



Communications Protocol

13/11/2023

Data acquisition and control of Axiom equipment is performed by home automation systems or personal computers using a special protocol developed by Axiom. This document explains the protocol itself and the low level transport.

1. Transport

1.1. RS-232

The lead must be a straight wired cable, except on the AX4752 and prior models this was a “null modem”, which is a 9 pin female ‘D’ connectors at both ends (pin connections 2 and 3 swapped at one end). **Only RX, TX & 0V (pin 5) are used.** Baud Rate = **9600**.

Each byte in the command is encoded into two ASCII encoded hexadecimal characters. A linefeed character signals the end of the command. The carriage return character that is typically used in combination with the line feed is optional and is ignored. XON and XOFF (characters 11h and 13h) are used for flow control where the XOFF state times out after approximately 1.5 seconds in order to prevent an inadvertent system lock up.

If the device that is receiving a command is capable of transmission, it should transmit the command back to the originator. This is required because of connection issues with RS-232. For example a PC's TX is connected to an amplifier's RX and the amplifier's TX is connected to a keypad and the PC's RX. For the PC to communicate successfully with the keypad, all that arrives on the amplifier's RX must be sent back out its TX. To avoid possible endless cycles, the device originally sending a command must ignore the command if it is received again.

1.2. USB

Previously the CDC class (abstract control model) was employed to emulate RS-232, enabling software to interact with devices supporting USB theoretically without need of change. However from the AX400/AX800 generation onwards a proprietary protocol is in use.

1.3. Infrared

The infrared interface allows Axiom devices to be controlled using the Axiom Communications Protocol. It is not suitable for two-way communications unless combined with another interface such as RS232.

The modulation frequency is 44.1 kHz and all bursts of IR are 1ms in length. The length of the pauses between the IR bursts is what contains the data. The first pause is 3ms and represents the start bit. All following pauses are either 1ms or 2ms, which represent logic 0 and logic 1 respectively. Each IR stream takes the following form:

<start bit><product ID><command data><checksum><stop bit>

The product ID is a three bit code and is 001 for Axiom Amplifiers.

The checksum is built using the command data: the sum of each byte in the command plus the checksum must equal zero. The stop bit is represented by a 15ms pause before there is any further IR activity. A receiver can however decide as soon as 4ms after IR activity has ceased that the packet is complete.

1.4. Ethernet

On Ethernet equipped devices, commands can also be sent via Ethernet to TCP port 17037. Each byte in the command is encoded into two ASCII encoded hexadecimal characters. A line feed character signals the end of the command. The carriage return character that is typically used in combination with the linefeed is optional and is ignored. A telnet session can be used for experimentation purposes.

1.5. Expansion bus

The expansion bus is based on the I²C bus and allows multiple devices to be attached to a common bus. All transmissions are performed using multi-master mode and take the following form:

<start bit><address byte><command data><stop bit>

The address byte is a requirement of the I²C bus. 7 bit address mode is used and the R/W bit is always 0 (always writing). Address 30h is used for the Axiom protocol (02h was used by older devices until this address was declared as being reserved). Other addresses are used such as 3Ah (for Ethernet packets) and 20h (for transmission preambles to ensure collision free access to the bus).

1.6. Ethernet expansion

The expansion bus is now obsolete and has been replaced by “expansion bus over Ethernet”. Commands are sent and received by multicasting to and from IP address 239.195.188.235 and UDP port 17037. The commands are in the same form as on TCP port 17037.

2. Command Protocol

Command structure: <command><zone>[<data>[<data>[...]]]

Command:

The command byte can have the following values:

Command (Hex Value)	Description
00	No Operation
01	Standby / Power
02	Mute
03	Source Selection
04	Volume
05	Bass
06	Treble
07	Balance
08	Request Protocol Version
09	Send All Parameters
0A	Report Error (obsolete)
0B	Emulate key press on Keypad (obsolete)
0C	Amplifier special features
0D	Maximum volume
0E	Preset Selection / Status (obsolete)
0F	Link zone pair (obsolete)
10	Media Favourites
11	Volume Up
12	Volume Down
13	Auto distributed source assignment advisory
14	Request Device information
15	Firmware update
16	Auto power on/off
17	Digital input/output options
18	Dynamic zone linking
19	Master volume
1A	Unused
1B	Preset Parameters
1C	Zone name
1D	Preamp volume mode
1E	Preset Selection / Status
1F	No longer in use
20	Preset Sound set up
21	Equalisation
22	Request device log entry
23	Preset alarm control
24	Request PCM capabilities
25	PCM Stream (obsolete)
26	Keypad port options
27	Set time zone, date and time
28	Video Source Selection
29	Source Name and Options
2A	Preset Name
2B	Request preset name
2C	Source Up
2D	Source Down
2E	Zone assignment
2F	Request zone assignments
30	Link zones

31	Audio delay
32	Source Gain
33	Unused
34	Clipping notification
35	IR routing assignments
36	Party mode select/deselect (obsolete)
37	Party mode configuration (obsolete)
38	Zone name request
39	Request extended device information
3A	Network settings
3B	Media Servers
3C	List Sources
3D	Media Control
3E	Media Status
3F	Media Status request
40	Report message
41	Request time
42	Settings management
43	Miscellaneous device settings and management
44	Zone gain
45	User accounts (obsolete)
46	Source specific metadata
47	Source specific metadata request
48	Power on volume level
49	Request keypad zone assignment
4A	Keypad port/zone mapping
4B	KPE key event
4C	KPE LED control
4D	Keypad port occupancy
4E	Arbitrary data storage
4F	Distributed Source Definition
50	Distributed Source Audio Delay
51	Register Service
52	Extended Media Control
53	Extended Media Status
54	Extended Media Status Request
55	Service Status
56	Source Mapping
57	Arbitrary and extensible command handling
58	Resolve address
60 – 6F	Reserved (undocumented commands)
70 – 7F	Unused
88	Response to Request Protocol Version
94	Response to Request Device information
A2	Response to Request device log entry
A4	Response to Request PCM capabilities
A5	Response to PCM Stream (obsolete)
AF	Response to Request zone assignments
B9	Response to Request extended device information
C1	Response to Request time
C9	Response to Request keypad zone assignment
CD	Response to Keypad port occupancy

The commands are also used as notifications. If an amplifier is switched on by pressing the power/standby button on the amplifier's front panel, it will notify the other devices by sending the standby/power command.

Zone:

This byte is used for checking if the command is applicable to the device receiving the command and if so, for optionally selecting a “sub-device”, e.g. a bank or part of a device.

The lower 5 bits of the zone byte equal **00h..1Fh** for zones 0 to 31. The upper 3 bits can represent a sub-zone or extended zones.

The sub-zone codes for the AX4750I and AX4752 amplifiers are as follows:

- 000 standard amplifier
- 001 page preset amplifier (obsolete) - add 20h
- 010 standard preamplifier (obsolete) - add 40h
- 011 page preset preamplifier (obsolete) - add 60h

Bits 7 is set for extended zones 32..95. Bits 5..7 represent three zone ranges:

- 000 common zone range (0..31)
- 100 extended zones 32..63
- 110 extended zones 64..95

Special zone values:

- **00h** = Zone 96.
- **FFh** = all zones. This could be used for setting all zones to a specific value, for example. However some commands are always sent to all zones.
- **FEh** = all local zones. The device receiving the command will respond as it would in the “all zones” case but only for its own zones.
- **FDh** = interface dependent zone. This represents the zone associated with the interface that the command is received on. Some interfaces such as keypad ports can have a zone associated with them and using this value allows the same command to be used for controlling different zones. If there is no zone associated with the interface then the command is ignored.
- **FCh** = unassigned zone. Represents when a zone should be automatically selected.
- **FBh** = disabled zone. Represents when an output on an amplifier is not used and should not appear in the system's zones.
- **FAh** = all used zones. Usable by commands relating to a service (ie. extended media player), where it indicates all zones using that service.
- **F0h** = AMM master/default zone. Used for communications with the AMM when a zone ID is required.
- **F1h** = AMM internal zone. For media manager device that is internalised in another product.
- **F2h – F4h** = AMM slave zones. For additional media manager devices.

Data:

Some commands use a unique 16 bit device ID, which is assigned to the device during manufacturing. The number of data bytes is dependent on the command. If a multi-byte parameter containing large numbers (e.g. the device ID) then the left most bytes in the string (the ones received first) are the most significant (big-endian format).

If fewer than the expected number of data bytes are received, the AX400/AX800 and later generation devices treat the command as a request and will return the requested data.

