

## MP3orator

### Embedded Digital Audio Player

Datasheet Rev.: 2.1  
Date: 28.03.2009

#### Features

- Atmel **ATmega644P** RISC CPU @ 6MHz
- VLSI **VS1011E** MP3/WAV Decoder
- Decodes MPEG 1 & 2 audio layer 3 (ISO 11172-3), WAV and PCM files
- Supports VBR (variable bitrate) for MP3
- High-quality stereo DAC with no phase error between channels
- Active de-click circuit on audio outputs
- Stereo earphone driver capable of driving a 300ohm load
- **SDcard** removable storage device, on SPI
- Remote controlled via UART, I2C and/or keyboard
- Operating Voltage: 3.3VDC
- Pre-programmed with application firmware for standalone operation. Controlled via keypad and/or serial port control. FAT16 with multi-level directories supported.

#### Ordering Information

##### Art.-No. 01.0138

MP3orator, OEM board

##### Contact:

##### Elektronik-Atelier Kallen

Giacomettistrasse 33A  
CH-3006 Berne / Switzerland

Web: [www.avrcard.com](http://www.avrcard.com)

Email: [info@avrcard.com](mailto:info@avrcard.com)

Tel: +41-31-832 1441

Fax: +41-31-560 4110

#### Specifications

##### AUDIO

|                       |   |
|-----------------------|---|
| Decoder Type          | VLSI VS1011E  |
| Clock                 | 14.7456 MHz   |
| Decoded Audio Formats | MPEG 1 & 2 audio layer 3 (ISO 11172-3), WAV and PCM |
| DAC Resolution        | 16 bits   |
| THD                   | 0.1%  |
| Dynamic Range         | 88dB  |
| Output Voltage        | 1.6...2.1VPP @ 30 Ohms                              |

##### CPU

|       |                            |
|-------|----------------------------|
| Type  | Atmel ATmega644 8-bit RISC |
| Clock | 6 MHz                      |

##### STORAGE DEVICE

|                |        |
|----------------|--------|
| Type           | SDcard |
| Operating Mode | SPI    |

##### I/O

|          |                                 |
|----------|---------------------------------|
| Serial   | UART with TXD, RXD              |
| Two-wire | I2C with SDA, SCL               |
| Parallel | 12 I/O Ports, freely assignable |

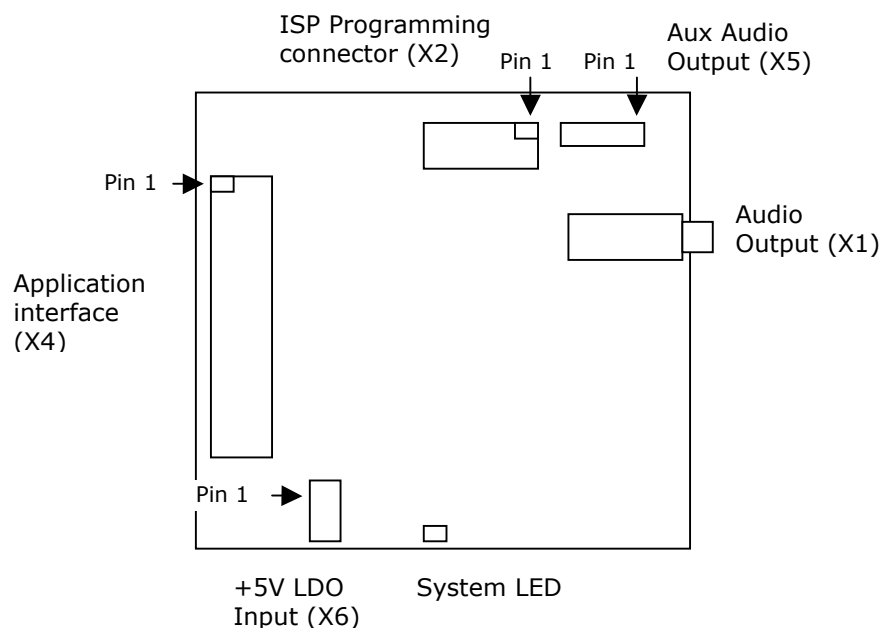
##### Power Supply

|             |                         |
|-------------|-------------------------|
| Requirement | 3.3V or 5VDC, max. 40mA |
|-------------|-------------------------|

