INSTREAMER

Instreamer

Network audio encoder for commercial, industrial and security applications

Technical Documentation

Firmware V3.05
Released   4th March 2008
Supports:

- **INSTREAMER (legacy)**
- **INSTREAMER 100**
- **IP Audio Module (IPAM 100)**

BAR IX

Table of Contents

1 Introduction

1.1 About the "Instreamer" firmware

The "Instreamer" firmware is designed to serve as a versatile, network-enabled analog and digital audio-to-Ethernet converter for commercial audio distribution.

The "Instreamer" firmware converts audio from any analog or digital device into G.711 (8 bit), PCM (16 bit) or high-quality MP3 streams. The audio is encoded in real-time, and the generated audio stream can be distributed, via an IP-based network or the Internet, to one or more receivers or Shoutcast / Icecast-servers.

Barix devices running the "Instreamer" firmware can be easily managed via a web browser interface using PCs, web pads, PDAs or other web-enabled devices. SNMP remote monitoring capabilities allow for building a manageable distributed audio network. With serial and Ethernet control APIs, open IP-standards, and the standard encoding formats, the device can also be integrated with other components, controlled by automation systems, or used with Barix Exstreamers to create more flexible, more cost-effective distributed audio systems.

The built-in serial interface allows data to be relayed to another Barix device, a PC or a server using the "Serial Gateway" functionality of the "Instreamer" firmware.

With an optional stick-on transmitter, additional IR-enabled devices can be remote controlled via the network connection, enabling users to control their audio sources without being in the same location as the device.

1.2 Features

- Generates MP3 streams at adjustable bit rates (VBR) from analog or digital (optical/coaxial S/P DIF) sources

- Generates G.711 (aLaw/uLaw) streams at 8 or 24 kHz sample rate from an analog source

- Generates PCM (16 bit) streams at 8 or 24 kHz sample rate from an analog source

- Supported stream connections: HTTP, BRTP, RTP, SIP, Raw UDP, Raw TCP, Icecast and Icecast ID3 source, Shoutcast source

- Supports stream authentication (HTTP, Shoutcast, Icecast)

- 10/100 Mbit Ethernet connection supports automatic network configuration (BOOTP, DHCP, AutoIP and IPzator) as well as manual static IP configuration

- Features SonicIP® announcing the IP address on power up over the audio outputs

- Control and configuration using a standard web browser

- Remote monitoring using SNMP

- Remote controllable using HTTP, TCP and UDP

- Supports IR remote control command relaying (Network to IR out)

- Supports Serial Port relaying (Serial gateway over Network)

1.3   Installing the device

For the installation of the Barix Instreamer 100 or the legacy Barix Instreamer please refer to the corresponding "Quick Install Guide".
A printed version is included in the box and can also be downloaded from our site www.barix.com.

1.4   Additional documents

Technical specifications can be found in the corresponding product sheet which can be downloaded from our site www.barix.com.

For configuration information please download the "Instreamer Manual" from our website.

1.5   Preloaded Firmware

Barix preloads all Instreamer family devices with the current "Instreamer" firmware release version.

1.6   About this Technical Documentation

Links to chapters

References to chapters (e.g. X Chapter name) are red and underlined and serve as direct links when viewed in Adobe Acrobat Viewer.
Click on the link to jump to the referenced chapter, click on the left arrow iconto jump back to where you came from.

Bookmarks pane in Adobe Acrobat

The complete "Table of Contents" is available in Adobe Acrobat Viewer.
Click on the "Bookmarks" pane tab on the left side of Adobe Acrobat Viewer to open it.
Click on any bookmark to directly jump to the corresponding part of the manual.

Chapter overview

This technical documentation is divided into the following chapters:

- 2 Software Application Interface (explaining the details on available APIs and web interfaces)

- 3 Hardware and Connectors (explaining the details on the hardware connetcors)

2 Software Application Interface

2.1 Control Interface Description

- `0x`nn means a hexadecimal number.
- ↵ means `0x0D 0x0A 0x00` on answers. On requests ↵ could be one or more of the following codes/bytes: `0x0D, 0x0A, 0x00`.
- The answers are only echoed to the origin source of the command (not to the other interfaces).
- The answer can by selected by concatenate the `L` command to the command. If no special answer is requested the file ack.ack will be sent back.
- The answer files can be edited and changed to your needs (see 2.12 Own skins and web interface).
- The standard answers are designed as XML.
- All strings and everything else are case sensitive.
- All commands are asynchronous to the stream.
- One command mustn't exceed 1024 bytes even it is concatenated.

2.2 Concatenate Control Commands

- To concatenate control commands use `&`. The commands will be executed from left to right in sequence (not parallel). The ↵ must only be placed at the end of the whole command and not after each separate command.
- To start talking and set volume to 12 use: `c=83&v=12`↵
- This is useful in the init sequence, in UDP commands or the define the answer. The init sequence is based on the serial command interface.

2.3 Formats supported

- Streams MP3, G.711 (uLaw/aLaw 8/24 kHz) and PCM (Motorola 16Bit 8/24 kHz).
  MP3Pro files can be streamed but without the additional quality of MP3Pro.
- Encodes to MP3 (VBR only), G.711 (uLaw/aLaw 8/24 kHz) and PCM (Motorola 16Bit 8/24 kHz).
- The file extension of the audio file in the web application must be for uLaw 8 kHz u8, for uLaw 24 kHz u24, for aLaw 8 kHz a8, for aLaw 24 kHz a24, for PCM 8 kHz p8, for PCM 24 kHz p24 and for MP3 mp3. All extensions are case sensitive except mp3.

2.4   Principles of the CGI WEB interface

- The browser should support frames.

- Use GET method in forms.

- Respect the common character set for URL's.

- Example for CGI WEB commands: `http://x.x.x.x/rc.cgi?c=84` (command for RESETTALK on Instreamer x.x.x.x)

2.5   Principles of the SERIAL interface

- Default settings of the serial control interface: 9600 baud, 8 data bits, 1 stop bit, no parity

- Each command must be terminated with an ASCII code less than a space 0x20 (like carriage return or/and line feed).

- If the command is correct and could be executed the answer OK is sent back with attached carriage return (ASCII 0x0D) and line feed (ASCII 0x0A).

- ERROR with attached carriage return (ASCII 0x0D) and line feed (ASCII 0x0A) is sent back when:

  - a byte got lost

  - an invalid syntax is used

  - the time between two characters exceeds 10 seconds

  - the command is unknown or can't be executed

- There is no time to wait between two commands or characters.

- The serial connector pin out is described in chapter 3.3 Serial Port. It's the same as on a standard PC 9pol. DSub.

- If IR once is used, the serial command interface can't be used anymore.

- If the serial gateway functionality is used, the serial command interface can't be used anymore.

2.6   Principles of the UDP command interface

- The standard UDP interface port for control commands is 12301.

- Each UDP packet must be terminated with a ASCII 0x00, ASCII 0x0D (carriage return) or ASCII 0x0A (line feed). If it is not terminated, the last character will be discarded.

- If the command queue is full, busy.ack will be returned. This could happen if too many other commands are being executed at the same time.

2.7 Principles of the TCP interface

- The standard TCP interface port for control commands is 12302.

- Each command must be terminated with a ASCII 0x00, ASCII 0x0D (carriage return) or ASCII 0x0A (line feed).

