



NUGGET

Programming Manual

Version 1.5

Compliant with Firmware Version 2.0.0-x and 5.0.0-x

(2 / 2011)

The information contained herein is confidential and may not be divulged to any person or entity or reproduced, disseminated or disclosed, in whole or in part. By receipt of this material including any exhibits, attachments and spreadsheets the recipient agrees that the information contained herein shall be kept confidential and shall not, without the prior written consent of Doremi Laboratories, be disclosed by the recipient in any manner.

Table of Contents

1	Control Protocol.....	4
1.1	Communication Format.....	4
1.2	Standard Command Block Format.....	5
1.3	Extended Command Block Format.....	6
1.4	Communications Protocol.....	6
1.5	Controlling the Nugget using the Serial Port.....	7
1.5.1	Wiring of the Standard RS422 Cable.....	8
1.5.2	Wiring of the NUGGET RS422-PC Cable.....	9
1.5.3	Wiring of the Nugget RS422-Mac Cable.....	10
1.6	Controlling the Nugget using the Ethernet Port.....	11
1.6.1	Using UDP for Control.....	11
1.6.2	Using TCP/IP for Control.....	12
1.6.2.1	Description.....	12
1.6.2.2	Simple messages.....	12
1.6.2.3	More complex messages.....	13
1.7	List of Commands.....	14
1.8	File Commands.....	20
1.9	Back to Back Playback.....	22
1.10	Retrieving the File List.....	22
1.11	Playlist Commands.....	23
2	Nugget Administration File.....	24
2.1	Global Machine Settings.....	24
2.2	Specific Channel Settings.....	25

Software License Agreement

Doremi's software license agreement is available at the following location:

<http://doremilabs.com/warrantiesLABS.html>

1 Control Protocol

THIS SECTION IS NOT FINISHED. IF YOU ARE A DEVELOPPER AND YOU NEED HELP

The Nugget can be controlled using the RS422 port or using TCP/IP. The Nugget uses the standard P2 protocol or otherwise known as the Sony 9-pin protocol for non file related functions and it uses the Odetics protocol to load and browse files.

In this protocol, the CONTROLLING DEVICE and CONTROLLED DEVICE are defined as follows:

The CONTROLLING DEVICE means the equipment that controls a DDR, VTR, etc. The CONTROLLED DEVICE means the equipment that is controlled like the Nugget.

1.1 Communication Format

Asynchronous, bit serial signal

- Full duplex communication channels
- Data signaling rate: 38.4kb/s (kbits per second)

The composition of bits is defined as follows.

1 START Bit + 8 DATA Bits + 1 PARITY Bit + 1 STOP Bit

Start Bit	D 0 (LSB)	D1	D2	D3	D4	D5	D6	D7	Parity (odd)	Stop Bit
-----------	-----------	----	----	----	----	----	----	----	--------------	----------

ODD PARITY: The total of logic "1"s in D0 to D7 and PARITY equals an odd number.

1.2 Standard Command Block Format

The data communication between the CONTROLLING DEVICE and the CONTROLLED DEVICE is performed in the following format:

CMD-1 (MSD)	DATA Count (LSD)	CMD-2	DATA-1	DATA-15 (MAX)	CHECKSUM
-- 1 Byte --	-- 1 Byte --	-- 1 Byte --	-- 1 Byte --	-- 1 Byte --	-- 1 Byte --

CMD-1 (4 BIT'S)

Indicates the value according to the function and direction of the command.

CMD-1 FUNCTION

- 0: SYSTEM CONTROL
- 1: RETURN TO 0, 2, 3, 4 OF CMD-1
- 2: TRANSPORT CONTROL
- 3: SYSTEM SPECIFIC
- 4: PRESET & SELECT CONTROL
- 6: SENSE REQUEST
- 7: SENSE RETURN

DIRECTION

- Master → Slave
- Master ← Slave
- Master → Slave
- Master → Slave
- Master → Slave
- Master → Slave
- Master → Slave
- Master ← Slave

Master = CONTROLLING DEVICE
Slave = CONTROLLED DEVICE

DATA COUNT (4 BIT'S)

Indicates the number of DATA bytes (max.15 bytes) inserted between CMD-2 and CHECKSUM.

Ex. DATA COUNT = 4

Which means one byte data is added respectively on the DATA-1, 2, 3 and 4. COMMAND block becomes as follows.

CMD-1	DATA Count	CMD-2	DATA-1	DATA-2	DATA-3	DATA-4	CHECKSUM
-------	------------	-------	--------	--------	--------	--------	----------

CMD-2 (1 BYTE)

Designates the command.

Refer to the "COMMAND TABLE" for definitions.

Ex. CMD-1=0, CMD-2=0C

This means a LOCAL DISABLE command.

DATA-1 to 15 (1 BYTE each)

Data bytes indicated by the command. Refer to the command table for data formats.

Ex. 41.30.DATA1: EDIT PRESET command

-- DATA COUNT

Since Data Count is "1", DATA-1 should be added after the command

DATA-1=58 means bits 6,4 and 3 are set to one. This is equivalent to insert Video only.