Command	Data field length	Content (Hex value)
Standby / Power (01)	1 byte	00h A Standby (power off) 01h A Power On 02h B Standby (obsolete) 03h B Power On (obsolete) 04h Toggle A Standby/Power On 05h Toggle B Standby/Power On (obsolete) 06h A and B Standby (power off) 07h A and B Power On
Mute (02)	1 byte	00h mute 01h unmute 02h toggle mute
Source Selection (03)	1 or 2 bytes	00h select S5 (CD) 01h select S6 (Tape) 02h select S7 (Tuner) 03h select S4 (Aux) 04h select S8 (Utility) 05h select S1 (SAT) 06h select S2 (DVD) 07h select S3 (Video) 08h..0Fh select S9..S16 10h Airplay 11h Reserved 12h Media Player 1 13h Media Player 2 20h..3Fh select Distributed source 1..32 Bit 6 is set if Video source is not to be switched, i.e. 40h select S5 (CD) audio only 41h select S6 (Tape) audio only 42h select S7 (Tuner) audio only 43h select S4 (Aux) audio only 44h select S8 (Utility) audio only 45h select S1 (SAT) audio only 46h select S2 (DVD) audio only 47h select S3 (Video) audio only 48h..4Fh select S9..S16 audio only Bit 7 is set to turn the zone on. A second byte may be reported when a source local to a zone on an amplifier is also a distributed source. In this case either the local source ID or the distributed source ID can be used on the zone so when an amplifier is reporting a source selection then it will include both sources.
Volume (04)	1 or 2 bytes	First byte (v1): 00h – A0h (0 – 160), represents 80 dB range in 0.5 dB steps. In practice only steps of 4 have been used resulting in 41 individual volumes in steps of 2 dB. Second optional byte (v2): 00h – C8h (0 – 200), represents 100 dB range in 0.5 dB steps. Currently steps of 2 are used resulting in 101 individual volumes in steps of 1 dB. v2 is an extension and not used by all amplifiers. If v2 provided then v1 = 00 when v2 in 00h – 17h, v1 = A0h when v2 in B4h – C8h, else v1 = floor((v2 – 20) / 4) * 4. If a command containing only v1 is received by a device that supports extended volume then it must calculate the v1 for its current volume setting and only apply the new v1 if different. The missing v2 is then determined by v2 = v1 + 20 when v1 > 0, otherwise v2 = 0.
Bass (05)	1 byte, signed	F4h – 0Ch (-12 – 12)
Treble (06)	1 byte, signed	F4h – 0Ch (-12 – 12)
Balance (07)	1 byte, signed	ECh – 14h (-20 – 20)

Request Protocol Version (08)	0 bytes in command, 1 byte in response	01h in response for the current version
Send All Parameters (09)	0 bytes	Avoid sending to all zones because this could generate an unreasonable amount of traffic, particularly on large amplifier stacks. This command is deprecated and does not support any of the newer settings. Try to use requests for specific parameters if possible.
Report Error (obsolete) (0A)	1 or 4 bytes	00h no error Amplifier errors: 01h no acknowledge 02h SCL line stuck low 03h SDA line stuck low 04h bus not free 05h overflow in register 06h overflow in receive buffer 07h receive aborted 08h write collision Keypad errors: 10h frequency error 11h too many different period lengths 12h IR function too long 13h compressed IR function too long 18h EEPROM comms error 19h EEPROM erase error 1Ah EEPROM full In case four bytes of data are transmitted, 2 nd byte is an extra information byte for the error, 3 rd and 4 th bytes are the device ID.
Emulate key press on Keypad (obsolete) (0B)	1 byte	Keypad key code in lower 5 bits, number of repeats in upper 3 bits. FFh means stop repeating the current key.
Amplifier special features (0C)	1 or 2 bytes	Bit 0: loudness enable (1) / disable (0) Bits 1..5 = reserved, set to 0. Bit 6: Sound played back on preset A. Obsolete, replaced by Preset Parameters . Bit 7: Sound playback enable supported when 1. Obsolete, replaced by Preset Parameters . <u>Second byte</u> Bit 0: mono (1) / stereo (0). Bit 1: mono / stereo control supported when 1. Bit 2: power on volume enable (1) / disable (0). See Power on volume level . Bit 3: power on volume supported when 1. Bit 4: low pass filter enable (1) / disable (0) Bit 5: low pass filter supported when 1. Bit 6..7: low pass filter cut off frequency, where 00 = 80Hz, 01 = 120Hz, 10 = 180Hz, 11 = 240Hz
Maximum volume (0D)	1 or 2 bytes	See data parameters in Volume .
Preset Selection / Status (obsolete) (0E)	1 byte	Bits 0-3 command the preset selection: 0 = default (preset input selects between standard and preset modes) 1 = select standard mode 3 = select preset 0 (paging preset) 4 = select preset 1 5 = select preset 2 6 = load preset 1 parameters

		<p>7 = load preset 2 parameters Bits 4-7: current preset status:</p> <p>0 = standard mode 1 = preset 0 2 = preset 1 3 = preset 2</p>
Link zone pair (obsolete) (0F)	1 or 2 bytes	<p>Zone to be linked to the command's zone. FFh for no zone linking. Otherwise 0..95 This command always links a pair of zones. It is not possible to link more than two zones together but there is no restriction on the number of linked zone pairs. The second byte specifies options for the zone linking:</p> <ul style="list-style-type: none"> • Bit 0 is set for source linking • Bit 1 is set for volume linking • Bit 2 is set for standby (power on/off) linking <p>If no second byte is received then only source tracking is the default. Volume linking also implies mute and Amp/Preamp volume tracking mode and offsets are also synchronised. Obsolete, use the Link zones command instead.</p>
Media Favourites (10)	3+ bytes	<p>Two bytes for the unit ID, one byte for favourite index, followed by strings for favourite name, service URL and image URL. Each string is null terminated. The favourite index begins at 0 and is FFh to request the list. Sent to all zones.</p> <p>Device type specifics and quirks:</p> <p>Amplifier</p> <ul style="list-style-type: none"> • No strings are listed for last entry in the list. This means that favourite index 254 (FEh) is unusable. The list must always be terminated by the no strings entry. • Entries must be sent in ascending order. Modifying an entry requires that all subsequent entries be sent also, else they will be removed. • The service URL can refer to an HTTP server for Internet radio or to a network share, in which case it must correspond to an entry in the media servers list (see Media Servers). <p>Media manager</p> <ul style="list-style-type: none"> • The favourite index is treated as an ID and does not update from changes to other favourite entries. • The list is read-only. • Requests can be made for specific entries by omitting the strings. • Responses do not include empty entries. • The service URL string is always empty.
Volume Up (11)	0 or 1 byte, signed	If there is no parameter or the parameter is 00h then the increment is a single step up in volume. Otherwise the parameter is the number of steps to increase volume. If the parameter is negative then the volume is decreased. Older amplifiers do not support this parameter.
Volume Down (12)	0 or 1 byte, signed	If there is no parameter or the parameter is 00h then the decrement is a single step down in volume. Otherwise the parameter is the number of steps to decrease volume. If the parameter is negative then the volume is increased. Older amplifiers do not support this parameter.
Auto distributed source assignment advisory (13)		<p><sender unit ID><host type><host unit ID><distributed source ID><source ID>[<distributed source ID><source ID>[.]]</p> <p>This advisory is sent when an amplifier automatically assigns sources hosted by other devices to its own distributed sources table. The host device may use this information for updating its own automatic assignments or to change responses to certain types of the List Sources command. The automatic distributed source assignment process does not use Distributed Source Definition commands since the automatic assignments are considered transitory. <sender unit ID> is the two byte unit ID of the amplifier that is sending the advisory. <host type> is a</p>

