

# **Industrial MP3 player SD with build-in amplifier 2x10W**



WWW.DIGINN.PL

## Table of contents

1. Short description of Embedded MP3 Player .....	3
2. Pinout .....	4
2.1 Player dimensions .....	6
3. Functional description of MP3 player .....	7
4. MP3 Player modes .....	9
5. Connecting MP3 Player .....	11
5.1 Using buttons: .....	11
5.2 Connecting to RS232 port: .....	11
6. Preparing a SD card.....	12
7. RS232 .....	14
7.1 Data frame .....	15
7.2 Response from MP3 player .....	16
7.3 Commands.....	18
8. Technical specification:.....	22

## 1. Short description of Embedded MP3 Player



MP3 Player is designed for playing audio files in MP3 format. All files are stored in external SD card plugged in into this module. This audio player has build-in amplifier with output power 2x10 Watt.

Technical information:

- supported MP3 formats – type MPEG I, II, III,
- amplifier in class D,
- designer to speakers from 4 to 8 ohm,
- 21 level volume regulation (automatically saved),
- sampling rate MP3: 44.1 kHz,
- several different operation modes,
- simple player control: from microcontroller or buttons,
- supported file format type: FAT, FAT32,
- supported memory type: SD or SDHC,
- support for RS232 interface,
- power supply 8-18V,
- in low power mode, current consumption ca. 700 uA,
- aluminium case.

## 2. Pinout

MP3 Player has 4 configurations pins, 12 control pins, speakers output connector, power input connector, SD card connector and 2 status LEDs.

Below there's a pinout of MP3 Player:

a) front side:



Configuration pins: CONF1 – CONF4:

Configuration pins for player mode. Detailed specification in chapter 3.

Power supply connector IN:

Pin 1 : GND –

Pin 2 : VCC +

Speaker output connector SPK:

Pin 1 : output, right channel -

Pin 2 : output, right channel +

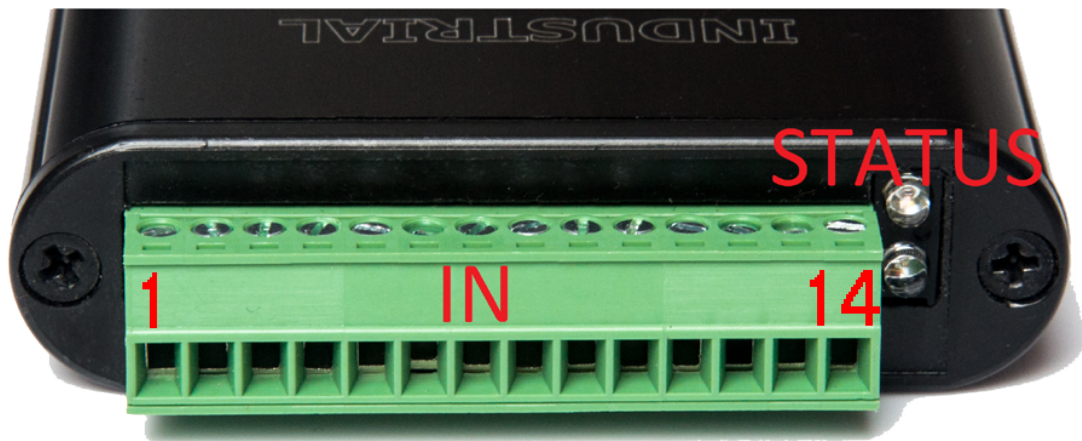
Pin 3 : output, left channel +

Pin 4 : output, left channel -

SD card connector:

Connector for 'big' SD card.

b) back side:



Inputs from IN1 to IN14:

Pin 1 to 10: function depends on operation mode, details in chapter 4,

Pin 11: RS232 mode RXD,

Pin 12: RS232 mode RXD,

Pin 13: ground GND,

Pin 14: VCC +, **use only with IN1 to IN10.**

LED STATUS diode:

Represent the current state of audio player:

- a) green led turned on – power supply is ok,
- b) green led is blinking – power supply is too low or too high,
- c) green led is turned off – only in low power mode, player is turned off,
- d) blue led turned off – player in idle state,
- e) blue led is blinking – song is playing.

## 2.1 Player dimensions



All dimensions are in centimeters.



