
JKP Documentation

Release 2.2

Jeff Koftinoff

February 09, 2011

CONTENTS

1	JKP Packet Grammar	1
2	Address Quadlet	3
3	ASCII Address Composition	5
4	Address Components	7
4.1	section	7
4.2	subsection	7
4.3	subsubsection	7
4.4	metaaddress	8
4.5	subaddress	8
4.6	item	10
5	Address Schema	11
5.1	Meta Messages	11
5.2	Device Identity	12
5.3	Media Source and Sink Formats Enumeration	14
5.4	Media Source and Sink Enumeration	15
5.5	Stream Source and Sink Formats Enumeration	16
5.6	Stream Source and Sink Enumeration	17
5.7	Connection of streams and media elements	18
5.8	Connection of talker and listener streams	18
5.9	Media Source and Sink Metering	19
5.10	Media Source and Sink Control	20
6	Examples	23

JKP PACKET GRAMMAR

A JKP Packet may contain one or more messages, in the following general ABNF form:

```
1 ; a jkp packet contains
2 jkp-packet = *message
3
4 ; a message is a 24 bit size in a quadlet, followed by an
5 ; address in a quadlet, with 0 to many parameters
6 ; depending on the specific address. All parameter
7 ; lengths must be in quadlets
8
9 message = size address [parameters]
10
11 ; an address is a quadlet
12 address = int32
13
14 ; generic parameters is one or more quadlets
15 parameters = 1*int32
16
17 ; the size field of a message is 24 bit value in a quadlet
18 size = int24
19
20 ; the various parameter types can be:
21
22 oui = int64
23 mac = int48
24 int16 = DOUBLET
25 int24 = NUL 3OCTET
26 int32 = QUADLET
27 int48 = 2NUL 6OCTET
28 int64 = OCTLET
29
30 ; strings are utf8 and NUL terminated and aligned to quadlets.
31 string = 4NUL
32 string =/ *(4utf8char) 4NUL
33 string =/ *(4utf8char) 1utf8char 3NUL
34 string =/ *(4utf8char) 2utf8char 2NUL
35 string =/ *(4utf8char) 3utf8char 1NUL
36
37 QUADLET = 4OCTET
38 DOUBLET = 2NUL 2OCTET
39 OCTLET = 8OCTET
40
41 ; a utf8char is 8 bits and does not include NUL
```

```
42 utf8char = %x01-%xFF
43 NUL = %x00
```

ADDRESS QUADLET

Every message contains a single address quadlet and the data values required for the address, if any.

A single packet may contain multiple values for multiple addresses.

Some addresses can be used as modifiers for the packet itself, for instance to specify a time to execute the commands, or to specify authentication tokens, or to specify required acknowledgment details.

ASCII ADDRESS COMPOSITION

On the “wire”, addresses are always 32 bit binary and every unique logical address in a device’s schema would be represented by a unique 32 bit pattern.

For readability for diagnostics and testing, the 32 bit binary address value can be mapped directly to or from an ASCII representation.

The ASCII representation of an address is in the form of concatenated strings and integer identifiers. There are 4 forms defined:

1. `/[section]/[metaaddress]`
2. `/[section]/[subsection]/[metaaddress]`
3. `/[section]/[subsection]/[subsubsection]/[subaddress]`
4. `/[section]/[subsection]/[subsubsection]/[item]/[subaddress]`

Where `section`, `subsection`, `subsubsection`, `subaddress`, and `metaaddress` are all string tokens representing an integer, and `item` is an integer value from 0-2046 inclusive represented as ASCII decimal.

Forms 1 and 2 are only used when `section` is `meta`, aka `016`.

Form 3 is used when `item` is `204710`, `2047` meaning “No item”.

Form 4 is used when `item` is between `010` and `204610` inclusive.