		<p>two byte type consisting of the type and subtype fields from the Request Device information response. <host unit ID> is the two byte unit ID of the device that hosts the source. <distributed source ID> is 20h..3Fh and where the host's source has been placed into the distributed source table (as far as the sender device is concerned). <source ID> is 00h..1Fh and is the source ID of the source on the host device. Multiple pairs of distributed source and source IDs can be sent. See Source Selection for source IDs.</p> <p>Sent to all zones.</p>
Request Device information (14)	0 or 1 bytes in command, 5+ bytes in response	<p>Request: if a data byte exists, bit 0 is set if devices should not reply on the expansion bus. This enables information retrieval of devices directly connected to a serial port, usb, etc. If bit 1 is set devices should only reply on the port that received that command. This prevents flooding the stack with responses. If bit 2 is set all zones on the unit responding are listed after the data.</p> <p>Response: First byte is the device type, the following are defined:</p> <ul style="list-style-type: none"> • 00h = amplifier • 01h, 02h = reserved • 03h = video matrix • 04h = media manager • 05h = virtual zone host <p>Second byte is the firmware version (major version number only).</p> <p>Third byte is device specific, the following codes are defined for Axiom amplifier types:</p> <ul style="list-style-type: none"> • 80h = AX4750 • 81h = AX4752 • 83h = AX-451/452-AV • 84h = AX-800DAV (original firmware branch) • 86h = AX-400DA (3 analogue inputs, original firmware branch) • 89h = AX-400DA (4 analogue inputs, original firmware branch) • 8Ah = AX-1250 • 8Bh = reserved for prototype • 8Fh = AX-Mini4 (variant 1) • 90h = AX-800-X (new firmware branch) • 91h = AX-400-X (3 analogue inputs, new firmware branch) • 92h = AX-400-X (4 analogue inputs, new firmware branch) • 96h = AX-Mini1 • 97h = AX-Mini4 (variant 2) • 99h = reserved for prototype • 9Ah = reserved for prototype • 9Ch = AX-Mini4 (variant 3) • 85h, 87h, 88h, 8Ch, 8Dh, 8Eh, 93h, 98h, 9Bh, 9Dh = reserved <p>Fourth & fifth bytes are a unique ID for the device (MSD first). If more than 5 bytes are returned, the remaining data are specific to the type of device. For the amplifier type this data is the optional zone list. For the media server type this data is a single flags byte indicated by the base value 0xC0. If bit 0 is set then there is a host device unit id following. If bit 1 is set then there is a zone list following. For the virtual zone host type this data is the two byte internal device type plus two other reserved bytes. Afterwards is the optional zone list.</p> <p>The zone list is added to the response if bit 2 is set in the options byte in the request. The zones are not encoded as they are in the zone byte and are in {00h...5Fh}. Also see Request extended device information.</p>
Firmware update (15)	3+ bytes	<p>First two bytes for the unit ID.</p> <p>Third byte:</p> <ul style="list-style-type: none"> • 00h: Downloading. Fourth byte is percentage (0..100). • 01h: Success • 02h: Failure. Following bytes is an error message. • 03h: Firmware update information (Axiom only). Bytes 4..7 are update version, bytes 8..11 are current version (major, minor, micro, beta),

		<p>remaining bytes are the URL.</p> <ul style="list-style-type: none"> 80h: Download from URL. Following bytes is the URL and must contain "http://" or "https://" to be accepted. Command not accepted on RS232 or IR ports. 81h: reserved for internal use. 82h: Cancel update. 83h: Request firmware update information (Axium only). <p>Sent to all zones.</p>
Auto power on/off (16)	4 or 6 bytes.	<p>This command is for configuring auto power on and off which enables zones to be turned on or off when audio is detected on their selected sources. First two bytes for the unit ID.</p> <p>Third byte is a bitfield of options, where</p> <ul style="list-style-type: none"> Bit 0: auto power on, 0 = disable, 1 = enable. Bit 1: auto standby, 0 = disable, 1 = enable. <p>Fourth byte is the auto standby time in 2ⁿ seconds, where n is the value of the byte. Optionally two bytes can follow which specify sources that can automatically activate the preset trigger 1 and 2 functions. The same source encoding is used as in Source Selection, FFh disables the preset trigger function. Sent to all zones.</p>
Digital input/output options (17)	2, 3 or 4 bytes.	<p>This command is for configuring options for the digital inputs/outputs, if supported. First two bytes for the unit ID. Third byte is a device specific ID representing the digital I/O port. Fourth byte is a device specific source ID for the digital I/O port. Sent to all zones.</p>
Dynamic zone linking (18)	1 or more bytes	<p>Dynamic zone linking differs from the fixed zone linking created by the Link zones command in that it is intended for use by end users whereas the fixed zone linking is for permanent links between zones. The command is not available on all amplifiers. The first data byte is the operation and is followed by a list of 0 or more zones, depending on the type of command. Operations are:</p> <ul style="list-style-type: none"> 80h = List zones in group request. Returns a response which lists all zones in group belonging to the zone following the operation or list all groups if no zone is specified. 00h = List zones in group response. Responses always list at least two zones. 81h = List all ungrouped zones request. 01h = List all ungrouped zones response. 82h = List zones in group with options request. 02h = List zones in group with options response. Similar to the 00h response except that each zone is followed by an options byte where bits 0..3 are the same as in the 1xh command explained below. Bits 4..7 are reserved and set to 0. 1Xh = Link all zones following the operation. The settings of the last zone in the list will be applied to the first zones in the list. The 4 bits specified by X have the following meaning: <ul style="list-style-type: none"> Bit 0: 1 = enable implicit removal from group on source change or on standby (if bit 1 = 0) and only applies to the first zone or zones in the list which assume the settings of the last zone ("follow zone" mode), 0 = explicit removal only. Bit 1: 1 = no power/standby linking, 0 = link power/standby. Bit 2: 1 = volume is unaffected by the Master volume command, 0 = enable normal volume tracking behaviour. Bit 3: reserved and set to 0. 20h = Remove zones in zone list from their groups. 21h = Remove zones in zone list from their groups by dissolving the groups entirely. 22h = Delete all zone linking groups. An extra data byte is expected which must equal C7h. <p>Unlike the Link zones command, clients are not expected to handle the dynamic zone linking and the amplifier will report changes to zone settings because of the zone linking. Sent to all zones.</p>

Master volume (19)	1 or 2 bytes	The master volume is the highest volume of a dynamically linked zone group. Setting the volume adjusts the volumes of all the zones in the group while maintaining their offsets with respect to each other, if possible. The zone in the command can be any zone in the group. The command causes Volume commands to be issued for each zone in the group. This command should therefore be used with some care so that excessive traffic is not generated. See data parameters in Volume .
Preset Parameters (1B)	12 - 25 bytes	<p>Like the preset select command, this command should be sent to all zones. The first byte is the preset to set up using the same constants as the preset select command, where preset A = 1, preset B = 2, ..., preset O = 15.</p> <p>Three bytes for the auto activation schedule in local time:</p> <ul style="list-style-type: none"> Hour (0..23) Minute (0..59) Weekdays (bit 0 = Sun, bit 1 = Mon, ..., bit 6 = Sat). Bit 7 is set to deactivate auto activation. <p>One byte for favourite selection to apply to any media sources, where 00=no favourite, 01..FFh = favourite index. Previously a reserved byte so only used if 25 or more data bytes used otherwise must be 00.</p> <p>3 bytes reserved, set to 00.</p> <p>4 bytes for zone selection: represents the first 32 zones.</p> <p>Optionally 3 more bytes for DND (do not disturb). First two bytes are the time window (time from, time to) where bits 2..6 represent the hour and bits 0..1 represent the minute in quarter hour resolution (minute in {00, 15, 30, 45}). The third byte represents the preset to substitute when inside the time window (1..15 for preset A..O and 0 = DND unused). DND is only used for presets selected by the preset triggers.</p> <p>Optionally 1 byte for flags:</p> <ul style="list-style-type: none"> Bits 0..1: preset sound selection (4 sounds) Bit 2: 1 = enable sound playback on preset activation by preset trigger activation, 0 = disable. Bit 3: 1 = enable sound playback on preset activation by Preset Selection / Status command, 0 = disable. Bit 4: 1 = enable sound looping, 0 = disable sound looping. Bit 5: reserved, set to 0. Bit 6: 0 = preset trigger contact normally closed, 1 = normally open. Ignore if bit 7 = 0. Bit 7: 1 = preset trigger contact close/open selection supported. <p>Optionally 8 bytes for zone 32..95 selection.</p> <p>Optionally 1 byte for more flags (read as 00h if not provided):</p> <ul style="list-style-type: none"> Bit 0: 1 = zone selection available for preset A, 0 = all zones are always selected for preset A. Bit 1: 1 = bits 2..4 valid, 0 = bits 2..4 ignored. Bit 2: 1 = preset is activated by trigger 1, 0 = not activated. Bit 3: 1 = preset is activated by trigger 2, 0 = not activated. Bit 4: 1 = preset settings loaded by trigger, 0 = restore settings after trigger.
Zone name (1C)	Unlimited	Data field contains the string. Character encoding is UTF-8. To request the zone name see Zone name request .
Preamp volume mode (1D)	1 byte	A signed number in the same units as the volume command. -128 (80h) represents independent mode, otherwise tracking mode is specified and the offset is equal to Preamp – Amp volume.
Preset Selection / Status (1E)	1 or 2 bytes	<p>Supports up to 15 presets (preset A .. O) and is intended to always be transmitted to all zones (zone byte should be FFh).</p> <p>1st byte commands the preset selection:</p> <p>00h = default (preset trigger input selects between standard and preset modes)</p> <p>01h = select preset A (formerly “page preset”)</p> <p>02h = select preset B (formerly “preset 1”)</p> <p>...</p>