BIT-7	BIT-7	BIT-5	BIT-4	BIT-3	BIT-2	BIT-1	BIT-0
-------	-------	-------	-------	-------	-------	-------	-------

	INSERT	ASSEMBLE	VIDEO		TC	A2	A1
MSB							LSB

CHECKSUM (1 BYTE)

Lower eight bits of the sum of the bytes in the command block.

1.3 Extended Command Block Format

The standard command block allows only for up to 15 data bytes (Data Count coded on 4 bits), the Nugget supports the extended command block that allows up to 2 bytes for data count which means over 65000 data bytes are supported, but rarely any application would require more than 256 Data Bytes. This is the structure:

Ext. Indicator	CMD-1	Data Count LSD Byte	Data Count MSD Byte	CMD-2	DATA-1	DATA-255 (MAX)	CHECKSUM
-1Byte -		-1Byte -	-1Byte -			-1Byte -	-1Byte -

- **Extended Command Indicator (4 BIT'S):** This indicator is always the hex value E
- **CMD-1 (4 BIT'S):** CMD-1 has the same description as in the Standard Command Block format
- **Data Count LSD (1Byte):** This is the lower byte of the data count
- **Data Count MSD (1Byte):** This is the higher byte of the data count, set to 00 for data bytes count below 256.
- **CMD-2 (1Byte):** CMD-2 has the same description as in the Standard Command Block format
- **Checksum (1Byte):** CMD-2 has the same description as in the Standard Command Block format

To send the command 24.31.00.01.02.03.5b in extended mode, you need to send:
E2.04.00.31.00.01.02.03.1D

The extended command block is mostly used to get the long Name associated with an Odetics Clip ID.

1.4 Communications Protocol

The CONTROLLING device (master) should initiate the communication with the CONTROLLED device (slave). The slave should return a response within 9 msec. The response may be:

- NAK + Error Data: Undefined command or communication error
- COMMAND + Data: if Command requested data
- ACK: if Command did not request data

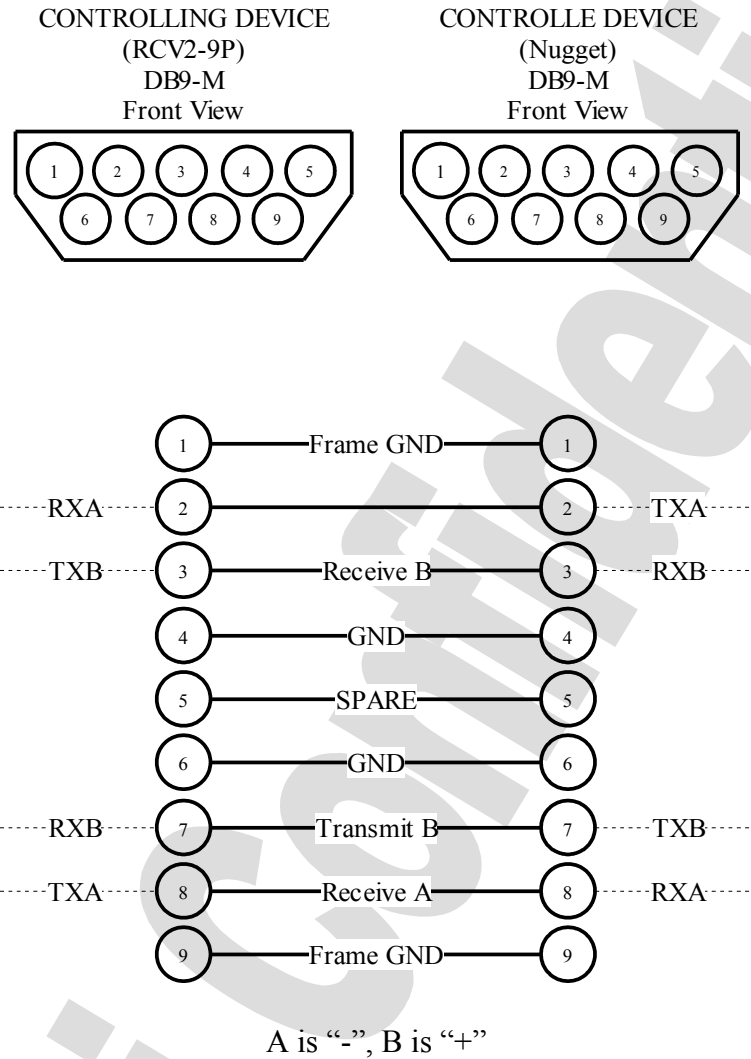
The master should not send another command until receiving a response from the slave device. The master must also insure that no more than 10 msec lapses between bytes in a command block. The master must immediately stop sending data when it receives a NAK + Error Data message. If the Error Data contains "Undefined Command" the master may immediately send another command, otherwise it must wait at least 10 msec before sending another command. When the master does not receive a response from the slave within the 10 msec timeout, it may assume that communications have ceased and take appropriate measures.