ADDRESS COMPONENTS

4.1 section

Bits 29-27 of the address specify the `section` that the address is in. The following values are defined and reserved:

Table 4.1: section codes

<code>meta</code>	<code>0₁₆</code>	Meta Messages
<code>device</code>	<code>1₁₆</code>	Device specific values
<code>enum</code>	<code>2₁₆</code>	Enumeration of properties and state
<code>connection</code>	<code>3₁₆</code>	Media and stream connection management
<code>control</code>	<code>4₁₆</code>	Media control
<code>snapshot</code>	<code>5₁₆</code>	State storage and recall
<code>ext</code>	<code>7₁₆</code>	Future expansion

4.2 subsection

Bits 26-24 of the address specify the `subsection` that the address is in. The following values are defined and reserved:

Table 4.2: subsection codes

<code>sys</code>	<code>0₁₆</code>	System oriented values
<code>talker</code>	<code>1₁₆</code>	AVTP talker values
<code>listener</code>	<code>2₁₆</code>	AVTP listener values
<code>controller</code>	<code>3₁₆</code>	AVTP controller values
<code>processing</code>	<code>4₁₆</code>	Processing values
<code>ext</code>	<code>7₁₆</code>	Future expansion

4.3 subsubsection

Bits 23-21 of the address specify the `subsubsection` that the address is in. The following values are defined and reserved:

Table 4.3: subsection codes

identity	0 ₁₆	Identity oriented values
media	1 ₁₆	Media oriented values
mediafmt	2 ₁₆	Media format values
stream	3 ₁₆	Stream oriented values
streamfmt	4 ₁₆	Stream format values
block	5 ₁₆	Processing block values
ext	7 ₁₆	Future expansion

4.4 metaaddress

When `section` is `meta`, then the bits 10-0 of the address specify the metaaddress. The following values are defined and reserved:

Table 4.4: metaaddress codes

time/actuate	30 ₁₆	Message modifier to specify 802.1AS time to actuate the message
request/id	70 ₁₆	Message modifier to request an acknowledgement of the packet with an identifier
request/status	71 ₁₆	Message modifier that contains the acknowledgement of a packet
io/report	90 ₁₆	Meta message to request values based on a address and address mask

4.5 subaddress

When `section` is not `meta`, then the bits 20-11 of the address specify the subaddress. The following values are defined and reserved:

Table 4.5: subaddress codes

id	0 ₁₆
count	1 ₁₆
active	2 ₁₆
health	3 ₁₆
type	4 ₁₆
capabilities	5 ₁₆
description	6 ₁₆
name	7 ₁₆
properties	8 ₁₆
format	9 ₁₆
map	A ₁₆
connect_tx	10 ₁₆
disconnect_tx	11 ₁₆
get_tx_state	12 ₁₆
get_tx_connection	13 ₁₆
connect_rx	14 ₁₆
disconnect_rx	15 ₁₆
get_rx_state	16 ₁₆
version	20 ₁₆
guid	21 ₁₆
vendor	22 ₁₆
vendoroui	23 ₁₆
Continued on next page	

Table 4.5 – continued from previous page

bootid	24 ₁₆
name	25 ₁₆
description	26 ₁₆
modelid	27 ₁₆
modelname	28 ₁₆
typeenum	29 ₁₆
typetxt	2A ₁₆
wink	2B ₁₆
signal	2C ₁₆
mac	2D ₁₆
gmid	2E ₁₆
talkercap	2F ₁₆
listenercap	30 ₁₆
controllercap	31 ₁₆
level	40 ₁₆
panpot	41 ₁₆
position	42 ₁₆
mute	43 ₁₆
invert	44 ₁₆
pad	45 ₁₆
scale	46 ₁₆
phantom	47 ₁₆
preamp	48 ₁₆
trim	49 ₁₆
wait	4A ₁₆
fade	4B ₁₆
pfl	4C ₁₆
afl	4D ₁₆
meter/map	52 ₁₆
meter/format	51 ₁₆
meter/values	50 ₁₆
eq	60 ₁₆
dynamics	61 ₁₆
send	62 ₁₆
busassign	63 ₁₆
buslevel	64 ₁₆
effect	65 ₁₆
matrix/level	66 ₁₆
matrix/mute	67 ₁₆
matrix/invert	68 ₁₆
matrix/eq	69 ₁₆
matrix/delay	6A ₁₆
playback/select	80 ₁₆
playback/mode	81 ₁₆
record/select	82 ₁₆
record/mode	83 ₁₆
timecode/mode	84 ₁₆
timecode/source	85 ₁₆
timecode/format	86 ₁₆
state/clear	90 ₁₆
state/store	91 ₁₆
state/recall	92 ₁₆
Continued on next page	

Table 4.5 – continued from previous page

state/current	93 ₁₆
persist/clear	A0 ₁₆
persist/store	A1 ₁₆
persist/recall	A2 ₁₆
persist/onboot	A3 ₁₆

4.6 item

Non-meta addresses in form 4 contain an 11 bit value in the 11 least significant bits of the address quadlet.