		<p>0Fh = select preset 0 (formerly “preset 14”) Bit 7 1 = preset load function, which loads the preset settings but stays in non-preset mode, 0 = enter preset mode aka. “restore after trigger” mode. Bit 6 selects sound playback. Bit 5 locks all settings when set. Prevents inadvertent changes to the preset via front panel, IR or otherwise.</p> <p>2nd optional byte specifies the current preset status: 00h = standard mode 01h = preset A selected 02h = preset B selected ... FFh = unspecified</p>
Preset Sound set up (20)	1 byte in request. 7 bytes+ in response. 5 bytes+ in command.	<p>Request: <preset trigger 1..2> Response: <80h + preset trigger 1..2><2 byte unit ID><sound ID 0, 1..4><flags><list of zones + volume pairs> Command: <preset trigger 1..2><sound ID 0, 1..4><flags><list of zones + volume pairs></p> <ul style="list-style-type: none"> • preset trigger is 1 or 2. Preset trigger 1 uses preset 0 and preset trigger 2 uses preset 1. • sound ID is 1..4 or 0 for no sound. • flags: bit 0: 0 = contact normally closed, opened on activation (default and original behaviour), 1 = contact normally open and closed on activation (normal push button behaviour). • Zones are 0..95. • Volumes use the same encoding as the volume command. • The zone/volume pairs are required for both trigger 1 and 2 so that the volumes can be applied to the appropriate zones. The zones are excluded if not listed in the command. • The zone/volume pairs returned in responses only include the zones hosted by the responding amplifier. <p>Unsupported by Axium products. Sent to all zones.</p>
Equalisation (21)	0 or 1 byte in request. 28+ bytes in command.	<p>Request:</p> <ul style="list-style-type: none"> • [1 byte optional] Band/channel, bits 0..3 = 0..n-1 (band 1..n), bits 6..7: 0=both channels, 1=left channel or 2=right channel. If not specified then all bands are requested. If band specified and both channels requested then responses for each channel are sent when different, otherwise a single response for both channels is sent. <p>Response/Command:</p> <ul style="list-style-type: none"> • [1 byte] Band/channel, bits 0..3 = 0..n-1 (band 1..n), bits 6..7: 0=both channels, 1=left channel or 2=right channel. • [1 byte] Flags – bit 0: 0=Band disabled (bypassed), 1=Band enabled, bits 1..3: 0=band (peak/bell) filter, 1=low pass filter, 2=high pass filter, other values unused. • [1 byte, signed] Combined peak gain in dB (-100..100) • [2 bytes] Frequency (30..32000) • [1 byte, signed] Gain in dB/10 (-120..120) • [2 bytes] Q (100..10000). Filter Q factor 0.1-10. • [4 bytes, signed, 4.28 fixed point] a1 coefficient • [4 bytes, signed, 4.28 fixed point] a2 coefficient • [4 bytes, signed, 4.28 fixed point] b0 coefficient • [4 bytes, signed, 4.28 fixed point] b1 coefficient • [4 bytes, signed, 4.28 fixed point] b2 coefficient • [[20 bytes] a1, a2, b0, b1, b2 [...]] <p>n represents the maximum number of EQ bands supported by the amplifier. Biquad coefficients are calculated by an external application for the expected operating sample rate(s) and can be used to implement any type of filter. Multiple sets of coefficients are supplied if the amplifier type requires this such</p>

		as for supporting multiple operating sample rates. The combined peak gain of all bands is also calculated by the external application and is rounded up to the nearest whole number in dB. It is calculated across all bands and channels, therefore the same number is expected to appear in commands for each band and channel. The Frequency, Gain and Q parameters are only stored and not used by the amplifier.
Request device log entry (22)	3 - 5 bytes in request, 11 or more bytes in response when entry exists, 4 bytes when not.	<p>Request contains:</p> <ul style="list-style-type: none"> • Device ID (2 bytes) • Log entry ID low byte • Log entry ID high byte (optional) • Number of entries to return (optional) <p>Responses are returned with:</p> <ul style="list-style-type: none"> • Device ID (2 bytes) • Log entry ID low byte (1 byte) • Log entry type (1 byte), FFh for no entry • Hour (0..23) • Minute (0..59) • Second (0..59) • Weekday (0 = Sun, 1 = Mon, ... 6 = Sat). Lower 3 bits only. • Day (1..31) • Month (1..12) • Year (0..255, where 0 = 2000) <p>Further bytes may be sent as parameters. The number and format depend on the log entry type. Multiple responses are returned if the request asks for more than 1 log entry.</p>
Preset alarm control (23)	2 bytes	<p>Like the preset select command, this command should be sent to all zones. For historical reasons, the first byte uses the following values to select the preset:</p> <p>0Fh = preset A 00h = preset B 01h = preset C ...</p> <p>0Dh = preset O FFh = all presets</p> <p>The second byte has the following meanings:</p> <p>00h = disable alarm 01h = enable alarm 02h = sleep (re-activate alarm in 5 minutes)</p>
Request PCM capabilities (24)	3 bytes in request, 10 or more bytes in response	<p>Request:</p> <ul style="list-style-type: none"> • Device ID (2 bytes) • Purpose (1 byte) <p>Response:</p> <ul style="list-style-type: none"> • Device ID (2 bytes) • Purpose (1 byte) • Available memory (4 bytes) • Max packet size for PCM stream command (2 bytes) • Supported format (1 byte) • [Additional supported formats, optionally terminated with 00h] • PCM sound file size (only if formats terminated with 00h) (4 bytes)
PCM Stream (obsolete) (25)	8 bytes in request header, 7 bytes in response (3 bytes before AX-800DAV)	<p>Request:</p> <ul style="list-style-type: none"> • Device ID (2 bytes) • Purpose (1 byte) • Format in first packet, otherwise 00 (1 byte) • Length of stream in first packet, otherwise position in stream (4 bytes). • PCM data <p>Response:</p> <ul style="list-style-type: none"> • Device ID (2 bytes) • Purpose (1 byte) • Position in stream (4 bytes)

Keypad port options (26)	2+ bytes	<p>First two bytes for the unit ID. This is followed by a series of bytes which are the options for each keypad port. Each byte is a bitfield where:</p> <ul style="list-style-type: none"> • Bit 0: 0 = allow all zone commands received on the port, 1 = ignore all zone commands. • Bit 1: 0 = allow all zone commands received on the port to affect all zones, 1 = all zone commands received on the port only affects the port's assigned zone. • Bits 2..7 are reserved and should be set to 0. <p>If there are only two data bytes then the command acts as a request. Unsupported by Axiom products. Also see Keypad port/zone mapping.</p>
Set time zone, date and time (27)	12 or 19 bytes	<ul style="list-style-type: none"> • Standard time offset in minutes (2 bytes) • Daylight saving time (DST) offset in minutes (2 bytes) • DST to standard time changeover month (1..12) • DST to standard time changeover week (1 = first week in month, 2 = second week, 5 = last week in month) • DST to standard time changeover weekday (0 = Sun, 1 = Mon, ..., 6 = Sat) • DST to standard time changeover hour (0..23) • Standard time to DST changeover month • Standard time to DST changeover week • Standard time to DST changeover weekday • Standard time to DST changeover hour <p>If 19 or more bytes of data:</p> <ul style="list-style-type: none"> • Hour (0..23). Bit 7 set if in DST, bit 6 set if standard time. If neither bit is set, time is local time (either DST or standard time). • Minute (0..59) • Second (0..59) • Weekday (0 = Sun, 1 = Mon, ..., 6 = Sat). Lower 3 bits only. • Day (1..31) • Month (1..12) • Year (0..255, where 0 = 2000) <p>The offset values are 16 bit signed values in minutes (high byte first). The following rule applies: local time = UTC + offset If no daylight saving applies, all changeover parameters are 0. Changeover times are in the standard local time.</p>
Video Source Selection (28)	1 byte	<p>00h select S5 (CD) 01h select S6 (Tape) 02h select S7 (Tuner) 03h select S4 (Aux) 04h select S8 (Utility) 05h select S1 (SAT) 06h select S2 (DVD) 07h select S3 (Video) 08h..0Fh select S9..S16</p>
Source Name and Options (29)	Min 4 bytes for command, 0 or 1 bytes for request.	<p>First byte is the source ID using the same encoding as the Source Selection command. Bytes 2 – 4 represented a short name of up to three characters which was used for devices with limited display capabilities. These short names are no longer in use and so byte 2 should be set to 00h in which case bytes 3 and 4 are available for other functions. Presently byte 4 contains the following flags:</p> <ul style="list-style-type: none"> • Bit 0 indicates the “independent” source name state if set. • Bit 1 overrides independent / non-independent source name filtering if set. • Bit 2 disables the source if set. • Bit 3 indicates the source cannot be used as a distributed source if set. • Bit 4 indicates the source has locked independency if set. • Bit 5 indicates the undefined value for the device specific source data. When clear the undefined value is 00h, and when set it is FFh. • Bit 6 indicates the source has additional metadata that is available via the Source specific metadata command.