1.5 Controlling the Nugget using the Serial Port

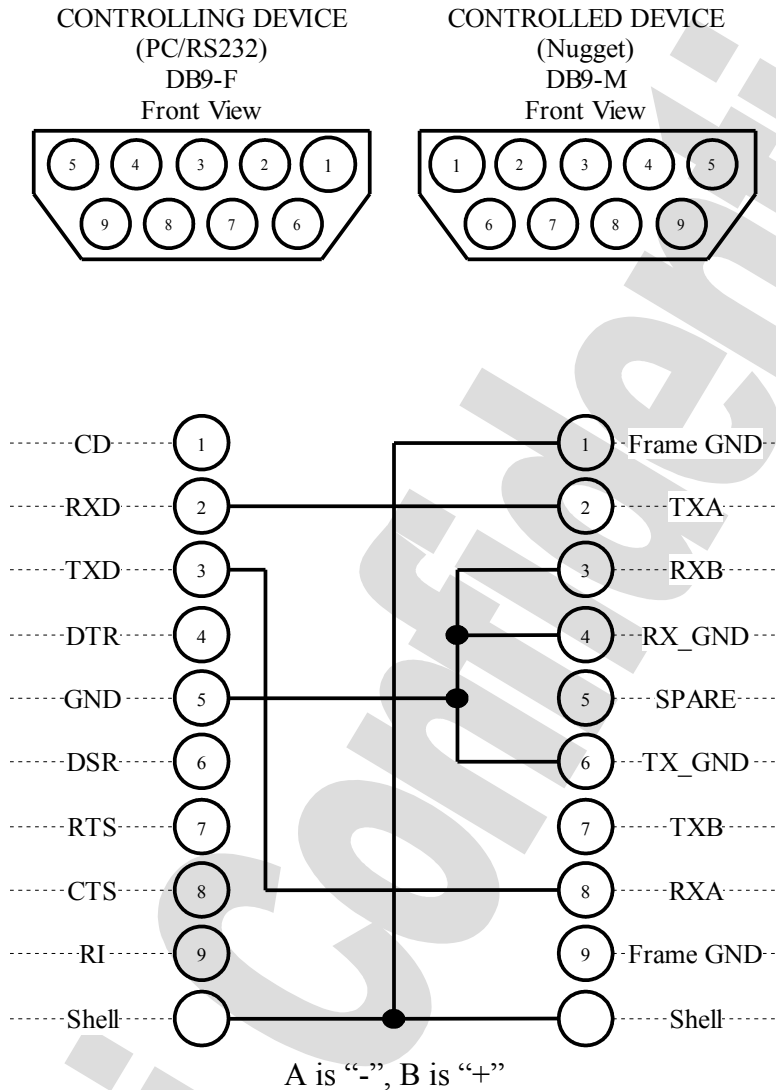
To control the nugget using the serial port, you need to have the proper RS422 channel.

- If you are using a controller with an RS422 serial port, use the standard RS422 cable.
- If you are using a PC with an RS232 port, use the RS422-PC cable
- If you are using a Mac with a mini-DIN8 RS422 cable, use the RS422-MAC cable

1.5.1 Wiring of the Standard RS422 Cable



1.5.2 Wiring of the NUGGET RS422-PC Cable

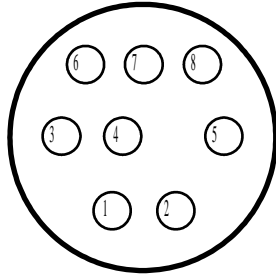


Wiring List: Nugget (1) to Nugget Shell to PC Shell
 Nugget (2) to PC (2)
 Nugget (8) to PC (3)
 Nugget (3 + 4 + 6) to PC (5)

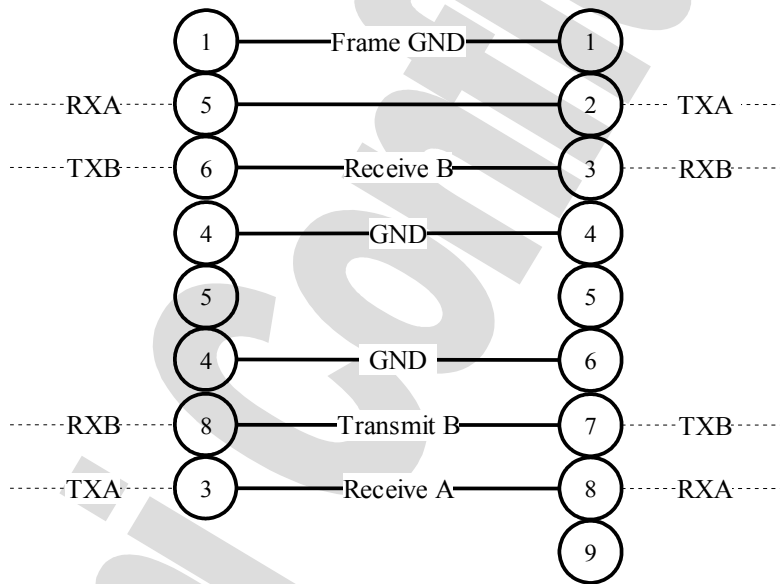
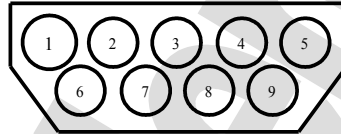
Note: For a true RS422 connection, you can use an adapter made by KK Systems (Part Number K422-99). This adapter connects to the RS232 port on the PC and provides an RS422 connection on the other side. To connect the K422-99 to the NUGGET, use a standard RS422 cable (See wiring of the standard RS422 cable). A true RS422 connection allows for a better connection and longer cables.