- The answers are the same as on the SERIAL interface.

2.8 Principles of the TCP Serial Gateway

- Open a TCP connection to the local port (default 12302).

- The serial port parameters can be set in the configuration.

- Each byte transmitted to this TCP port is sent to the serial port.

- Each byte received on the serial port is sent to this TCP port.

- Only one TCP port at once can be used.

- As long as this TCP port is open the serial command interface is disabled.

- If IR functionality is once used (for receiving and/or transmitting), don't use the serial gateway without resetting the device .

2.9 Control, SERIAL, UDP, TCP and CGI WEB interface

Note: Although the Instreamer is an input device, it also has a headphone monitor, hence the commands to control the volume.

Element	Description	CGI command	SERIAL, TCP or UDP command
ANSWERS	Standard answer file ack.ack will be sent if nothing else is specified with the L command. The file nosupport.ack will be sent on an unknown command. To change the answer concatenate the command GETDYNFILE and chose the required answer file. ex. set volume: `v=4&L=volume.ack`	see the files in 2.12 Own skins and web interface	
MUTE	Toggle between mute and volume. MUTE stores the last volume and resets to this value on turn on volume. VOLUMEINC, VOLUMEDEC, VOLUME > 0 as well as FORCEMUTEOFF unmutes the device.	c=8	0x63 0x3D 0x38 0x00 (c=8)
VOLUMELOCK	After this command the volume can't be changed until you unlock it.	c=11	0x63 0x3D 0x31 0x31 0x00 (c=11)
VOLUMEUNLOCK	Unlock volume.	c=12	0x63 0x3D 0x31 0x32 0x00 (c=12)
SETASDEFAULT	Store current values (volume, volume lock, mute, bass, treble,	c=13	0x63 0x3D 0x31 0x33 0x00 (c=13)

Element	Description	CGI command	SERIAL, TCP or UDP command
	loudness level, loudness on) as default on startup.		
FACTORYDEFAULTS	Set factory default values for the current runtime configuration.	c=14	0x63 0x3D 0x31 0x34 0x00 (c=14)
VOLUMEINC	Increment volume one step.	c=19	0x63 0x3D 0x31 0x39 0x00 (c=19)
VOLUMEDEC	Decrement volume one step.	c=20	0x63 0x3D 0x32 0x30 0x00 (c=20)
LINEIN	Select Line In as input.	c=28	0x63 0x3D 0x32 0x38 0x00 (c=28)
FORCEMUTEON	mute on forced (no toggle), details see MUTE command	c=40	0x63 0x3D 0x34 0x30 0x00 (c=40)
FORCEMUTEOFF	mute off forced (no toggle)	c=41	0x63 0x3D 0x34 0x31 0x01 (c=41)
MONOOUT	Set mono	c=48	0x63 0x3D 0x34 0x38 0x01 (c=48)
SETRTS	Sets output RTS to logic 1 (-12V) Set the parameter RTS usage to off first.	c=60	0x63 0x3D 0x36 0x30 0x00 (c=60)
RESETRTS	Sets output RTS to logic 0 (+12V) Set the parameter RTS usage to off first.	c=61	0x63 0x3D 0x36 0x31 0x00 (c=61)
SELSPDIFIN1	Selects the optical input	c=81	0x63 0x3D 0x38 0x31 0x00 (c=81)
SELSPDIFIN2	Selects the coaxial input	c=82	0x63 0x3D 0x38 0x32 0x00 (c=82)
RESETTALK	Stops talking/streaming except in "send always" mode	c=84	0x63 0x3D 0x38 0x34 0x00 (c=84)
SETCTS	Simulates a set on CTS	c=89	0x63 0x3D 0x38 0x39 0x00 (c=89)
RESETCTS	Simulates a reset on CTS	c=90	0x63 0x3D 0x39 0x30 0x00 (c=90)
FORCETALK	Starts talking/streaming.	c=91	0x63 0x3D 0x39 0x31 0x00 (c=91)
DEFAULTS	Set factory defaults without changing the network settings: own IP, Gateway and Netmask and reboot the device.	c=94	0x63 0x3D 0x39 0x34 0x00 (c=94)
DEVICERESET	Hard reboot of device.	c=99	0x63 0x3D 0x39 0x39 0x00 (c=99)
BOOTLOADER	Exist the application and start the bootloader.	c=100	0x63 0x3D 0x31 0x30 0x30 0x00 (c=100)
DISCOVER	If this command is received the device replies with the file discover.ack.	c=65535	0x63 0x3D 0x36 0x35 0x35 0x33 0x35 0x00 (c=65535
PASSWORD	Concatenate this command the rest of the command sequence if the command interface is password (level 3) protected. The password has to be added in plain text.	a=…	0x61 0x3D … 0x00 (a=…)
CONFIG	Configuration commands via the normal CGI WEB command interface. Append the setup elements to this command. (see 2.10)	C=	0x43 0x3D … 0x00 (C=…)
SETMETADATA	Sets shoutcast metadata into the audio stream. The format of the input data is an ASCII string.	E=	0x45 0x3D … 0x00 (E=…)
CFGENCODE	Sets the encoding quality. The high byte nibble defines the encoding quality 0..7. 0 for the lowest up to 7 for the highest quality. The low byte nibble defines the sampling frequency and data format: 3 = MPEG1 / 48 kHz 1 = MPEG1 / 44.1 kHz 5 = MPEG1 / 32 kHz 2 = MPEG2 / 24 kHz 0 = MPEG2 / 22.05 kHz 4 = MPEG2 / 16 kHz	e=…	0x65 0x3D … 0x00 (e=…)

Element	Description	CGI command	SERIAL, TCP or UDP command
	6 = u-Law 24kHz 7 = u-Law 8kHz 8 = a-Law 24kHz 9 = a-Law 8kHz 10 = PCM 24 kHz 11 = PCM 8 kHz ex. for MPEG2/24khz and encoding quality 6: hexadecimal 0x62 is decimal 98. The command then will be **e=98** **e=255** restarts the encoder without changing the settings.		
SETSERGATEWAY	Sets the destination IP address and port for the serial gateway and connect to it. Alternatively, closean existing connection by setting a zero IP address. If this command is executed via the serial command interface the connection will be closed 1 sec after the last transmitted or received byte. **g=[<ip>][:<port>]** **<ip>** is the destination IP address. 0.0.0.0 to close the connection. **<port>** is the destination port. If none, LOCALPORT is used.	g=…	0x67 0x3D … 0x00 (g=…)

SENDIR	Transmit an IR command.
	This command is word organized (2 bytes, high byte first). The words can be separated by a comma (no spaces). The word is interpreted as a hexadecimal value.
	The first word is the selection for the IR command data following. It is organized in bits.
	Low Byte: **0x01** = SONY IR modulation
	                0x02 = RC 5 IR modulation
	                0x04 = NEC IR modulation
	                0x80 = RAW IR modulation
	
	Bit 8-14:0 for base band
	          1 for 37.5 kHz modulation frequency
	          2 for 60 kHz modulation frequency
	
	Bit 15: 0 for transmit on serial port 0 (Serial IR Dongle)
	              1 for transmit on IR OUT
	
	Example: **8102** for RC 5 IR modulation with 37.5 kHz on IR OUT.
	
