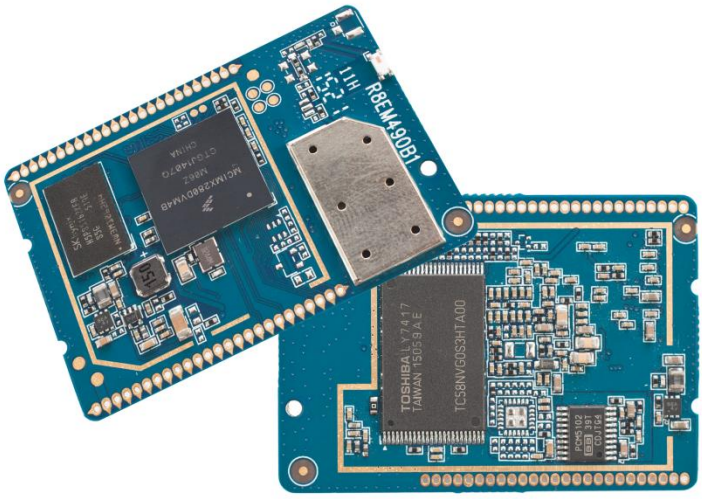
LinTech

Wireless Audio Module
HBM10

Specification v4.2.0



Contents

LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

Changes

Date	Version	Changes	Author
2016-06-27	4.2.0	Firmware 1. Improved volume control and AirPlay audio playback 2. Auto disable WLAN interface if Ethernet link is detected 3. New radio playback features: • Play and navigate with the multimedia keys through a playlist • Define a favourite station • Enhanced *query* command 3. New sound features: • Get and set the master volume • Get and set the volume for the status tones 4. Improved network toggle button handler 5. Measure network bandwith with iperf3 6. Fix some network issues Documentation 1. New HTTP API v1.5 2. Add note about volume control in ch. 7.2 3. Add note about delay concerning AUDIO_STATUS_PIN in ch. 7.4 4. Add note about PWM-driven LEDs in ch. 6.3	Jörg Krause
2016-05-02	4.1.2	Firmware 1. Fix duplicated Ethernet MAC addresses 2. Fix LED blinking 3. Fix I2S BLCK	Jörg Krause
2016-04-11	4.1.1	Firmware 1. Fix streaming issue with some mp3 playlists 2. Fix streaming issue with Windows Media Player Documentation 1. Add note about supported sampling rates of WM8804 in ch. 7.2	Jörg Krause
2016-03-21	4.1.0	Firmware 1. Add support for setting the brightness of the LEDs 2. Improve support for MP3 audio files 3. Improve update over the air 4. Fix factory reset sent via HTTP-API 5. Fix storing volume Documentation 1. Add section 14.4 to HTTP API	Jörg Krause
2016-03-02	4.0.1	Firmware 1. Fix HTTP command “Radio Stop” 2. Fix an AirPlay bug, causing unnecessary resynchronizations 3. Fix an incompatibility bug with some UPnP media control points	Jörg Krause
2016-02-29	4.0.0	Firmware 1. Add support for remote control of audio playback with keys 2. Improve performance and stability	Jörg Krause

LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

		3. Feature “Serial to WifiBridge” is now optional and not enabled by default Documentation 1. Another fix for the sample request in section 14.2.1	
2016-02-15	3.3.6	Firmware 1. Fix storing volume changes followed by a power-cut 2. Fix wrong AirPlay volume after changing UPnP/Radio volume Documentation 1. Fix sample request in section 14.2.1	Jörg Krause
2016-01-27	3.3.5	Firmware 1. Improve switching between different audio streams by temporarily turning AirPlay off, when UPnP or radio playback is about to begin 2. *HBM10-ETH*: Fix UPnP not available on Ethernet Documentation 1. Fix some key names and descriptions in ch. 4	Jörg Krause
2016-01-15	3.3.4	Firmware 1. Fix radio playback after network reconnection 2. Fix ”Next Track”-Bug using AirPlay on iOS 9.2 3. Disable DHCP server in network client mode	Jörg Krause
2015-12-23	3.3.3	Firmware 1. Fix radio unintentionally starting a stream after reboot although radio stream was stopped	Jörg Krause
2015-12-14	3.3.2	Firmware 1. Fix UPnP stuttering in case a Control Point becomes suddenly unreachable 2. Fix audio noise in quite passages	Jörg Krause
2015-12-01	3.3.1	Firmware 1. HTTP API v1.3 2. Expose API version in Zeroconf service description 3. Fix UART issue outputting characters twice 4. Fix UPnP not working on Windows 5. Fix AUDIO_STATUS for internet radio 6. Improve network management for wifi + ethernet 7. Fix missing HTTP response for action setconfig, in case of changing the device name only without setting a network configuration Documentation 1. HTTP API v1.3 2. Fix current consumption values 3. Improve reference schematics sketches	Jörg Krause
2015-10-21	3.3.0	Firmware 1. Internet radio support 2. HTTPS support Documentation 1. New HTTP API v1.2 2. Module Pin Definition: Audio KEYS are optional 3. HTTP API: Add note to 14.2.2	Jörg Krause
2015-10-19	3.1.3a	Firmware 1. Fix another power-cut issue	Jörg Krause
2015-09-30	3.1.3	Firmware 1. Fix bricking issue in case of a power-cut during boot	Jörg Krause

LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

		process 2. Fix hostname issue causing trouble with some routers 3. Improve HTTPs JSON parser robustness 4. Backport support for HTTP API v1.1 5. Bump Linux Kernel to 4.1 LTS Documentation 1. Align device name description in ch. 13 to with hostname fix. 2. New ch. 8.1	
2015-08-18	3.1.2a	Firmware 1. Append last four digits of the mac address to wifi ssid Documentation 1. Fix default values in ch. 13 and 14	Jörg Krause
2015-08-06	3.2.0	Firmware 1. HBM10-ETH with Ethernet support 2. Serial to Wi-Fi bridge Documentation 1. New chapters: Ethernet, Pin Headers, Serial to WiFi Bridge, AT Commands 2. New HTTP API v1.1 3. Several improvements all over the places	Jörg Krause
2015-07-06	3.1.2	Firmware 1. Fix clock synchronization with AirPlay 2. Pin *AUDIO_STATUS* also works now for closing UPnP connections 3. Improve compatibility with WHAALE app 4. Change default device and SSID name to “HBM10” Documentation 1. Fix wrong values for MCLK frequencies in the table of ch. 7.4	Jörg Krause
2015-05-26	3.1.1	1. Fix non-working *AUDIO_STATUS* in certain cases when streaming with UPnP	Jörg Krause
2015-05-20	3.1.0	1. Add module pin *AUDIO_STATUS* (pin #8)	Jörg Krause
2015-05-19	3.0.0	1. Enable GPIOs 2. Add an icon for UPnP device rendering	Jörg Krause
2015-05-18	2.0.1	1. Fix issue when updating firmware with the iOS app	Jörg Krause
2015-05-04	2.0.0	1. Initial version	Jörg Krause

LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

1 Introduction

The **HBM10** is a low-cost and powerful wireless audio System-on-Module (SOM) bundled on a small 34mm x 50mm PCB. Its complete reference design drastically reduces the development time to start your own application on a custom carrier board.

The **HBM10-ETH** is additionally equipped with an 10/100 Mbps Ethernet transceiver for extended possibilities to setup your network.

The integrated and ready to use Linux-powered software stack supports audio streaming wirelessly with AirPlay and UPnP/DLNA to all kind of home audio equipment including home theater systems, A/V receivers, radios, wireless speakers and portable music players.

Furthermore, with an HBM10 you can turn your home audio system into an wireless internet radio player.

1.1 Target Applications

- Network music stations
- HiFi-systems
- Light and sound systems
- iPod docks
- Portable audio system
- Boom-boxes
- Network audio loudspeakers
- Wireless media adapters
- Complete radio and audio products

1.2 Software Features

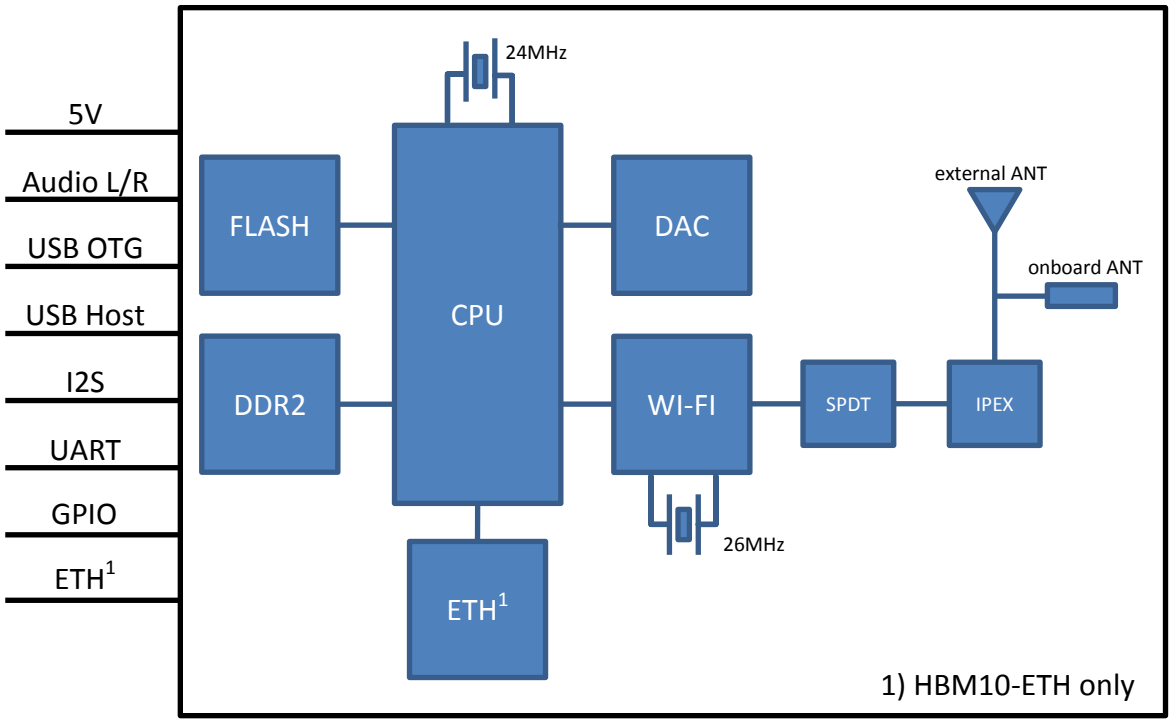
- Linux Kernel 4.4
- Bootloader (USB Firmware recovery)
- Buildroot rootfs (pre-installed in NAND Flash)
- Audio Streaming Protocols:
    - AirPlay
    - UPnP/DLNA
    - OpenHome
    - HTTP
- Firmware Update over the Air
- Remote Control through HTTP API
- Internet radio player
- Audio playback control keys support
- Serial to Wifi Bridge *(optional)*

LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

1.3 Board Features

	HBM10	HBM10-ETH
CPU	ARM9@454 MHz + Security Co-processor 128-bit AES hardware decryption	
RAM	512 Mbit DDR2	
Flash	1 Gbit NAND	
Ethernet	—	10/100 Mbps
Interfaces	I2S, I2C, UART, GPIOs	
Dimension	33.8mm x 49.5mm x 5mm	
Approvals & Certifications	CE, FCC, RoHS	

1.4 Module Block Diagram



LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

2 System Specification

Platform	OS	Linux 4.4
	CPU	ARM9 454 MHz + Security Co-processor 128-bit AES hardware decryption
	Wi-Fi	BCM43362
Memory	NAND FLASH	1 Gbit
	RAM	512 Mbit
Wi-Fi	Frequency Band	2.4 GHz
	Frequency Range	2.412 GHz ~ 2.484 GHz
	Channels	1 - 13
	Protocols	IEEE 802.11b IEEE 802.11g IEEE 802.11n
	Max data rates	*802.11b*: 11 Mbps *802.11g*: 54 Mbps *802.11n*: 150 Mbps
	Security	*Encryption*: None, WEP, WPA, WPA2 *Ciphers*: CCMP, TKIP
	Network Modes	Access Point Client
	Antenna	Onboard SMT antenna, 50 Ω (*default*) IPEX connector to external antenna (*optional*)
	EVM	*802.11n*: -30 dB
	Maximum transmit power	*802.11b*: 16 dBm, EVM: 28 % *802.11g*: 14 dBm, EVM: 28 % *802.11n*: 12 dBm, EVM: -30 db
	RSSI	*802.11b*: -90 dBm *802.11g*: -70 dBm *802.11n*: -70 dBm
Audio	Protocols	AirPlay, UPnP/DLNA, OpenHome, HTTP
	Formats	MP3, AAC, Vorbis, Opus, PCM, WMA, AC3, FLAC, ALAC, APE, WavPack
	Container	MP4, MKV, OGG, WAV, AIFF, ASF
	Audio Data Lengths	16 and 24 bit
	Sampling Frequency	8 to 192 kHz (Wi-Fi: up to 48 kHz)
	SNR	> 110dB
Interfaces	UART	1x
	USB 2.0	1x OTG 1x Host (optional)
	Audio	1x I2S 1x Line out (R, L)
	I2C	1x
	Power	*Input*: 5 V

LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

		Output: 3.3 V
	Reset	1x
	GND	4x
	LED	2x
	KEYS	1x
	LRADC	1x
	ANT	1x
	GPIO	Up to 14x
Environment	Operation Temperature	-10 to 70 ℃
	Operation Humidity	10 to 90 %
	Storage Temperature	-40 to 100 ℃
	Storage Humidity	5 to 95 %
Performance	Boot Strap	~15 Sec
	Power Dissipation	< 3 W
	Current Consumption	typ: 200 mA @ 5V max: 800 mA @ 5 V
Operating Condition	VDD	5 V ± 5%
	VDD_DAC	3.3 V ± 3%