##### Mechanical

|              |              |
|--------------|--------------|
| Size (L x W) | 46 x 48.5 mm |
| Weight       | 30 g         |

## Connector Specifications



## Application Interface

| X4: Application |              |        |  |                               |
|-----------------|--------------|--------|--|-------------------------------|
| No.             | Signal       | Type   | Description  | Standard Firmware Function    |
| 1               | <b>+3.3V</b> | POWER  | DC Input voltage, 3.3V, or output voltage when LDO is used |                               |
| 2               | <b>TXD</b>   | OUT    | UART Serial Interface, transmit data                       | Serial command interface      |
| 3               | <b>RXD</b>   | IN     | UART Serial Interface, receive data                        |                               |
| 4               | <b>SCL</b>   | IN     | I2C Interface, clock                                       | SHIFT signal (IN)             |
| 5               | <b>SDA</b>   | IN/OUT | I2C interface, data  | BUSY signal, active low (OUT) |
| 6               | <b>GND</b>   | POWER  | Ground   |                               |
| 7               | <b>PD4</b>   | IN/OUT | I/O Port   | PLAY (active low)             |
| 8               | <b>PD5</b>   | IN/OUT | I/O Port   | SKIP (active low)             |
| 9               | <b>PD6</b>   | IN/OUT | I/O Port   | Volume Up (active low)        |
| 10              | <b>PD7</b>   | IN/OUT | I/O Port   | Volume Down (active low)      |
| 11              | <b>PA0</b>   | IN/OUT | I/O Port / ADC Input                                       | Keypad row 0                  |
| 12              | <b>PA1</b>   | IN/OUT | I/O Port / ADC Input                                       | Keypad row 1                  |
| 13              | <b>PA2</b>   | IN/OUT | I/O Port / ADC Input                                       | Keypad row 2                  |
| 14              | <b>PA3</b>   | IN/OUT | I/O Port / ADC Input                                       | Keypad row 3                  |
| 15              | <b>PA4</b>   | IN/OUT | I/O Port / ADC Input                                       | Keypad column 0               |
| 16              | <b>PA5</b>   | IN/OUT | I/O Port / ADC Input                                       | Keypad column 1               |
| 17              | <b>PA6</b>   | IN/OUT | I/O Port / ADC Input                                       | Keypad column 2               |
| 18              | <b>PA7</b>   | IN/OUT | I/O Port / ADC Input                                       | Keypad column 3               |
| 19              | <b>nRES</b>  | IN/OUT | CPU Reset, active low                                      |                               |
| 20              | <b>PC5</b>   | OUT    | Parallel to onboard LED                                    |                               |

## Audio

| <b>X5: Aux Audio</b> |               |             |                            |
|----------------------|---------------|-------------|----------------------------|
| <b>No.</b>           | <b>Signal</b> | <b>Type</b> | <b>Description</b>         |
| 1                    | <b>AGND</b>   | POWER       | Audio Signal Ground        |
| 2                    | <b>RIGHT</b>  | OUT         | Right channel audio output |
| 3                    | <b>LEFT</b>   | OUT         | Left channel audio output  |

## ISP Programming Interface

| <b>X1: Programming</b> |               |             |                         |
|------------------------|---------------|-------------|-------------------------|
| <b>No.</b>             | <b>Signal</b> | <b>Type</b> | <b>Description</b>      |
| 1                      | MISO          | OUT         | SPI bus                 |
| 2                      | +3.3V         | POWER       | Digital power           |
| 3                      | SCK           | IN          | SPI bus                 |
| 4                      | MOSI          | IN          |                         |
| 5                      | RESET         | IN          | CPU Reset Input/Output  |
| 6                      | GND           | RESET       | Signal and power ground |

## Device Operation

---

### Power Supply

The MP3orator module can be powered either from

- a) a regulated 5VDC source through X6, or
- b) a regulated 3.3VDC source, through X4. In this case, X6 must be disconnected.

### Signal Levels

All signal levels present on the Application Interface connector X4 are 3.3V logic levels.

A 100 Ohm series resistor is present on each I/O line to provide a minimum protection against ESD. If long wires are connected, or if other signal levels are to be applied, additional signal conditioning circuitry is required.

### Keyboard Commands

#### **PLAY**

---

Plays the file which has previously been preset via serial interface by the SF command. If no filename has been preset, the first file in the root directory is played.

#### **SHIFT**

---

When this signal is asserted, the value 16 is added to the key value of the keyboard.

