 second to initialise, (BUSY is high during this time). Therefore each PICAXE program should always commence with this sequence

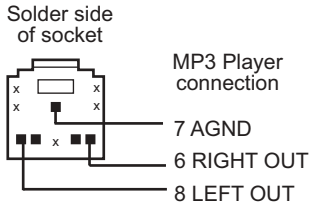
```
symbol TXD = 1   ' select the correct PICAXE serial output pin
high TXD         ' set serial output to default high state
pause 1000       ' pause 1 second
```

This puts the serial line into the correct default (high) status and allows the player to complete initialisation before serial data is transmitted to it from the PICAXE chip. Important - if the initial 'high' command is omitted it is likely that the first 'serout' command to the player will not be responded to. When using a very large memory capacity compact flash card the power up delay may need to be extended.

Amplifier & Speakers:

The audio output from the MP3 player is at line voltage (1.8V pp). This can then be fed into any amplifier/speaker arrangement. For experimentation and testing it is recommended that a 'computer style' set of speakers are used. These are low cost and convenient to use, as they are designed to operate at line voltage.





These speakers generally have a 3.5mm stereo plug for connection. The diagram shows how to connect a suitable 3.5mm socket (e.g. part CON039) to the MP3 player. This allows the speakers to be easily connected to the board.
NEVER CONNECT A PICAXE DOWNLOAD CABLE INTO THIS AUDIO SOCKET

MP3 File Information:

- The MP3 player does not support recording of files onto the compact flash card. It is purely a MP3 playback device.
- The MP3 player does not contain any interface mechanism for downloading of the MP3 files onto the compact flash card. Therefore the MP3 files must be downloaded onto the compact flash card before use in the MP3 player. This is normally achieved by use of a separate USB style 'memory card reader' (e.g. part USB020).
- The compact flash card must be formatted in standard PC 'FAT16' format. No folders are supported, i.e. all files must be in the root directory of the card.
- Each MP3 file must be named with a decimal file number from 001.MP3 through to 511.MP3. Each file name must start with three digits (e.g. 010.MP3 not 10.MP3). Only the first three digits are used by the player – the rest of the filename is not relevant (e.g. 001songname.mp3 is acceptable). Do not use the filename 000.MP3, as this is not supported within the PICAXE version firmware.
- When a small number of files are used the filenames 001.MP3 to 127.MP3 are recommended, as these can be directly addressed with a single byte command.
- The MP3 files can be in any up bitrate format, up to and including 320kbps. Note that higher bitrates give better quality, but create a large file size.

Serial Commands:

The full datasheet (pages 6-12) provides detailed information for all the serial commands. However the most important commands are summarised here:

Decimal	Hex	Command
1 to 127	$01 to $FE	Direct play (file numbers 001 to 127 only)
128 to 159	$80 to $9F	Set sound volume (very low– 128 ($80) is mute)
160 to 184	$A0 to $B8	Set sound volume (audible output range) *Always leave a small pause after volume change.*
240	$F0	Stop playing
248	$F8	Pause
249	$F9	Resume (after pause)
224,xx	$E0,xx	Play file 000 to 255 (where xx = 000 to 255)
225,xx	$E1,xx	Play file 256 to 511 (where xx = file number – 256)

Note that there are two ways to play files numbered 001 to 127 – either directly addressed or via the $E0 prefix. Therefore, when only a small number of files are used, it is convenient to ensure they are numbered between 001 and 127 so that they can be directly addressed. File numbers 128 to 255 can only be played via the $E0 prefix, and files numbered 256 to 511 must use the $E1 prefix.