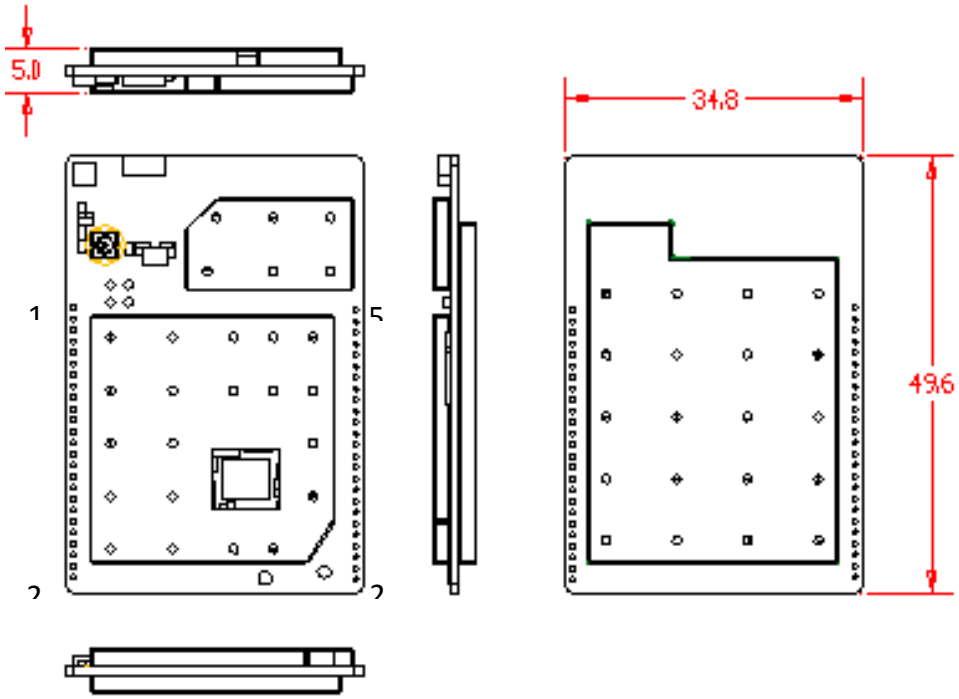
LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

3 Mechanical Specifications

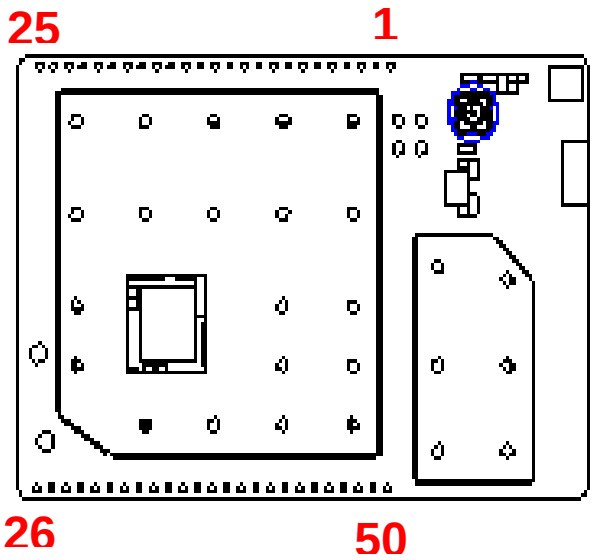
3.1 Dimension

Parameter Typical Units: 50pins

- Dimension (LxWxH): 34.8 x 49.6 x 5 mm
- Dimension tolerances: ±0.2 mm



3.2 Footprint View



LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

4 Module Pin Definition

Pin	Name	I/O	Function	Notes
1	KEY_WIFI_MODE	I	Wi-Fi AP/Client mode (short pressed) FACTORY_RESET (long pressed)	
2	HW RESET	I	HW_RESET	
3	FEC_LED	O	ETH	4
4	GND	#	DIGITAL GROUND	
5	GPIO12	I/O	GPIO	2
6	GPIO13	I/O	GPIO	
7	GPIO14	I/O	GPIO	
8	AUDIO_STATUS	O	Audio status	
9	KEY_VOLUME_UP	I	Key – Volume up	
10	GPIO1	I/O	GPIO	2
11	GPIO2	I/O	GPIO	
12	GPIO3	I/O	GPIO	
13	GPIO4	I/O	GPIO	
14	GND	#	DIGITAL GROUND	
15	GPIO5	I/O	GPIO	1, 2
	LRADC	I	Low-rate ADC	1, 3
16	GPIO6	I/O	GPIO	1, 2
	UART0_TX	O	UART – Tx	1
17	GPIO7	I/O	GPIO	1, 2
	UART0_RX	I	UART – Rx	1
18	GPIO8	I/O	GPIO	1, 2
	UART0_CTS	O	UART – CTS	1
19	GPIO9	I/O	GPIO	1, 2
	UART0_RTS	I	UART – RTS	1
20	VDDIO_3V3	#	I/O voltage for GPIO	
21	LED1	O	Status LED 1	
22	LED2	O	Status LED 2	
23	GPIO10	I/O	GPIO	1, 2
	I2C_SDA	I/O	I2C – SDA	1, 3
24	GPIO11	I/O	GPIO	1, 2
	I2C_SCL	I/O	I2C – SCL	1, 3
25	GND	#	GND	
26	BATTERY	#	Battery input	3
27	USB_5V	#	USB 5V IN	
28	AGND	#	ANALOG GND	
29	LINE_OUT_R	O	Line out Right	
30	LINE_OUT_L	O	Line out Left	
31	KEY_VOLUME_DOWN	I	Key – Volume down	
32	I2S_LRCLK	O	I2S – LRCLK	
33	I2S_MCLK	O	I2S – MCLK	
34	I2S_BCLK	O	I2S – BITCLK	
35	I2S_DOUT	O	I2S – DOUT	
36	I2S_DIN	I	I2S – DIN	
37	FEC_A3V3	#	ETH – 3V3 supply	4
38	ETH0_RXP	I	ETH – RXP	4
39	ETH0_RXN	I	ETH – RXN	4

LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

40	ETH0_TXP	O	ETH – TXP	4
41	ETH0_TXN	O	ETH – TXN	4
42	USB0_DM	I/O	USB OTG – D-	
43	USB0_DP	I/O	USB OTG – D+	
44	USB1_DM	I/O	USB Host – D-	3
45	USB1_DP	I/O	USB Host – D+	3
46	GND	#	DIGITAL GND	
47	KEY_PLAY_PAUSE	I	Key – Play/Pause	
48	KEY_STOP	I	Key – Stop	
49	KEY_NEXT	I	Key – Next	
50	KEY_PREV	I	Key – Previous	

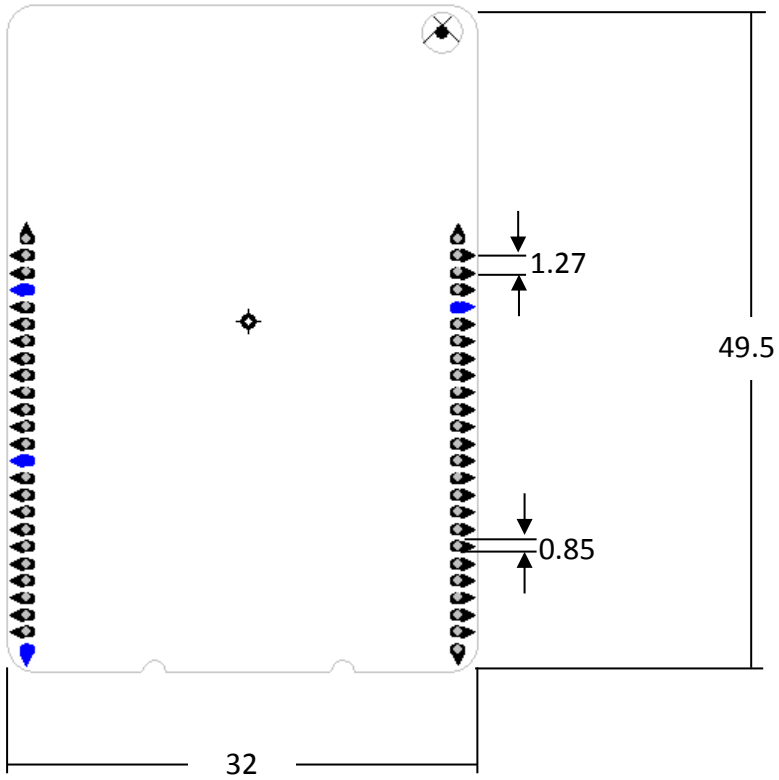
Notes:

1. Pins 15 – 19 and pins 23 – 24 are multiplexed
2. All GPIOs can be fully customized, see ch 10.4 Customization GPIOs,
3. Optional, not enabled by default
4. Only available on module HBM10-ETH

LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

5 Application Information

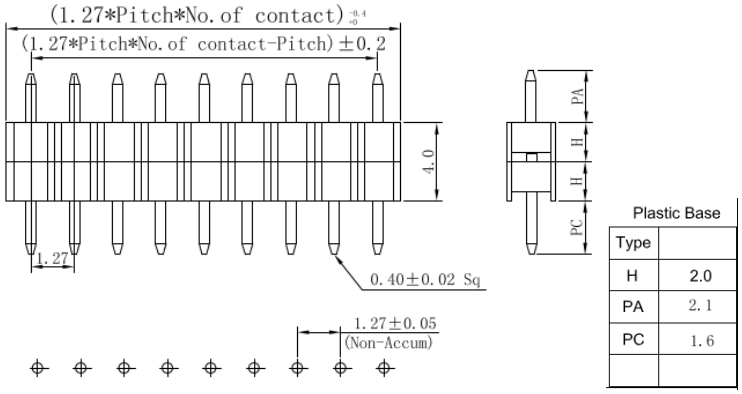
5.1 Recommended host circuit board PCB pattern



Recommended Host (customer) PCB Pattern

5.2 Pin Header

A pin header with a pitch of 1.27 mm is required.

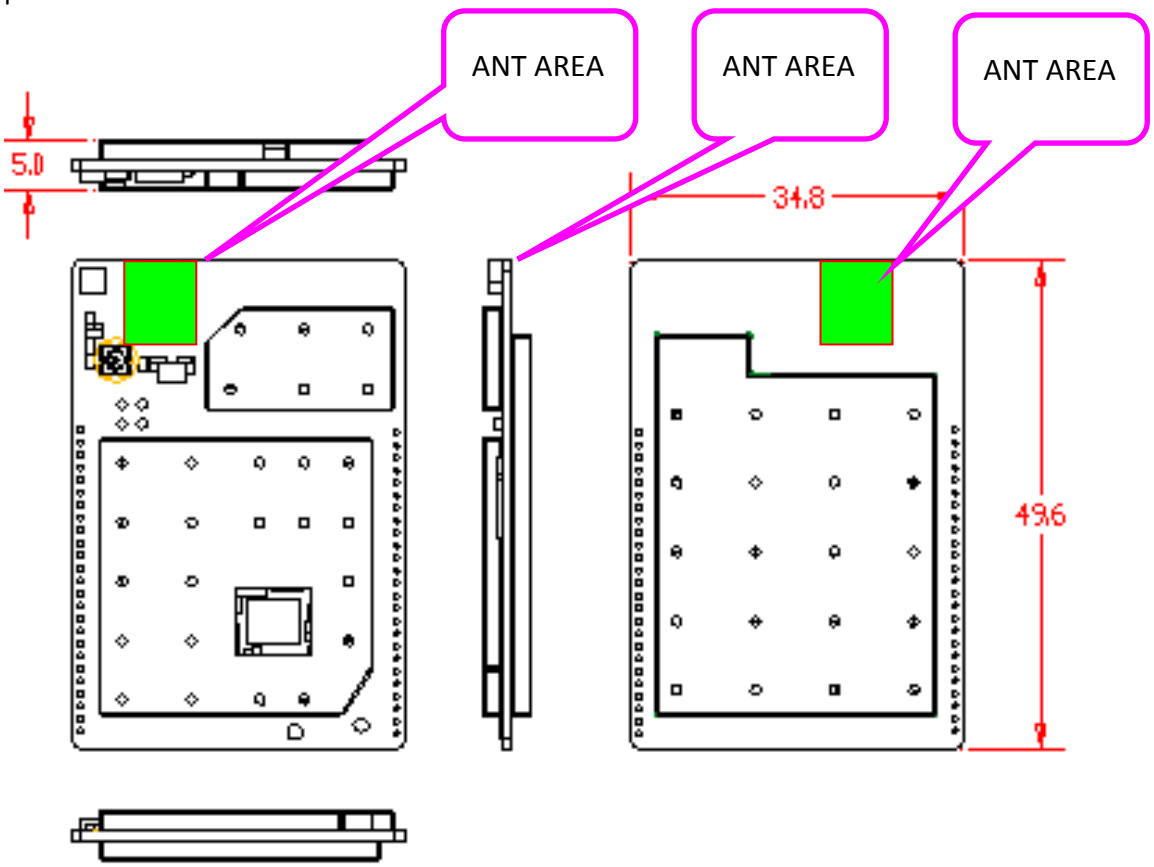


Plastic Base	
Type	
H	2.0
PA	2.1
PC	1.6

LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

5.3 Host PCB layout recommendations

The HBM10 module has an onboard antenna. Please make sure that the radio can achieve its best RF performance.