		<ul style="list-style-type: none"> Bit 7 indicates that the source is a proxy for a source provided by a secondary device. This has implications for distributed sources since the secondary device provides the audio stream. <p>Byte 3 is for device specific source data. Bytes 5 onwards specifies the source name. Character encoding is UTF-8. The request form of the command can request the names of all sources (no data bytes) or a single source (one data byte). If a request is sent to all zones then amplifiers will only send independent source names to one of the zones that they occupy as sending all names to all zones will generate excessive traffic.</p>
Preset Name (2A)	Min 1 byte	The first byte is the preset to set up using the same constants as the preset select command, where preset A = 1. Following bytes is the preset name. Character encoding is UTF-8.
Request preset name (2B)	1 byte	The preset to set up using the same constants as the preset select command, where preset A = 1. The response is the "Preset Name" command.
Source Up (2C)	1 byte	<p>Bits 0..1:</p> <p>00h = audio source up with video source sync 01h = audio source up 02h = video source up</p> <p>Bits 2..3:</p> <p>00h = cycle through all sources 01h = cycle through all local sources only 02h = cycle through all local physical sources only (no media players)</p> <p>Bit 7: indicates that the zone should turn on if set</p>
Source Down (2D)	1 byte	See Source Up command data.
Zone assignment (2E)	2+ bytes	Device ID followed by a list of zone numbers
Request zone assignments (2F)	0 or 2 bytes in request, <i>n</i> bytes in response	<p>Request: device ID of device to request the zones from. If no device ID is specified, the device receiving the command will answer and the command will not be relayed to another interface.</p> <p>Response: device ID followed by a list of zone numbers.</p>
Link zones (30)	2+ bytes	<p>Links a list of zones into a group. This is intended for configuring permanent zone links for audibly coupled areas or where multiple channels are used for a single area. For user defined zone linking, use the Dynamic zone linking command. This command should be sent to all zones (zone byte = Ffh).</p> <p>The first byte specifies the options for the zone link group:</p> <ul style="list-style-type: none"> Bit 0 is set for source linking Bit 1 is set for volume linking Bit 2 is set for standby (power on/off) linking Bit 3 is set for merging zones so user interfaces show only the linked zone group instead of the individual zones. Implies that the zone names are linked. Bit 7 is set if the list of zones is preceded by a 4 byte group identifier. Group identifiers are not enabled in all amplifier firmwares. Bit 6 is reserved for internal purposes. Bit 5 can be set in requests for suppressing responses for ungrouped zones. Sending 30FF20h will request only the grouped zones. <p>The remaining bytes list the zones in the group and are in {0..95}. Zones can be removed from the group by defining new groups which exclude the zone or by defining groups which contain only the zone to exclude. Similarly if a group contains {A, B, C, D} and a new group is defined containing {A, C} then {B, D} will remain in the original group. A zone cannot belong to more than one group. Zone groupings are not affected by presets. All clients are expected to support this command and update their corresponding states when linked zones change their settings.</p>
Audio delay	1+ byte	Output delay of audio stream in 5ms steps. 00h = no delay. Multiple data bytes

(31)		can be used for specifying the delay for each source where the first byte represents S1, then S2, etc. If fewer bytes are used than sources, the remaining sources are set to the last delay in the command. (This ensures compatibility with commands with a single data byte that applied to all sources.) No delay can be configured for the media player source.
Source Gain (32)	2 bytes or 2+ bytes	First byte is the source ID as defined by the Source Selection command (00h – 0Bh) or FFh in which case this is followed by gains for all sources beginning with S1. Second byte is the gain applied to the source 00h – 12h (0dB – +18dB). On the AX400/AX800 generation of amplifier the source gain setting affected the gain of the analogue input which could cause clipping if set too high. Later amplifiers adjust the output level when the source is selected and affects analogue and digital inputs.
Clipping notification (34)	2 bytes	First byte is type of clipping event. The second byte is information pertaining to the event. The following events are currently defined: <ul style="list-style-type: none"> 01h = analogue input clipped. The following byte represents the source that is clipped. 02h = analogue input unclipped.
IR routing assignments (35)	6+ bytes 3 bytes in request	First two bytes is the unit ID. Third byte is the IR source (keypads, bus run, expansion bus, etc) where the first keypad is 0. The bus run is 4 on the AX-400DA and 8 on the AX-800DAV and AX-1250. The expansion bus is 5 and 9 respectively on the AX-400DA and AX-800DAV. Fourth byte is the zone to monitor for source selection in {0..95} or FFh for none. Then for each source starting from S1, two bytes which represent a bitfield of IR output enables. IR output 1 is at bit 0, IR sum output follows bit for last output, expansion bus follows bit for IR sum output. If the IR source is the expansion port or bus run, the zone is ignored and only two bytes for the IR output enables is used. Sent to all zones.
Party mode select/deselect (obsolete) (36)	1 byte	00h = deselect party mode (selects standard mode) 01h = select party mode Sent to all zones.
Party mode configuration (obsolete) (37)	3 bytes	First two bytes is the unit ID of the party master. Third byte is the source selected for its party mode output. Sent to all zones.
Zone name request (38)	0 bytes	Requests the zone name. See also Zone name .
Request extended device information (39)	2 bytes in command, response length is device specific	Command: two bytes for the unit ID. Response: <ul style="list-style-type: none"> 4 bytes for 32 bit unit ID where bytes 3-4 match the standard 16 bit unit ID. 3 bytes for firmware version: major, minor and beta version (0 for release versions). 1 byte current temperature (signed). 1 byte peak temperature (signed). 4 bytes for IP address. 6 bytes for MAC address. 3 bytes manufacture date: year (0..255 = 2000..2255), month (1..12), day (1..31) 1 byte flags: <ul style="list-style-type: none"> bit 0: 1 = settings are locked, 0 = settings are not locked. bit 1: 1 = response was not broadcasted and is guaranteed to be from the device that was queried, 0 = response may be from any device in the system. Firmware micro version (optional) Also see Request Device information .
Network settings	Minimum 3 bytes	Two bytes for the unit ID, followed by setting ID (3 rd byte) which is followed by

(3A)		<p>the setting data. Setting IDs:</p> <ul style="list-style-type: none"> • 01h = Network name • 02h = Time server name • 03h = 1 byte flags where bit 0: 0 = DHCP, 1 = Static, bit 1: 0 = Time server disabled, 1 = Time server enabled, bit 2: KNX enabled (read only), bit 3: OvrC connected (read only and applicable only for OvrC enabled devices), bit 4: 1 = Vantage Ethernet Bus enabled, 0 = disabled (applicable only if Vantage Ethernet Bus supported and is optional). 16 bytes IP addresses (4 bytes per address): IP address, subnet mask, DNS server, default router. • 04h = IP address of KNX interface (4 bytes), KNX interface type (1 byte where 00 – disabled). • 05h = Device GUID (16 bytes). Words in GUID are stored in little endian format! • 06h = Amplifier stack assignment. 1 byte for the amplifier stack (0..255). Amp stack 0 is on UDP port 17037, stacks 1..255 are on UDP ports 17090..17344. • 07h = Reserved. • 08h = Airplay enable, 1 byte 0 = auto, 1 = enable, 2 = disable. • 09h = WiFi settings. 1 byte for routing information and in responses is set to the same byte set in the request. 1 byte for status flags: Bits 0-2 = signal strength bars, bit 3 = connected. 1 byte for the SSID length (0-32 bytes). Then the SSID of the specified length, and then finally a null terminated password. The password is excluded in responses to the 0x89h request. • 0Ah = WiFi access point list. 1 byte for routing information and in responses is set to the same byte set in the request. This is followed by list of SSID's. Each SSID is prefixed with 1 byte for status flags and 1 byte for the length in the same format as the WiFi settings above. • 0Bh = Network status. One byte for each network interface where the primary network interface (the one being used) is listed first. Bits 0..3: 0 = no network link, 1 = half duplex and/or 10 Mbit Ethernet, 2 = full duplex 100 Mbit Ethernet, 3 = WiFi up, 4 = WiFi down, 5..7 = reserved. • 81h = Request network name • 82h = Request time server name • 83h = Request IP addresses and flags • 84h = Request KNX IP BAOS 771/772 address. • 85h = Request device GUID. • 86h = Request amplifier stack assignment. • 87h = Reserved. • 88h = Request Airplay enable status. • 89h = Request WiFi settings. A routing information byte is expected and is copied in the response. • 8Ah = Request list of WiFi access points. A routing information byte is expected and is copied in the response. • 8Bh = Request network status. <p>Sent to all zones.</p>
Media Servers (3B)	3+ bytes	<p>Two bytes for the unit ID, one byte for list entry index, followed by strings for UNC path, user name and password. Each string is null terminated. No strings are listed for last entry in the list. If an alias name is used this is written into the UNC path field and is appended to the path with a colon separating it and the alias. The list entry index begins at 0 and is FFh to request the list. Entries must be sent in ascending order.</p> <p>Only supported by amplifier device types.</p>
List Sources (3C)	0 or 1 byte in request 1+ bytes in response	<p>Lists the available sources on the zone. Notifications are sent if the available sources change.</p> <p>The data byte represents flags for which sources to list. If the flags are specified then the first byte of the response is the flags byte (this is so the receiver knows what exactly the list of sources represents):</p> <p>Bit 0 = list non-independent local sources*.</p>