	The format of the IR command data is dependent on the IR
		j=…	0x70 0x3D … 0x00 (j=…)

Element	Description	CGI command	SERIAL, TCP or UDP command
	modulation: SONY IR and RC 5: The bits (0 or 1) are transmitted as nibbles of the words in the IR command. If the number of bits is odd than an f is added. The sync and end of transmission are discarded Example: `j=0101,1010001001010f` NEC: Bytes are transmitted. The bit values are swapped so that LSB format is presented to the IR. Example: `j=0104,00FE7887` RAW: The first word after the selection word, is the active `1` (IR on) time in 100 us. The second word is the active `0` (IR off) time in 100 us. The next is again for active `1` and so on until a word has the value 0000. This is to build every possible IR command. Example: `j=0180,0018,0004,000c,0008,0000`		
GETDYNFILE	The response is the dynamic file stored in a cob file with that name (see 2.12 Own skins and web interface). Example: `L=getstate.ack`	L=…	0x4C 0x3D … 0x00 (L=…)

QUIETCONFIG
 	
!!! ONLY FOR EXPERTS!!!
	Quietly change configuration commands via the normal CGI WEB command interface. The commands are updated in RAM only without resetting the device and the changes lost if the device is rebooted. Append the setup elements to this command. (see) cf. C= command.
	
	QUIETCONFIG specifically supports the change of the UDP listen ports and was the main reason for adding the command.
	Examples of these commands via the Serial port:
	1) To change the UDPRXPORT to 3035
	    `Q=L=&W499=3035`
	2) To change the UDPPRIOPORT to 9090
	    `Q=L=&W287=9090`
	3) To change the UDPCMDPORT to 12399
	    `Q=L=&W791=12399`
	
	Not all commands are changeable on the fly, for example changing the TCP ports will have no affect. As a general rule if there is already a separate command to perform the change, e.g Volume control then QUIETCONFIG will not work.
	If you are not sure please ask Barix for advice.
		Q=	0x51 0x3d … 0x00 (Q=…)

Element	Description	CGI command	SERIAL, TCP or UDP command
STRING2SERIAL	Sends a string to the serial port. Neither a zero terminator nor a `CR` nor a `LF` will be added to the end of the string. Zero character and ampersand (`&`) are impossible to send this way.	S=...	0x53 0x3D...0x00 (S=...)
SENDTCPSTRING	Sends the attached string through the TCP interface. The answer is the string itself.	T=…	0x53 0x3D (T=…)
VOLUME00 VOLUME01 --- VOLUME20	Set the volume for the headset Set minimal volume level (volume off). Set volume level 1. Set maximal volume level. Corresponding values sent to the codec for 0..20: 0, 76, 80, 84, 88, 92, 96, 99, 102, 106, 108, 110, 112, 114, 116, 118, 120, 124, 125, 126, 127 One step is showed as 5%. The level 0 equals the 0%.	v=0 v=1 --- v=20	0x76 0x3D 0x30 0x00 (v=0) 0x76 0x3D 0x31 0x00 (v=1) --- 0x76 0x3D 0x32 0x30 0x00 (v=20)

2.10 Configuration via the command interfaces

The difference between the command and the configuration interface is only the prefix used, i.e. setup.cgi instead of rc.cgi for the cgi web interface.

Element	Description	CGI command	SERIAL, TCP or UDP command
SETCONFIG	Sets the configuration The expected string is exactly the one the HTML browser generates for the used forms for the configuration. (L= see GETDYNFILE) (see table in chapter 2.15 Configuration via HTML Pages for the names) The device will respond with HTTP status 200 OK and if referenced with the optional <file>. Then it will reboot. Examples for set the IP address to 192.168.1.22: for Serial: C=L=&B0=192&B1=168&B2=1&B3=22 for cgi: setup.cgi?L=uinetwork.html&B0=192&B1=168&B2=1&B3=22 or rc.cgi?C=L=uinetwork.html&B0=192&B1=168&B2=1&B3=22	L=<file>&…	0x43 0x3D 0x4C … (C=L=…)
GETCONFIG	Gets the configuration See the file getconfig.ack for the answer.	L=getconfig.ack	(L=getconfig.ack)

2.11 Setup

The factory default setup is contained in the binary file config.bin. This file can be edited with a hex editor. Be careful if you do changes. This file will be loaded to the EEPROM on factory default.

General Terms (EEPROM Organization)

- IP addresses are always stored with the highest byte at the lowest address.
- Strings are coded in ASCII and terminated with 0x00. The Length includes the termination.
- Values are stored in little endian format (Intel) (low byte first)
- All Values are integer.
- Signed values are stored in 2-complement.
- Unused bytes must be set to 0x00.

In the following table the column Byte shows the byte number in the 1016 bytes of configuration. The first byte has the number 0.

If a password is set then the device only will answer to this commands if the password in the command is set correct.

Parameter	Byte [dec]	Dyn. Name	Length [Byte]	Default Value	Short Description
Own IP	0	B0, B1, B2, B3	4	0.0.0.0	Static IP address of the device. 0.0.0.0 for DHCP. 0.0.1.0 disable AutoIP 0.0.2.0 disable DHCP 0.0.4.0 disable BOOTP 0.0.8.0 disable IPzator add this special IP addresses for disabling multiple protocols
Gateway IP	4	B4, B5, B6, B7	4	0.0.0.0	Gateway IP address. 0.0.0.0 for no gateway
Netmask	8	N8B0, N8B1, N8B2, N8B3	1	0	Subnetmask. The value is the count of the zero bits counted from the lowest byte. (ex. 8 for 255.255.255.0)
Password	72	S72	8		Password as String. Used in the Telnet Setup. All 0 means no password.
IFMODE0	80	B80b0-1, B80b2-3, B80b4-5, B80b6-7 or B80	1	0x4C	Definition of the bits in that byte for the serial port 0:

Parameter	Byte [dec]	Dyn. Name	Length [Byte]	Default Value	Short Description									
					Function	7	6	5	4	3	2	1	0	
					RS232-C							0	0	
					7 Bit					1	0			
					8 Bit					1	1			
					no parity			0	0					
					even parity			1	1					
					odd parity			0	1					
					1 Stopbit	0	1							
					2 Stopbit	1	1							
BAUDRATE0	81	B81	1	2	Baudrate for the serial port 0. (7 = 300, 6 = 600, 5 = 1200, 4 = 2400, 3 = 4800, 2 = 9600, 1 = 19200, 0 = 38400, 9 = 57600, 8 = 115200)									
FLOWCONTROL0	82	B82	1	0	Flow control for the serial port 0. (0 = no, 1= Software XON/XOFF, 2= Hardware RTS/CTS)									
LOCALPORT	86	W86	2	0	Port for the serial gateway (0 for disable). If the Gateway Destination is set this parameter is used as source port (0 for random source port).									
GATEWAYDSTIP	88	B88, B89, B90, B91	4	0.0.0.0	Gateway destination IP address. If this IP address is 0.0.0.0 and the GATEWAYDSTPORT is 0 then the serial gateway is disabled.									
GATEWAYDSTPORT	92	W92	2	0	Gateway destination Port (see GATEWAYDSTIP).									
Version Major	116	B116	1	1	Version Major value (do not change)									
Version Minor	117	B117	1	4	Version Minor value (do not change)									
Setupex Length	120	W120	2	894	Length of the extended setup (always 894)									
Password Level 1	122	S122	8		Password stored as MD5 hash (first 8 bytes) used for save configuration via web, all 0 means no password									
Password Level 2	130	S130	8		Password stored as MD5 hash (first 8 bytes) used for view the configuration via web, all 0 means no password									
Password Level 3	138	S138	8		Password stored as MD5 hash (first 8 bytes) used for control/commands, all 0 means no password									
Password Level 4	146	S146	8		Password stored as MD5 hash (first 8 bytes), all 0 means no password									
Password Level 5	154	S154	8		Password stored as MD5 hash (first 8 bytes), all 0 means no password									
Password Level 6	162	S162	8		Password stored as MD5 hash (first 8 bytes), all 0 means no password									
Listening password selector	204	S204	1		Value from 0 to 7. 0 means listening is not protected with a password and 1-6 mean Password level 1 to level 6 is used. Value too high is treated as 0.									
Ice/Shoutcast password	205	S205	24		Password stored as plaintext, with zero terminator. If length is 24, zero terminator is not present.									
Volume	244	B244	1	10	Volume 0..20									
A/D Amplifier Gain	249	B249	1	0	A/D amplifier gain 0=-3, 15=19.5 dB, one step is 1,5 dB (only for the line input)									
Sync Port	253	W253	2	0	Sync Port if device is Sync Master. 0 means use algorithm (see Exstreamer technical documentation) else use this number as fixed port (useful for firewalls).									
MP3 Configuration	255		1	0x00	This bits define the mp3 header configuration.									