Recommended Host Circuit Board Design underneath the Module

NOTES

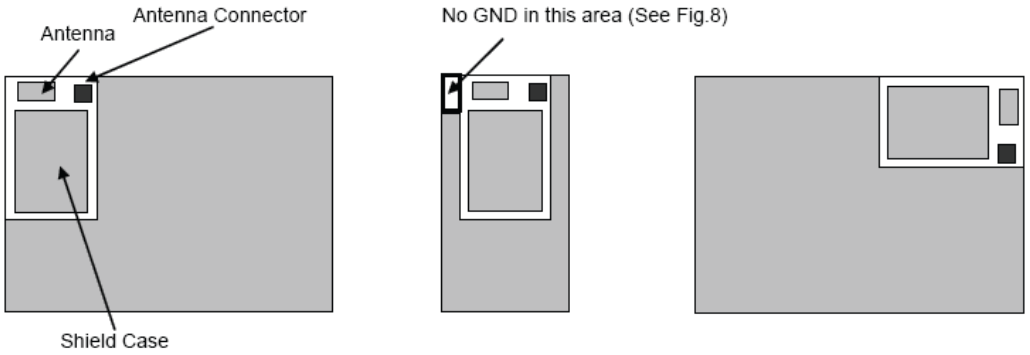
1. **Due to the surface mount antenna on the module, the green area on all layers of the customer circuit board should be free of any metal objects. Specifically, there should be no ground plane, traces or metal shield case.**
2. **The wireless signal including Wi-Fi applications is mostly affected by the surrounding environment, such as trees, and other obstacles. Metal absorbs a certain radio signal. In practical application, the data transmission distance is affected.**
3. **Please do not use metal housings.**

LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

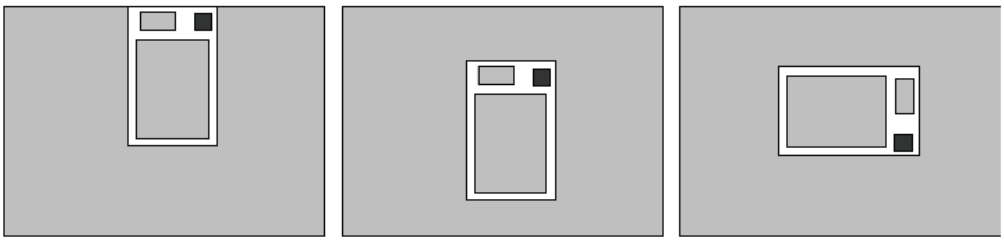
5.4 Module placement

For optimum EIRP, the customer is advised to use the recommended module placement on their host circuit board (see below).

Location in x-y plane

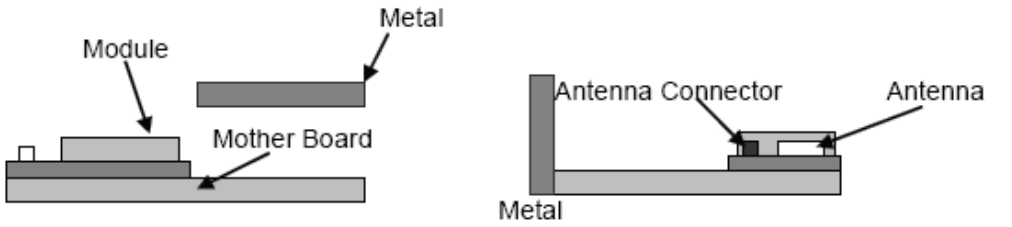


Recommended Placement in xy-plane

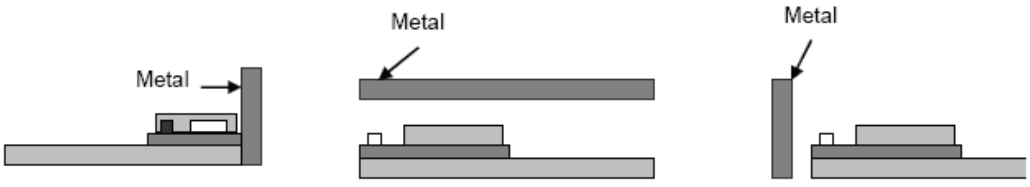


Locations Not Recommended in xy-plane

Location in z-plane



Recommended Locations in z-plane

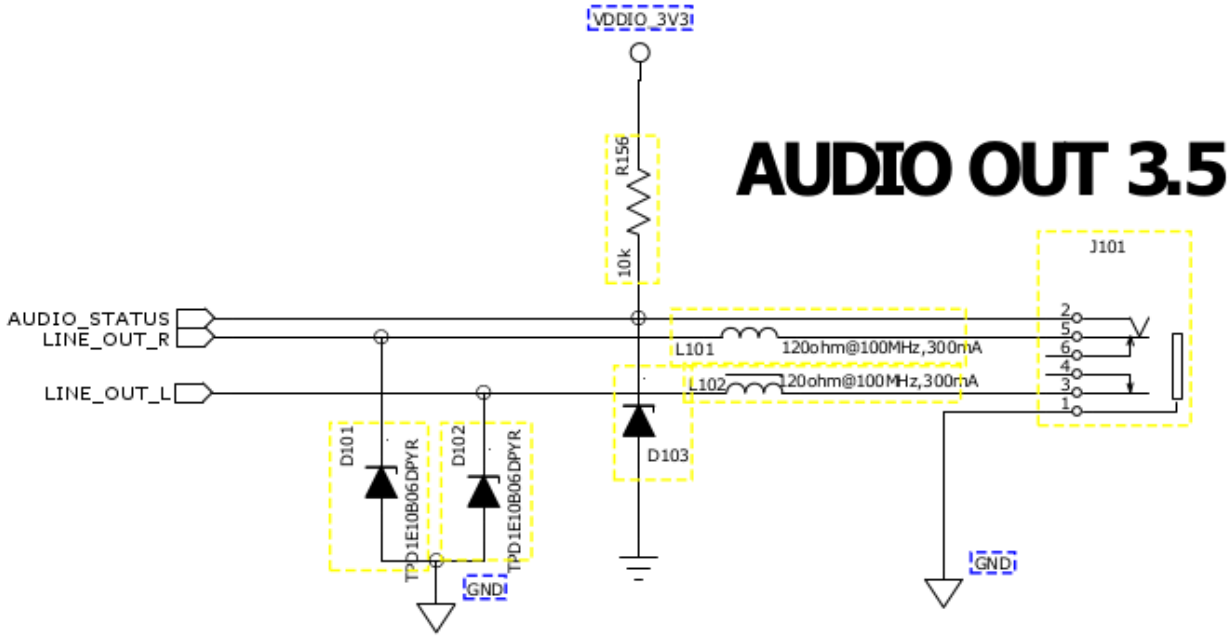


Locations Not Recommended in xy-plane

LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

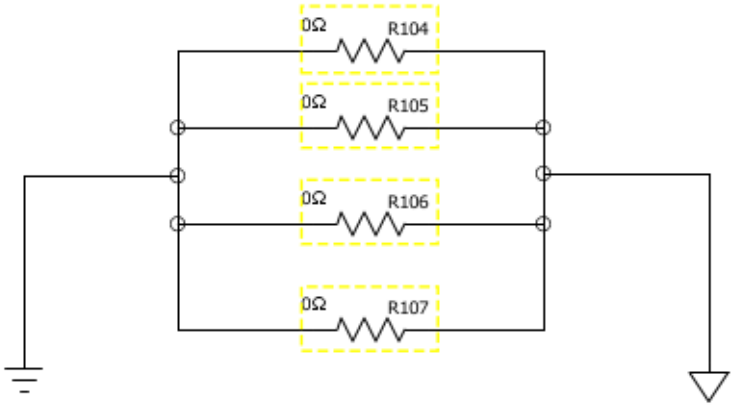
6 Reference schematic

6.1 Audio



Audio out 3.5mm

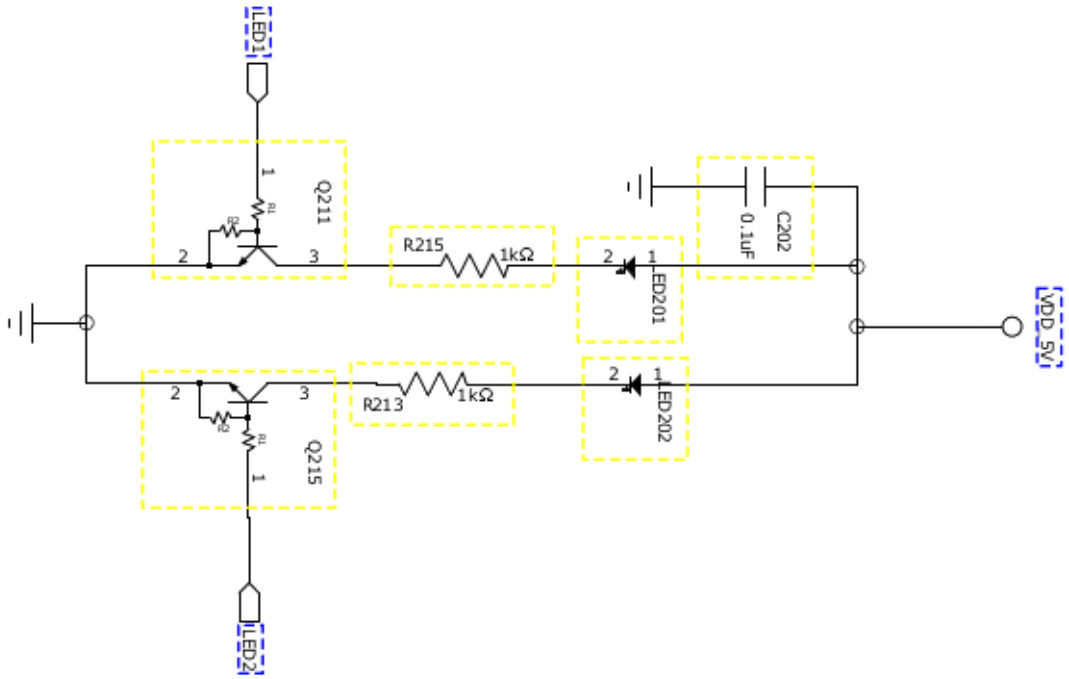
6.2 Ground



AGND and GND must be at different planes

LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

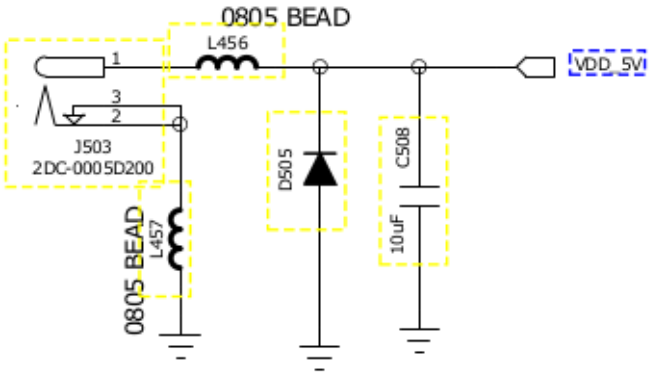
6.3 LED



LED

Note that the LEDs are PWM driven.

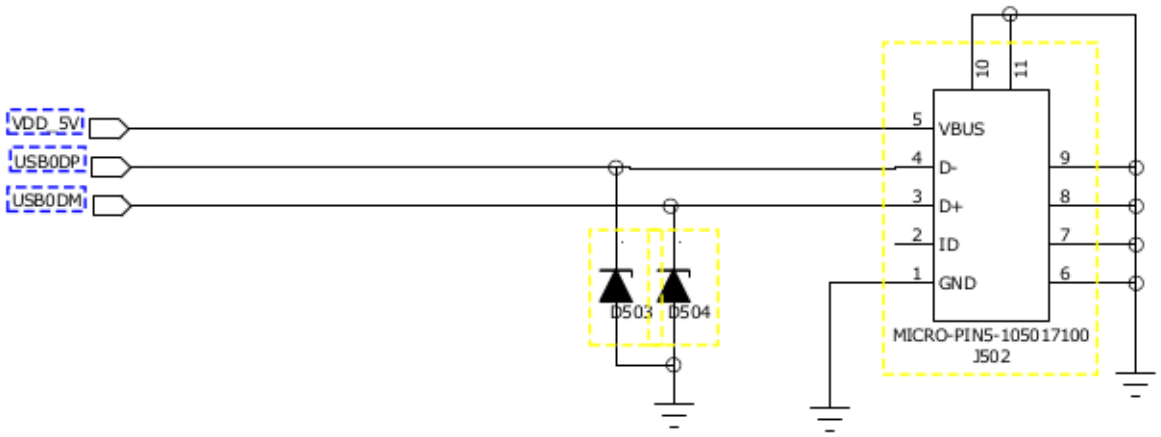
6.4 5V-Power in



5V-Power in

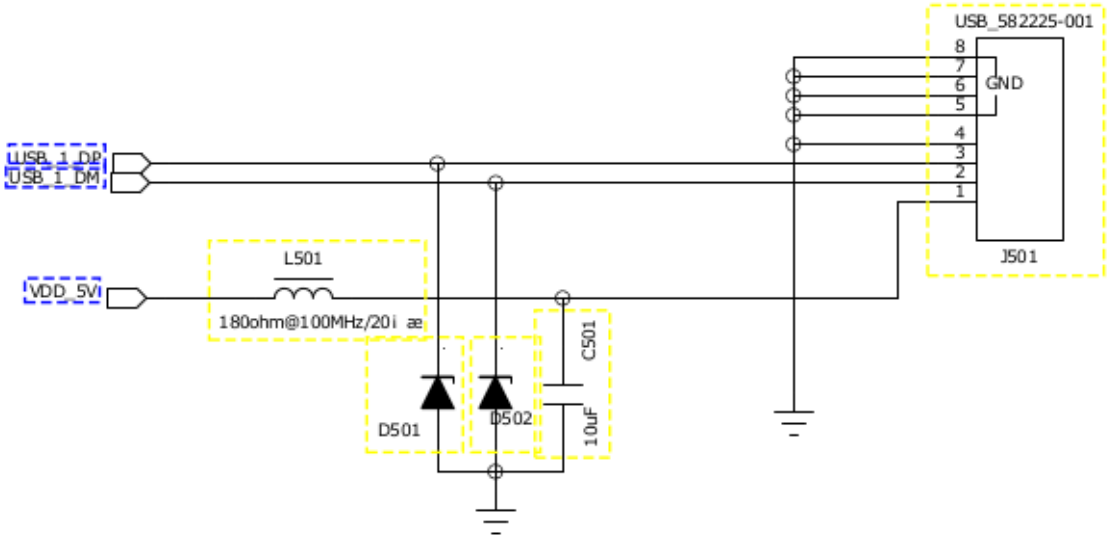
LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