If all bits of the item are 1, ie. a value of 2047₁₀, then this means that the address is in form 3 and has no specific item.

ADDRESS SCHEMA

5.1 Meta Messages

All messages that start with “/meta/” modify the handling of the messages following them in the same packet.

5.1.1 /meta/time/actuate

Time to actuate messages.

Address Code: 80000030₁₆

Parameters: One octlet containing 64 bit PTP time..

5.1.2 /meta/request/id

Request identifier for request and related response.

Address Code: 80000070₁₆

Parameters: One quadlet.

5.1.3 /meta/request/status

Status response.

Address Code: 80000071₁₆

Parameters: One quadlet containing the status code.

5.1.4 /meta/io/report

Report values of address pattern.

Address Code: 80000090₁₆

Parameters: One quadlet containing the address mask, one quadlet containing the address to compare.
TBD: only specific address masks will be required to be supported.

5.2 Device Identity

Device identity properties

5.2.1 `/device/sys/identity/version`

1722.1 Protocol Version.

Address Code: 880107FF₁₆

Parameters: One quadlet containing version number.

5.2.2 `/device/sys/identity/guid`

Device's GUID.

Address Code: 88010FFF₁₆

Parameters: One octlet containing the device's EUI-64.

5.2.3 `/device/sys/identity/vendor`

Vendor's human readable name.

Address Code: 880117FF₁₆

Parameters: One UTF8 string.

5.2.4 `/device/sys/identity/vendoroui`

Vendor's OUI.

Address Code: 88011FFF₁₆

Parameters: One quadlet containing the vendor's assigned OUI.

5.2.5 `/device/sys/identity/bootid`

Boot identifier.

Address Code: 880127FF₁₆

Parameters: One octlet containing a different number upon every device start up.

5.2.6 `/device/sys/identity/name`

user settable device name.

Address Code: 88003FFF₁₆

Parameters: One UTF8 string.

5.2.7 /device/sys/identity/description

Human readable product description.

Address Code: 880037FF₁₆

Parameters: One UTF8 string.

5.2.8 /device/sys/identity/modelid

Model identifier.

Address Code: 88013FFF₁₆

Parameters: One octlet.

5.2.9 /device/sys/identity/modelname

Human readable device model name.

Address Code: 880147FF₁₆

Parameters: One UTF8 string.

5.2.10 /device/sys/identity/typeenum

Enumerated device types.

Address Code: 88014FFF₁₆

Parameters: One quadlet.

5.2.11 /device/sys/identity/typetxt

Human readable device type.

Address Code: 880157FF₁₆

Parameters: One UTF8 string.

5.2.12 /device/sys/identity/wink

Device notification wink request.

Address Code: 88015FFF₁₆

Parameters: None.

5.2.13 /device/sys/identity/signal

Device notification signalled.

Address Code: 880167FF₁₆

Parameters: None.

5.2.14 /device/sys/identity/mac

mac-addr on this device.

Address Code: 88016FFF₁₆

Parameters: One octlet for each 802.1AS capable network on the device.

5.2.15 /device/sys/identity/gmid

mac-addr of current 802.1AS grand master ID.

Address Code: 880177FF₁₆

Parameters: One octlet for each 802.1AS capable network port on the device.

5.2.16 /device/sys/identity/talkercap

Talker Capabilities.

Address Code: 88017FFF₁₆

Parameters: One bitmapped quadlet.

5.2.17 /device/sys/identity/listenercap

Listener Capabilities.

Address Code: 880187FF₁₆

Parameters: One bitmapped quadlet.

5.2.18 /device/sys/identity/controllercap

Controller Capabilities.

Address Code: 88018FFF₁₆

Parameters: One bitmapped quadlet.