1.5.3 Wiring of the Nugget RS422-Mac Cable

CONTROLLING DEVICE
Mac DIN8-M
Front View



CONTROLLED DEVICE
(Nugget) DB9-M
Front View



A is “-”, B is “+”

1.6 Controlling the Nugget using the Ethernet Port

The NUGGET can handle IP based network communication using connectionless (UDP) or connection-oriented messages (TCP). The NUGGET uses a very simple protocol on top UDP and TCP.

Important note: The byte ordering in the structure is big endian: When filling or reading a data structure from a computer using little endian byte ordering (eg. Intel ix86), the programmer must manually swap the structure member.

This is a list of the ports used on the Nugget:

- TCP port 5000 is used to control the Nugget
- FTP port 21 is used to import and export settings files, playlists, etc..
- FTP port 21 is used to import media file to the Nugget.
- UDP port 0x8080 is used for sending/receiving "sony9p" or "Doremi" commands (Transports commands, retrieving clip list, ...).

1.6.1 Using UDP for Control

BYTE ORDERING IS BIG ENDIAN

The NUGGET receives UDP messages sent on port 0x8080, and replies on the same port. Every message should be accompanied with a 10 byte header which has the following format:

```
typedef struct {
    unsigned short type;
    unsigned short size;
    unsigned short msgType;
    unsigned short seqID;
    unsigned short seqNum;
} ComHdr;
```

the type field can have one of the values of the following enums :

```
enum {
    Sony9P_Protocol = 1,
    ComputerLink = 2,
    Mgmt_Protocol = 3
};
```

Example : To send a Sony 9 pin Play command, you need to construct the following :

0x0001, 0x0003, 0x0000, 0x0000, 0x0000, 0x20, 0x01, 0x21

The first word indicates that the message is a Sony9P_Protocol message, the second word indicates the size of the message excluding the 10 byte header, the following 3 words are "don't care". Then the play message 20.01.21

you will receive an ack

0x0001, 0x0003, 0x0000, 0x0000, 0x0000, 0x10, 0x01, 0x11

1.6.2 Using TCP/IP for Control

1.6.2.1 Description

At boot time, the NUGGET opens two listening socket on port 5000 (0x1388). Operations are initiated using a simple protocol that uses messages of 12 bytes structured as follows:

```
typedef struct
{
    int32 type;           // 32 bits
    int32 param1;        // 32 bits
    int32 param2;        // 32 bits
} cnxn_msg;
```

This structure is the header used for all TCP/IP communication, both requests and replies. However, the header might be followed by one or more bytes depending on the *type* of the message described below.

1.6.2.2 Simple messages

- **Nop** (No operation): This message does not do anything. It is basically used to prevent communications time out.
 - **Request message**
 - type* is set to the value 0x616c6976 ('aliv')
 - param1* is not used and should be set to zero
 - param2* is not used and should be set to zero
 - data* no data should follow the header
 - **Reply message**
 - There is no reply to this message
 - **Example**
 - Outgoing: 61 6c 69 76 00 00 00 00 00 00 00 00
 - Incoming: (none)
- **Version**: This message retrieves the protocol version used on the NUGGET. The present document describes protocol 1.0
 - **Request message**
 - type* is set to the value 0x76657220 ('ver')
 - param1* is not used and should be set to zero
 - param2* is not used and should be set to zero
 - data* no data should follow the header
 - **Reply message**
 - type* is set to the value 0x76657220 ('ver')
 - param1* contains the protocol version number
 - param2* contains the protocol revision number
 - data* none
 - **Example**
 - Outgoing: 76 65 72 20 00 00 00 00 00 00 00 00
 - Incoming: 76 65 72 20 00 00 00 01 00 00 00 00

- **Sony 9 pin protocol embedded message**
 - **Request message**

type is set to the value 0x73397020 ('s9p')

param1 is initialized with the sony 9 pin message length (checksum included).

param2 is not used and should be set to zero

data the sony 9 pin command should be immediately after the header.
 - **Reply message**

type is set to the value 0x73397020 ('s9p')

param1 is initialized with the sony 9 pin message length (checksum included).

param2 is not used

data the sony 9 pin reply follow immediately the header.
 - **Example**

Outgoing: 73 39 70 20 00 00 00 03 00 00 00 20 00 20

Incoming: 73 39 70 20 00 00 00 03 00 00 00 10 01 11

1.6.2.3 More complex messages

- **Get GOP:** This message is used to fetch the currently loaded frame from the Nugget. This command uses a 2 steps scheme. First the Nugget is notified of a getGop request, it acknowledges the request, then the application sends a getData message and the Nugget sends all the data.
 - **Request message**

type is set to the value 0x63676770 ('cggp')

param1 is not used and should be set to zero

param2 is not used and should be set to zero

data no data should follow the header
 - **Reply message**

type can be one of the following values:

Accepted 0x63676770 ('cggp'): The request is accepted, the application can send the getData message to get the GOP data. Param1 contains the size of the GOP the application can download.