Parameter	Byte [dec]	Dyn. Name	Length [Byte]	Default Value	Short Description
		B255b0, B255b1, B255b2, B255b3, B255b4-5, B255b4-5, B255b6, B255b7			This values can be added (the function is activated by set the bit): 0x01: set for disable CRC in MP3 frame header 0x02: set for disable MS-Stereo encoding 0x04: set for bitreservoir kept empty 0x08: not used 0x10: for emphasis 50/15 s 0x30: for emphasis CCITT J.17 0x40: set for original stream 0x80: set for not copyright protected
Device Name	256	S256	21		Name of the device
Media Configuration	277	B277b0, B277b1, B277b2, B277b3, B277b4, B277b5, B277b6, B277b7	1	0x00	This values can be added (the function is activated by set the bit): 0x01: not used 0x02: not used 0x04: not used 0x08: not used 0x10: not used 0x20: not used 0x40: not used 0x80: no SonicIP
Buffer Underrun Mode	281	B281	1	0	This parameter defines what should happen if a buffer underrun is detected. 0=disconnect 1=skip
Stream Packet Strategy	282	B282	1	0	This parameter defines how the packets are generated: 0=send as fast as possible (if a frame is received from the encoder) 1=optimal package size (send if the packet is full or after 200ms)
WEB Server Port	283	W283	2	0	This parameter defines on which port the device's web server are on. (0=default HTTP port 80)
Shoutcast private	285	B285	1	0	0=public (icy-public=1), 1=private (icy-public=0). Takeninto account only for Shoutcast source streaming.
UDP Priority Rx Port	287	W287	2	0	This parameter defines the UDP Priority Receiver Port. A stream sent to this port always will be played unless another priority stream already will be received.
TCP Priority Rx Port	289	W289	2	0	This parameter defines the TCP Priority Receiver Port. A stream sent to this port always will be played unless another priority stream already will be received.
icy-url / SIP user	291	S291	61		icy-URL for Icecast servers or SIP user for SIP mode.
Mode	352	B352	1	6	5=send on CTS, 6=send always, 7=send on level
Radio Path	353	S353	33	/xstream	Path for the internet radio connection (example URL for an internet radio connection to the Instreamer http://a.a.a.a/xstream where a.a.a.a is the IP address of the serving device)
Init Sequence	386	S386	64		String of commands (like serial command interface) which is executed after each startup of the device.
Destination IP 1	453	B453, B454,	4	0.0.0.0	Destination IP for the 1st connection (on type Raw UDP 0.0.0.0 is the subnet broadcast IP

Parameter	Byte [dec]	Dyn. Name	Length [Byte]	Default Value	Short Description
		B455, B456			address)
Destination IP 2	457	B457, B458, B459, B460	4	0.0.0.0	Destination IP for the 2nd connection (on type Raw UDP 0.0.0.0 is the subnet broadcast IP address)
Destination IP 3	461	B461, B462, B463, B464	4	0.0.0.0	Destination IP for the 3rd connection (on type Raw UDP 0.0.0.0 is the subnet broadcast IP address)
Destination IP 4	465	B465, B466, B467, B468	4	0.0.0.0	Destination IP for the 4th connection (on type Raw UDP 0.0.0.0 is the subnet broadcast IP address)
Destination IP 5	469	B469, B470, B471, B472	4	0.0.0.0	Destination IP for the 5th connection (on type Raw UDP 0.0.0.0 is the subnet broadcast IP address)
Destination IP 6	473	B473, B474, B475, B476	4	0.0.0.0	Destination IP for the 6th connection (on type Raw UDP 0.0.0.0 is the subnet broadcast IP address)
Destination IP 7	477	B477, B478, B479, B480	4	0.0.0.0	Destination IP for the 7th connection (on type Raw UDP 0.0.0.0 is the subnet broadcast IP address)
Destination IP 8	481	B481, B482, B483, B484	4	0.0.0.0	Destination IP for the 8th connection (on type Raw UDP 0.0.0.0 is the subnet broadcast IP address)
Encoding Quality	485	B485	1	0	Encoding quality 0..7, 0 for the lowest up to 7 for the highest quality.
Sampling Frequency	486	B486	1	0	Sampling frequency and data format 3 = MPEG1 / 48 kHz 1 = MPEG1 / 44.1 kHz 5 = MPEG1 / 32 kHz 2 = MPEG2 / 24 kHz 0 = MPEG2 / 22.05 kHz 4 = MPEG2 / 16 kHz 6 = u-Law 24kHz 7 = u-Law 8kHz 8 = a-Law 24kHz 9 = a-Law 8kHz 10 = PCM 24 kHz 11 = PCM 8 kHz
Pre Trigger Start	488	W488	2	0	Number of bytes that will be streamed before the trigger occured
Post Trigger Play	490	W490	2	1024	Amount of time [ms] that will be streamed after the trigger has switched off
Trigger Level	492	W492	2	1000	Audio receiving level that defines when to start streaming
Input Mode	494	B494	1	0x01	Input Mode 0x01 = Line In 0x04 = S/PDIF optical 0x05 = S/PDIF coaxial 0x80 = Mono (set this flag for mono else it is stereo)
UDP TX Source Port	495	W495	2	0	UDP stream source port, 0 for use corresponding destination port, else fixed
TCP Command IP	497	B497,	4	0.0.0.0	If 0.0.0.0, the TCP command port is opened to listen. Otherwise, an active connection is