The BASIC command for use with these commands is 'serout' e.g.

```
symbol TXD = 1              ' set serial output pin
high TXD                    ' output pin high
pause 1000                  ' power up delay
serout TXD,T2400,($B8)      ' set volume to maximum
pause 100
serout TXD,T2400,($01)      ' play file number 01
pause 3000
serout TXD,T2400,($F8)      ' pause
pause 1000
serout TXD,T2400,($F9)      ' resume
pause 3000
serout TXD,T2400,($F0)      ' stop playback
```

Example Program 1 – Serial Output only

This program plays ten seconds of music from two different files (file 001 and 002). Naturally these files must be longer than ten seconds for this program to function correctly. It only requires a single PICAXE output.

```
symbol TXD = 1                  ' PICAXE serial output pin

init:
      high TXD                  ' serial pin high
      pause 1000                ' wait 1 second
main:
      serout TXD,T2400,($01)    ' play file 001
      pause 10000               ' wait 10 seconds
      serout TXD,T2400,($F0)    ' stop command
      pause 10                  ' short delay
      serout TXD,T2400,($02)    ' play file 002
      pause 10000               ' wait 10 seconds
      serout TXD,T2400,($F0)    ' stop command
      pause 1000
      goto main
```

Example Program 2 - Serial Output and BUSY feedback

This program plays a file until it has been completed. It then automatically re-plays the file after a 1 second delay. Note that the BUSY signal from the MP3 player is a on/off signal, high level when the file is playing.

```
symbol TXD = 1                  ' PICAXE output pin
symbol BUSY = input2            ' PICAXE input pin

init:
      high TXD                  ' serial pin high
      pause 1000                ' wait 1 second
main:
      serout TXD,T2400,($01)    ' play file 001
loop:
      if BUSY = 1 then loop
      pause 1000
      goto main
```

Example Program 3 – Serial Output with Serial Response

This program reads the error byte ($F6) from the player and displays the returned value on screen (in variable b1).

```
symbol TXD = 1                  ' PICAXE serial output pin
symbol RXD = 2                  ' PICAXE serial input pin

init:
      high TXD                  ' serial pin high
      pause 1000                ' wait 1 second
main:
      serout TXD,T2400,($F6)    ' send 'get errors' command
      serin RXD,T2400,b1        ' receive reply byte
      debug b1                  ' display on screen
      pause 1000                ' wait 1 second
      goto main
```

Firmware Update and Serial Calibration

The firmware on the MP3 player can be updated to the latest version by inserting a compact flash card with the latest firmware file (file must be called YAMPPIND.BIN) upon powerup. This must be the only file on the Compact Flash card. For these upgrade procedures it is presumed that the PICAXE chip is connected to the MP3 player (via the potential divider) and speakers are connected to the MP3 player.

The two following files are available for download on the software page at www.picaxe.co.uk

YAMPPIND.BIN latest firmware version
YAMPPIND.CALIB latest calibration firmware

Upgrade procedure:

1) Download this program onto PICAXE chip.

```
symbol TXD = 1 ' NB set output pin to pin you have used!
high TXD
stop
```

2) Power down PICAXE chip/MP3 player.
3) Copy the YAMPPIND.BIN file onto compact flash card & insert into MP3 player.
4) Power up and leave for 5 seconds.
5) Remove power. Remove compact flash card. Delete .BIN file from card. Copy MP3 files onto card & reinsert.
6) Power up and use as normal.

The MP3 player can also be calibrated to the serial baud rate from the PICAXE chip via use of calibration firmware and a special test program (the unit is pre-calibrated prior to despatch and so calibration is normally optional). This can be useful when using PICAXE chips with internal resonator, as the internal resonator of the MP3 player is then 'synchronised' to the PICAXE chip's internal resonator.

Calibration Procedure:

1) Download this program onto PICAXE chip.

```
symbol TXD = 1 ' NB set output pin to pin you have used!
high TXD
stop
```

2) Power down PICAXE chip/MP3 player.
3) Copy the YAMPPIND.CALIB file onto compact flash card & rename to YAMPPIND.BIN Insert into MP3 player.
4) Power up and leave for 5 seconds.
5) Remove power. Remove compact flash card. Delete .BIN file from card. Copy MP3 files onto card & reinsert.
6) Power up (card must be inserted) and download this program.

```
symbol TXD = 1          ' NB set output pin to pin you have used!
high TXD                ' output pin high
pause 2000              ' power up delay
serout TXD,T2400,("UUUUUUUUUUUUUUU")  ' NB must use capital U characters
stop
```

7) Power down. Wait 5 seconds. Power up. Wait until a repeated beep sound is heard on speakers. Power down.