		<p>Bit 1 = list independent local sources*. Bit 2 = list distributed sources hosted by other devices*. Bit 3 = list distributed sources hosted by the device. Bit 4 = list distributed sources with local equivalent sources (must be used with bit 3 but not with bits 0 and 1). Bit 5 = include local sources even if disabled. Bit 7 must be set so flags in requests and responses can be distinguished from sources. *Option enabled by default when no flags byte is specified in request. A response that contains no sources should have FFh appended to it so that it is not confused with a request. Responses should always include the flags byte. See Source Selection for source IDs.</p>
Media Control (3D)	2+ bytes	<p>First byte is the source ID followed by play control command and command dependent data: 01h = Play: Optional 2 bytes for 16 bit track index (numbered from 0). 02h = Pause/Resume. 03h = Stop. 04h = Previous track. 05h = Next track. 06h = Repeat mode: Byte values: 0 = no repeat, 1 = repeat track, 2 = repeat all, 3 = cycle available repeat modes. 07h = Select. 2 bytes for the 16 bit item index. Selecting a track will play it. Selecting a folder will change to it. 08h = Shuffle mode: Byte values: 0 = no shuffle, 1 = shuffle tracks, 2 = toggle 09h = Open server. Normally not required since same as Select. 0Ah = Select favourite. Next byte is favourite index/ID (00h..FEh, see Media Favourites). 0Bh = Add currently playing content to favourites list. 0Ch = Remove currently playing content from favourites list. Next byte is the favourite index/ID. 0Dh = Closes the session and returns to the root menu.</p>
Media Status (3E)	2+ bytes	<p>First byte is the source ID followed by status parameter and data: 00h = Play status flags: 1 byte (1 = player available, 2 = loading directory, 4 = playback active, 8 = playback paused, 10h = reserved, 20h = repeat track, 40h = repeat disc, 80h = shuffle) 01h = Directory length: 2 bytes for directory count, 2 bytes for file count, 2 bytes for previous directory index (after going back a directory only, otherwise FFFFh) and 1 byte flags where: bit 0 = presence of “..” directory. 02h = Playing track index: 2 bytes for 16 bit index, numbered from 0. Always sent after the directory length. Only sent if playing or paused. 03h = Directory name: data is UTF-8 string. 04h = Directory item name: 2 bytes for 16 bit index followed by UTF-8 string. 05h = Playing track artist: data is UTF-8 string. 06h = Playing track album: data is UTF-8 string. 07h = Playing track title: data is UTF-8 string. 08h = Cover art file name. If there is no protocol specified, then the image is retrieved from http://<host>/artwork/<file name>. Only sent to network interfaces. 09h = Playing track position: 2 bytes for the value in seconds. 0Ah = Playing track length: 2 bytes for the value in seconds. Note that the track length can change if it is not available via metadata and it needs to be estimated using the file size and average byte rate. 0Bh = Buffer status in percent, FFh when not filling buffer. 0Ch = Bitfield of available functions: 1 or 2 bytes. First byte 01h = play, 02h = pause/resume, 04h = stop, 08h = prev track, 10h = next track, 20h = repeat track, 40h = repeat disc, 80h = shuffle. If a second byte is specified then 01h = favourites available. 0Dh = Favourite ID of currently playing content. FFh if not a favourite.</p>

		Automatic notifications for any changes to play status are sent to all zones on the unit implementing the source.
Media Status request (3F)	2+ bytes	First byte is the source ID followed by one or two bytes which is a bitfield of parameters to query (bit 0 = play status flags, bit 1 = directory length, etc). If the request is sent to all zones, the responses are sent to all zones using the source. Otherwise the responses are sent to the zone sending the request. If parameter 4 (directory item name) is selected, 4 bytes are required for start – end indexes after the parameter bitfield.
Report message (40)	2+ bytes	First byte is message type, followed by UTF-8 string. 00h = Information 01h = Error 02h = Success Sent to whichever zones and interfaces are relevant.
Request time (41)	1 byte in request, 7 bytes in response	Requests the time. 00h = Immediately, 01h = the next time it changes. The response contains the time in UTC if available: <ul style="list-style-type: none"> • Hours (0..23 decimal) • Minutes (0..59 decimal) • Seconds (0..59 decimal) • Bits 2:0: Weekday (0 = Sun, 1 = Mon, ..., 6 = Sat), bit 3: 1=UTC, 0=local time, bit 4: 1=DST, 0=standard time (applies only when bit 3 = 0). • Day (1..31) • Month (1..12) • Year (0..255, where 0 = 2000)
Settings management (42)	3+ bytes	Two bytes for the unit ID, followed by instruction: 00h = Query settings lock state. 01h = Lock settings. 02h = Unlock settings. If a command, this must be followed by 4 bytes for the password. If status is being reported only the instruction byte is sent. 03h = Restore default settings. Must be followed by two bytes 55h and AAh. 04h = Reboot the device. Must be followed by two bytes 55h and AAh. 05h = WiFi set up (for internal use only). Following byte gives the particular command or notification. Sent to all zones.
Miscellaneous device settings and management (43)	4+ bytes	Two bytes for the unit ID, followed by an Option byte (00h if unused), and then a Command byte. Any additional bytes are Command+Option specific. Commands: <ul style="list-style-type: none"> • 01h: delete PCM sound given by the option byte. • 02h: disable compression. Option byte is unused. For testing specific amplifiers only, setting is not stored permanently. • 03h: enable compression. See above. • 04h: set/report parameter specified by option byte. The following parameters exist: <ul style="list-style-type: none"> ◦ 01h: Service tag. Not used by all amplifiers. ◦ 02h: Ethernet Bus mode. Data is 00h..06h or FFh. ◦ 03h: Auto distributed source assignments enable. Data is 00h = disable, 01h = enable. Applies to all amplifiers so the Device ID is ignored when setting this. ◦ 04h: HDANYWHERE video switch settings. First data byte is switch type followed by source selections per video input and then amplifier zone selections per video output. ◦ 05h: Onkyo/Integra/Pioneer designated zone assignment. ◦ 06h: Onkyo/Integra/Pioneer designated source assignment. ◦ 07h: Onkyo/Integra/Pioneer available zones count (read-only). ◦ 08h: Onkyo/Integra/Pioneer available sources list (read-only). ◦ 09h: Onkyo/Integra/Pioneer preset name. First data byte is the preset number (1..40). Following bytes specify the preset name (UTF-8). ◦ 0Ah: Olson time zone id. eg. Pacific/Auckland ◦ 0Bh: Onkyo/Integra/Pioneer device priority and identifier (read-only).