Parameter	Byte [dec]	Dyn. Name	Length [Byte]	Default Value	Short Description
Address		B498, B499, B500			established. The port number is governed by TCP Command Port (W793).
Destination Port 1	511	W511	2	0	Destination Port for the 1st connection (0 for use UDP Rx Port, if this is 0 use default 3030)
Destination Port 2	513	W513	2	0	Destination Port for the 2nd connection (0 for use UDP Rx Port, if this is 0 use default 3030)
Destination Port 3	515	W515	2	0	Destination Port for the 3rd connection (0 for use UDP Rx Port, if this is 0 use default 3030)
Destination Port 4	517	W517	2	0	Destination Port for the 4th connection (0 for use UDP Rx Port, if this is 0 use default 3030)
Destination Port 5	519	W519	2	0	Destination Port for the 5th connection (0 for use UDP Rx Port, if this is 0 use default 3030)
Destination Port 6	521	W521	2	0	Destination Port for the 6th connection (0 for use UDP Rx Port, if this is 0 use default 3030)
Destination Port 7	523	W523	2	0	Destination Port for the 7th connection (0 for use UDP Rx Port, if this is 0 use default 3030)
Destination Port 8	525	W525	2	0	Destination Port for the 8th connection (0 for use UDP Rx Port, if this is 0 use default 3030)
Connection Type 1	527	B527	1	1	Type for the 1st connection (0=Internet Radio, 1 = Raw UDP, 2 = Raw TCP, 3=not used, 4=Icecast source, 5=Shoutcast source, 6=RTP)
Connection Type 2	528	B528	1	0	Type for the 2nd connection (0=Internet Radio, 1 = Raw UDP, 2 = Raw TCP, 3=not used, 4=Icecast source, 5=Shoutcast source, 6=RTP)
Connection Type 3	529	B529	1	0	Type for the 3rd connection (0=Internet Radio, 1 = Raw UDP, 2= Raw TCP, 3=notused, 4=Icecast source, 5=Shoutcast source, 6=RTP)
Connection Type 4	530	B530	1	0	Type for the 4th connection (0=Internet Radio, 1 = Raw UDP, 2 = Raw TCP, 3=not used, 4=Icecast source, 5=Shoutcast source, 6=RTP)
Connection Type 5	531	B531	1	0	Type for the 5th connection (0=Internet Radio, 1 = Raw UDP, 2 = Raw TCP, 3=not used, 4=Icecast source, 5=Shoutcast source, 6=RTP)
Connection Type 6	532	B532	1	0	Type for the 6th connection (0=Internet Radio, 1 = Raw UDP, 2 = Raw TCP, 3=not used, 4=Icecast source, 5=Shoutcast source, 6=RTP)
Connection Type 7	533	B533	1	0	Type for the 7th connection (0=Internet Radio, 1 = Raw UDP, 2 = Raw TCP, 3=not used, 4=Icecast source, 5=Shoutcast source, 6=RTP)
Connection Type 8	534	B534	1	0	Type for the 8th connection (0=Internet Radio, 1 = Raw UDP, 2 = Raw TCP, 3=not used, 4=Icecast source, 5=Shoutcast source, 6=RTP)
Icy-genre	535	S535	31		Genre of the encoded stream. Takeninto account only when Shoutcast source is selected. Holds 30 characters + 1 zero terminator = 31 bytes
UDP Command Port	791	W791	2	12301	Receiving port for the UDP command interface (0 for disable)
TCP Command Port	793	W793	2	12302	Listening port for the TCP command interface (0 for disable)
Command CTS opened	795	S795	64	c=84	Command sequence executed if CTS will be opened
Command CTS closed	859	S859	64	c=91	Command sequence executed if CTS will be closed
Preset	923	B923	1	0	User-specific storage, this parameter has no functionality. It canbe used by the user for the web interface.
SNMP Target Trap IP	924	B924, B925, B926, B927	4	0.0.0.0	SNMP Target IP (0.0.0.0 for disable SNMP)

Parameter	Byte [dec]	Dyn. Name	Length [Byte]	Default Value	Short Description
Low Level Left	928	W928	2	0	Trap will be triggered if the quasi peak an the left channel falls below this level. (0 disables Trap)
Low Level Right	930	W930	2	0	Trap will be triggered if the quasi peak an the right channel falls below this level. (0 disables Trap)
High Level Left	932	W932	2	0	Trap will be triggered if the quasi peak an the left channel falls above this level. (0 disables Trap)
High Level Right	934	W934	2	0	Trap will be triggered if the quasi peak an the right channel falls above this level. (0 disables Trap)
Trap Repeat Left	936	W936	2	0	The Trap for the left channel will be repeated after this number of seconds (0 disable)
Trap Repeat Rigth	938	W938	2	0	The Trap for the right channel will be repeated after this number of seconds (0 disable)
Silence Timeout Left	940	W940	2	0	Silence for the left channel will be detected after this number of seconds
Silence Timeout Right	942	W942	2	0	Silence for the left channel will be detected after this number of seconds

2.12 Own skins and web interface

With the Instreamer App Development Kit you can design your own web pages (skin) and modify the answers to your needs. This kit is placed in the folder "**webuidevkit**" inside the firmware update package available on www.barix.com. The "**instreamerapp**" folder holds the files you need for the web pages. You can simply edit these files and/or add new ones. The web interface (and the firmware) need at least the files shown in the table further below:

The filenames mustn't start with rc.cgi or setup.cgi.

Don't exceed 64 kByte of data per file. Note that a bigger .cob file needs per 64 kByte one flash page of 64 kByte.

To generate a .cob file start the batch instreamerapp.bat. Upload the generated .cob file into the device to the web application page (overwrite).

For the upload go to the configuration page of the device and click on the button Update. Follow the instructions there. If the device has rebooted and the update page is showed type **http://x.x.x.x/updateex.html** in the address field of the browser where **x.x.x.x** is the IP address of the device.

Free targets can be found in 2.13 Memory Page Usage. The target field is case sensitive. If you upload a .cob file to used pages the current content will be overloaded by the new one. The web server in the device sees all the targets (.cob files) as one directory. If two files in different .cob files have the same name then the one from the lower page is chosen. After the upload reboot the device and reload the modified page in the browser to see the changes. Sometimes it's needed to close the browser to see the changes depending on the browser's cache strategy.

File	Dyn	Description

Version File

File	Dyn	Description
INSTREAMERAPPVERSION		for the version number and the history

Answer files (see 2.15 Configuration via HTML Pages for the dynamic marks contained in the files and the exact description therefore).

File	Dyn	Description
status		current state and configuration
ack.ack		standard answer for commands
busy.ack		standard answer for ignored commands (UDP)
discover.ack	✓	answer for the DISCOVER command
getconfig.ack	✓	complete configuration
getcts.ack	✓	actual state of CTS
getrts.ack	✓	actual state of RTS
getstate.ack	✓	outputs the state of the device
nosupport.ack		answer for unknown and/or unsupported commands
instreamer.m3u	✓	M3U (Moving Picture Experts Group Audio Layer 3 Uniform Resource Locator) playlist file for the Instreamer internet radio station. If this file is accessed over the web interface, it contains references generated from every line of the streaming table that is set to Internet Radio. Lines with 0.0.0.0 IP address configured will contain the device's IP address. Otherwise the configured IP address from the table line will appear in the playlist. The port appearing in the playlist is 80 for 0 configured, otherwise the number configured.
instreamer.pls	✓	PLS playlist file for the Instreamer internet radio station. Contains only one line with Instreamer IP address and the web server port.
instreamer.ram	✓	RAM (Real Audio Metadata) playlist file for the Instreamer internet radio station. Contains only one line with Instreamer IP address and the web server port.
instreamer.asx	✓	ASX (Advanced Stream Redirector) playlist file for the Instreamer internet radio station. If this file is accessed over the web interface, it contains references generated from every line of the streaming table that is set to Internet Radio. Lines with 0.0.0.0 IP address configured will contain the device's IP address. Otherwise the configured IP address from the table line will appear in the playlist. The port appearing in the playlist is 80 for 0 configured, otherwise the number configured.