### 3. Functional description of MP3 player

The way of playing mp3 files depends on player configuration. To select a required configuration you need to use config pins CONF1, CONF2, CONF3, CONF4. To change a player configuration mode you need to switch up or down required pin. Configuration pin in down position mean OFF state and configuration pin in up position mean ON state.



After purchase all configuration pins are in OFF state.

There's a few different modes available in MP3 Player:

CONF pins configuration				Player mode
4	3	2	1	
OFF	OFF	OFF	OFF	- normal MP3 Player work in standard mode. There is available 6 different buttons: START, STOP, NEXT, PREV, VOL+, VOL-
OFF	OFF	OFF	ON	- normal 2 MP3 Player work in standard mode. There is available 6 different buttons: START, STOP, NEXT, PREV, VOL+, VOL- When song is ended, go to next file and play it.
OFF	OFF	ON	OFF	- continuous After power up music is played all the time, one track after another.
OFF	OFF	ON	ON	- special After pressing buttons there is played only file with name: 1.mp3, 2.mp3 etc... Pressing another button does not interrupt played file.

OFF	ON	OFF	OFF	- RS232 Fully management MP3 Player using commands.
OFF	ON	OFF	ON	- special 2 After pressing buttons there is played only file with name: 1.mp3, 2.mp3 etc... Pressing another button does interrupt currently played file and start another file.
OFF	ON	ON	OFF	- random When button is pressed random file is played.
OFF	ON	ON	ON	- normal, low power MP3 Player work in standard mode. There is available 6 different buttons: START, STOP, NEXT, PREV, VOL+, VOL-
ON	OFF	OFF	OFF	- normal 2, low power MP3 Player work in standard mode. There is available 6 different buttons: START, STOP, NEXT, PREV, VOL+, VOL- When song is ended, go to next file and play it.
ON	OFF	OFF	ON	- special, low power After pressing buttons there is played only file with name: 1.mp3, 2.mp3 etc... Pressing another button does not interrupt played file.
ON	OFF	ON	OFF	-special 2, low power After pressing buttons there is played only file with name: 1.mp3, 2.mp3 etc... Pressing another button does interrupt currently played file and start another file.
ON	OFF	ON	ON	- random, low power When button is pressed random file is played.



## 4. MP3 Player modes

MP3 Player can play files in few different modes. Each mode has a different method to play files.

**In every mode volume level is automatically saved and restored after power down.**

Below there's a full description of all modes:

a) normal – in this mode player wait for user action. You need to use control pins IN1 to IN6 in this way:

- IN 1 - START – player starts to play one time mp3 file,
- IN 2 - STOP – stop playing files,
- IN 3 - NEXT – jump to next file,
- IN 4 - PREV – jump to previous file,
- IN 5 - VOL+ - volume up (21 volume levels), default volume level 11,
- IN 6 - VOL- - volume down (21 volume levels), default volume level 11.

b) normal – in this mode player wait for user action. After pressing a START button and when song is ended, player starts play next song. You need to use control pins IN1 to IN6 in this way:

- IN 1 - START – player starts to play one time mp3 file,
- IN 2 - STOP – stop playing files,
- IN 3 - NEXT – jump to next file,
- IN 4 - PREV – jump to previous file,
- IN 5 - VOL+ - volume up (21 volume levels), default volume level 11,
- IN 6 - VOL- - volume down (21 volume levels), default volume level 11.

c) continuous – player after finding first available mp3 file on sd card, starts to play. In this mode active buttons are: NEXT, PREV, VOL+ and VOL-. Default volume level in this mode is 19.

d) special – Pressing a button will cause playing a file at special name:

- IN 1 – playing filename 1.mp3,
- IN 2 – playing filename 2.mp3,

- IN 3 – playing filename 3.mp3,
- IN 4 – playing filename 4.mp3,
- IN 5 – playing filename 5.mp3,
- IN 6 – playing filename 6.mp3,
- IN 7 – playing filename 7.mp3,
- IN 8 – playing filename 8.mp3,
- IN 9 – playing filename 9.mp3,
- IN 10 – playing filename 10.mp3,

Default volume level in this mode is 19. Playing another file is available only if currently played file has reached the end of file.

e) RS232 – Steering MP3 module is available after sending a special command. More details in chapter 7.