		<ul style="list-style-type: none"> ◦ 0Ch: Volume ramp up speed. Data is 00h = instantaneous, otherwise milliseconds per dB (e.g. 255 = 25.5 seconds to ramp up 100 dB). ◦ 0Dh: Slave - master zone pairing. First data byte is slave zone, second is master zone, both numbered from 0. ◦ 11h - 13h: AMM voice service configuration (for internal use only). • 05h: check setting for conflicts result. The setting is specified by option byte. The following settings exist: <ul style="list-style-type: none"> ◦ 01h: Auto distributed source assignments enable status. Data is 00h = no conflict, 01h conflict. ◦ 02h: Distributed source definition. First data byte is the distributed source 00h..1Fh, followed by the conflict code: 00h = no conflict, 01h = the device does not define the distributed source but the request does, 02h = the device does define it but the setting doesn't, 03h = the definition differs in way of the device hosting it, 04h = the definition differs in way of the local source being distributed, 05h = the definition differs in way of the name. • 06h: Media manager role negotiation. • 82h: request compression enable/disable status. See 02h/03h above. • 84h: request parameter specified by option byte. See 04h above. • 85h: request a check for settings conflicts. See 05h above. This command is used to compare the given setting with the setting used by the device receiving the command. The data used is in the same format expected of the command responsible for the setting – ie for the following settings: <ul style="list-style-type: none"> ◦ 01h: Data is 00h = disable, 01h = enable ◦ 02h: See the Distributed Source Definition command format • 86h: Media manager role negotiation requests.
Zone gain (44)	1 byte	F4h – 0Ch (-12 dB – +12dB)
User accounts (obsolete) (45)	3+ bytes	<p>Two bytes for the unit ID, one byte for list entry index, followed by strings for user name and password. Each string is null terminated. 1 byte follows the strings which specifies the resource being protected. Currently only 01h is defined for the web application which uses the following further parameters.</p> <ul style="list-style-type: none"> • 12 bytes for zone enable (zone numbers 0..95). Bit 0 of the 12th byte corresponds to zone number 0 and bit 7 of the 1st byte corresponds to zone 95. Note that zone number 0 is zone 96. • 3 bytes for source enable (S1..S16, Media player). Bit 0 of the 3rd byte corresponds to S1 and bit 7 of the 2nd byte corresponds to S16. Bit 2 of the 1st byte is the media player, bits 0 and 1 are reserved. • 2 bytes for preset enable (presets B..O). Bit 0 of the 2nd byte corresponds to preset B. (Preset A is always enabled.) • 1 byte for other enable where bit 0 enables the Settings button. <p>No parameters follow the list entry index for the last entry in the list. The list entry index begins at 0 and is FFh to request the list. Entries must be sent in ascending order.</p>
Source specific metadata (46)	2+ bytes	First byte is source ID. Second byte is metadata entry index beginning at 0. Remaining bytes is the metadata in UTF-8 format. Support for this command and the number of supported metadata entries is device specific.
Source specific metadata request (47)	2 bytes	First byte is source ID. Second byte is metadata entry index beginning at 0. Support for this command and the number of supported metadata entries is device specific.
Power on volume level (48)	1 or 2 bytes	See data parameters in Volume . Use the Amplifier special features command to enable/disable this feature.
Request keypad zone assignment (49)	4 bytes in request, 5 bytes in response	Intended to be only sent by keypads connected to amplifier keypad ports. Data is a 4 byte unique identifier which is returned in the response followed by the zone (0..95 or FFh). If the amplifier has received the same unique identifier in a request from another keypad, the amplifier will not respond and the keypad must time out the request and retry using another identifier. Sent to all zones. Request not relayed to other interfaces and response is

		returned only on interface receiving the request.
Keypad port/zone mapping (4A)	2+ bytes	<p>Used for setting up the zones assigned by the Request keypad zone assignment command.</p> <p>First two bytes for the unit ID. This is followed by a series of bytes which are the zones for each keypad port and are 0..95 or FFh. If FFh, the corresponding output zone is used. For example if the amp is on zones 11 to 18 and the data bytes are "0B 0B 0B FF FF FF FF FF" then keypads on ports 1..4 are all assigned to zone 11 and keypads on ports 5..8 are assigned zone 15..18. If there are only two data bytes then the command acts as a request.</p> <p>Unsupported by Axium products. Also see Keypad port options.</p>
KPE key event (4B)	2 bytes	<p>Used for extending the built-in functionality of the KPE keypad and is reported on key press, long press and release events.</p> <p>First two bytes represent a key code where buttons 1..8 (top left to bottom right) are 01h..80h. The "set" button is 100h. Buttons can be combined in which case the key codes are ORed together. These codes are ORed with the following:</p> <ul style="list-style-type: none"> • 1000h = press event • 2000h = long press event • 4000h = release event after short press • 8000h = release event after long press <p>Events can be aggregated together if they occur in quick succession after one another. This command can be sent to the KPE without data for querying the current key.</p> <p>Unsupported by Axium products.</p>
KPE LED control (4C)	4 bytes	<p>Two bytes per LED group. The first byte of each group is a bitfield which selects if the LED is controlled by the KPE (0) or if the LED is controlled by commands (1). The second byte is a bitfield of the active LEDs. The first group is for the button LEDs and uses the same key codes as the KPE key event command. The second group is for the top row of LEDs where 01h is the left most LED and 80h is the right most LED. The command can be sent to the KPE without data for querying the LED status.</p> <p>Unsupported by Axium products.</p>
Keypad port occupancy (4D)	2 bytes in request, 3 bytes in response	<p>First two bytes for the unit ID. The 3rd byte of the response is a bitfield of keypad ports that are occupied by a keypad which sends Request keypad zone assignment commands. The response will be a few seconds after the request.</p> <p>Unsupported by Axium products.</p>
Arbitrary data storage (4E)	2+ bytes	<p>First two bytes provide an ID for the string of data in the range 0000h to 3FFFh. This is followed by the data to be stored (length restricted only by available RAM and storage space). If the MSB of the ID is set (i.e. ID is in the range 8000h to BFFFh) and there are no data bytes then the entry will be deleted. Note: storage space is limited so although there are 16384 possible IDs, this does not imply that 16384 strings can be stored. To reduce communications overhead, responses to requests are only returned on the interface that received the request.</p> <p>Unsupported by Axium products.</p>
Distributed Source Definition (4F)	0 or 1 bytes for requests. 2 or 18+ bytes in command.	<p>Requests:</p> <ul style="list-style-type: none"> • 0 data bytes: request for all distributed source definitions. • 1 data byte: request for a specific distributed source definition. <p>Commands/Responses:</p> <ul style="list-style-type: none"> • 2 data bytes (where second byte = 00h): unused distributed source. • 18+ data bytes: distributed source definition where first byte is 00h..1Fh for the distributed source (00h = distributed source 1). Bytes 2..17 is the device GUID (see Network settings) of the device hosting the source. Byte 18 is the physical source on the host device using device specific IDs. Remaining bytes is the UTF-8 name of the distributed source. <p>Commands must be sent to all zones, but requests may specify a target zone. Generally commands are responded to only on the connection the request was received on. However, if the request is for a specific zone, then response will also be sent to the amplifier stack.</p>

Distributed Source Audio Delay (50)	4+ bytes for command. 0 bytes for request.	First 4 bytes is a big endian bit field specifying which of the 32 distributed sources are specified by the command. This field will only contain valid distributed sources when generated by an amplifier. Remaining bytes specify the audio delay as defined by the Audio delay command.
Register Service (51)	3 or 7 bytes in command. 1+ bytes in request.	<p>Bytes in requests:</p> <ul style="list-style-type: none"> • Source ID – see Source Selection • Flags (optional) • 4 random bytes (used for auto-determined media player option) <p>Bytes in responses:</p> <ul style="list-style-type: none"> • Source ID • Flags • Service ID • 4 random bytes (when auto mode is requested) • The unit id of the AMM handling the registration (since FW 2.0) <p>The media player must send a register service command when initialising. Upon a successful registration, the service id will be anything but FFh, which is an error. A set of 4 random bytes may be included in a request which are to be used for distinguishing the results of a response when the automatic media player mode is used. The bytes are ignored if the auto flag is not set. The unit id is necessary for cases in which there are multiple AMM's present.</p> <p>Flags:</p> <p>Bit 7 is set if it is a response.</p> <p>Bit 6 is set in responses if the service is to unregister.</p> <p>Bit 5 is set in responses if the unit id is present.</p> <p>Bit 4 is set to power the zone upon successful registration.</p> <p>Bit 3 is set if media player events are to be sent directly to the connection this is received on.</p> <p>Bit 2 is set if media player events are to be sent by multicast to the amplifier stack port – see Ethernet expansion and Amplifier stack assignment.</p> <p>Bit 1 is set if the media player is to be controlled directly without any zone or source considerations.</p> <p>Bit 0 is set if the media player is to be auto-determined.</p> <p>If auto-determination is used, then the source in the response is the actual media player selected. If the zone is not already using the source, it will be selected. The random ID must be specified when auto-determination is used.</p>
Extended Media Control (52)	2+ bytes	<p>Extension of Media Control, but the service id is used instead of the source id. Additionally the following commands are modified as follows:</p> <p>07h = Select. Next two bytes is the 16 bit item index. Selecting a track will play it while replacing the playlist with the contents of the directory. Selecting a folder will change to it.</p> <p>0Dh = Navigate to root. Does not result in service closure since the extended media player supports media service navigation whilst playback is active.</p> <p>Additional commands:</p> <p>0Eh = Select directory item options. Similar to select but is instead used to show additional options for the selected item. 2 bytes for the 16 bit item index.</p> <p>10h = Select item in playlist. 2 bytes for the 16 bit item index.</p> <p>11h = Add item to playlist. 2 bytes for the 16 bit directory item index.</p> <p>12h = Remove item from playlist. 2 bytes for the 16 bit item index.</p> <p>13h = Clear playlist.</p> <p>14h = Replace playlist. 2 bytes for the 16 bit directory item index.</p> <p>15h = Add item to playlist and start playing it. 2 bytes for the 16 bit directory item index.</p> <p>16h = Apply rating to the currently playing content. Byte values: 0 = neutral, 1 = positive, 2 = negative.</p> <p>17h = Reserved.</p>