Configuration file

File	Dyn	Description
config.bin		factory default settings. The file is binary and an exact mirror for the EEPROM. See 2.11 Setup for the organization. Edit this file with a hex editor if you need your own factory default settings.

Pictures

File	Dyn	Description
4to0.gif		needed for apply the configuration for waiting for the reboot of the device
active.gif		used as refreh button on the control page
barix.gif		used in uicfg.html
barix0.gif		used in uimenu0.html
d0.gif		pixel used for showing logical states
d1.gif		pixel used for showing logical states

File	Dyn	Description
d2.gif		pixel used for showing logical states
instreamer.gif		picture for the control interface, used in skin.html
menu.gif		picture for the menu buttons in the configuration, used in uicfg.html
menu0.gif		picture for the menu buttons in the control interface, used in uimenu.html
favicon.ico		Barix symbol

HTML pages (see 2.15 Configuration via HTML Pages for the dynamic marks included in the files)

File	Dyn	Description
index.html		main page of the web server, included the five frames: skin, info, playlist, songlist, empty. empty is a hidden frame that receives the answer of the CGI commands. Can be overridden by start.html. This allows the creation and execution of an alternative cob file which can be loaded into any of the available flash web pages. (Free pages are listed in 2.13 Memory Page Usage. The existing Barix application cob files can be left in the flash.
notauthorized.html		showed if the user isn't authorized to view a page
skin.html		for the control interface of the device, used in index.html
status	✓	shows the actual states and configuration of the device
toomanyusers.html		for the answer if too many users try to connect to the internet radio
uiaudio.html	✓	configuration pages for the corresponding settings
uicontrol.html	✓	
uiio.html	✓	
uinetwork.html	✓	
uisecurity.html	✓	
uiserial.html	✓	
uistreaming.html	✓	
uicfg.html	✓	shows the current loaded versions of the device
uiconfig.html	✓	main configuration, contains the main frames for the configuration
uidefaults.html	✓	set factory defaults
uifaudio.html		specific configuration page, contains the frames for the corresponding configuration pages
uifcontrol.html		
uifdefaults.html		
uifio.html		
uifloader.html		
uifnetwork.html		
uifreboot.html		
uifsecurity.html		
uifserial.html		
uifstreaming.html		
uifupdate.html		
uifmenu.html		frame page for the skin and version
uifstatus.html		frame page for the device status
uihaudio.html		help for the corresponding configuration pages
uihcontrol.html		

File	Dyn	Description
uihdefaults.html		
uihio.html		
uihloader.html		
uihnetwork.html		
uihreboot.html		
uihsecurity.html		
uihserial.html		
uihstreaming.html		
uihupdate.html		
uihstatus.html		help for the device status page
uilogout.html		logout page
uimenu.html	✓	frame for menu
uimenuline.html	✓	menu for the configuration buttons on the configuration pages
uimenu0.html		menu for the configuration button on the control page
uiraudio.html		showed after pressing apply or a reboot of the device is needed until the device has rebooted
uircontrol.html		
uirdefaults.html		
uirio.html		
uirloader.html		
uirnetwork.html	✓	
uirnetwork0.html	✓	
uirnetwork00.html		
uirreboot.html		
uirsecurity.html		
uirserial.html		
uirstreaming.html		
uirupdate.html	✓	
uirdefaults1.html		showed after the device is set to factory defaults an has successfully rebooted
uireboot.html		reboot the device
uirreboot1.html		showed after the device is rebooted an has then successfully rebooted
uistatus.html	✓	control page for IO and talk
uistatusl.html	✓	status page that shows the states
uiupdate.html	✓	update the device
update.html		forwarding page to hide the command for the update

Java Script

util.js		javascript functions for the HTML configuration pages (range checks)

2.13 Memory Page Usage

A page is 64 kByte of flash memory. Free pages can be used for additional resources.

Instreamer (1MB Flash) (Note: 0xC00000 = 0xD00000 = 0xE00000 = 0xF00000)

Page / Target	Content	Address for Rescuekit
8K (WEB0)	instreamware.rom (Firmware)	
WEB1	reserved	0xC10000
WEB2	xt05.bin (BIOS)	0xC20000
WEB3	sg.bin (Util library)	0xC30000
WEB4	sonicip.cob (Sonic IP Resources)	0xC40000
WEB5	instreamerapp.cob (Web Application)	0xC50000
WEB6	instreamerapp.cob continued (Web Application)	0xC60000
WEB7	free (see 2.12 Own skins and web interface)	0xC70000
WEB8	free (see 2.12 Own skins and web interface)	0xC80000
WEB9	temporary used for updates	0xC90000
WEB10	free (see 2.12 Own skins and web interface)	0xCA0000
WEB11	free (see 2.12 Own skins and web interface)	0xCB0000
WEB12	free (see 2.12 Own skins and web interface)	0xCC0000
WEB13	free (see 2.12 Own skins and web interface)	0xCD0000
WEB14	temporary used for updates	0xCE0000

2.14 Dynamic Web Page

Web pages can include dynamic values. Dynamic Web Pages are built in HTML or XML or in an other text file format that exclude the binary character 0x00, i.e. the dynamic page can be an HTML file. It's possible to use scripts or everything else allowed in the given document's file format.

Initial Dynamic Mark

In order to indicate that Web page is dynamic, it has to contain the special initial dynamic mark `&L(0,"*");` in the first 500 Bytes and before any other dynamic value is used. The initial mark can also have decimal number as its optional third parameter. Example of such initial mark is `&L(0,"*",1);`.

The third parameter is parsed bitwise and has the following meaning:

- If bit 7 is set then the code page IBM437 will be used instead of the standard HTML code page.
- If bit 4 is set the access will be exclusive (only one user at a time, tested by its IP address). The user has to logout or the software does an automatic logoff 20 min after the last access to such a page. Only one password level can have the exclusive feature (doesn't matter which one).
- Bits 1-3 are used as password level (1-6) for the file corresponding to the password level parameters in the configuration.
  Example for level 5: `&L(0,"*",10);`
- If bit 0 is set, then the content length will not be included in the HTTP header. Page is sent faster by saving the time needed to calculate the content length.

Syntax of Dynamic Marks

Dynamic marks can be used to put dynamic values in Web pages. All dynamic marks have the following syntax:
`&L<name>(<id>,<format>[,par]);`
A dynamic mark always starts with `&L` and it is always case sensitive.

- `<name>` selects a group of dynamic values. Defined is the "Setup" group for all configuration parameters and the "State" group for actual parameter states. Remaining parameters are included in parentheses, with the right parenthesis followed by a semicolon.
- `<id>` determines the desired function.
- `<format>` is a C-style format string (refer to the ANSI documentation).
- `<par>` are optional additional parameters. If additional parameters are needed, it is mentioned in the function lists below.

Note: The string ") ;" is not allowed inside a dynamic mark.
To have this construct inside the format string, use ") \ ;" (in an unknown escape sequence, only the '\' will be removed).
To have a "%" sign (percent sign) inside the format string, use "%%" (two signs without space).