f) special 2 – same as ‘special’ mode except one thing. Playing another file is available at any time.

g) random – pressing a START button will select a random file to play. In this mode We can use:

- IN 1 – START – play a random file, second press is available only if current file is ended,
- IN 2 – START 2 – play a random file, second press will interrupt currently played file and go to next random file,
- IN 5 - VOL+ - volume up (21 volume levels), default volume level 11,
- IN 6 - VOL- - volume down (21 volume levels), default volume level 11.

h) normal, low power – same as „normal”, when nothing is played, player goes to low power mode,

i) normal 2, low power – same as „normal 2”, when nothing is played, player goes to low power mode,

j) special, low power – same as „special”, when nothing is played, player goes to low power mode,

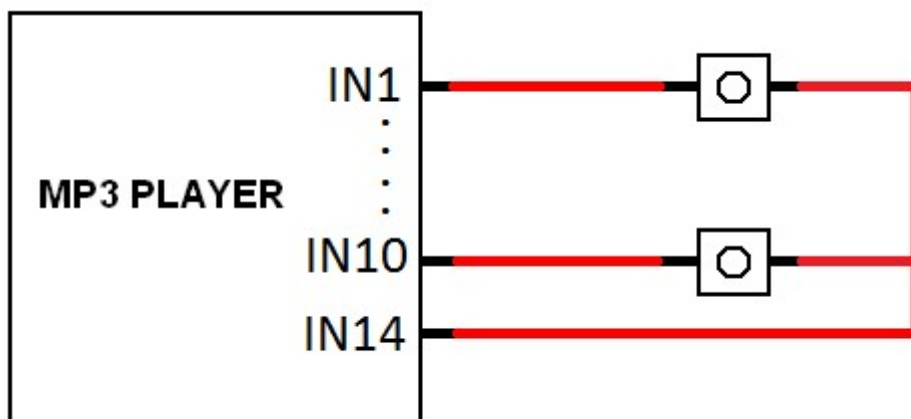
k) special 2, low power – same as „special 2”, when nothing is played, player goes to low power mode,

l) random, low power – same as „random”, when nothing is played, player goes to low power mode.

## 5. Connecting MP3 Player

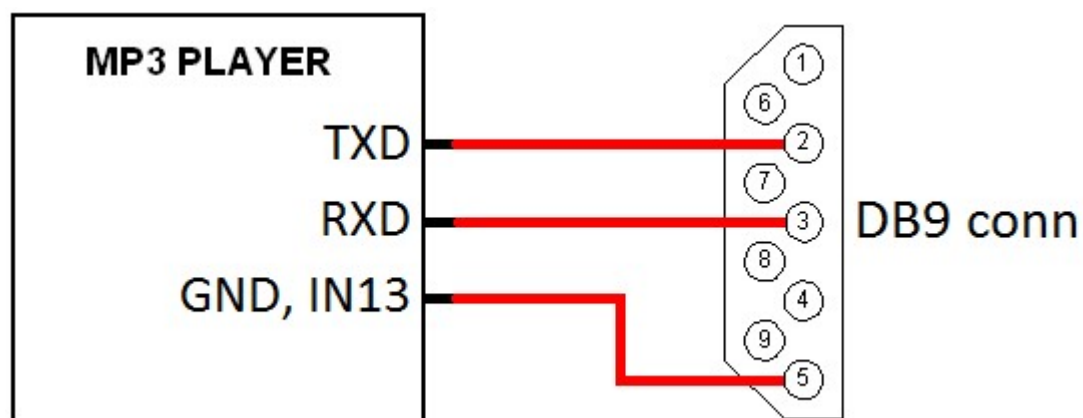
We can use control pins in two different ways: as a buttons and we can connect a microcontroller. Below you can find example of both possibilities.

### 5.1 Using buttons:



Pin IN14(+) need to be connected to all inputs of all buttons. Pins from IN1 to IN 10 need to be connected to second buttons input.