6.5 USB OTG



USB OTG

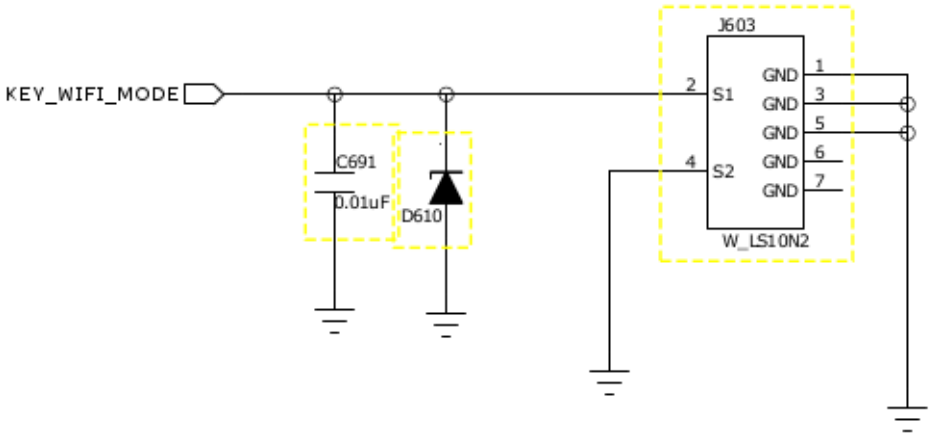
6.6 USB Host



USB Host

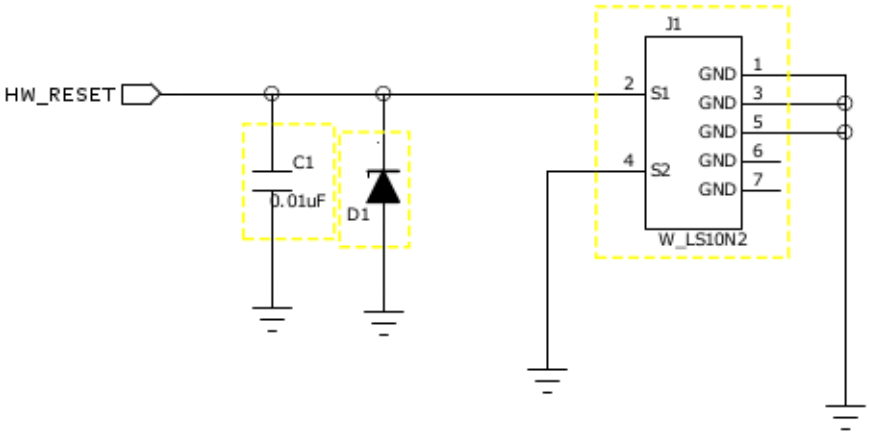
LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

6.7 Key WiFi-Mode



Key WiFi-Mode

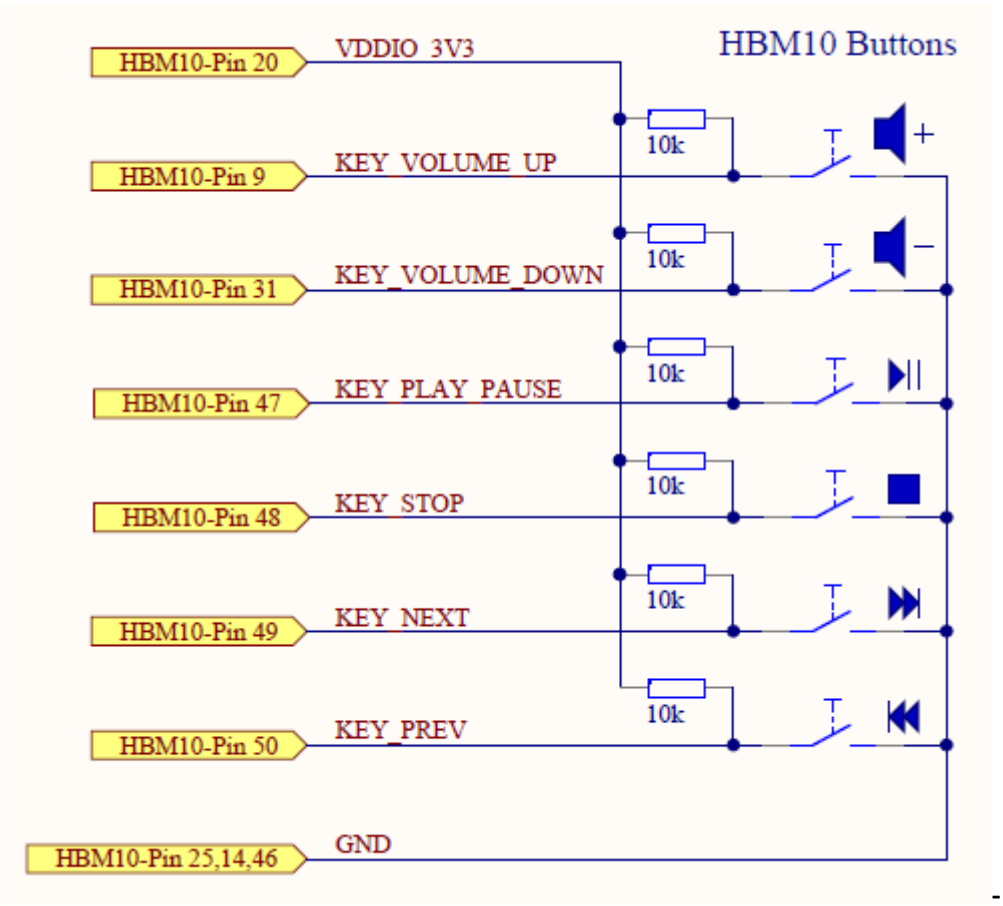
6.8 HW-Reset



Key HW-Reset

LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

6.9 Audio Control Keys

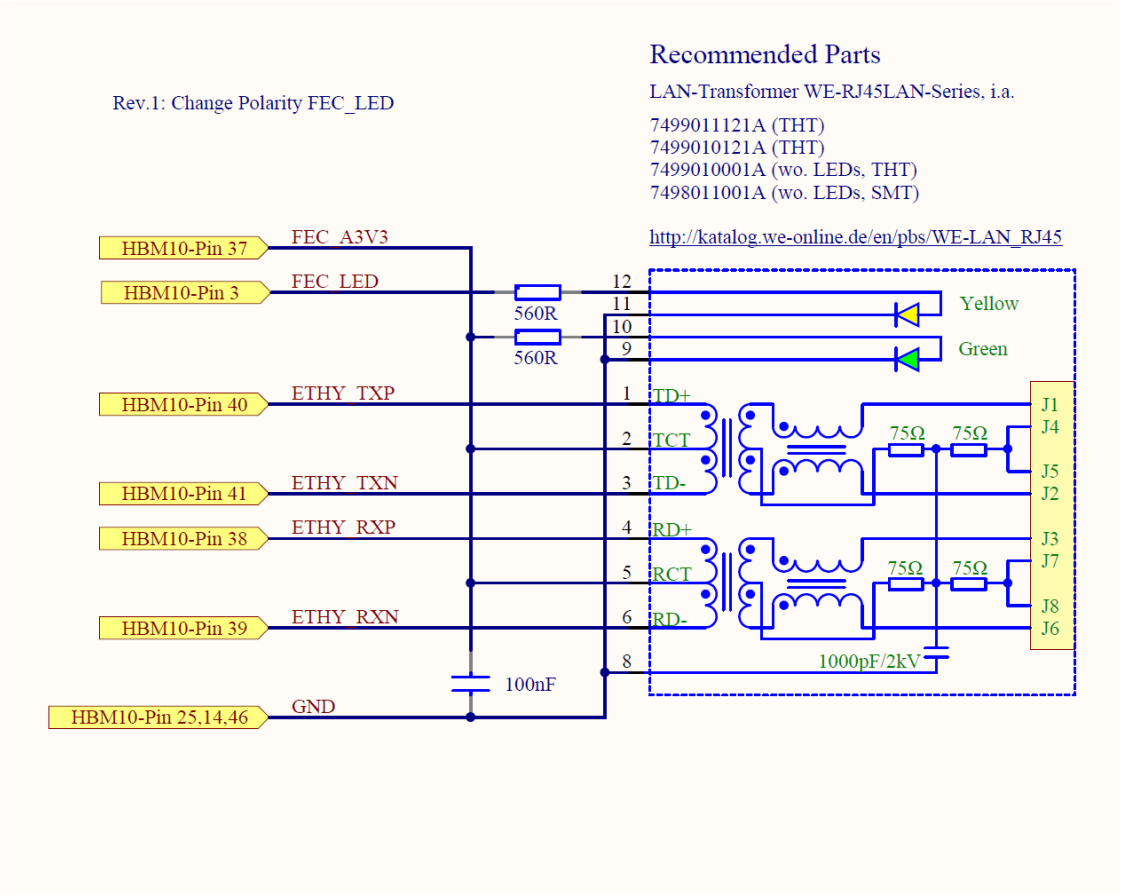


Keys Audio Control

LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

6.10 Ethernet

NOTE: Ethernet is only available for the HBM10-ETH module.



LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

7 Reference design

The LinTech GmbH provides a complete reference design for using the HBM10 on a custom carrier board. The use of this reference design is the cost effective alternative to own development of your professional Hi-Fi audio applications.

Key features:

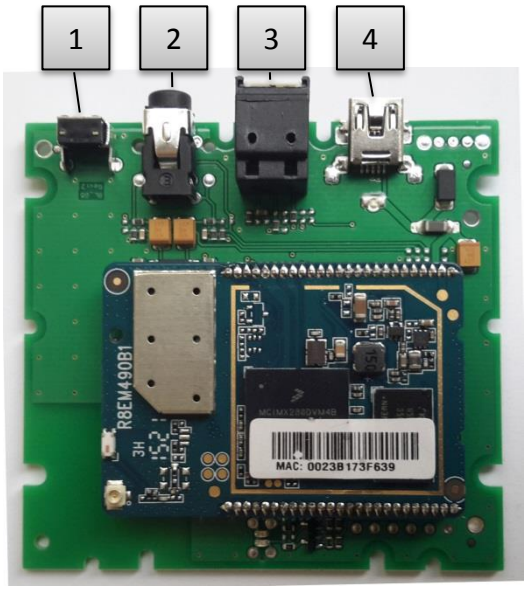
- stream music from various audio sources over WLAN
- remote control and configuration with apps

Supported audio streaming protocols:

- AirPlay
- UPnP/DLNA
- OpenHome

7.1 Hardware

	Port	I/O	Spec
1	Key	I	Toggle AP/Client mode (short pressed) Factory reset (long pressed)
2	Audio Line-Out **analog**	O	Audio jack 3.5mm
3	Audio Line-Out **digital**	O	Optical S/PDIF (TOS Link)
4	Power supply 5V, 800mA	I	3,5mm DC 5V



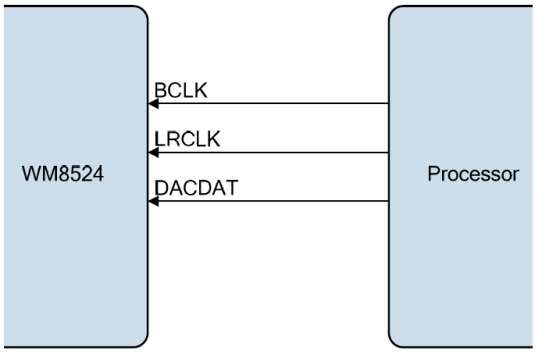
Dimensions: 81mm × 70mm

LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

7.2 Audio Output

The reference design features a Wolfson WM8524 stereo DAC with integral charge pump and hardware control interface together with a Wolfson WM8804 S/PDIF transceiver. The analogue output level for the reference design is set to **$2V_{rms}$** typical for 0dBFS.