Now follow the Firmware Upgrade procedure described above to restore the normal firmware.

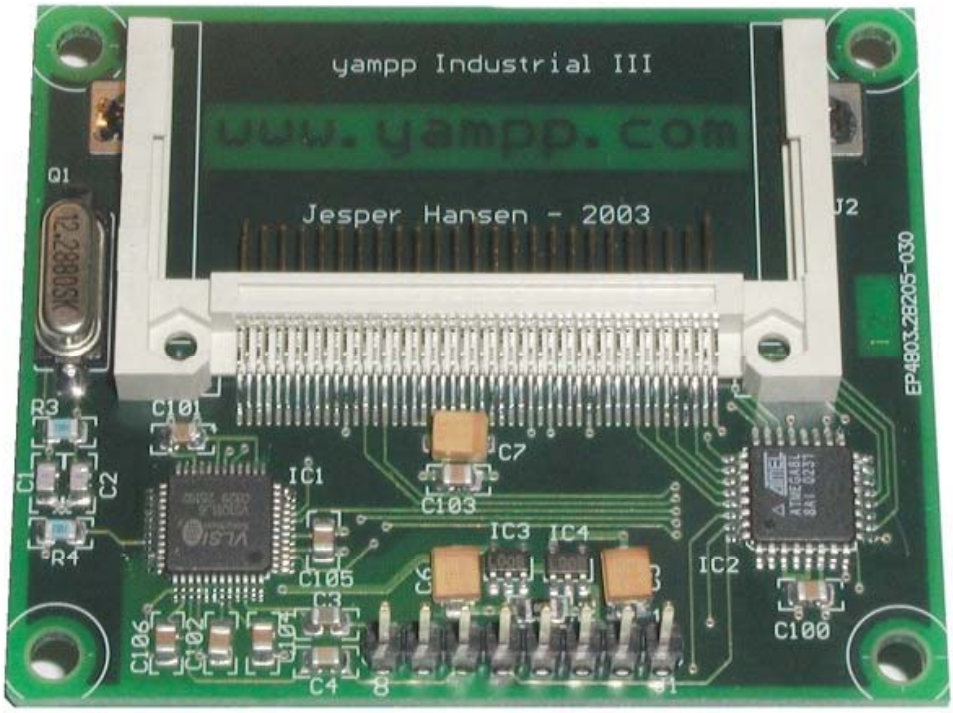
When the MP3 player starts 'beeping' the serial calibration settings have been saved into the MP3 players EEPROM memory. The player is therefore now permanently calibrated to that PICAXE chip.

yampp Industrial III

PICAXE Compatible Firmware Specification

Rev. B
(hardware rev A)
(PICAXE compatible firmware version)

2005-03-02



jelu handelsbolag

Bergvägen 68
S-18641 Vallentuna
Sweden

Table of Contents

Document History 1
Company Information 1
Features 1
Overview 1
Technical Data 2
Playing Time 2
Usage 2
Serial Interface 3
Serial Commands 3
yampp Extensions 4
Firmware Update 4
Connectors 5
Board Layout 5

Document History

Date	Author	Revision	Comments
2004-01-02	Jesper	A	Initial Revision
2005-03-02	CPS/JH	B	PICAXE Revision

Company Information

JELU HB is a privately owned company located in Vallentuna, just north of Stockholm, Sweden. We're specializing in MP3 technology, both for leisure and for commercial applications. We have been designing MP3 devices since 1999. Many of the designs are used in commercial applications such as alarm and evacuations systems, presentation and kiosk displays, theme parks, various entertainment systems as well as in busses and trains.
For more information please contact Jesper Hansen at jesper@jelu.se
Also see the company and player websites: http://www.jelu.se/and http://www.yampp.com/

Features

The yampp Industrial III (YI3) is the newest commercial MP3 player based on the yampp series of MP3 players. See www.yampp.com for details.
The PICAXE firmware version is optimised for use with PICAXE microcontrollers.