### 5.2 Connecting to RS232 port:



To RS232 DB9 connector, please connect:

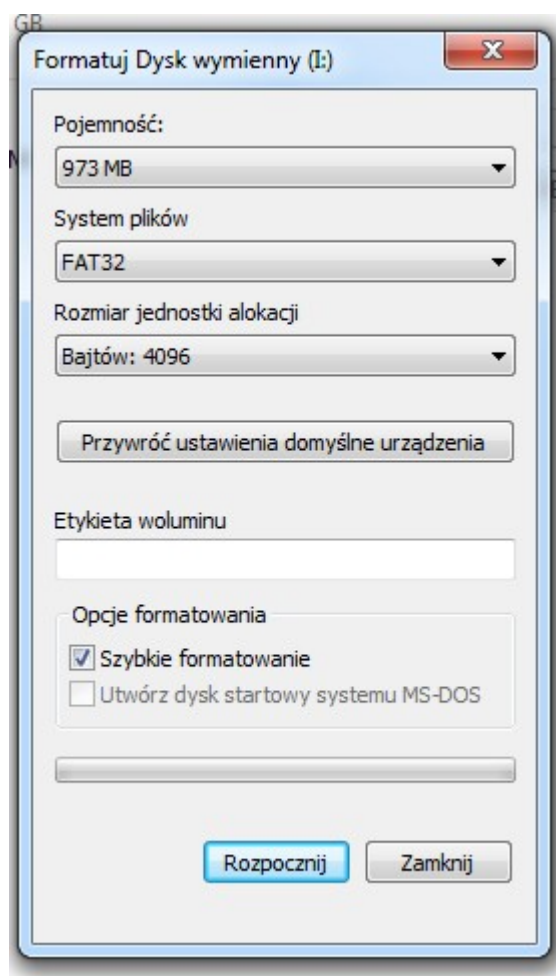
- DB9, pin 5 – player ground, pin IN 13,
- DB9, pin 3 – player RXD, pin IN 11,
- DB9, pin 2 – player TXD, pin IN 12.

## 6. Preparing a SD card

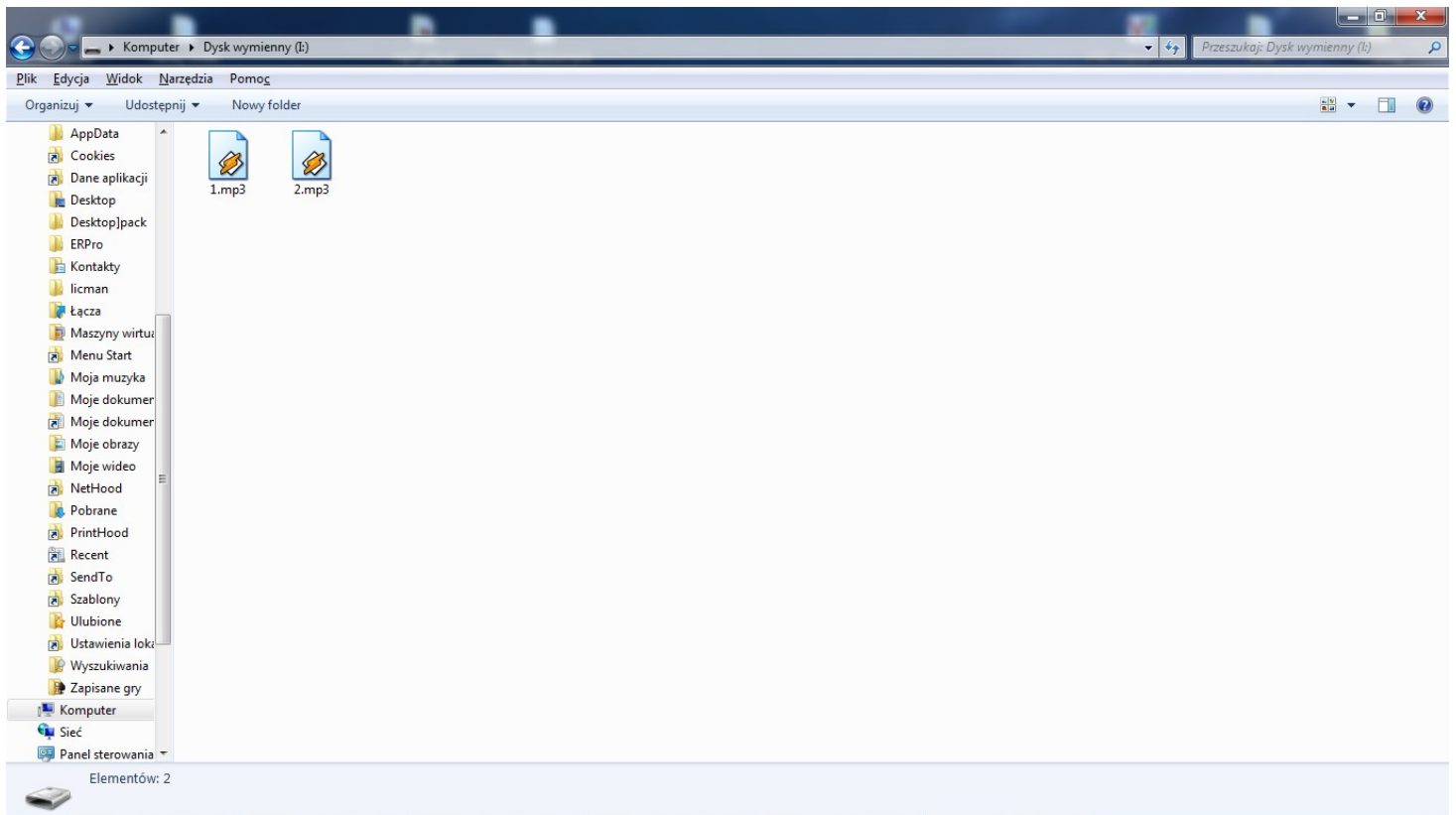
MP3 Player support SD memory with FAT 16/32 file systems. In this case you need to prepare SD card memory using computer with Windows OS.