The Serial Audio Interface (SAIF) interface of the i.MX28 processor transmits the PCM audio data to the WM8524 and WM8804 both operating in slave mode:



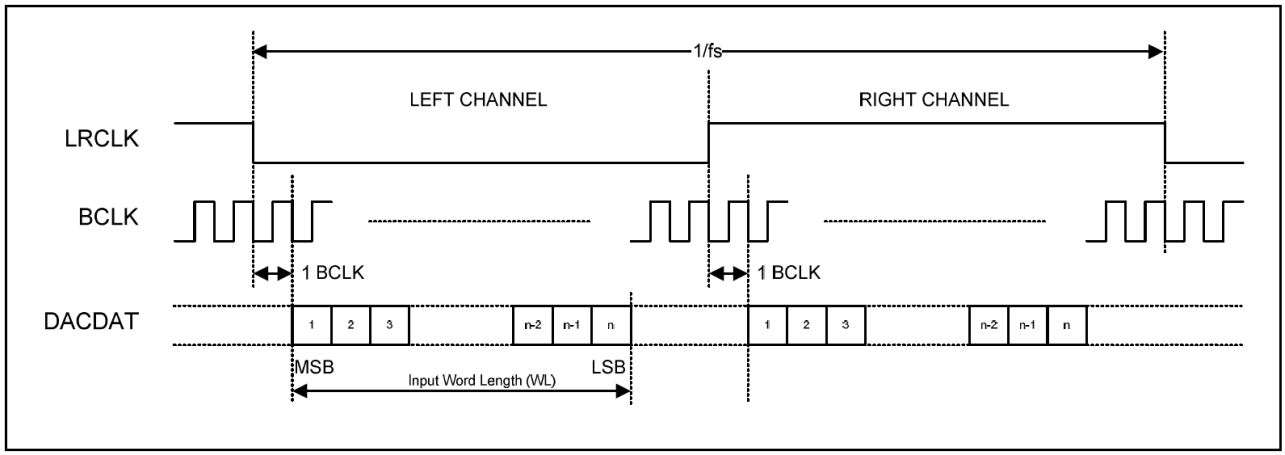
DAC operating in slave mode

I2S Audio Format

The supported interface format for PCM audio data transmission is I2S. The MSB is sent first. A word length of 24 bits is used.

Audio data for each stereo channel is clocked with the BCLK signal. Data is time multiplexed with the LRCLK, indication whether the left or right channel data is present. The LRCLK is also used as a timing reference to indicate the beginning or end of the data words. The minimum number of BCLKs per LRCLK period is two times the number of 24 bits.

The MSB of the output data changes on the first falling edge of BCLK following an LRCLK transition, and may be sampled on the next rising edge of BCLK. LRCLK is low during the left samples and high during the right samples.



I2S Audio Format

LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

The Audio Interface supports a MCLK to LRCLK ratio of 192*fs and 384*fs and sampling rates of 8kHz to 192KHz[1]. The BCLK base rate is 48*fs.

Sampling Rate (kHz) LRCLK	MCLK (MHz)		BCLK (MHz)
	192*fs	**384*fs**	**48*fs**
8	–	3.072	0.384
32	–	12.288	1.536
44.1	–	16.9344	2.1168
48	–	18.432	2.304
88.2	–	33.8688	4.2336
96	–	36.864	4.608
176.4	33.8688	–	8.4672
192	36.864	–	9.216

MCLK Frequencies and Audio Sample Rates

Volume Control

The audio volume is controlled by a software control with a dynamic range from -57.2 to -6.2 dB and a resolution of 256 corresponding to a step of 0.2 dB.

The volume is stored periodically. However, to make sure a volume change withstands an abrupt power-cut a delay of 10 seconds is necessary.

7.3 Status LEDs

Two status LEDs are used to indicate the current device status – a blue and a red led:

Status	LED	
	Blue	**Red**
Bootloader initializes hardware	On	On
Bootloader starts Linux	On	Off

[1] The WM8804 S/PDIF transceiver supports sampling rates of 32kHz to 192KHz.

LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

Booting Linux	Regular Double Flash	Off
Device Mode		
AP mode	Off	On
Client mode	On	Off
Connecting to network	Regular Single Flash	Off

7.4 Audio Status

The module pin AUDIO_STATUS (pin #8) represents the audio output status for AirPlay and UPnP:

Level	**Function**
Low	No audio output
High	Audio output active

The level toggles from Low to High if audio streaming is started, e.g. the “Play” button is pressed on a remote device. The level toggles from High to Low when audio streaming is stopped, e.g. by pressing the “Stop” button.

Note that it might take some seconds before the pin goes from the High to the Low state after a “Stop” button is pressed. For example AirPlay on iOS takes 8 seconds before the streaming session is closed; BubbleUPnP on Android takes 3 seconds.

7.5 Toggle Network Mode

If the module is integrated into a network the key “*KEY_WIFI_MODE*“ can be used to toggle the network mode. A short button press will switch the network state from “*Network-Client mode*” to “*Access-Point mode*” and vice versa.

Note that after a button press event the key is block for 10 seconds to allow the network configuration to be setup properly. Furthermore, the network toggle button press event is ignored in case of a plugged-in Ethernet cable.

LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

8 Remote Control

For easy remote device configuration the HBM10 module runs an unsecured HTTP server as well as an secured HTTPS server. The following table shows the network configuration:

Protocol	IP		Port
	Access Point	Network Client	
HTTP	192.168.2.1	DHCP	8989
HTTPS	192.168.2.1	DHCP	4949

For connecting to the HTTPS server, the client needs to accept the self-signed certificate. The recommended security protocol is TLS v1.2, the recommended cipher is “AES256-GCM-SHA384”. You can test the secured connection with the OpenSSL client tool, e.g:

```
openssl s_client –tlsv1_2 –cipher AES256-GCM-SHA384 –connect 192.168.2.1:4949
```

8.1 Network Configuration

If the module is not already integrated into a network it will boot into *Access-Point mode* after power-up unless, in case of the HBM10-ETH, an Ethernet cable is plugged in. This is also the default mode after a factory reset. The module can be easily integrated into a network with a few HTTP requests as described in ch. 13 *“HTTP API”*.

The modules internal DHCP client will use a distinct hostname for identifying itself in the network. The hostname is based on the model name and the last four digits of the MAC address, e.g. HBM10-9710.

When trying to connect to the networks router the modules *Status LED 2* (pin #22) turns off and the *Status LED 1* (pin #21) is flashing periodically. If the association with the router succeeds and the module is successfully integrated into the network the *Status LED 1* will turn on permanently. If the association with the router fails it will return into *Access-Point mode*. In this case the *Status LED 1* will stop flashing and the *Status LED 2* will turn on permanently (indicating the Access Point mode).

Possible reasons for failing to connect to the network are:

- wrong network key
- DHCP client does not receive an IP address within 60 seconds
- connection to the networks access point is lost

If the module is integrated successfully into a network it is running in *Network-Client mode* and can be accessed by the IP address provided by the networks DHCP.

LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

8.2 HTTP API Documentation

The HTTP API allows you to interact with a HBM10 module connected to a remote control through HTTP requests.

This documentation represents version 1.5 of the HBM10 HTTP API. Accordingly, all API endpoints will be prefixed with "`api/v15`". Additional versions may be introduced in the future, and will be accompanied by a different prefix. To ease maintenance of customer HTTP clients HBM10's HTTP server provides backward compatibility to previous API versions.

The HTTP server is accessed through POST requests to one of the following endpoints:

API endpoint	Description
`configure.action`	Device configuration
`upgrade.action`	Firmware update over the air
`radio.action`	Radio station playback
`leds.action`	LEDs control

Parameters has to be passed into these endpoints through the request body as a JSON object. The Appendix A includes a complete reference for all supported request.

8.3 AirLino® Configurator App

The AirLino® Configurator App is an easy to use configuration tool designed to run on iOS and Android mobile devices. Currently, the app implements the HBM10 HTTP API v1.0 to control the device remotely and can be used to perform all necessary steps to use the HBM10:

- integrate the device into an existing wireless network
- change the device name
- update the firmware over the air
- reset to factory settings
- radio station playback control

Once you have integrated your devices into an existing wireless network you are able to enjoy an unique listening experience by one or more devices at the same time - wirelessly!

LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

How it works:

Connect your mobile device (iOS or Android) under Settings > Wi-Fi to the audio receiver network named “HBM10-xxxx”, where xxxx signifies the last for hexadecimal digits of the MAC address, (e.g. “HBM10-A54C”)

1. Start the “AirLino® Configurator” app and select the device named “HBM10”
2. If desired, you can provide the audio receiver with an individual device name or select one of the default preset options.
3. Select your home network name from the scanned network list to integrate the HBM10 module into your network. After receiving an IP address from the home networks base station the Module is now available on your wireless network.
4. You can now close the app and return with your phone back to your home network.

LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

9 Internet radio

The HBM10 module offers the ability to listen to streaming audio of radio stations worldwide.

9.1 Features

The HBM10 radio player's main features are:

- play HTTP streams encoded with MP3 or Ogg/Vorbis
- group radio stations in playlists and store them on the device
- remote control through the HTTP API[2]

9.2 Playback

To make the HBM10 playback a radio station is very easy. You just need to request a valid URL[3] and optionally a station name with the **play** command to the HBM10 radio player. The **stop** command immediately stops the radio stream.

If the module is powered-down or rebooted while it has been playing a radio station, it will resume the stream automatically the next power-up cycle.

The current playback status can be retrieved with the **query** command.

9.3 Playlists

Radio stations can be group into playlists and are stored directly on the HBM10 module. So you have access to all your playlist even if you change the remote client.

A playlist is stored with the **saveplaylist** command. Every playlist has an unique ID in the range of 0 to 128. You can give each list a short description, e.g. "News" or "Rock". Each playlist can contain up to 128 radio stations.

A playlist can be fetched with the **getplaylist** command. The playlist is specified by its ID. If the playlist is present, the radio player will return the corresponding playlist with its description and a list of stations.

A playlist is removed from the HBM10 with the **rmplaylist** command.

[2] See the Appendix for a full list and description of the provided HTTP API commands.
[3] Note that the player does not support multimedia playlists like M3U or PLS – however the appropriate URL can be extracted from such files.

LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

10 Audio playback control

The HBM10 module can be extended with keys to control the audio playback:

- Volume Up
- Volume Down
- Play/Pause
- Stop
- Next
- Previous

10.1 AirPlay Remote Control

Enabling remote control for AirPlay requires an AirPlay server supporting the Digital Audio Control Protocol (DACP). Once a key event has been triggered on the module, e.g. the “Stop” key was pressed by the user, an appropriate HTTP request is sent from the AirPlay client to the AirPlay server. Depending on the network connectivity it may need several dozen to hundred milliseconds until the server has received and processed the command sent by the client – for example in case of a “Stop” command, the server will end the playback stream.

10.2 UPnP Remote Control

Besides controlling the audio playback with the keys any compliant UPnP media control point, e.g. BubbleUPnP can be used to. Therefor the UPnP media control point has to be connected to the HBM10 who is acting as an UPnP media renderer.

LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

11 Network Tools

One of the most common network problems is insufficient or unreliable bandwidth. Bandwith limitation can cause packet loss, delays, and jitters. In addition, if the required sending and receiving bit rates exceed the bandwith limitations of the network, network congestion will occur and eventually results in a poor audio experience.

11.1 iPerf3

iperf is a commonly used network testing tool to measure the bandwidth of a network.

The HBM10 module runs an iperf3 server on port 5210. Note that the server has to be enabled using the HTTP API as described in section 14.6.

Running an iperf3 client on another device in the same network allows to measure the bandwidth between the two endpoints. The example below shows the results of an iPerf3 test run between a client and the HBM10 module measuring the UDP bandwidth through a IEEE802.11g network.

```
$ iperf3 -c 192.168.1.139 -u -i 1 -t 10 -b 54M
Connecting to host 192.168.1.139, port 5201
[  4] local 192.168.1.167 port 58274 connected to 192.168.1.139 port 5201
[ ID] Interval           Transfer     Bandwidth       Total Datagrams
[  4]   0.00-1.00   sec   728 KBytes  5.96 Mbits/sec  91
[  4]   1.00-2.00   sec   704 KBytes  5.77 Mbits/sec  88
[  4]   2.00-3.00   sec   704 KBytes  5.77 Mbits/sec  88
[  4]   3.00-4.00   sec   664 KBytes  5.44 Mbits/sec  83
[  4]   4.00-5.00   sec   656 KBytes  5.37 Mbits/sec  82
[  4]   5.00-6.00   sec   688 KBytes  5.64 Mbits/sec  86
[  4]   6.00-7.00   sec   608 KBytes  4.98 Mbits/sec  76
[  4]   7.00-8.00   sec   696 KBytes  5.70 Mbits/sec  87
[  4]   8.00-9.00   sec   680 KBytes  5.57 Mbits/sec  85
[  4]   9.00-10.00  sec   704 KBytes  5.77 Mbits/sec  88
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval           Transfer     Bandwidth       Jitter    Lost/Total
Datagrams
[  4]   0.00-10.00  sec  6.67 MBytes  5.60 Mbits/sec  5.044 ms  0/854 (0%)
[  4] Sent 854 datagrams

iperf Done.
```


LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

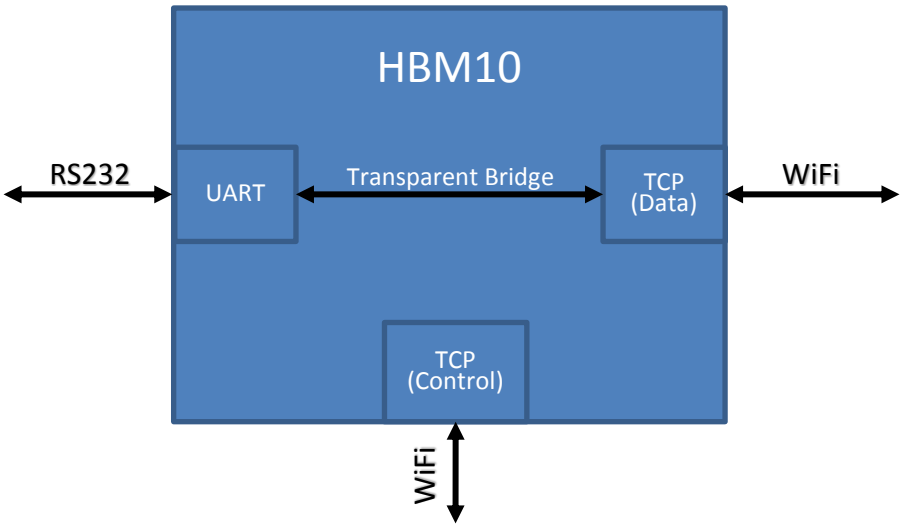
12 Serial to WiFi Bridge

NOTE: This feature is optional and not enabled by default. Please contact the LinTech support team: lintech@lintech.de.