The PICAXE firmware version is also compatible with the command set used by Quadravox players, making it easy to replace these.

The YI3 has the following feature :

+ Small size, only 50x60 mm. Card protruding 14 mm on long edge.
+ High quality 4-layer PCB for low EMI.
+ CompactFlash card based. Using standard PC formatted FAT16 cards. Card can be up to 4GB, allowing more than 70 hours of continuous, unrepeated near CD quality music (at 128 kbps).
+ Easy and fast firmware upgrade through a file on the CF card.
+ Serial asyncronous interface for remote control of the player (Like RS-232, but at 3.3V levels).
+ BUSY status output line (high signal when file is playing).
+ Flexible 4.5-15V DC supply voltage.
+ Line level output for connection to an external amplifier.
+ Low power consumption. Approx. 40 mA (operating, with Sandisk 128 MB card).
+ Plays all MP3 bitrates up to and including 320kbps.

Overview

The yampp Industrial III, is like is sister card, the yampp Industrial, mainly intended as a replacement for mechanical playback units such as Cassette Decks or CD-ROMS. The mechanical nature of these devices often makes them slow and/or unreliable and require regular maintenace or replacement.
The yampp Industrial III is a complete Solid-State design, with no moving parts, offering high reliability and reducing maintenance to an absolute minimum.
The player firmware has an extensive command set, which makes it highly flexible. A few features like autostart e.t.c. allows it to be used autonomous in some environments, but it's mainly intended for being controlled by an external PICAXE microcontroller.

The player can be controlled via the serial port. The communication standard is like normal RS-232, but at 3.3V level with active low signalling.

Technical Data

Supply voltage	4.5-15 V DC
Power Supply Current	40 mA (operating)
Audio Output Level	1.8V pp
Output Load	30 Ohm
Total Harmonic Distorsion	0.1 %
Dynamic Range	90 dB
S/N Ratio	87 dB
Channel Separation	70 dB

Playing Time

The playing time depends on the CF card size and the bitrate used. 16 or 32 kbps is generally used for voice or medium quality music, while 128, 192 or 256kbs is used for music in near CD quality.
The times shown in the table below is total playing time for a certain CF card size. This time can of course be divided between a number of separate files.

	16 kbps	32kbps	64kbps	128kbps	256kbps
4MB	0:35	0:17	0:09	0:04	0:02
16MB	2:20	1:10	0:35	0:17	0:09
64MB	9:19	4:40	2:20	1:10	0:35
256MB	37:17	18:38	9:19	4:40	1:10
1GB	149:08	74:34	37:17	18:38	9:19
4GB	596:08	297:36	149:08	73:52	37.16

Approximate playing times for some common CF cards and bitrates (hours:minutes)

Usage

To start the player, simply insert a CF card containing MP3 files into the player.
The player currently only supports files in the root directory (limiting the number of files to 511).
The player only looks at the first 3 characters of the filename, so both long and short filenames can be used.
The files should therefore be named "001xxxxx.mp3" to "511xxxxx.mp3", where x can any valid filename string (or empty).

Serial Interface

The player has a standard asynchronous serial interface. It's like a RS-232 interface, but operating at 3.3V levels (active low). Do not connect 5V systems directly to the 3.3V RXD input.
The interface speed defaults to 2400 bps. The data format is 8-bit, no parity, 1 stop.

The serial port will report the status on all commands executed on the player. Status for playing commands will be returned when the playing is complete.