The whole mark is replaced by the dynamic value formatted with the **`<format>`** string. Only one value is allowed per dynamic mark. The length of the dynamic mark mustn't exceed 500 characters. The resulting string from the dynamic mark must not exceed 500 characters.

A dynamic mark can be contained in an another dynamic mark. Only one recursion step is allowed and correct "escaping" has to be applied. Example:

```
&LSetup(3,"%s",419,B,!0,"<meta http-equiv=refresh content=\"&LSetup(1,\"%u\",419)\;; url=info.html\">");
```

Note the special "\" before the semicolon of the dynamic mark inside. This is because the escape sequence is interpreted as only a semicolon and is needed in order to include the prohibited sequence ") ;" inside a dynamic mark.

2.15 Configuration via HTML Pages

The HTML pages for the device configuration use the functionality for dynamic web pages (see 2.14 Dynamic Web Page). All of the configuration parameters are placed in HTML forms and are transferred by the method GET. Some of the values are checked by java script to prevent wrong values. Not all of the configuration parameters have to be present in a form. It is possible to have only a part of the configuration on a web page. The form has to start with the following two tags:

```
<form method=GET action=setup.cgi target="answer"><input type="hidden" type="text" name=L
value=uirnetwork.html>
```

The target of the form could be changed. The answer after transmitting the form will be the HTML page uirnetwork.html. For another HTML page change this value. If this value isn't available only the HTTP status 200 OK will be sent back.

The following example shows how to implement a form field for the configuration value of the highest byte in the 'own IP address'.
The input element name is a defined string, which has to be handled with care (see more about this below). The type character B stands for an unsigned value (see table below). 0 is the address of the expected configuration parameter (see table in 2.11 Setup).
The value is a dynamic mark (see table below).

```
<input name=B0 size=3 maxlength=3 value=&LSetup(1,"%u",0); onChange=IPCheck(this)>
```

In the next example the name selects the configuration parameter 'CTS close command' in the setup (see table in 2.11 Setup).

```
<input name=S535 size=20 maxlength=20 value='&LSetup(4,"%s",535);'>
```

To set a password also the name Sxxx is used with the address of the corresponding password level. The max. length if the plain password is 24. The delete the password use 24 or more characters (ex. 25 spaces). The device builds the MD5 hash over the plain password and stores the lower 8 bytes. All 8 bytes set to 0 means no password set.

This example shows how to implement a form field for the configuration of the Netmask. The names for the bytes of the Netmask are N8B0, N8B1, N8B2 and N8B3. 8 is the address of the Netmask in the common setup (see table in 2.11 Setup). The value after the B is the byte number of the byte in the Netmask starting with 0 for the first byte at the left. This special handling for Netmask is needed because the Netmask is stored in one byte and not like the IP address in 4 bytes.

```
<input name=N8B0 size=3 maxlength=3 value=&LSetup(2,"%u",8,0); onChange=netMaskCheck(this)>
```

The next example shows how to implement a form field for the configuration of the parameter 'Flow control' as a selection. If the value of the configuration parameter is equal to the second last parameter in the dynamic mark it will be replaced by the last parameter of the dynamic mark.

```
<select size=1 name=B82>
  <option value=0 &Lsetup(3,"%s",82,B,0,"selected");>none</option>
  <option value=1 &LSetup(3,"%s",82,B,1,"selected");>Software (XON/XOFF)</option>
  <option value=2 &LSetup(3,"%s",82,B,2,"selected");>Hardware (RTS/CTS)</option>
</select>
```

This example shows how to implement radio buttons for the configuration parameter 'Sonic IP'. The function of the dynamic marks are equal to the example above.

```
<input type=radio name=B277b7 value=0&LSetup(3,"%s",277,b7,0," checked");>Yes<input type=radio name=B277b7
value=1&LSetup(3,"%s",277,b7,1," checked");>No
```

To transmit the new configuration data to the device the submit input type of the form is used.

```
<input type=submit value=' Apply '>
```

By pressing the Apply button the new configuration data will be transferred to the device. It will store the new data to its configuration memory (EEPROM). After this it sends the answer (see above) to the browser and then it reboots itself to apply the new configuration.

Dynamic Marks For Group State

<id>	Type	Description

1	Function	Print string if equal
		3. [par]: state variable
		    5 volume lock (0 = no lock, 1 = locked)
		   22 mode (0 = unknown, 1 = streaming)
		   23 status (0 = idle, 1 = play, 2 = pause)
		   45 streaming mode (0 = none, 3 = forced talk, 4 = receive)
		4. [par]: Type (see id 1 in 'Dynamic Marks for Group Setup' below)
		5. [par]: value to compare. The prefixes !, > or < are allowed to change the comparison (no spaces between)
		6. [par]: string for output if state value is equal to 5. [par]

2	Function	Print Byte
		3. [par]: state variable
		   1 current IP address
		   2 LAN MAC address
		   3 current netmask (stored as 4 bytes like the current IP address)
		   4 current gateway IP address

<id>	Type	Description
		5 current DNS IP address
		6 current MAC address
		4. [par]: offset in bytes for the state variable
		ex. `&LState(3,"%u",1,0);` for the highest byte of the current IP address
		ex. `&LState(3,"%H",2,0);` for the MAC address
3	Function	Print state value
		3. [par]: state variable
		1 quasi peak value left in
		2 quasi peak value right in
		3 quasi peak value left out
		4 quasi peak value right out
		6 hardware type (0 = unknown, 1 = Instreamer)
		7 codec type (0 = no codec, 1 = MAS3509 (decoder), 2 = MAS3587 (encoder))
		8 mode (0 = unknown, 1 = streaming)
		9 status (0 = idle, 1 = play, 2 = pause)
		13 state of CTS (0 = close, 1 = open)
		14 state of RTS (0 = close, 1 = open)
		15 state of talk (0 = off, 2 = forced)
		16 send stream (0 = no, 1 = yes)
		18 amount of bytes in the streaming buffer
		19 number of encoded bytes (double word)
		21 current volume
		31 CTS (0 = off, 1 = on)
		39 current uptime in milliseconds (double word)
		40 current uptime in seconds (double word)
		45 streaming mode (0 = none, 3 = forced talk, 4 = receive)

Dynamic Marks for Group Setup

<id>	Type	Description
1	Function	Print setup value
		3. [par]: Address (decimal) of the value in the setup
		4. [par]: Type of the value (B for unsigned byte, W for word, D for double word, c for char/signed byte, b for bit numbered from 0 to 7 ex. b3 for the fourth bit). If this parameter isn't available the type will be B.
		ex. `&LSetup(1,"%08lx",315,D);` as hexadecimal value with 8 characters an leading zeros
		ex. `&LSetup(1,"%lu",311,D);` as unsigned long decimal value
2	Function	Print Netmask Byte
		3. [par]: Address (decimal) of the value in the setup
		4. [par]: Byte number of the netmask IP address byte starting with 0 for the first left byte and incremented by one for the next bytes

<id>	Type	Description
3	Function	Print string if equal 3. [par]: Address (decimal) of the value in the setup 4. [par]: Type (see id 1 above) 5. [par]: value to compare. The prefixes !, > or < are allowed to change the comparison (no spaces between) 6. [par]: string for output if value at address is equal to 5. [par]
4	Function	Print string 3. [par]: Address (decimal) of the value in the setup
5	Byte (integer)	Firmware Version Major
6	Byte (integer)	Firmware Version Minor
7	Byte (integer)	Bootloader Version Major
8	Byte (integer)	Bootloader Version Minor
9	Function	Prints the version out of a standard version file in a *.cob application 3. [par]: name of the version file 4. [par]: 1 for major version number (byte), 0 for minor version number (byte)
10	Byte (integer)	year of the firmware build (only decade)
11	Byte (integer)	month of the firmware build
12	Byte (integer)	day of the firmware build
13	Byte (integer)	Song Major
14	Byte (Integer)	Song Minor
15	Byte (integer)	XT Major
16	Byte (integer)	XT Minor

See chapter for the addresses of the configuration parameters.