The HBM10 module can be used as a transparent bridge to carry serial (UART) traffic over an 802.11 wireless link. AT commands as described in the AT Command Reference are used to manage the configuration.

12.1 Block Diagram

The HBM10 is running a TCP Data Server listening on port 8990 and a TCP Control Server listening on port 8991.



The Data Server transparently bridges data between the UART interface and the TCP port 8990. The data is sent to the other interface as received – no processing or formatting is done. The default setting for the UART interface is 9600 8N1.

The Control Server listens on TCP port 8991 for incoming AT commands as described in the AT Command Reference.

12.2 Workflow

The UART interface is opened and ready to user after the board powers up. If present, the UART interface uses the serial settings stored by the user otherwise the default settings for initialization.

To use the UART interface as a transparent bridge a TCP client has to establish a connection to the Data Servers port. Any data sent to the UART interface before a TCP connection is established will be lost.

If any configuration is requested, a TCP client has to establish a connection to the Control Servers port. This can be done at any point of time after the board powers up.

LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

13 Customization

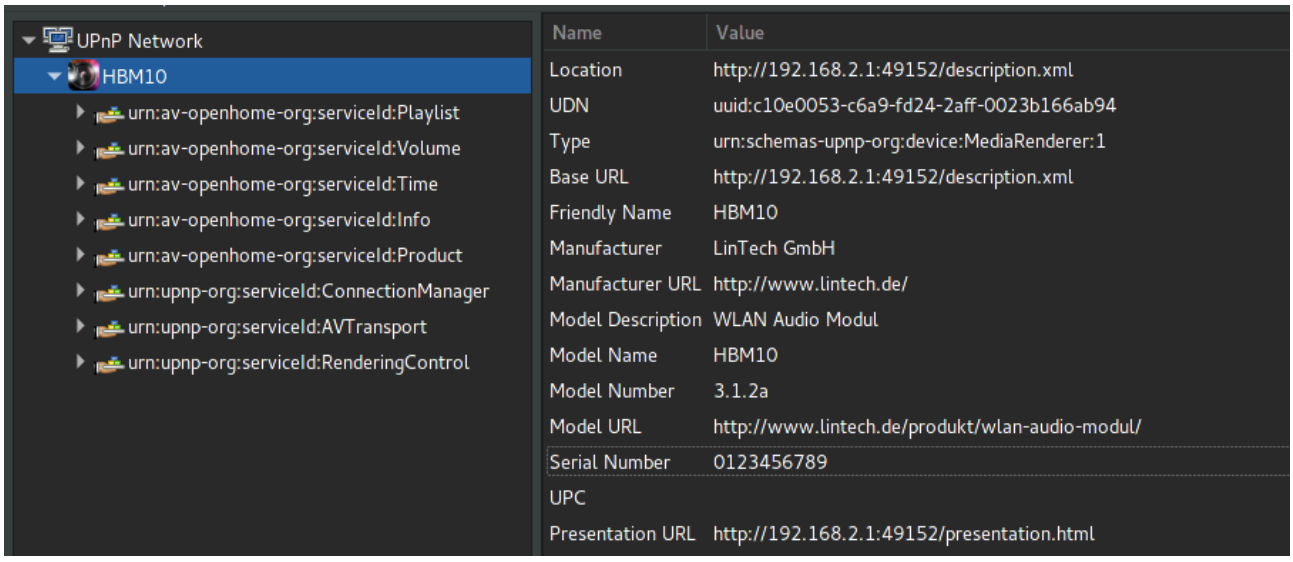
The device can be customized to represent your application. For customizing any of the values please contact the LinTech support team: lintech@lintech.de.

13.1 Device

The device parameters are mainly used for service description, e. g. by the UPnP protocol.

Parameter	Default	Description
Name	HBM10	A friendly name for identifying the device within the network.
Manufacturer		
Name	LinTech GmbH	The manufacturer name for the device.
URL	http://www.lintech.de	The manufacturer URL.
Model		
Name	HBM10	A model name for the device.
Description	WLAN Musikempfänger	A short description of the model.
URL	http://www.lintech.de/produkt/airlino-wlan-airplay-dlna-musikempfaenger/	The model URL.

These values are advertised by the device and can be viewed by most UPnP control points:



LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

13.2 Wi-Fi

Currently, the SSID (Service Set Identifier) is the only customizable Wi-Fi parameter.

Parameter	Default	Description
SSID	HBM10-XXXX	The SSID (Service Set Identifier) name used as Access Point appended with the last four digits of the MAC address, eg. HBM10-A97B.

13.3 Audio

The default status tones used to signal the boot finished and the network connection event can be turned off.

Parameter	Default	Description
Status Tones	on	Turn status tones on/off.

13.4 GPIOs

The HBM10 module offers up to 15 GPIOs. For each pin the behavior can be customized.

Parameter	Default	Description
Direction	Input	Configure the GPIO as input or output.
Value	0	The value to drive for GPIOs configured as output: • 0 = Low • 1 = High
Pull Up Resistor	Disabled	Enables/disables integrated on-chip pull up resistors.
Voltage	3.3V	Select between: • 1.8V • 3.3V
Drive Strength	Low	Select between: • Low • Medium • High
IRQ Enable	Disabled	Enable/Disable interrupts

LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

Level/Edge Sensivity	Edge detection	Select between: • Edge detection • Level detection
Polarity	Low or falling edge	Select between: • Low or falling edge • High or rising edge

LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

14 HTTP API

The current HTTP API version is 1.5. The API is valid for both the HTTP and the HTTPS server.

The HTTP API allows you to interact with an HBM10 module through HTTP requests. Accordingly, all API endpoints will be prefixed with "`api/v15`". Additional versions may be introduced in the future, and will be accompanied by a different prefix.

The HTTP server is accessed through HTTP requests to one of the following endpoints:

API Endpoint	Description
configure.action	Device configuration
upgrade.action	Firmware Update over the Air
radio.action	Internet radio control
leds.action	LEDs control
sound.action	Sound control
iperf.action	iPerf3 control

Parameters has to be passed into these endpoints through the request body as a JSON object.

HTTP Request Rules

For proper operation the following requirements has to be met:

- All HTTP request are `'POST'` requests.
- The `'Content-Type'` is `'application/json'`.
- The `'charset'` is UTF-8.

The syntax for the body parameters description follows the rules for the Extended Backus–Naur Form (EBNF):

- Words inside double quotes (" ... ") represent terminal strings.
- Square brackets ([...]) surround optional items.
- Curly brackets ({ ... }) surround items that can repeat zero or more times.
- A vertical line (|) seperates alternatives.
- `String = ? all visible ASCII characters ?`

Sample Requests using HTTPS

The sample requests use the network tool "curl" to provide examples to connect to the HBM10 with the HTTP protocol. To connect the module using HTTPS curl has to be called with the parameter "**-k**" and the URL should be **https**://<IP>:**4949**/..., e.g.

```
curl -k -H 'Content-Type: application/json; charset=UTF-8'
     -d '{"action": "query"}'
     -X POST https://1921.168.2.1:4949/api/v15/configure.action
```


LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

14.1 Configure

14.1.1 Get Device Status

Description

Get the current device and network configuration.

Method

POST

URL

http://<IP>:8989/api/v15/configure.action

Request Parameters

Parameter	Description	Value
action	Action type	`"query"`

Response Parameters

Parameter	Description	Value
networkinfo	*Wireless network information*	
└ ***apinfo***	└ *AP mode information*	`Wifi object`
└ ***clientinfo***	└ *Client mode information*	`Wifi object`
└ **wifimode**	└ Current wifi mode	`"ap" | "client"`
└ ***ipaddressinfo***	└ *IP information*	`IPAddress object`
ethinfo	*Ethernet network information*	
└ ***ipaddressinfo***	└ *IP information*	`IPAddress object`
deviceinfo	*Device information*	`DeviceInfo object`

JSON Objects

Wifi

Parameter	Description	Value
wifissid	Service Set Identifier	`String`
wifipwd	Authentification password	`String`
encrypt_type	Encryption type **Parameter** / **Description** NONE / No encryption	`"NONE" | "WEP" | "WPA"`

LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

	WEP: WEP encrypted WPA: WPA2 or WPA encrypted	
encrypt_subtype	Encryption subtype **Parameter** / **Description** WPA2: WPA2 only WPA: WPA only WPA2 WPA: WPA2 and WPA	`"WPA2" | "WPA" | "WPA2 WPA"`
group_cipher	Group cipher	`String`
pairwise_ciphers	Pairwise cipher	`String`

IPAddress

Parameter	Description	Value
type	IP address assignment method	`"DHCP"`
subnetmask	Subnet mask	`Reserved`
primarydns	Primary DNS	`Reserved`
seconddns	Second DNS	`Reserved`
gateway	Gateway	`Reserved`
ipaddress	IP address	`Reserved`

DeviceInfo

Parameter	Description	Value
model	Model name	`String`
devicename	Friendly name of the device	`String`
softwarever	Firmware version	`String`
macaddress	Wifi interface's MAC address	`String`
configured	Wifi interface is integrated into a network	`"true" | "false"`
serialnumber	Serial number	`Reserved`
hardwarever	Hardware revision	`String`
airplaypwd	AirPlay password	`Reserved`
devicepwd	Device password	`Reserved`

Sample Request

```
curl -H 'Content-Type: application/json; charset=UTF-8'                    \
     -d '{"action": "query"}'                                              \
     -X POST http://192.168.2.1:8989/api/v15/configure.action
```


LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

Sample Response

AP mode

```
{
    "networkinfo": {
        "apinfo": {
            "wifissid": "HBM10-AB94",               // SSID
            "wifipwd": "",
            "encrypt_type": "",
            "encrypt_subtype": "",
            "group_cipher": "",
            "pairwise_ciphers": ""
        },
        "clientinfo": {
            "encrypt_type": "",
            "encrypt_subtype": "",
            "wifipwd": "",
            "group_cipher": "",
            "wifissid": "",
            "pairwise_ciphers": ""
        },
        "wifimode": "ap",                           // AP mode
        "ipaddressinfo": {
            "type": "",
            "subnetmask": "",
            "secondarydns": "",
            "gateway": "",
            "primarydns": "",
            "ipaddress": ""
        }
    },
    "ethinfo": {
        "ipaddressinfo": {
            "type": "DHCP",                         // DHCP (Dynamic IP)
            "subnetmask": "",
            "seconddns": "",
            "gateway": "",
            "primarydns": "",
            "ipaddress": "192.168.178.101"          // IPv4 address
        }
    },
    "deviceinfo": {
        "airplaypwd": "",
        "serialnumber": "2976295828",               // Serial number
        "hardwarever": "R8EM49490B1",               // Hardware version
        "macaddress": "00:23:b1:66:ab:94",          // MAC address
        "model": "HBM10",                           // Model name
        "devicepwd": "",
        "configured": "false",                      // Device is not configured
        "devicename": "HBM10",                      // Device name
        "softwarever": "4.2.0"                      // Software version
    }
}
```


LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

Client mode

```
{
    "networkinfo": {
        "apinfo": {
            "encrypt_type": "",
            "encrypt_subtype": "",
            "wifipwd": "",
            "group_cipher": "",
            "wifissid": "",
            "pairwise_ciphers": ""
        },
        "clientinfo": {
            "wifissid": "My Home",                  // SSID
            "wifipwd": "",
            "encrypt_type": "WPA2-PSK",             // Encryption
            "encrypt_subtype": "",
            "group_cipher": "TKIP",                 // Group cipher
            "pairwise_ciphers": "CCMP"              // Pairwise cipher
        },
        "wifimode": "client",
        "ipaddressinfo": {
            "type": "DHCP",                         // DHCP (Dynamic IP)
            "subnetmask": "",
            "seconddns": "",
            "gateway": "",
            "primarydns": "",
            "ipaddress": ""
        }
    },
    "ethinfo": {
        "ipaddressinfo": {
            "type": "DHCP",                         // DHCP (Dynamic IP)
            "subnetmask": "",
            "seconddns": "",
            "gateway": "",
            "primarydns": "",
            "ipaddress": "192.168.178.101"          // IPv4 address
        }
    },
    "deviceinfo": {
        "airplaypwd": "",
        "serialnumber": "2976295828",               // Serial number
        "hardwarever": "R8EM49490B1",               // Hardware version
        "macaddress": "00:23:b1:66:ab:94",          // MAC address
        "model": "HBM10",                           // Model name
        "devicepwd": "",
        "configured": "true",                       // Configured
        "devicename": "HBM10",                      // Device name
        "softwarever": "4.2.0"                      // Software version
    }
}
```


LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

14.1.2 Wifi Scan

Description

Get a list of scanned networks.