		<p>18h = Request service to load the next chunk of the browser listings.</p> <p>19h = Request service to load the next chunk of the playlist listings.</p>
Extended Media Status (53)	2+ bytes	<p>Extension of Media Status, but the service id is used instead of the source id. Additionally the following status parameters are modified as follows:</p> <p>01h = Directory length: 2 bytes for directory count, 2 bytes for file count, 2 bytes for previous directory index (after going back a directory only, otherwise FFFFh) and 1 byte flags where: bit 0 = presence of “..” directory, bit 1 = service supports partial results in which case the command has an extra 2 bytes for the total item count. After the flags is another byte for a loading id which may be used to detect content changes.</p> <p>02h = Playing track index: 2 data bytes for 16 bit index, numbered from 0. 1 byte for a loading id which may be used to detect content changes in cases where the index doesn’t change or isn’t available. Only sent if playing or paused.</p> <p>0Ch = Bitfield of available functions: 2 bytes where bit 0 = play, bit 1 = pause/resume, bit 2 = stop, bit 3 = prev track, bit 4 = next track, bit 5 = repeat track, bit 6 = repeat disc, bit 7 = shuffle, bit 8 = favourites, bit 9 = rating, bit 10 = playlist, bit 11 = can add to playlist, bit 12 = can remove from playlist, bit 13 = has more item options.</p> <p>Additional status parameters:</p> <p>0Eh = Content source details. Data comprises 3 null-terminated UTF-8 strings which are as follows: service name, service art, and station/playlist name.</p> <p>10h = Playlist length: 2 bytes for the item count and 1 byte flags where bit 0 is set if the service supports partial results in which case the command has an extra 2 bytes for the total item count. After the flags is another byte for a loading id which may be used to detect content changes.</p> <p>11h = Playlist item name. 2 bytes for 16 bit index followed by UTF-8 string.</p> <p>12h = Directory item art URL. Has 4 bytes for start and end indexes. End index is relevant since often the same art will be used for multiple subsequent items.</p> <p>13h = Playlist item art URL. Has 4 bytes for start and end indexes.</p> <p>14h = Playing track rating. Byte values: 0 = neutral, 1 = positive, 2 = negative.</p> <p>15h = Playing track playlist index. 2 bytes for 16 bit index.</p>
Extended Media Status Request (54)	4+ bytes	<p>Extension of Media Status request, but the service id is used instead of the source id. Additionally the command requires 3 or 4 flag bytes instead of 1 or 2.</p> <p>If any of the parameters 04h (directory item name), 11h (playlist item name), 12h (directory item art) or 13h (playlist item art) are included, then 4 additional bytes are required for the start and end indexes.</p>
Service Status (55)	2+ bytes	<p>Used as a notification for the status of a service. Bytes are the service id, message type/flags. Any remaining bytes are specific to the message type.</p> <p>Message Type/Flags:</p> <ul style="list-style-type: none"> • 01h for a service status update. Following bytes are the 2 byte unit id and then a byte indicating the status (00h = unavailable, 01h = unused, 02h = dormant, 03h = active) • 02h for a service membership update. Following bytes are the 2 byte unit id and then the list of zones that are registered to the service • 40h for a deprecated service membership update (similar to to 02h, but without the unit id) • Bit 7 is set to indicate a request (currently for 01h only) <p>Sent to all zones.</p>
Source Mapping (56)	4+ bytes	<p>Defines how the digital outputs of the AMM are used, i.e. the source and zone of an amplifier that they are connected to.</p> <p>First 2 bytes specify the unit id. Third byte is the digital output. A request may use FFh to indicate that all outputs are being queried. The fourth byte is the amplifier source in the usual encoded form. If FFh then no source mapping is set. Any remaining bytes are the zones for which the mapping applies.</p>

		Sent to all zones.
Arbitrary and extensible command handling (57)	4+ bytes	<p>The first byte indicates the Purpose and the second byte is the Command. Any additional bytes are Command+Purpose specific.</p> <p>The Purpose byte indicates what the nature of the command is so it can be interpreted and handled as appropriate (00h = no purpose, 01h = generic, 02h = onkyo/integra/pioneer).</p> <ul style="list-style-type: none"> Onkyo/Integra/Pioneer purpose: <ul style="list-style-type: none"> 01h: Tuner preset selection. Has up to additional 2 data bytes – the first is mandatory as an instruction (00h = current value, 01h = set parameter value, 02h = step up value, 03h = step down value), and an optional second parameter is the preset number (0 = no preset, 1 – 40 [01h – 28h] = specific preset). When the current value instruction is used, the omission of the parameter byte makes it a request. If the command is for a feature not supported by the zone, a response will be generated with the instruction byte set to FFh. 02h: Bass. Same format as above. The parameter is the bass level (-10 – 10 [F6h – 0Ah]). 03h: Treble. Same format as above. The parameter is the treble level (-10 – 10 [F6h – 0Ah]). 04h: Balance. Same format as above. The parameter is the balance level (-10 – 10 [F6h – 0Ah]). 05h: Listening mode. Same format as above. The parameter is the same value used as the "LMD" - Listening Mode Command in the ISCP documentation.
Resolve address (58)	1+ bytes	<p>The first byte indicates the type of address to be resolved and what it is to be resolved into. Types:</p> <p>00: A GUID is specified in the following 16 bytes and in the response the GUID is followed by a 6 byte MAC address and flags byte where bit 0: 0 = MAC address belongs to the device with GUID, 1 = MAC address belongs to a proxy.</p> <p>The response uses the same command number as the request but just returns the expected data.</p> <p>Sent to all zones.</p>

Notes:

- AX450I, AX4750I and all later amplifiers ignore standby/power B commands.
- 'Power On' commands imply that the zone is not muted if the zone was previously off. A mute command must follow the power on command if it should be muted.
- The Volume Up and Volume down commands increment or decrement the amplifiers volume at each command string reception. Therefore to achieve a volume ramp, the commands must be sent continuously for the duration of the ramp.
- All tracking modes require some intelligence on the part of the receiver. This means that a device transmitting a command does not need to account for all tracking modes and settings that it is perhaps not aware of. This is important because not all devices have sufficient memory to store all the settings for all zones. Furthermore, it means that devices only need to query the parameters for zones that they are interested in, thus reducing traffic when a device goes on line or is set to a zone.
 - Zone linking behaviour:
The "Link zone pair" command tells the zone specified in the zone byte "A" to follow the settings in the zone to be linked "B". However it also implicitly tells zone "B" to follow "A". Likewise if a link zone command is transmitted that either terminates the zone linking or tells either zones "A" or "B" to link to another, then both zones "A" and "B" are to stop tracking each other. Similar behaviour is expected when tracking the settings, if zone "A" changes its source, volume, mute, Preamp tracking mode/offset or standby settings, devices on zone "B" are expected to synchronise their internal settings and without sending notifications of any changes on zone "B". It is not the responsibility of the device on zone "A" that is making the change to inform zone "B" of any changes and it is not the responsibility of devices on zone "B" to inform of changes resulting from a change on zone "A".

The “Link zones” command tells all the zones in the command to be linked into one group.

Commands sent to any of the zones in the group cause all the zones in the group to implicitly track the change. Devices therefore need to know about all the zones in the group, even if they are otherwise not used by the device. In addition, devices reporting its linked zones using this command must be able to list zones in the group that are not used by the device.

- The “no operation” command is useful for resetting the receive status of the amplifier after long periods or when random signals could have been transmitted to it (e.g. when booting the computer controlling the amplifier). The zone and data byte should be 00h.