Put your SD card into free SD card reader to computer or by using USB<=>SD adapter. Click a right mouse button on new installed removable disk and select ‘format’ option. Choose an FAT32 partition type and press button ‘start’.

Example:



After successfully device format we can put mp3 files into removable disk:



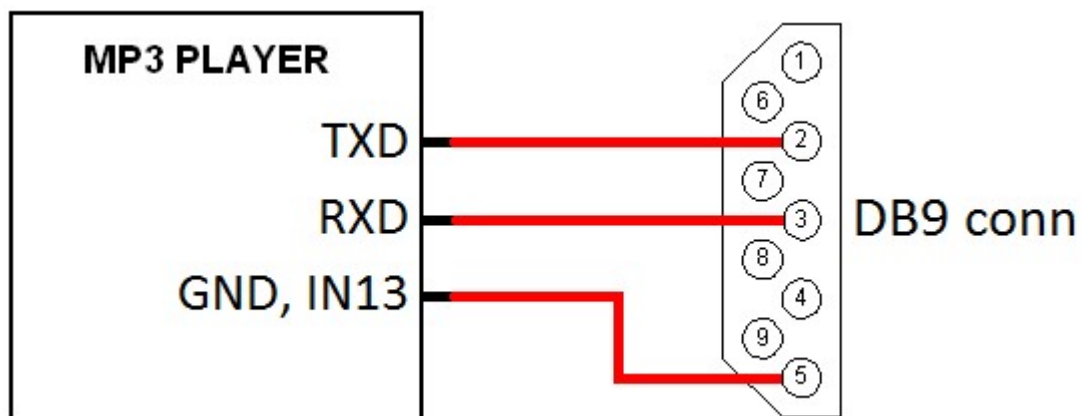
MP3 Player doesn't support any files other than .mp3 extension and folders other than root.

WARNING:

It is not recommended to format your SD card in PC with MAC OS.

## 7. RS232

In this mode player needs to be connected to RS232 port some other device or port RS232 in your personal PC. To communication player use 2 pins RXD and TXD.



To RS232 DB9 connector, please connect:

- DB9, pin 5 – player ground, pin IN 13,
- DB9, pin 3 – player RXD, pin IN 11,
- DB9, pin 2 – player TXD, pin IN 12.

## 7.1 Data frame

Data frame to mp3 player is designed in this way:

- start – 0xAA,
- data count (bytes),
- data,
- end – 0x15.



Data frame (response) from mp3 player is designed in this way:

- start 0x15,
- data count (bytes),
- data,
- end –0xAA.



Configuration of serial port MP3 module:

- baudrate– 115200 bps,
- data bits – 8,
- stop bit – 1,
- without parity check.

After receiving a data frame, mp3 player always send a response frame.

Data of commands sended to/from mp3 player are always in lower case letters starting from 'a' to 'z'.