#### **SKIP**

---

Immediately starts playback of the file with the next sequence number. If no file with a corresponding number is found, the playback stops. If the number 9999 is reached, no action is taken.

The file sequence number is reset to 1 after power-up. It can be set to any value in the range 1...9999 with the PN or SN commands.

In order to playback a file, the MP3orator must find a file with the name xxxx.mp3, where xxxx is a number string in the range of 0001...9999.

#### **VOLUME UP**

---

Increases the playback volume by 2dB.

#### **VOLUME DOWN**

---

Decreases the playback volume by 2dB.

## KEYBOARD

Any keypress immediately starts playback of the file with the corresponding sequence number. In order to playback a file, the MP3orator must find a file with the name xxxxx.mp3, where xxxxx is a number string in the range of 10000...10031.

If no file with a corresponding number is found, no action is taken and any playback in progress will be continued.

Relation between keyboard matrix and file played:

|       | Col 0     | Col 1     | Col 2     | Col 3     |
|-------|-----------|-----------|-----------|-----------|
| Row 0 | 10012.mp3 | 10011.mp3 | 10000.mp3 | 10010.mp3 |
| Row 1 | 10013.mp3 | 10009.mp3 | 10008.mp3 | 10007.mp3 |
| Row 2 | 10014.mp3 | 10006.mp3 | 10005.mp3 | 10004.mp3 |
| Row 3 | 10015.mp3 | 10003.mp3 | 10002.mp2 | 10001.mp2 |

## Serial Command Interface

The serial interface can be connected to a host microcontroller or a PC with a terminal application. In case of the PC, a suitable RS-232 level converter must be used on the TXD/RXD lines. The communications parameters are:

9600 bps, 8 databits, 1 stopbit, no parity

The available commands are described from page 6 onwards.

## Startup

### Normal Startup

After power-on, the BUSY signal will be asserted and the MP3orator module will send initialization messages over the serial port as follows:

```
INIT: MP3annunciator V2.2
INIT: DSP HW
INIT: DSP SW
INIT: IDE
INIT: Ready
```

After the 'Ready' message, the BUSY signal will be deasserted, and the module is ready to receive commands over the serial interface.

### Startup Errors

```
ERROR: IDE init failed, need Reset
SDcard not inserted, not properly formatted or defect.
```

## Playback Control Commands

| <b>PF:</b>   | <b>Play File</b>  |
|--|---|
| Argument:  | Filename  |
| Description:   | Plays the *.mp3 file with name given in argument.   |
| Response:  | PLAY: <filename>  |
| Errors:  | ERROR: not found.   |
| <p>Filenames must be 1...8 characters long, conforming to 8.3 standard. Files are searched in the current directory (see command for further information in directories). Extension must not be entered in the argument.</p> |   |
| <b>PN:</b>   | <b>Play Number</b>  |
| Argument:  | 1...4-digit file number   |
| Description:   | Plays *.mp3 file with name corresponding to number given in argument. The number must be in the range 0...9999.<br>The number argument is padded with '0' characters to a 4-digit filename. This filename is then split into path and filename as follows:<br>Digit 1: name of first directory level<br>Digit 2: name of second directory level<br>Digits 1-4: filename |
| Example:   | The file 1023.mp3 will be found in the directory<br>\1\0\1023.mp3   |
| Special Case:  | When the file number is 5 digits long, the file will be searched in the root directory.   |
| Response:  | PLAY: <filename>  |
| Errors:  | ERROR: not found.   |
| <b>PR:</b>   | <b>Play Number in Current Directory</b>   |
| Argument:  | 1...4-digit file number   |
| Description:   | Plays *.mp3 file with name corresponding to number given in argument. Number must be in range 0...9999.<br>The number argument is padded with '0' characters to a 4-digit filename. This filename is then searched in the current directory.  |
| Response:  | PLAY: <filename>  |
| Errors:  | ERROR: not found.   |
| <b>SF:</b>   | <b>Set File Name</b>  |
| Argument:  | Filename  |
| Description:   | Presets *.mp3 file with name given in argument. File will be played when PLAY signal is asserted.   |
| Response:  | FILE: <filename> ready.   |
| Errors:  | ERROR: not found.   |
| <b>SN:</b>   | <b>Set File Number</b>  |
| Argument:  | 1...4 digit file number   |
| Description:   | Presets *.mp3 file with name corresponding to number given in argument. Number must be in range 0...9999. File will be played when PLAY signal is asserted.<br>The number argument is padded with '0' characters to a 4-digit filename. This filename is then searched in the current directory   |
| Response:  | FILE: <filename> ready.   |
| Errors:  | ERROR: not found.   |