Denied 0x45727220 ('Err'): The request is denied, the application **must not** send the getData message. A query is denied when the Nugget is not able to send a GOP – eg the disks are not ready, or not mounted.

param1 contains the GOP size when accepted, not used when denied

param2 not used

data none
 - **getData Request message**

type is set to the value 0x64617461 ('data')

param1 is not used and should be set to zero

param2 is not used and should be set to zero

data no data should follow the header
 - **getData Reply message**

There is no reply message. The GOP data is sent.
 - **Example request granted**

Outgoing: 63 67 67 70 00 00 00 00 00 00 00 00

Incoming: 63 67 67 70 00 02 10 00 00 00 00 00

Outgoing: 64 61 74 61 00 00 00 00 00 00 00 00

Incoming: xx xx xx xx [...] xx xx xx (GOP data)
 - **Example request denied**

Outgoing: 63 67 67 70 00 00 00 00 00 00 00 00

Incoming: 45 72 72 20 00 00 00 00 00 00 00 00

Outgoing: **the getDATA message should not be sent**

1.7 List of Commands

Command	Return	Comment
00.11 Device Type Request	12.11 Device Type Return	NUGGET Emulation: NTSC=A0.50, PAL=A1.50 BVW75: NTSC=20.25, PAL=21.25 DVW500: NTSC=B0.00, PAL=B1.00
10.01 ACK	10.01	ACK
Transport Commands		
20.00 STOP	10.01	Stop command
20.01 PLAY	10.01	Plays immediately from a stop, otherwise, it plays after 4 frames
20.02 REC	10.01	This command is kept for future recording capability using ASI input
20.0F Eject	10.01	No effect.
20.10 Fast Fwd	10.01	Speed is 20x.
20.11 JOG FWD at speed 0 (PAUSE)	10.01	Force the unit to still or pause
2X.13 SHUTTLE FWD at speed 0 (PAUSE)	10.01	Force the unit to still or pause
20.14 STEP FWD	10.01	Move on frame ahead.
20.90 Disable Chase	10.01	Disable Chase Mode
20.91 Enable Chase	10.01	Enable Chase Mode
24.31 CUE UP WITH DATA	10.01	Cues the slave to the indicated time. Time is formatted as follows: DATA1 DATA2 DATA3 DATA4 Frames Seconds minutes hours 10 1 10 1 10 1 10 1 This is how time is represented in all commands and responses using a time code. The numbers indicate that the 10s value is stored in the high nibble and the 1s value in the low nibble. This is not to be confused with the 80-bit SMPTE timecode or with the VITC timecode.
24.9D Goto Field/Frame (Depending on the Video Format)	10.01	Cues the slave to the indicated field or frame number specified in Hex. See Sense Video Format 60.AE to see if the command should be a Goto Field or Goto Frame DATA1 is Most Significant. 24.9D.00.00.00.00 locates to the start of the disk. 24.9D.00.00.00.20 locates to field number 32 ⇔ 00:00:00:16 in TC or to frame number 32 ⇔ 00:00:01:02 depending on the video format
System Specific		
34.09 Hardware Control	No return	DATA1.DATA4=41.53.41.50 (ASAP), RESET NUGGET. DATA1.DATA4=48.41.4c.54 (ASAP), Power Off DATA1.DATA4=53.54.4f.50 (ASAP), Software Stop

Preset Commands		
40.09 Video Capture	10.01	Captures the current video frame the Nugget's memory (UYVY format). Use in conjunction with 61.B2.10 and the GetGOP command
40.10 In Preset	10.01	Acknowledged but ignored
40.11 Out Preset	10.01	Acknowledged but ignored
40.14 In Preset with Data	10.01	Set the IN point of the open file to 00;00:00:00
44.14	10.01	Set the IN point of the open file to the timecode in DATA1..DATA4
48.14 or A8.14	10.01	Load a file defined by DATA1..DATA8
4C.14	10.01	Load a file defined by DATA5..DATA12 and set the IN point at timecode defined by DATA1..DATA4
40.15 Out Preset with Data	10.01	Set the OUT point to the current position
44.15	10.01	Set the OUT point to the timecode defined by DATA1..DATA4
40.20 In Flag Reset	10.01	Clears the In DATA Flag.
40.21 Out Flag Reset	10.01	Clears the Out DATA Flag.
40.30 Edit Preset	10.01	Acknowledged but ignored
41.33 Servo Ref Select	10.01	DATA1= 00 Auto Mode DATA1 = 01 Sync In DATA1 = Any other value Internal.
41.36 Timer Mode Select	10.01	Data1 = 00 Timecode Data1 = 01 A-Time
40.40 Auto Mode OFF	10.01	Sets Auto Mode OFF
40.41 Auto Mode ON	10.01	Sets Auto Mode ON and resets the In Data & Out Data Flags.
41.B7 Preset Loop Mode	10.01	Set loop mode ON (DATA1=FF) or OFF (DATA1=0)
41.BD.30	10.01	Save Settings. Save current settings as default.
44.B9.00 Set Video Startup	10.01	When DATA1=00, All 3 params can be set with one command. DATA2=Video Standard DATA3=Color Space DATA4=Aspect Ratio When DATA1=10, 11, 12 and 20 individual params can be set. AS indicates Auxiliary Settings.
42.BA Set Audio Startup	10.01	DATA1=10 Set Audio Rendering DATA2=0 OFF DATA2=FF ON DATA1=11 DATA2=0 OFF DATA2=01 Auto Detect DATA2=02 PCM DATA1=12 -127 < DATA2 < 127 Signed integer indicating audio delay
44.BB.00 Set Video Params	10.01	When DATA1=00, All 3 params can be set with one command. DATA2=Video Standard DATA3=Color Space DATA4=Aspect Ratio Set Video Standard Set Color Space Set Aspect ratio Set Auxiliary Settings
42.BB.10.VS		
42.BB.11.CS		
42.BB.12.AR		
42.BB.20.AS		
44.BC.00 Set Audio Params	10.01	DATA1=00, DATA2=Audio Rendering, DATA3=Audio SPDIF, DATA4= Audio Delay
42.BC.10	10.01	DATA1=10, DATA2=Audio Rendering ON or OFF
42.BC.11	10.01	DATA1=11, DATA2=Audio SPDIF OFF, AUTO_DETECT or PCM
42.BC.12	10.01	DATA1=11, DATA2=Audio Delay from -127 to 127
45.BD.01 Set Next IP	10.01	Set new IP address DATA2...DATA5 on next reboot
49.BD.01 Set Next IP/Mask	10.01	Set new IP address in DATA2...DATA5 and Subnet Mask in