## 7.2 Response from MP3 player

Every valid received frame is confirmed with special message sent from mp3 player.

Available responses from mp3 player:

a) ACK:

Received command was properly received and executed. Player send a response frame do controller: 0x15, 0x03, 0x61, 0x63, 0x6B, 0xAA (0x15, 0x03, 'a', 'c', 'k', 0xAA).

b) NACK:

Received command was properly received. Execution of command failed. Player send a response frame do controller: 0x15, 0x04, 0x6E, 0x61, 0x63, 0x6B, 0xAA (0x15, 0x04, 'n', 'a', 'c', 'k', 0xAA).

c) FRAME ERROR:

Frame was incorrectly received. Player send a response frame do controller: 0x15, 0x05, 0x66, 0x72, 0x65, 0x72, 0x72, 0xAA (0x15, 0x05, 'f', 'r', 'e', 'r', 'r', 0xAA).

d) UNKNOWN COMMAND:

Received command is not supported by MP3 player. Player send a response frame do controller: 0x15, 0x07, 0x75, 0x6E, 0x6B, 0x6E, 0x6F, 0x77, 0x6E, 0xAA (0x15, 0x07, 'u', 'n', 'k', 'n', 'o', 'w', 'n', 0xAA).

e) STATUS:

Sends actual status of mp3 player. Response frame contains status data:

0x15	0x0F	/st1/st2/st3/st4/st5/st6/st7/	0xAA
------	------	-------------------------------	------

Status from st1 to st7 are one bytes states which contains '0' or '1' in ascii format. Meaning of status data:

- st1 – sd memory found and properly initialized, example: '0' sd memory is missing '1',
- st2 – state of playing a file, '1' music is playing, '0' music is not playing,
- st3 – '1' if player is in pause mode, '0' otherwise.

Status from st4 to st7 is not used and always are '0'.

Every status is separated by: ‘/’ sign.

Example of status frame: 0x15, 0x0F, 0x2f, 0x31, 0x2f, 0x30, 0x2f, 0x30, 0x2f, 0x30, 0x2f,  
0x30, 0x2f, 0x30, 0x2f, 0x30, 0x2f, 0xAA (0x15, 0x0F, '/', '1', '/', '0', '/', '0', '/', '0', '/', '0',  
/, '0', /, '0', /, 0xAA).

f) FILE NAME:

Sends actual selected file name.

Player send a response frame do controller: 0x15, 0x05+x, 0x6E, 0x61, 0x6D, 0x65, 0x3A, 0xAA (0x15, 0x05+x, 'n', 'a', 'm', 'e', ':', 0xAA).

## 7.3 Commands

Every command is placed in 'data' part of frame. Every command contains lower case letters.

It is very important to write a proper data bytes count in 'data count' part of frame.

Supported commands:

### 1. PLAY CONTINUOUS:

Data frame:

0xAA, 0x06, 0x70, 0x6C, 0x61, 0x79, 0x5F, 0x63, 0x15 (0xAA, 0x06, 'p', 'l', 'a', 'y', '\_', 'c', 0x15)

Command will play all MP3 file one after the other.

Expected response:

- ACK

- NACK

### 2. PLAY SINGLE:

Data frame:

0xAA, 0x06, 0x70, 0x6C, 0x61, 0x79, 0x5F, 0x73, 0x15 (0xAA, 0x06, 'p', 'l', 'a', 'y', '\_', 's', 0x15)

Command will play only one MP3 file.

Expected response:

- ACK

- NACK

### 3. STOP:

Data frame:

0xAA, 0x04, 0x73, 0x74, 0x6F, 0x70, 0x15 (0xAA, 0x04, 's', 't', 'o', 'p', 0x15)

Command will stop play MP3 file.

Expected response:

- ACK

- NACK

#### 4. NEXT:

Data frame:

0xAA, 0x04, 0x6E, 0x65, 0x78, 0x74, 0x15 (0xAA, 0x04, 'n', 'e', 'x', 't', 0x15)

Command will select a next file.

Expected response:

- ACK

- NACK

#### 5. PREVIOUS:

Data frame:

0xAA, 0x04, 0x70, 0x72, 0x65, 0x76, 0x15 (0xAA, 0x04, 'p', 'r', 'e', 'v', 0x15)

Command will select a previous file.