|              |  |
|--------------|--|
| <b>SP:</b>   | <b>Stop Playback</b>   |
| Argument:    | (none)   |
| Description: | Stops playing of present audio file. Player will return to idle state.   |
| Response:    | ACTION: stopped.   |
| <b>PS:</b>   | <b>Pause Playback</b>  |
| Argument:    | (none)   |
| Description: | Pauses playing of present audio file. Player will return to paused state. Audio playback may be resumed with CO command.                                     |
| Response:    | ACTION: paused.  |
| <b>CO:</b>   | <b>Continue Playback</b>   |
| Argument:    | (none)   |
| Description: | Resumes playing of present audio file at file position where it has been paused.   |
| Response:    | PLAY: <filename>: <Bytes played>: <Bytes Total>.   |
| <b>FO:</b>   | <b>Fade Out</b>  |
| Argument:    | (none)   |
| Description: | Fades out the currently played audio file and stops playback when the lowest volume level is reached. The fade out speed can be defined with the FT command. |
| Response:    | ACTION: Fading out...<br>ACTION: End of File.  |
| <b>FT:</b>   | <b>Set Fade Out Time</b>   |
| Argument:    | 1...5-digit numeric value  |
| Description: | Sets the fade out time in milliseconds per 2dB volume step.  |
| Response:    | SYSTEM: Fade Interval set to xxxx ms/2dB.  |
| <b>ST:</b>   | <b>Show Player Status</b>  |
| Argument:    | (none)   |
| Description: | Displays player status.  |
| Response:    | Idle   Playing   Paused   Error  |

## File Handling Commands

|              |   |
|--------------|---|
| <b>DI:</b>   | <b>Directory Information</b>  |
| Argument:    | (none)  |
| Description: | Displays directory listing of current directory.                    |
| Response:    | DIR: path: <path of current directory><br>DIR: <filename><br>...    |
| <b>FI:</b>   | <b>File Information</b>   |
| Argument:    | (none)  |
| Description: | Displays file information of currently playing or preset *mp3 file. |
| Response:    | FILE: <filename>>: <Bytes played>: <Bytes Total>                    |

**CD: Change Directory**

---

Argument: Directory name  
 Description: Changes the current directory to the one given in the argument. Directory name must be 1...8 characters long. Wildcards are not allowed.  
 Special characters:  
 .. changes to the parent directory.  
 Restriction: A maximum of 4 directory levels is currently supported.  
 Response: DIR: <current directory>  
 Errors: ERROR: not found.

**GD: Get Current Directory**

---

Argument : (none)  
 Description: Displays the current working directory.  
 Response: DIR: <current directory>

**Sound Control Functions****VU : Volume Up**

---

Argument : (none)  
 Description: Increases playback volume by 2dB.  
 Response: VOL: <hex value>

**VD: Volume Down**

---

Argument: (none)  
 Description: Decreases playback volume by 2dB.  
 Response: VOL: <hex value>

**VS: Volume Set**

---

Argument: 2-digit Hex value  
 Description: Sets playback volume level to hex value given in argument. Value 0 corresponds to maximum volume or 0dB, value BF corresponds to minimum volume, value FF switches off the audio part of the VS1011 decoder.  
 Response: VOL: <hex value>

**Maintenance Functions****UD: Update Flash**

---

Argument: (none)  
 Description: Invokes firmware upload mode.  
 Response: Confirm Firmware Upload.  
 Enter Y to confirm, or any other character to abort. Please refer to the following chapter for instructions on download process.

## Loading of New Firmware to MP3orator

### Prerequisites

1. Image file with firmware in Intel-Hex format (\*.hex)
2. Download tool for PC: `flashloader.exe`
3. Serial connection. For this, a level shifter is required to convert the 3.3V logic levels on the MP3orator to RS-232 levels.