Method

POST

URL

http://<IP>:8989/api/v15/configure.action

Request Parameters

Parameter	Description	Value
action	Action type	`"wifiscan"`

Response Parameters

Parameter	Description	Value
aplist	*List of scanned networks*	
└ **[network]**	└ List of network objects	`List of Network objects`

JSON Objects

Network

Parameter	Description	Value
bss	Basic Service Set Identification	`String`
ssid	Service Set Identifier	`String`
channel	Wireless channel	`String`
signal_level	Signal level	`String`
encrypt_type	Encryption type **Parameter** / **Description** NONE — No encryption WEP — WEP encrypted WPA — WPA2 or WPA encrypted	`"NONE" | "WEP" | "WPA"`
encrypt_subtype	Encryption subtype **Parameter** / **Description** WPA2 — WPA2 only WPA — WPA only WPA2 WPA — WPA2 and WPA	`"WPA2" | "WPA" | "WPA2 WPA"`

LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

group_cipher	Group cipher	String
pairwise_ciphers	Pairwise cipher	String

Sample Request

```
curl –H 'Content-Type: application/json; charset=UTF-8'                    \
     –d '{"action": "wifiscan"}'                                           \
     –X POST http://192.168.2.1:8989/api/v15/configure.action
```

Sample Response

```
{
    "aplist": [
        {
            "encrypt_type": "NONE",                 // Open network
            "encrypt_subtype": "",
            "ssid": "Some SSID 1",                  // SSID
            "group_cipher": "",
            "pairwise_ciphers": "",
            "bss": "00:12:23:34:45:56",             // MAC address
            "signal_level": "-46",                  // Signal level in dBm
            "channel": "6"                          // Channel
        },
        {
            "encrypt_type": "WPA",                  // Secured network
            "encrypt_subtype": "WPA2 WPA",          // WPA+WPA2
            "ssid": "Some SSID 2",                  // SSID
            "group_cipher": "TKIP",                 // Group cipher
            "pairwise_ciphers": "CCMP TKIP",        // Pairwise cipher
            "bss": "00:14:6c:53:4f:52",             // MAC address
            "signal_level": "-24",                  // Signal level in dBm
            "channel": "6"                          // Channel
        }
    ]
}
```


LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

14.1.3 Set Config

Description

Configure the device

Method

POST

URL

http://<IP>:8989/api/v15/configure.action

Request Parameters

Parameter	Description	Value
action	Action type	`"setconfig"`
deviceinfo	*Device information*	
└ **devicename**	└ Device name	`String`
networkinfo	*Network information*	
└ **wifimode**	└ Wifi mode	`"client"`
└ ***clientinfo***	└ *Network client configuration*	
└ **encrypt_type**	└ Encryption type **Parameter** / **Description** NONE — No encryption WEP — WEP encrypted WPA — WPA2 or WPA encrypted	`"NONE" | "WEP" | "WPA"`
└ **wifipwd**	└ Wifi password	`String`
└ **wifissid**	└ SSID name	`String`
└ ***ipaddressinfo***	*IP address information*	
└ **type**	└ IP address type	`"DHCP"`

Sample Request

```
curl -H 'Content-Type: application/json; charset=UTF-8'                    \
     -d '{"action": "setconfig",                                           \
          "deviceinfo": {"devicename": "Kitchen"},                         \
          "networkinfo": {"wifimode": "client",                            \
                          "clientinfo": {"encrypt_type": "WPA",            \
                                         "wifipwd": "12345678",            \
                                         "wifissid": "MyHomeNetwork"},     \
                          "ipaddressinfo": {"type": "DHCP"}}}'             \
     -X POST http://192.168.2.1:8989/api/v15/configure.action
```


LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

Sample Response

```
{ "returncode": "success" }
```


LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

14.1.4 Factory Reset

Description

Reset the device to factory settings.

Method

POST

URL

http://<IP>:8989/api/v13/configure.action

Request Parameters

Parameter	Description	Value
action	Action type	`"resetdefault"`

Sample Request

```
curl -H 'Content-Type: application/json; charset=UTF-8'                    \
     -d '{"action": "resetdefault"}'                                       \
     -X POST http://192.168.2.1:8989/api/v15/configure.action
```

Sample Response

```
{ "returncode": "success" }
```


LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

14.1.5 Reboot

Description

Reboot the device

Method

POST

URL

http://<IP>:8989/api/v15/configure.action

Request Parameters

Parameter	Description	Value
action	Action type	`"reboot"`

Sample Request

```
curl -H 'Content-Type: application/json; charset=UTF-8'                    \
     -d '{"action": "reboot"}'                                             \
     -X POST http://192.168.2.1:8989/api/v15/configure.action
```

Sample Response

```
{ "returncode": "success" }
```


LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

14.2 OTA Upgrade

14.2.1 Firmware Update

Description

Start a firmware update by setting an URL from where the device will fetch the firmware. This only works in network client mode.

Method

POST

URL

http://<IP>:8989/api/v15/otaupgrade.action

Request Parameters

Parameter	Description	Value
action	Action type	`"seturl"`
otaaddress	URL where to fetch the update file from	`String`
filesize	File size of the update file	`String`

Sample Request

```
curl -H 'Content-Type: application/json; charset=UTF-8'                    \
     -d '{"action": "seturl"                                               \
          "otaaddress": "http://dl.lintech.de/download/upgrade/HBM10/3.3.6" \
          "filesize": "25438208"}'                                         \
     -X POST http://192.168.1.159:8989/api/v15/otaupgrade.action
```

Sample Response

```
{ "returncode": "success" }                  // Update starts
```


LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

14.2.2 Firmware Update Status

Description

Requests the firmware update status. This only works in network client mode.

NOTE: Querying the firmware update status should not be done in periods less then 1 second.

Method

POST

URL

http://<IP>:8989/api/v15/otaupgrade.action

Request Parameters

Parameter	Description	Value
action	Action type	`"querystatus"`

Sample Request

```
curl -H 'Content-Type: application/json; charset=UTF-8'                    \
     -d '{"action": "querystatus"}'                                        \
     -X POST http://192.168.1.159:8989/api/v15/otaupgrade.action
```

Response Parameters

Parameter	Description
status	Status code: -1: update not running 1: downloading 2: download finished 3: updating 4: checksum verification 5: failure occurred
downfilename	File name currently downloading
downprogress	Download progress in percent
errinfo	Error code: -1: update not running 1: file not exist 2: download error 3: checksum error 4: flash error 5: update error

LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

Sample Response

```
{
    "status": "1",                          // Status code
    "downfilename": "airlino.swu",          // Current downloading file name
    "downprogress": "44",                   // Download process in percent
    "errinfo": "-1"                         // Error code
}
```


LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

14.3 Internet Radio

14.3.1 Radio Play

Description

Starts to play a radio HTTP stream (MP3 or Ogg/Vorbis).

Method

POST

URL

http://<IP>:8989/api/v15/radio.action

Request Parameters

Parameter	Description	Value
action	Action type	`"play"`
station	*Radio station*	
└ **name**	└ Station name	`String`
└ **url**	└ Station URL	`String`

Response Parameters

Parameter	Description	Value
returncode	Response message **Parameter** / **Description** success: Plays radio stream error: Radio stream cannot be played, e.g. URL is not a valid	`"success" | "error"`

Sample Request

```
curl -H 'Content-Type: application/json; charset=UTF-8'                    \
     -d '{"action": "play",                                                \
          "station": {                                                     \
              "name": "Inforadio",                                         \
              "url": "http://inforadio.de/livemp3"                         \
          } }'                                                             \
     -X POST http://192.168.1.159:8989/api/v15/radio.action
```

Sample Response

```
{ "returncode": "success" }                    // Plays the stream
```


LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

14.3.2 Radio Stop

Description

Stops to play a radio HTTP stream.

Method

POST

URL

http://<IP>:8989/api/v15/radio.action

Request Parameters

Parameter	Description	Value
action	Action type	`"stop"`

Sample Request

```
curl -H 'Content-Type: application/json; charset=UTF-8'                    \
     -d '{ "action": "stop" }'                                             \
     -X POST http://192.168.1.159:8989/api/v15/radio.action
```


LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

14.3.3 Radio Query

Description

Query the current station.

If metadata is available for the current radio station it is appended to the station information. Note that the field *station.name* contains the string set through the HTTP-API, whereas the field *station.meta.name* contains a string fetched from the metadata.

Method

POST

URL

http://<IP>:8989/api/v15/radio.action

Request Parameters

Parameter	Description	Value
action	Action type	`"query"`

Response Parameters

Parameter	Description	Value
state	Radio playback state **Parameter** / **Description** 1 – Not playing 2 – Playing 3 – Playing, but paused	`Integer`
listid	Playlist identification number	`Integer`
station	*Radio station*	`Object`
└ **name**	└ Station name	`String`
└ **url**	└ Station URL	`String`
└ ***meta***	└ Station metadata	`Object`
└ **now_playing**	└ Now playing	`String`
└ **name**	└ Name	`String`

`listid` is only set if a playlist is played.

`meta` is only set if metadata are available for the current radio station.

LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

Sample Request

```
curl –H 'Content-Type: application/json; charset=UTF-8'                    \
     –d '{ "action": "query" }'                                            \
     –X POST http://192.168.1.159:8989/api/v15/radio.action
```

Sample Response

```
{  "state": 2,
   "station": {
       "url": "http://stream.berliner-rundfunk.de/brf/mp3-128/internetradio",
       "name": "Berliner Rundfunk Livestream",
       "meta": {
           "name": "Berliner Rundfunk Livestream",
           "now_playing": "KARAT - DER BLAUE PLANET"
       }
   }
}
```


LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

14.3.4 Radio Get Playlist

Description

Fetch a playlist from the device's internal memory.

Method

POST

URL

http://<IP>:8989/api/v15/radio.action

Request Parameters

Parameter	Description	Value
action	Action type	`"getplaylist"`
listid	Playlist identification number	`Integer`

Response Parameters

Parameter	Description	Value
description	Short description	`String`
stationlist	*List of stations*	
└ **[station]**	└ List of radio station objects	`List`

Sample Request

```
curl -H 'Content-Type: application/json; charset=UTF-8'                    \
     -d '{ "action": "getplaylist", "listid": 1 }'                         \
     -X POST http://192.168.1.159:8989/api/v15/radio.action
```

Sample Response

Playlist does not exists or is empty:

```
{ }   // empty JSON object
```

Playlist exists:

```
{
   "description": "News",
   "stationlist": [
       { "name": "Inforadio",
         "url": "http://inforadio.de/livemp3" },
       { "name": "Deutschlandfunk",
         "url": "http://dradio-ogg-dwissen-l.akacast.akamaistream.net/7/192/
135496/v1/gnl.akacast.akamaistream.net/dradio_ogg_dwissen_l" }
   ]
}
```


LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

14.3.5 Radio Save Playlist

Description

Store a radio station playlist to the device's internal memory. A playlist with the same ID will be overwritten.

Method

POST

URL

http://<IP>:8989/api/v15/radio.action

Request Parameters

Parameter	Description	Value
action	Action type	`"saveplaylist"`
listid	List identification number	`Integer`
playlist	*Playlist*	
└ **description**	└ Short description	`String`
└ ***stationlist***	└ *List of stations*	
└ **[station]**	└ List of radio station objects	`List`

Response Parameters

Parameter	Description	Value
returncode	Response message **Parameter** / **Description** success: playlist is stored in memory error: playlist was **not** stored, e.g. due to a JSON parser error	`"success" | "error"`

Sample Request

```
curl -H 'Content-Type: application/json; charset=UTF-8'                    \
     -d '{"action": "saveplaylist",                                        \
          "listid": 1,                                                     \
          "playlist": {                                                    \
              "description": "News",                                       \
              "stationlist": [                                             \
                  { "name": "Inforadio",                                   \
                    "url": "http://inforadio.de/livemp3" },                \
                  { "name": "Deutschlandfunk",                             \
                    "url": "http://dradio-ogg-dwissen-l.akacast.akamaistream.net
/7/192/135496/v1/gnl.akacast.akamaistream.net/dradio_ogg_dwissen_l" }       \
          ]}}'                                                             \
```


LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

```
     -X POST http://192.168.1.159:8989/api/v15/radio.action
```

Sample Response

```
{ "returncode": "success" }
```


LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

14.3.6 Radio Remove Playlist

Description

Deletes a specific playlist from the internal memory.

Method

POST

URL

http://<IP>:8989/api/v15/radio.action

Request Parameters

Parameter	Description	Value
action	Action type	`"rmplaylist"`
listid	List identification number	`Integer`

Response Parameters

Parameter	Description	Value
returncode	Response message **Parameter** / **Description** success – playlist is removed error – playlist cannot be deleted, e.g. the list does not exist	`"success" | "error"`

Sample Request

```
curl -H 'Content-Type: application/json; charset=UTF-8'                    \
     -d '{ "action": "rmplaylist", "listid": 1 }'                          \
     -X POST http://192.168.1.159:8989/api/v15/radio.action
```

Sample Response

```
{ "returncode": "success" }
```


LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

14.3.7 Radio Play Playlist

Description

Request to play a specific playlist from the internal memory. Playback will start with the first entry in the list. It is possible to step forward in the list with KEY_NEXT and step back in the list with KEY_PREV. The playlist will be played in repeat mode: if KEY_NEXT is pressed while the last entry of the list is currently played, the first entry in the list will be played next and if KEY_PREV is pressed while the first entry of the list is currently played, the last entry in the list will be played.

Method

POST

URL

http://<IP>:8989/api/v15/radio.action

Request Parameters

Parameter	Description	Value
action	Action type	`"playplaylist"`
listid	List identification number	`Integer`

Response Parameters

Parameter	Description	Value
returncode	Response message **Parameter** / **Description** success – Playback is about to begin error – playlist cannot be played, e.g. the list contains invalid an URL	`"success" | "error"`

Sample Request

```
curl -H 'Content-Type: application/json; charset=UTF-8'                    \
     -d '{ "action": "playplaylist", "listid": 1 }'                        \
     -X POST http://192.168.1.159:8989/api/v15/radio.action
```

Sample Response

```
{ "returncode": "success" }
```


LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

14.3.8 Get Favourite Station

Description

Get the current favourite radio station.