45.BD.08.	10.01	DATA6...DATA9 on next reboot Set Default Gateway. Send "45.bd.08.0a.01.01.80" to configure the gateway to 10.1.1.128
45.BD.10 Set Time	10.01	Set clock to DATA2.....DATA5 seconds past 1/1/1970
41.BE. Subtitle	10.01	DATA1=00 Subtitle ON DATA1=01 Subtitle OFF
42.BE.02	10.01	DATA2 is the subtitle position number (TBD)
42.BE.03	10.01	DATA2 is the subtitle foreground color number (TBD)
42.BE.04	10.01	DATA2 is the subtitle background color number (TBD)
Sense Commands		
61.0C Current Time Sense	74.04 LTC Time 74.00 Timer-1 Time 74.04 LTC Time 74.04 LTC Time DATA1=Frames DATA2=Seconds DATA3=Minutes DATA4=Hours Special bits returned: DATA1-BIT7: Color Flag DATA1-BIT6: DF Flag DATA2-BIT7: F2 Flag NTSC DATA4-BIT7: F2 Flag PAL	DATA1 = 1 request LTC DATA1 = 4 request Timer-1 DATA1 = 2 request VITC DATA1 = 3 request any TC
60.10 In Data Sense	74.10 In Data Time	Requests the In Data Time
60.11 Out Data Sense	74.11 Out Data Time	Requests the Out Data Time
61.20 Status Request	7X.20 Status Data return bytes	Requests status byte: DATA1 (bits 7..4): First byte requested DATA1 (bits 3..0): Number of bytes requested
60.33 Servo reference Sense	71.33 Servo Ref byte same as Servo Ref Select byte.	DATA1=01 → Sync IN DATA1=06 → Internal
60.36 Timer Mode Sense	71.36 Returns same as Timer Mode Select byte.	DATA1=00 → Time Code DATA1=01 → A-Time
60.95 Version Sense	74.95 Version number string	Requests the machine Version Number. 4 ASCII characters are sent that represents the version.
61.95 Menu Sense	7X.95 Menu Line string	Return message is either line 1 or line 2 of the LCD display. Format of message is described in Menu Sense Table below.
60.A2 Sense Current Frame	74.A2 current Frame	Returns the current frame number
6X.A7 Sense NUGGET Status	7x.A7	Requests status byte: DATA1 (bits 7..4) First byte requested DATA1 (bits 3..0) Number of bytes requested Return format: DATA4..DATA1: Current Time Code, DATA5: Timer Mode Setting, DATA6..: Status Bytes starting at first byte specified in Sense command.
60.AA Sense Odetics	71.AA	DATA1 = 00 → Emulation is not Odetics DATA1 = 01 → Emulation is Odetics
6X.AC	74.AC	
60.AD Sense number of Audio Channels	71.AA.	DATA1=Number of audio tracks in loaded file
60.AE Sense Video Mode	71.AE	Returns the value for the Video Mode:

		Hex value	Video Mode	Goto Field/Frame command
		00	PAL	Field
		01	NTSC	Field
		02	1080-59.94i	Field
		03	1080-60i	Field
		04	1080-25p	Frame
		05	1080-50i	Field
		06	1080-23.98p	Frame
		07	1080-24p	Frame
		08	720-59.94p	Field
		09	720-60p	Field
		0A	1080-29.97p	Frame
		0B	1080-30p	Frame
		0C	1080-47.95i	Field
		0D	1080-48i	Field
		0E	480-30p	Frame
		0F	480-29.97p	Frame
		10	576-25p	Frame
		11	480-24p	Frame
		12	480-23.98p	Frame
		13	576-24p	Frame
		14	720-29.97p	Frame
		15	720-30p	Frame
		16	720-25p	Frame
		17	720-23.98p	Frame
		18	720-24p	Frame
		19	720-59.97i	Field
		1A	720-60i	Field
		1B	720-50i	Field
		1C	720-47.95i	Field
		1D	720-48i	Field
		1E	480-59.94p	Frame
		1F	480-60p	Frame
		20	2K-23.98p	Frame
		21	2K-24p	Frame
		22	2K-47.95i	Field
		23	2K-48i	Field
60.B0 Sense Hardware 61.B0.00 Sense Protocol Version	7X.B0	Doremi Use Only		
60.B1 Sense Product Name	78.B1 Unit Name	ASCII string with Unit Name		
60.B2 Sense Serial Number	76.B2	ASCII string with Serial Number		
61.B2.10 Sense captured frame resolution	7B.B2.10	The command works but what are the 10 bytes returned.		
60.B3 Sense IP Address	78.B3	DATA1..DATA8 = IP & Subnet in Hex.		
60.B4 Sense Frame 0	74.B4	For 60.B4 DAT1..DATA4 indicate the first frame index For 61.B4,		
60.B5 Sense Last Frame of current clip 68.B5 Sense Last Frame of any clip	74.B5	Return DATA1...DATA4= Last Frame in hex		
60.B7 Sense Loop Mode	71.B7	DATA1= 00 OFF, 01=ON		
61.B9.10 Sense Video Startup	72.B9.10	DATA2=Video Standard		

61.B9.11	72.B9.11	DATA2=Color Space
61.B9.12	72.B9.12	DATA2=Aspect ratio
61.B9.20	72,B9.20	DATA2=Aux
61.BA.10 Sense Audio Startup	72.BA.10	DATA2=Audio rendering
61.BA.11	72.BA.11	DATA2=SPDIF
61.BA.12	72.BA.12	DATA2=Delay
61.BB.00 Sense current Format	75.BB.00	00.VS.CS.AR (Video Std., Color Space, Aspect Ratio)
61.BB.01 Sense current Bitrate	75.BB.01	00.VS.BR.CH (Video Std., bitrate, Chroma)
61.BB.02 Sense current video Range (Duration in frames=Last-First)	79.BB.02	DATA2-DATA5 First Video Frame DATA6-DATA9 Last Video Frame
61.BB.10 Sense current video format	72.BB.10	DATA2= Video Standard
61.BB.11 Sense current color space	72.BB.11	DATA2= Color Space
61.BB.12 Sense current aspect ratio	72.BB.12	DATA2= Aspect Ratio
61.BB.20 Sense current aux settings	73.BB.20	DATA2=Current Aux DATA3= Next Aux (after Eject)
61.BC.00 Sense Audio Settings	74.BC	DATA1=00 DATA2=Audio Rendering DATA3=Audio SPDIF DATA4= Audio Delay
61.BD.01 Sense Current IP/Netmask	79.BD.01	DATA2...DATA5=IP address for next reboot DATA6...DATA9=Subnet Mask for next reboot
61.BD.02 Sense Next IP/Netmask	79.BD.02	DATA2...DATA5=Current IP address DATA6...DATA9=Current Subnet Mask
61.BD.10 Sense Current time	75.BD.10	DATA2...DATA5= Number of seconds since 01/01/1970
61.BD.13 Sense Serial Communication	75.BD.13	DATA2...DATA5= Number of milliseconds since the last serial port I/O
E6.09.00.BE.10 Sense Clip Name using Clip ID or 69.BE.10	E7...BE.10 or 7...BE.10	The data bytes carry the long name associated with an Odetics Clip ID.
61.C1.01 Sense Current ID	78.C1	DATA1..DATA8 = Current ID name
Odetics Commands		
A0.01 Auto Skip	10.01	Skip the current clip in a playlist.
A8.02 Record Cue Up	10.01	Load a file defined by DATA1..DATA8
A0.04 In Preview Preset	10.01	Sets the In point for the preview clip to 0
A4.04	10.01	Sets the In point for the preview clip to the timecode in DATA1..DATA4
A8.04	10.01	Sets the preview clip to file defined by DATA1..DATA8 with In point =0
AC.04	10.01	Sets the preview clip to file defined by DATA5..DATA12 with In point = timecode defined in DATA1..DATA4
A0.05 Out Preview Preset	10.01	Sets the Out point for the preview clip to 0
A4.05	10.01	Sets the Out point for the preview clip to the timecode in DATA1..DATA4
A0.06 In Preview Reset	10.01	Resets the In Preview flag to 0
A0.07 Out Preview Reset	10.01	Resets the Out Preview flag to 0
A8.10 Erase ID	10.01	Erase the file defined in DATA1..DATA8
A0.14 List First ID	88.14 80.14	Return the ID of the first file on the server Returns 80.14 if no files are present
A0.15 List Next ID	88.15 80.15	Return the ID of the next file on the server Returns 80.15 if no next file is present
A8.18 ID Status	81.18	Bit0 of DATA1 indicates that the ID specified is on the server Bit1 of DATA1 indicates that the ID specified is loaded
A0.1C	84.1C	Return the available time on the Nugget. (Not implemented and returns 0)
A0.21	88.21	DATA1..DATA8 represent the Odetics ID of the device.