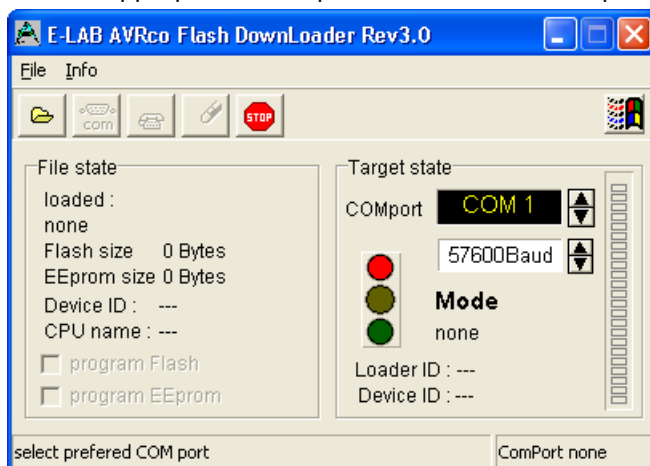
### Procedure

#### Hardware Setup

1. Connect MP3orator module via serial adapter to PC.
2. Apply power to the module

#### Prepare PC software

3. Download firmware file (\*.hex) from support website or receive it via email, and store it on PC.
4. Start a terminal program on the PC, set to 9600bit/s, 8 databits, 1 stopbit, no parity.
5. Check serial connection by entering the 'ST' command.
6. Enter the command 'UD' and confirm message with 'Y' to invoke the bootloader on the MP3orator module.
7. When the System LED on the MP3orator lits, close the terminal connection in order to release the COM port.
8. Start the download tool `flashloader.exe` on the PC.
9. Set the appropriate COM port and transmission speed to 9600 Baud:



10. Open the file selection dialog in the menu „File – Open File...“, select the firmware imagefile and select „Open“.
11. Click the button „com“. The status display on the right side should now show „Target connected“.
12. Start downloading by clicking on the “Telephone” button. The status display will now show „programming“, and the progress bar fills up. After the download is finished, the status message reads again „Target connected“.
13. Start firmware by clicking on the „Stop“ button.



**Check for Successful Download.**

4. Close the download tool
5. Open the terminal connection again and issue the 'SI' or any other command.

## Programming Guidelines

---

### CPU Port Assignments

| Port | Function | In/Out  | Assigned to              | Target        |
|------|----------|---------|--------------------------|---------------|
| PA0  | ADC0     | In/ Out | Application Interface X4 | X4.11         |
| PA1  | ADC1     | In/ Out |                          | X4.12         |
| PA2  | ADC2     | In/ Out |                          | X4.13         |
| PA3  | ADC3     | In/ Out |                          | X4.14         |
| PA4  | ADC4     | In/ Out |                          | X4.15         |
| PA5  | ADC5     | In/ Out |                          | X4.16         |
| PA6  | ADC6     | In/ Out |                          | X4.17         |
| PA7  | ADC7     | In/ Out |                          | X4.18         |
| PB0  | XCK      | Out     | VS1011 Decoder           | CS            |
| PB1  | T1       | Out     |                          | DCS           |
| PB2  | AIN0     | Out     |                          | Reset         |
| PB3  | AIN1     |         |                          |               |
| PB4  | SS       | Out     | Sdcard                   | CS            |
| PB5  | MOSI     | Out     | Sdcard, VS1011           | SDI, SI       |
| PB6  | MISO     | In      |                          | SDO, SO       |
| PB7  | SCK      | Out     |                          | SCK           |
| PC0  | SCL      | In/Out  | Application Interface X4 | X4.4          |
| PC1  | SDA      | In/Out  |                          | X4.5          |
| PC2  | TCK      |         |                          |               |
| PC3  | TMS      |         |                          |               |
| PC4  | TDO      |         |                          |               |
| PC5  | TDI      | Out     | System LED               | X4.20         |
| PC6  | TOSC1    | In      | SDcard                   | Card Detect   |
| PC7  | TOSC2    | In      |                          | Write Protect |
| PD0  | RXD      | In      | Serial Interface         | X4.2          |
| PD1  | TXD      | Out     |                          | X4.3          |
| PD2  | INT0     | In      | VS1011                   | DREQ          |
| PD3  | INT1     |         |                          |               |
| PD4  | OC1B     |         |                          |               |
| PD5  | OC1A     |         |                          |               |
| PD6  | ICP      |         |                          |               |
| PD7  | OC2      |         |                          |               |

## Writing VS1011 register values

The VS1011 registers can be accessed over the SPI, controlled by the CS signal. Each register is one word (16 bits) wide. Please refer to the VS1011 datasheet for a complete description of the registers.