Expected response:

- ACK

- NACK

#### 6. PAUSE:

Data frame:

0xAA, 0x05, 0x70, 0x61, 0x75, 0x73, 0x65, 0x15 (0xAA, 0x05, 'p', 'a', 'u', 's', 'e', 0x15)

Command will switch player into pause mode. Same commands sended again will switch back player into normal mode.

Expected response:

- ACK

- NACK

#### 7. VOLUME:

Data frame:

0xAA, 0x05, 0x76, 0x6F, 0x6C, 0x3A, 0x37, 0x15 (0xAA, 0x05, 'v', 'o', 'l', ':', '7', 0x15)

Volume change command. Supported volume values: from '0' to '9'. '0' mean totally mute, '9' means maximum output volume. After power up default volume level is 5. Volume value need to be provided after ':' sign.

Expected response:

- ACK

- NACK

## 8. FIND:

Data frame:

0xAA, 0x04+xx, 0x66, 0x6E, 0x64, 0x3A, 0XX, 0XX, ..., 0x15 (0xAA, 0x04+xx, 'f', 'n', 'd', ':', ..., 0x15)

Command will do search of MP3 files. It works only when player is not playing any file. Also full file name needs to be provided in this command, example: "music1.mp3". When mp3 player found a file there is send ACK response command otherwise NACK response is send.

Expected response:

- ACK
- NACK

## 9. STATUS

Data frame:

0xAA, 0x06, 0x73, 0x74, 0x61, 0x74, 0x75, 0x73, 0x15 (0xAA, 0x06, 's', 't', 'a', 't', 'u', 's', 0x15)

Command will send a status of mp3 player,

Expected response:

- STATUS

## 10. GET FILE NAME

Data frame:

0xAA, 0x05, 0x66, 0x6E, 0x61, 0x6D, 0x65, 0x15 (0xAA, 0x05, 'f', 'n', 'a', 'm', 'e', 0x15)

Command will send a actual selected file name,

Expected response:

- NACK
- NAME

## 11. AMP GAIN 0

Data frame:

0xAA, 0x05, 0x67, 0x61, 0x69, 0x6E, 0x30, 0x15 (0xAA, 0x05, 'g', 'a', 'i', 'n', '0', 0x15)

Command will disable additional output power GAIN.

Expected response:

- NACK

## 12. AMP GAIN 1

Data frame:

0xAA, 0x05, 0x67, 0x61, 0x69, 0x6E, 0x31, 0x15 (0xAA, 0x05, 'g', 'a', 'i', 'n', '1', 0x15)

Command will enable additional output power GAIN.

Expected response:

- NACK

## 8. Technical specification:

Parameter	Comment	Min	Typ	Max	Unit
<b>Power supply</b>		8	-	18	V
<b>Current consumption max.</b>				6	A
<b>Current consumption low power</b>			700		uA
<b>Signal to noise SNR</b>	PO = 10 W, RL = 8 $\Omega$		98		dB
<b>Gain low</b>			9		dB
<b>Gain high</b>			15		dB
<b>Total Harmonic Distortion + Noise</b>	PO = 5 W into 8 $\Omega$ , f = 1 kHz, PVDD = 12 V		0.01		%
<b>Output amplifier power</b>	RL = 8 $\Omega$ , THD = 1%, f = 1 kHz, 20 kHz BW, PVDD = 15 V		12		W
	RL = 8 $\Omega$ , THD = 1%, f = 1 kHz, 20 kHz BW, PVDD = 12 V		8		W
	RL = 8 $\Omega$ , THD = 10%, f = 1 kHz, 20 kHz BW, PVDD = 15 V		15		W
	RL = 8 $\Omega$ , THD = 10%, f = 1 kHz, 20 kHz BW, PVDD = 12 V		10		W
	RL = 4 $\Omega$ , THD = 1%, f = 1 kHz, 20 kHz BW, PVDD = 15 V		20		W
	RL = 4 $\Omega$ , THD = 1%, f = 1 kHz, 20 kHz BW, PVDD = 12 V		13		W
	RL = 4 $\Omega$ , THD = 10%, f = 1 kHz, 20 kHz BW, PVDD = 15 V		24		W
	RL = 4 $\Omega$ , THD = 10%, f = 1 kHz, 20 kHz BW, PVDD = 12 V		16		W