5.3 Media Source and Sink Formats Enumeration

5.3.1 /enum/talker/mediafmt/count

Count of the number of different talker media formats supported.

Address Code: 91400FFF₁₆

Parameters: One Quadlet.

5.3.2 /enum/talker/mediafmt/#/properties

Media format properties.

Address Code: 91404000₁₆ - 914047FE₁₆

Parameters: One UTF8 String.

5.3.3 /enum/listener/mediafmt/count

Count of the number of listener media formats supported.

Address Code: 92400FFF₁₆

Parameters: One quadlet.

5.3.4 /enum/listener/mediafmt/#/properties

The properties of one listener media format.

Address Code: 92404000₁₆ - 924047FE₁₆

Parameters: One UTF8 String.

5.4 Media Source and Sink Enumeration

5.4.1 /enum/talker/media/count

Count of talker media sources.

Address Code: 91200FFF₁₆

Parameters: One quadlet.

5.4.2 /enum/talker/media/#/type

Enumeration of media source type.

Address Code: 91202000₁₆ - 912027FE₁₆

Parameters: One quadlet.

5.4.3 /enum/talker/media/#/format

Media format of a talker media source.

Address Code: 91204800₁₆ - 91204FFE₁₆

Parameters: One quadlet which refers to a talker mediafmt item.

5.4.4 /enum/listener/media/count

Count of listener media sinks.

Address Code: 92200FFF₁₆

Parameters: One quadlet.

5.4.5 /enum/listener/media/#/type

Enumeration of media sink type.

Address Code: 92202000₁₆ - 922027FE₁₆

Parameters: One quadlet.

5.4.6 /enum/listener/media/#/format

Media format of a listener media sink.

Address Code: 92204800₁₆ - 92204FFE₁₆

Parameters: One quadlet which refers to a listener mediafmt item.

5.5 Stream Source and Sink Formats Enumeration

5.5.1 /enum/talker/streamfmt/count

Count of the number of talker stream formats supported.

Address Code: 91800FFF₁₆

Parameters: One quadlet.

5.5.2 /enum/talker/streamfmt/#/properties

The properties of one talker stream format.

Address Code: 91804000₁₆ - 918047FE₁₆

Parameters: One UTF8 String.

5.5.3 /enum/listener/streamfmt/count

Count of the number of listener stream formats supported.

Address Code: 92800FFF₁₆

Parameters: One quadlet.

5.5.4 /enum/listener/streamfmt/#/properties

The properties of one listener stream format.

Address Code: 92804000₁₆ - 928047FE₁₆

Parameters: One UTF8 String.

5.6 Stream Source and Sink Enumeration

5.6.1 /enum/talker/stream/count

Count of the number of talker streams supported.

Address Code: 91600FFF₁₆

Parameters: One quadlet.

5.6.2 /enum/talker/stream/#/format

The format identifier for one talker stream.

Address Code: 91604800₁₆ - 91604FFE₁₆

Parameters: One quadlet.

5.6.3 /enum/talker/stream/#/properties

The stream properties of one talker stream.

Address Code: 91604000₁₆ - 916047FE₁₆

Parameters: TBD.

5.6.4 /enum/listener/stream/count

Count of the number of listener streams supported.

Address Code: 92600FFF₁₆

Parameters: One quadlet.

5.6.5 /enum/listener/stream/#/format

The format identifier of one listener stream.

Address Code: 92604800₁₆ - 92604FFE₁₆

Parameters: One quadlet.

5.6.6 /enum/listener/stream/#/properties

The stream properties of one listener stream.

Address Code: 92604000₁₆ - 926047FE₁₆

Parameters: TBD.

5.7 Connection of streams and media elements

5.7.1 /connection/talker/stream/#/map

Map of talker media sources to talker stream source elements.

Address Code: 99605000₁₆ - 996057FE₁₆

Parameters: One quadlet describing the talker media source id for each element in the stream.

5.7.2 /connection/listener/stream/#/map

Map of listener media sinks to talker stream sink elements.

Address Code: 9A605000₁₆ - 9A6057FE₁₆

Parameters: One quadlet describing the listener media sink id for each element in the stream.