Names for the form elements

- If the value is an integer the first character is a B.

- If the value is a Netmask the first character is an N.

- If the value is a string the first character is an S.

- If the value is a long (4 byte) the first character is a D.

- If the value is a signed byte the character is a c.

- if the value is a word the character is a W.

The following decimal value in the name is the address of the configuration parameter (see).

To set a bit in a configuration parameter (ex. Mediaconfig) add the character b followed by the number of the bit (ex. 7 for the 8. bit in the byte) starting at 0.

To define the byte of the Netmask add the character B followed by the byte number (see <id> 2 in the table above).

Examples of names

- B0   first (left) byte of the configuration parameter 'own IP address'
- B1   second byte of the configuration parameter 'own IP address'
- N8B0   first (left) byte of the Netmask
- N8B1   name of the second byte of the Netmask
- S535   CTS close command
- B277b7   Sonic IP

2.16 Configuration Logout

The logout is placed in an HTML form and is transferred by the method GET. The form has to contain an element named L with the value for the answer page and a second element with the name D. This element is the indication for the logout.

```
<form action=setup.cgi method=get target=_top>

  <input type=hidden name=L value=logout.html><input type=hidden name=D><input type=submit value=" Logout ">

</form>
```

The target of the form could be changed.

The answer after transmitting the form will be the HTML page `logout.html`. For another HTML page change this value. If this value isn't available only the HTTP status `200 OK` will be sent back.

3 Hardware and Connectors

3.1 Green and Red LEDs

Two status LEDs provide information on the current device operation.

No Application loaded (only bootloader) or started with hold reset button during power up

The green LED is on and the red LED blinks.

Application starts (Barix boot-up sequence)

First the red goes on and the green LED blinks once.

If no IP address is configured, then during DHCP the red LED blinks with a continuous cycle . The green LED blinks five times and then pause four times.

If an error is detected the red LED remains on and the device resets itself after the green LED has indicated the error as follows:

Error	Green LED blinks
Corrupt application or IP address conflict	five times.
The Network hardware could not be initialized or a Corrupt MAC address	three times.

Application is running:

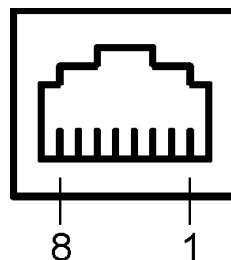On stop (not sending) the green LED is on and the red LED is off.

On sending the green LED blinks slowly and the red LED is off.

3.2 Ethernet

- Yellow LED (right LED): Link ok if on
- Green/Orange LED (left LED): green if 10Mbit, orange if 100Mbit

Connector RJ-45

| Pin	Name		Pin	Name
1	TX+		5	NC
2	TX-		6	RX-
3	RX+		7	NC
4	NC		8	NC

3.3 Serial Port

- V_{RxD} +/- 15V   TxD is short circuit protected.
- Max. Input Logic Threshold LOW 0.8V   Min. Input Logic Threshold HIGH 2.0V
- i_{max} (RTS): 10mA
- I_{max} (12VDC with supplied 12V power supply): 100mA. This pin is connected directly to the power supply.
- Transceiver: SIPEX SP3232E

Connector D-Sub (9 pin male)

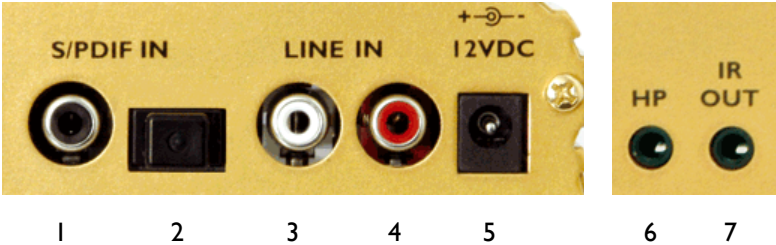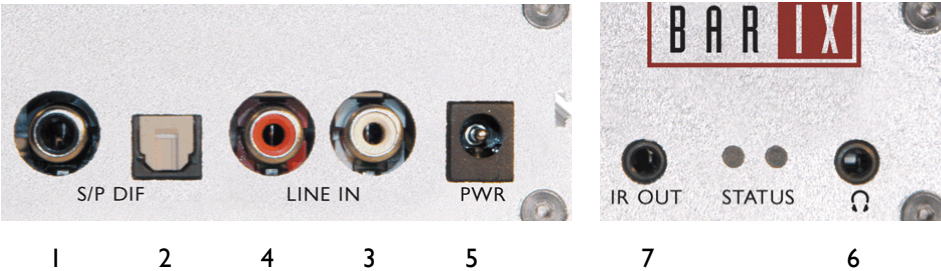| Pin	Name		Pin	Name
1	NC		6	NC
2	RxD		7	RTS
3	TxD		8	CTS
4	12VDC		9	NC
5	GND			
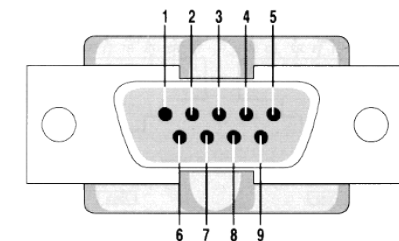
3.4 Codec

Built in as a Micronas MAS3587 Revision B2. For further information see Micronas' datasheets of the codec.

3.5 Other Connectors

Instreamer (legacy device)   Instreamer 100

Name	Nr	Description
S/PDIF IN coax.	1	coaxial S/PDIF input, 32 kHz, 44.1 kHz, 48 kHz
S/PDIF IN optic.	2	optical S/PDIF input, 32 kHz, 44.1 kHz, 48 kHz
LINE IN left	3	Line input left/right RCA, 2Vpp max. level (0dBFs), for
LINE IN right	4	Instreamer 100 4Vpp max. level (0dBFs)
Power	5	Power in, Center +9..24 VDC, Ring Ground, max. 4 Watt
HP	6	Headphone output, 3.5 mm Jack, 2Vpp max.
IR OUT	7	3.5 mm Jack Infared output for IR transmitter (Tip Anode, Ring and Ground Kathode of IR Emitter)

3.6 Thermal Characteristics

Type	Value
Working Temperature Range	0-50°C
Humidity	0-70% relative humidity, noncondensing

4   Legal Information

For information about our devices and the latest version of this manual please visit www.barix.com.

Barix AG
Seefeldstrasse 303
8008 Zurich

SWITZERLAND

Phone: +41 43 433 22 11
Fax: +41 44 274 28 49

Internet

web: www.barix.com

email: sales@barix.com

support: support@barix.com