Serial Commands

The player is compatible with the command set also used by the Quadravox MP3 players. In addition to this, it has a number of "extension" commands, for special configuration and control.
The player supports the following commands :

0x01-0x7F:	Play one of the 127 directly addressed sounds. Will play sounds with filenames "001*.mp3" to "127*.mp3".
0x80-0xB8:	Set common volume (both channels) to one of 56 levels (0x80 is mute). Returns <volume >. NB: 0x80 to 0x9F are not normally audible.
0xC0-0xCF:	Set balance left (by attenuating the right channel 0-15 levels) Returns <balance>
0xD0-0xDF:	Set balance right (by attenuating the left channel 0-15 levels) Returns <balance >
0xE0:	Play file number 0-255. The file number is defined by the next transmitted byte. Will play sounds with filenames "000*.mp3" to "255*.mp3".
0xE1:	Play file number 256-511. The the file number (minus 256) is defined by the next transmitted byte. Will play sounds with filenames "256*.mp3" to "511*.mp3".
0xE4:	Return the hardware revision. Returns <hardware>
0xE5:	Return the software revision. Returns <software>
0xE6:	yampp Extension. See next section.
0xE7:	not implemented
0xE8:	not implemented
0xE9:	not implemented
0xEA:	Unmute the Left channel. Returns <mode >
0xEB:	Mute the Left channel. Returns <mode >
0xEC:	Unmute the Right channel. Returns <mode >
0xED:	Mute the Right channel. Returns <mode >
0xEE:	Set extended balance; the second byte is +/- 56.(-1 = 255, -2 = 254 etc) Returns <balance >
0xEF:	Return the number of files detected, modulo 256
0xF0:	Stop playing.
0xF1:	Clear loop mode; play in progress will complete. Returns <mode >
0xF2:	Increment volume. Returns <volume >
0xF3:	Decrement volume. Returns <volume>
0xF4:	Set loop mode. Returns <mode >
0xF5:	Get status.
0xF6:	Get errors. Returns <errors >
0xF7:	S/W reset.
0xF8:	Pause.
0xF9:	Resume.
0xFA:	Jog balance left. Returns <balance >
0xFB:	Jog balance right. Returns <balance >
0xFC:	Get volume. Returns <volume >
0xFD:	Get balance. Returns <balance >
0xFE:	Get revision. Returns 4.
0xFF:	Get type. Returns 0x79 ('y') .

For each command, the status byte is returned (except where otherwise indicated).
The status byte has the following format :

Bit	Meaning
7	hardware error
6	alternate busy bit
5	no card in socket
4	pause
3	blank (requested file not found)
2	mute
1	loop
0	playing

yampp Extensions

For further functionality beyond the Quadravox compatibility, the yampp extension command (0xE6) can be used.

CMD	Databytes	Function
0x01	1	Set the level of the power on beep in 0-55 levels.
0x02	1	Set Logo display flag. (1=transmit logo string on RXD on powerup)
0x03	1	Set Song autostart 001-255. 000=off
0x04	1	Set Song autostart 256-511.
0x05	1	Set Saved Volume restore flag. (1=Restore on reset).
0x06	0	Save current volumes.
0x07	0	Restore saved volumes.
0x08	1	Set debug level. (0-3, 0=off, 3 begin most verbose.)

All commands will respond with 'y', unless otherwise indicated. If a command is not understood, a '?' will be returned.

Firmware Update

The player supports download of new firmware though a file on the CF card. The file should be named **YAMPPIND.BIN** and located in the root directory of the card.
The player will automatically replace the current firmware with the one on the CF card. Note that there is no version control, so you can also "downgrade" to a previous version. Also, you can easily upgrade to a custom version, especially designed for your needs.
To update, remove power from the player and insert the CF card with the new firmware. The player will transmit the following bytes on the serial port at 9600 bps, indicating the progess :

'B' - bootloader started.
'P' - yamppind.bin file found, programming started.
'D' - programming successful.