5.8 Connection of talker and listener streams

Manage the connection of talker and listener streams via Simple Connection Management Protocol

5.8.1 /connection/talker/stream/#/connect_tx

SCMP.

Address Code: 99608000₁₆ - 996087FE₁₆

Parameters: TBD.

5.8.2 /connection/talker/stream/#/disconnect_tx

SCMP.

Address Code: 99608800₁₆ - 99608FFE₁₆

Parameters: TBD.

5.8.3 /connection/talker/stream/#/get_tx_connection

SCMP.

Address Code: 99609800₁₆ - 99609FFE₁₆

Parameters: TBD.

5.8.4 /connection/listener/stream/#/connect_rx

SCMP.

Address Code: 9A60A000₁₆ - 9A60A7FE₁₆

Parameters: TBD.

5.8.5 /connection/listener/stream/#/disconnect_rx

SCMP.

Address Code: 9A60A800₁₆ - 9A60AFFE₁₆

Parameters: TBD.

5.8.6 /connection/listener/stream/#/get_rx_state

SCMP.

Address Code: 9A60B000₁₆ - 9A60B7FE₁₆

Parameters: TBD.

5.9 Media Source and Sink Metering

5.9.1 /control/talker/media/meter/map

Mapping of talker media sources to meter slots.

Address Code: A12297FF₁₆

Parameters: One quadlet per meter slot. The most significant 16 bits contains the talker media source number, and the least significant 16 bits contains some media specific or platform specific media source sub-item, for instance the audio media that comprises a 5.1 media source.

5.9.2 /control/talker/media/meter/format

Meter format code for each meter slot.

Address Code: A1228FFF₁₆

Parameters: One quadlet for every four meter slots. A format code is an octet with the following possible values: 0 = none, 1 = device specific format, 2 = octet indicating signal presence only when non-zero, 3 = unsigned octet representing an audio meter dbFS value from -192dbFS to +63dbFS inclusive, 5 = unsigned octet representing the peak luma of a video signal.

5.9.3 /control/talker/media/meter/values

Talker media source meter slot values.

Address Code: A12287FF₁₆

Parameters: One quadlet for every four meter slots. Each meter slot is an octet whose format is determined by the associated meter format code.

5.9.4 /control/listener/media/meter/map

Mapping of listener media sinks to meter slots.

Address Code: A22297FF₁₆

Parameters: One quadlet per meter slot. The most significant 16 bits contains the listener media sink number, and the least significant 16 bits contains some media specific or platform specific media source sub-item, for instance the audio media that comprises a 5.1 media sink.

5.9.5 /control/listener/media/meter/format

Meter format code for each meter slot.

Address Code: A2228FFF₁₆

Parameters: One quadlet for every four meter slots. A format code is an octet with the following possible values: 0 = none, 1 = device specific format, 2 = octet indicating signal presence only when non-zero, 3 = unsigned octet representing an audio meter dbFS value from -192dbFS to +63dbFS inclusive, 5 = unsigned octet representing the peak luma of a video signal.

5.9.6 /control/listener/media/meter/values

Talker media source meter slot values.

Address Code: A22287FF₁₆

Parameters: One quadlet for every four meter slots. Each meter slot is an octet whose format is determined by the associated meter format code.

5.10 Media Source and Sink Control

5.10.1 /control/talker/media/#/level

Set media source level in db.

Address Code: A1220000₁₆ - A12207FE₁₆

Parameters: One quadlet describing the level in dB multiplied by 100, allowing for 0.01 db precision.

5.10.2 /control/talker/media/#/mute

Set media source mute.

Address Code: A1221800₁₆ - A1221FFE₁₆

Parameters: One quadlet describing the mute status, 0=not muted, 1=muted.

5.10.3 /control/listener/media/#/level

Set media sink level in db.

Address Code: A2220000₁₆ - A22207FE₁₆

Parameters: One quadlet describing the level in dB multiplied by 100, allowing for 0.01 db precision.

5.10.4 /control/listener/media/#/mute

Set media sink mute.

Address Code: A2221800₁₆ - A2221FFE₁₆

Parameters: One quadlet describing the mute status, 0=not muted, 1=muted.

EXAMPLES

TBD