Method

POST

URL

http://<IP>:8989/api/v15/radio.action

Request Parameters

Parameter	Description	Value
action	Action type	`"getfavouritestation"`
station	*Radio station*	
└ **name**	└ Station name	`String`
└ **url**	└ Station URL	`String`

Response Parameters

Parameter	Description	Value
returncode	Response message **Parameter** / **Description** success – Station was set as favourite error – Station could not set as favourite	`"success" | "error"`

Sample Request

```
curl -H 'Content-Type: application/json; charset=UTF-8'                    \
     -d '{"action": "getfavouritestation",                                 \
          "station": {                                                     \
              "name": "Inforadio",                                         \
              "url": "http://inforadio.de/livemp3"                         \
          } }'                                                             \
     -X POST http://192.168.1.159:8989/api/v15/radio.action
```

Sample Response

```
{ "name": "Inforadio", "url": "http://inforadio.de/livemp3" }
```


LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

14.3.9 Set Favourite Station

Description

Set a radio station as the favourite station. This station will always be played after the module is ready after power up and is connected to the network.

To remove the favourite station issue a request with no station object set.

Method

POST

URL

http://<IP>:8989/api/v15/radio.action

Request Parameters

Parameter	Description	Value
action	Action type	`"setfavouritestation"`
station	*Radio station*	
└ **name**	└ Station name	`String`
└ **url**	└ Station URL	`String`

If no station is set the current favourite station is removed.

Response Parameters

Parameter	Description	Value
returncode	Response message **Parameter** / **Description** success – Station was set as favourite error – Station could not set as favourite	`"success" | "error"`

Sample Request

```
curl -H 'Content-Type: application/json; charset=UTF-8'                    \
     -d '{"action": "setfavouritestation",                                 \
          "station": {                                                     \
              "name": "Inforadio",                                         \
              "url": "http://inforadio.de/livemp3"                         \
          } }'                                                             \
     -X POST http://192.168.1.159:8989/api/v15/radio.action
```

Sample Response

```
{ "returncode": "success" }
```


LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

14.4 LEDs Control

14.4.1 Get LEDs configuration

Description

Get the current LEDs configuration.

Method

POST

URL

http://<IP>:8989/api/v15/leds.action

Request Parameters

Parameter	Description	Value
action	Action type	`"get"`

Response Parameters

Parameter	Description	Value
brightness	LEDs brightness level with: 1 <= *brightness* <= 255	`Integer`

Sample Request

```
curl -H 'Content-Type: application/json; charset=UTF-8'                    \
     -d '{ "action": "get" }'                                              \
     -X POST http://192.168.1.159:8989/api/v15/leds.action
```

Sample Response

```
{ "brightness": 30 }
```


LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

14.4.2 Set LEDs configuration

Description

Set the current LEDs configuration.

Method

POST

URL

http://<IP>:8989/api/v15/leds.action

Request Parameters

Parameter	Description	Value
action	Action type	`"set"`
brightness	LEDs brightness level with: 1 <= *brightness* <= 255	`Integer`

Response Parameters

Parameter	Description	Value
returncode	Response message **Parameter** / **Description** success / Success error / An error occured	`"success" | "error"`

Sample Request

```
curl -H 'Content-Type: application/json; charset=UTF-8'                    \
     -d '{ "action": "set", "brightness": 30 }'                            \
     -X POST http://192.168.1.159:8989/api/v15/leds.action
```

Sample Response

```
{ "returncode": "success" }
```


LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

14.5 Sound Control

14.5.1 Get Master Volume

Description

Get the current Master volume level.

Method

POST

URL

http://<IP>:8989/api/v15/sound.action

Request Parameters

Parameter	Description	Value
action	Action type	`"getmastervol"`

Response Parameters

Parameter	Description	Value
volume	Master volume level with: 0 <= *volume* <= 255	`Integer`

Sample Request

```
curl -H 'Content-Type: application/json; charset=UTF-8'                    \
     -d '{ "action": "getmastervol" }'                                     \
     -X POST http://192.168.1.159:8989/api/v15/sound.action
```

Sample Response

```
{ "volume": 255 }
```


LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

14.5.2 Set Master Volume

Description

Set the Master volume to a new value.

Note that the volume is stored periodically. To make sure a volume change withstands an abrupt power-cut a delay of 10 seconds is necessary.

Method

POST

URL

http://<IP>:8989/api/v15/sound.action

Request Parameters

Parameter	Description	Value
action	Action type	`"setmastervol"`
volume	Master volume level with: 0 <= *volume* <= 255	`Integer`

Response Parameters

Parameter	Description	Value
returncode	Response message **Parameter** / **Description** success / Success error / An error occured	`"success" | "error"`

Sample Request

```
curl -H 'Content-Type: application/json; charset=UTF-8'                    \
     -d '{ "action": "setmastervol", "volume": 255 }'                      \
     -X POST http://192.168.1.159:8989/api/v15/sound.action
```

Sample Response

```
{ "returncode": "success" }
```


LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

14.5.3 Get Status Tones Volume

Description

Get the current volume level of the status tones.

Note that the volume is stored periodically. To make sure a volume change withstands an abrupt power-cut a delay of 10 seconds is necessary.

Method

POST

URL

http://<IP>:8989/api/v15/sound.action

Request Parameters

Parameter	Description	Value
action	Action type	`"getstatusvol"`

Response Parameters

Parameter	Description	Value
volume	Master volume level with: 0 <= *volume* <= 255	`Integer`

Sample Request

```
curl -H 'Content-Type: application/json; charset=UTF-8'                    \
     -d '{ "action": "getstatusvol" }'                                     \
     -X POST http://192.168.1.159:8989/api/v15/sound.action
```

Sample Response

```
{ "volume": 0 }
```


LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

14.5.4 Set Status Tones Volume

Description

Set the volume for the status tones to a new value.

Method

POST

URL

http://<IP>:8989/api/v15/sound.action

Request Parameters

Parameter	Description	Value
action	Action type	`"setstatusvol"`
volume	Status tones volume level with: 0 <= *volume* <= 255	`Integer`

Response Parameters

Parameter	Description	Value
returncode	Response message **Parameter** / **Description** success / Success error / An error occured	`"success" | "error"`

Sample Request

```
curl -H 'Content-Type: application/json; charset=UTF-8'                    \
     -d '{ "action": "setstatusvol", "volume": 0 }'                        \
     -X POST http://192.168.1.159:8989/api/v15/sound.action
```

Sample Response

```
{ "returncode": "success" }
```


LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

14.6 iPerf Control

14.6.1 Enable iPerf3 Server

Description

Start a iPerf3 server daemon. If enabled the daemon will be started on bootup.

Method

POST

URL

http://<IP>:8989/api/v15/radio.action

Request Parameters

Parameter	Description	Value
action	Action type	`"enable"`

Response Parameters

Parameter	Description	Value
returncode	Response message **Parameter** / **Description** success / Success error / An error occured	`"success" | "error"`

Sample Request

```
curl -H 'Content-Type: application/json; charset=UTF-8'                    \
     -d '{ "action": "enable" }'                                           \
     -X POST http://192.168.1.159:8989/api/v15/iperf.action
```

Sample Response

```
{ "returncode": "success" }
```


LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

14.6.2 Disable iPerf3 Server

Description

Stop the iPerf3 server daemon. If disabled the daemon will not be started on bootup.

Method

POST

URL

http://<IP>:8989/api/v15/radio.action

Request Parameters

Parameter	Description	Value
action	Action type	`"disable"`

Response Parameters

Parameter	Description	Value
returncode	Response message **Parameter** / **Description** success / Success error / An error occured	`"success" | "error"`

Sample Request

```
curl -H 'Content-Type: application/json; charset=UTF-8'                    \
     -d '{ "action": "disable" }'                                          \
     -X POST http://192.168.1.159:8989/api/v15/iperf.action
```

Sample Response

```
{ "returncode": "success" }
```


LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

14.6.3 Get Status Of iPerf3 Server

Description

Get the status if the iPerf3 server daemon is currently running.

Method

POST

URL

http://<IP>:8989/api/v15/radio.action

Request Parameters

Parameter	Description	Value
action	Action type	`"status"`

Response Parameters

Parameter	Description	Value
enabled	Server status **Parameter** / **Description** true – Server is running false – Server is not running	`Boolean`

Sample Request

```
curl -H 'Content-Type: application/json; charset=UTF-8'                    \
     -d '{ "action": "status" }'                                           \
     -X POST http://192.168.1.159:8989/api/v15/iperf.action
```

Sample Response

```
{ "enabled": true }
```


LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

15 AT Commands Reference

NOTE: This feature is optional and not enabled by default. Please contact the LinTech support team: lintech@lintech.de.

15.1 AT Command Syntax

The “AT” or “at” prefix must be set at the beginning of each command line. To terminate a command line enter <CR>. Throughout this document, only the command lines are presented, <CR> is omitted intentionally.

Commands are usually followed by a response – <CR><LF>*response*<CR><LF>. Throughout this document, only the responses are presented, <CR><LF> are omitted intentionally.

The AT commands are case-insensitive and may be entered in either uppercase or lowercase letters and even can be mixed. Therefore, the following command lines are equivalent:

```
AT+UART?
at+uart?
At+Uart?
```

15.2 Result Codes

Result codes are messages sent from the Control Server to provide information about the execution of an AT command and the occurrence of an event. Two types of result codes are used:

- Final result codes
- Unsolicited result codes

A final result code marks the end of an AT command response. It is an indication that the Control Server has finished the execution of a command line. Two frequently used final result codes are **OK** and **ERROR**. Only one final result code will be returned for each command line.

The OK Final Result Code

The OK final result code indicates that a command line has been executed successfully by the Control Server. It always starts and ends with <CR><LF>.

The ERROR Final Result Code

The ERROR final result code indicates that an error occurs when the Control Server tries to execute a command line. After the occurrence of an error the Control Server will not process any remaining AT command. Like the OK final result code, the ERROR final result code always starts and ends with <CR><LF>. For common errors an error code follows the string “ERROR”, separated by a <SPACE> character, e.g.

```
ERROR 1
```


LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

The following error codes are supported:

Error Code	Description
1	*Unknown command.* The command is not supported or contains a typo.
2	*Syntax error.* The command syntax is wrong, e.g. not all necessary parameters are set.
3	*Invalid range.* One or more parameters are out of range.

Unsolicited Result Codes

Unsolicited result codes are currently not used, but may be introduced with a new AT command.

LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

15.3 Standard AT Commands

<table>
<tr><th>Command</th><th>Description</th></tr>
<tr><td>AT</td><td>Test command.
Response with OK when the control server is running.</td></tr>
<tr><td>A/</td><td>Repeat the last AT command.</td></tr>
<tr><td>ATE[<echo>]</td><td>Echo command.
Parameters:
<table>
<tr><th>Parameter</th><th>Type</th><th>Description</th></tr>
<tr><td>echo</td><td>Enum</td><td>0: Incoming characters will not be echoed.
1: Incoming characters will be echoed.</td></tr>
</table>
If <echo> is omitted, it defaults to 0.</td></tr>
<tr><td>AT&F</td><td>Factory defined configuration.
All configuration settings impacted by the AT&W command are reset to their default value.</td></tr>
<tr><td>AT&W</td><td>Stores the current configuration settings in non-volatile memory.
Parameters impacted by AT&W command:
<table>
<tr><th>Command</th><th>Parameter</th><th>Default</th></tr>
<tr><td>ATE</td><td><echo></td><td>0</td></tr>
<tr><td>UART</td><td><baud>
<data>
<parity>
<stop></td><td>9600
8
N
1</td></tr>
</table>
</td></tr>
</table>

LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de

15.4 Serial AT Commands

<table>
<tr><th>Command</th><th>Description</th></tr>
<tr><td>AT+UART=
<baud>,<data>,
<parity>,<stop></td><td>
Apply new UART settings.
Usage:
<table>
<tr><th>Parameter</th><th>Type</th><th>Description</th></tr>
<tr><td>baud</td><td>Integer</td><td>Supported baud rates:
<table>
<tr><td>300</td><td>28800</td></tr>
<tr><td>1200</td><td>38400</td></tr>
<tr><td>2400</td><td>57600</td></tr>
<tr><td>4800</td><td>115200</td></tr>
<tr><td>9600</td><td>230400</td></tr>
<tr><td>14400</td><td>460800</td></tr>
<tr><td>19200</td><td>921600</td></tr>
</table>
Note: Other baud rates may work too, but are not supported.</td></tr>
<tr><td>data</td><td>Integer</td><td>Supported data bits:
5, 6, 7 or 8</td></tr>
<tr><td>parity</td><td>Char</td><td>Supported parities:
N: No parity
O: Odd parity
E: Even parity</td></tr>
<tr><td>stop</td><td>Integer</td><td>Supported stop bits:
1 or 2</td></tr>
</table>
Example:
<pre>AT+UART=9600,8,N,1</pre>
</td></tr>
<tr><td>AT+UART?</td><td>
Read current UART settings.
Response:
<pre>+UART=<baud>,<data>,<parity>,<stop></pre>
Example:
<pre>+UART=9600,8,N,1</pre>
</td></tr>
</table>

LinTech GmbH Kommunikationstechnologien, Friedrich-Engels-Straße 35, D -13156 Berlin, Phone +49 (030) 54 94 72 60,
Fax +49 (030) 54 94 72 44, E-Mail LinTech@LinTech.de, http://www.LinTech.de