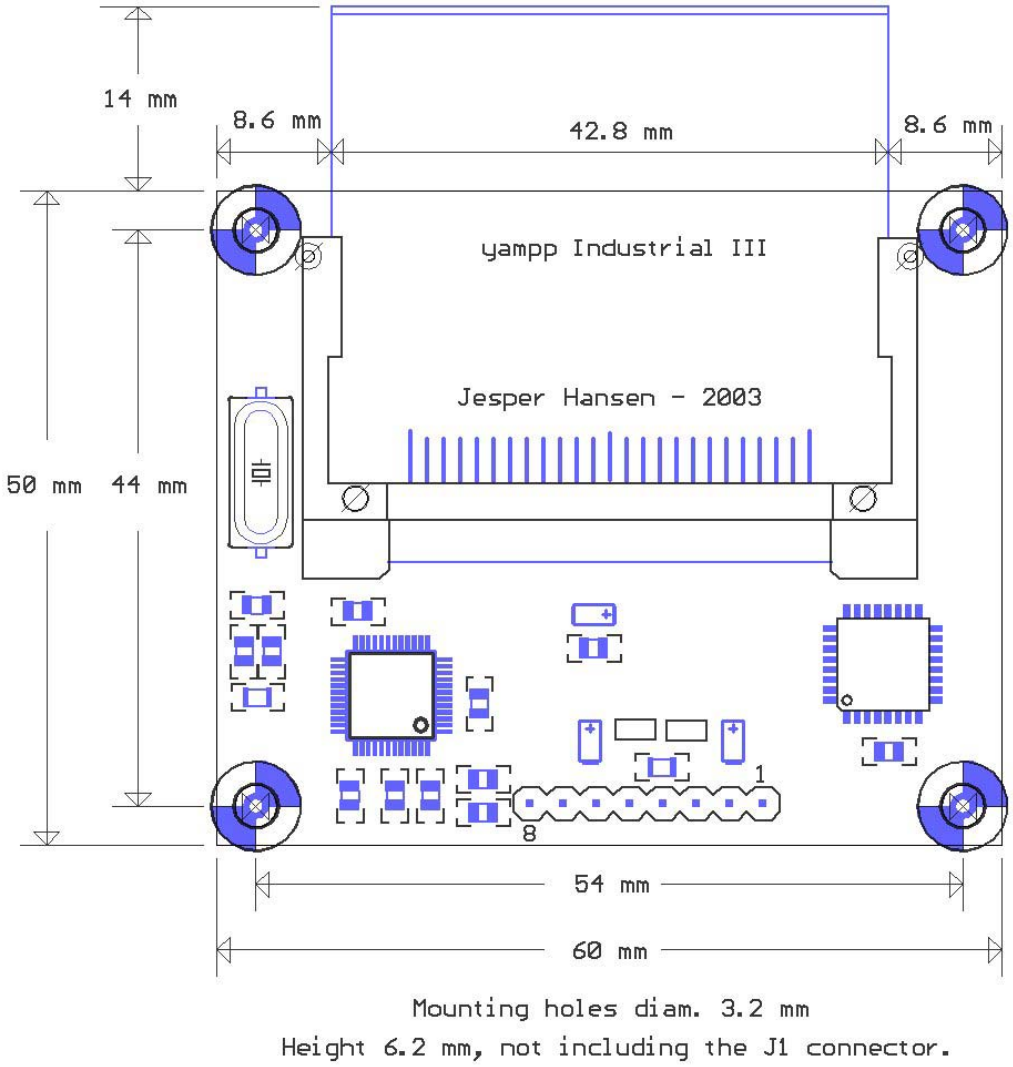
Now remove power and the CF card from the player and insert a card containing the MP3 files. When applying power to the player, it should respond with 'B' and 'A', indicating that the application has been started.

Connectors

All signals to and from the player are available on the 8-pin connector J1. This is a standard 0.1" (2.54mm) pin-header.

J1 Connector	
Pin #	**Function**
1	GND
2	4-15V DC input
3	TXD (YI3>PICAXE)
4	RXD (PICAXE>YI3)
5	BUSY Out
6	Right Channel Out
7	GNDA
8	Left Channel Out

Board Layout



Mounting holes diam. 3.2 mm
Height 6.2 mm, not including the J1 connector.