Following pseudocode shows how to access the registers:

```
Assert CS
Issue VS1011 write command: send 0x02 to SPI
Wait for data to be sent
Send register address to SPI
Wait for data to be sent
Send high byte of register value to SPI
Wait for data to be sent
Send low byte of register value to SPI
Wait for data to be sent
Deassert CS
Delay 5 microseconds after sending data
```

## VS1011 Initialization Procedure

The shaded paragraphs refer to VS1011 registers. Please see VS1011 datasheet for register addresses and access information.

### Initialize Host Interface Ports

1. Set CS (MP3/ChipSelect) as output
2. Set MOSI as output
3. Set SCK as output
4. Set SCK to low level
5. Set SPI clock speed to 2 MHz
6. Select clock phase positive going in middle of data
7. Set SPI to master mode

### Hardware reset of VS1011

1. Assert VS1011 reset
2. Delay 30ms
3. Deassert CS by setting to high level
4. Deassert DCS by setting to high level
5. Release VS1011 Reset
6. Delay 10ms (2.5ms according to datasheet)
7. Set volume to minimum:

```
VOL <- 0xFFFF
```

8. Set CLOCKF to compensate for a 24.576 MHz x-tal. Activate clock doubler. The value should be fClock/2000. The values shown are for the 12.288MHz crystal on the MP3 adapter board.

```
CLOCKF <- 6144 + 0x8000
```

9. Delay 1 ms
10. Wait for DREQ
11. Set slow sample rate for slow analog part startup

```
AUDATA <- 10 // 10 Hz
```

12. Delay 100 ms
13. Switch on the analog parts

```
VOL <- 0xFEFE
```

```
AUDATA <- 44101 // 44.1kHz stereo
VOL <- 0x2020 // or other default value
```

### Software Reset of VS1011

1. Delay 200ms
2. Set SW reset bit, set VS10xx native mode on SPI, allow tests (e.g. sine tests)

```
MODE <- SM_RESET + SM_SETTOZERO3 + SM_SDINNEW + SM_TESTS
```

3. Delay 2us
4. Rewrite SCI\_CLOCKF after soft reset

```
CLOCKF <- 6144 + 0x8000
```

5. Send 1024 nulls

```
Assert DCS
Send 0x53 to SPI
repeat
    Wait for DREQ
    Send 0x00 to SPI
    Wait for data to be sent to SPI
Until 1024 bytes are done
Deassert DCS
```

### Playing MP3 Data

Before playing a new MP3 stream, perform a software reset on the VS1011.

Below is an example pseudocode for playing a stream of MP3 data by reading blocks of data from a mass storage device and sending them to the VS1011.

```
Software reset VS1011
Open storage file for reading
For each data block do
    Read one data block of mp3 data
    For each 32 bytes in the data block do
        wait for DREQ
        Assert DCS
        Repeat
            Send next byte to SPI
            Wait until data byte was sent
        Until 32 bytes are done
        Deassert DCS
    Endfor
Endfor
Close File
```

## Appendix

---

### Notice to Users

The intended use of the MP3orator modules is described in this document. Other than the described uses are not permitted or only after consultation with the manufacturer.

MP3orator modules are not authorized for use as critical components in life-support devices or systems.

Life-support devices or systems are devices or systems intended for surgical implantation into the body or to sustain life, and whose failure to perform, when properly used in accordance with instructions for use provided in the labeling and user's manual, can be reasonably expected to result in significant injury.

No complex software or hardware system is perfect. Bugs are always present in a system of any size. In order to prevent danger to life or property, it is the responsibility of the system designer to incorporate redundant protective mechanisms appropriate to the risk involved.

All MP3orator modules are 100 percent functionally tested. Additional testing may include visual quality control inspections. Specifications are based on characterization of tested sample units rather than testing over temperature and voltage of each unit. MP3orator modules may qualify components to operate within a range of parameters that is different from the manufacturer's recommended range.

### Revision History

| Revision | Date       | Comments                               |
|----------|------------|--|
| 1.0      | 12.10.2006 | Initial Release                        |
| 1.1      | 29.06.2007 | Added PC5 to X4.20                     |
| 2.0      | 28.02.2008 | Added Description of serial commands   |
| 2.1      | 28.03.2009 | Added Description of keyboard commands |