		Nugget will reply with NUGGETV2 or 4E.55.47.47.45.54.56.32
--	--	--

Doremi Confidential

1.8 File Commands

Commands that are file specific are:

In Preset: 48.14 or 4C.14, use to load an existing file:

- **48.14:**
This command can be issued with a single ID parameter. If eight bytes of data are specified, the ID for the auto mode in preset will be set to the specified ID, and the time code position for the auto mode in preset will be set to the first field of video.
- **4C.14:**
This command can be issued with two parameters indicating the time code position and ID. If twelve bytes of data are specified, the first four bytes correspond to the time code position and the next eight bytes correspond to the ID.

Eject: 20.0F

- **20.0F**
This command places the video disk recorder in an IDLE mode where any "motion" command such as play, fast forward, record, etc., shall be aborted, and the currently loaded clip, if any, is unloaded. All preview presets shall be cleared and all preset status bits are cleared. If the NUGGET receives this command, the recording in progress shall be cleanly stopped. The clip is not deleted from the cache. When an eject command is issued, the stop status (status byte 1, bit 5) will be set high, and the rewind status (status byte 1, bit 3), fast forward status (status byte 1, bit 2), record status (status byte 1, bit 1), play status (status byte 1, bit 0), shuttle status (status byte 2, bit 5), jog status (status byte 2, bit 4), and variable play status (status byte 2, bit 3) shall be all set low.
The NUGGET will respond with an ACK (10.01)

Erase ID: A0.10 or A8.10

This command is used to erase and delete either a specified ID, or all existing IDs from the NUGGET storage. All time code positions for a deleted ID will be erased and deallocated. Note that the default ID (see section 3.2, IDs, page 12) can not be deleted. An attempt to delete the default ID will have no effect. The data provided for this command is variable.

- **A0.10**
Replicates the Initialize command described earlier in this document.
- **A8.10**
This command can be issued with a single ID parameter. If eight bytes of data are specified, the specified ID will be erased and deleted from the NUGGET storage. If the specified ID does not currently exist in the NUGGET storage, or if the default ID is specified, the erase ID command will have no effect.

List First ID: A0.14

- **A0.14**
This command will request the first ID in sorted order that currently exists in the NUGGET storage, returned using the ID listing response (88.14). This command will then advance the current listing position to the second ID in sorted order, so that a subsequent list next ID command will return the second ID.

List Next ID: A0.15

- **A0.15**
This command will request the next ID in sorted order that currently exists in the video disk recorder's storage, returned using the ID listing response (88.14). This command will then advance the current listing position to the next ID in sorted order, so that a subsequent list next ID command will return the next ID.

Sense current ID: 61.C1.01

- **61.C1.01**
The reply to this command is the name of the current ID.

Sense Time Left: A0.1C and A0.1D

- **A0.1C**
The reply to this command is 4 bytes representing the time left on the storage up to 99 hours, the format of the message is the same as Cue Up with Data.
- **A0.1D**
The reply to this command is 5 bytes representing the time left on the storage up to 9999 hours, the format of the message is the same as Cue Up with Data with DATA-5 being the 100th of hours byte. For 150h59m23s5f the reply would be 85.1d.05.23.59.50.01

Rename Current File: 38.11

- **38.11.4E.54.53.43.32.41.55.44**
This will rename the current file to "NTSC2AUD"

Sense Name associated with a Clip ID: E6.09.00.BE.10 or 69.BE.10

To Sense the long Name associated with a clip ID, you need to use the extended command block explained earlier in this manual. The command is E6.09.00.BE.10.[8 Bytes].CS, the 8 Bytes are the Clip ID. The reply to this command will carry the long Name.

1.9 Back to Back Playback

Auto mode is the mechanism by which advanced cueing information can be provided to the Nugget, thereby allowing continuous playback of video material.

When auto mode is enabled, the auto mode status (status byte 3, bit 7) will be set high. The in preset status (status byte 3, bit 0), out preset status (status byte 3, bit 1), preview in preset status (status byte 9, bit 0), and preview out preset status (status byte 9, bit 1) will be set low.

This is the sequence to run a seamless playlist:

- Auto Mode ON
- IN Preset (first clip), wait for the IN Preset flag to be set in the status register
- OUT Preset (first clip), wait for the OUT Preset flag to be set
- Preview IN Preset (next clip), wait for the IN Preview Preset flag
- Preview OUT Preset (next clip), wait for the OUT Preview Preset flag
- Play
- When the unit reaches the end of the current clip, it will seamlessly start the next clip by moving it from the preview clip to the current clip and clear the IN and OUT Preview flags.
- As soon as the Preview IN and OUT flags are cleared a new Preview IN and OUT preset commands can be sent to specify the next clip
- Etc....

A Stop command (20.00) will immediately abort a current auto play process, but will not reset the auto mode presets.

A Skip command (A0.01) will immediately abort auto play of the current video segment specified by the current auto mode preset, and perform the processing associated with the end of the current auto mode preset.

For more information about back to back playback, refer to the original Odetics manual provided by Odetics.

1.10 Retrieving the File List

Use the List First ID and List Next ID commands. The end of the list is reached when the List next ID command returns 80.15.

1.11 Playlist Commands

Commands that are playlist specific are:

Preset Auto Start: 49.BE.55, use to set an existing playlist to start automatically at boot.

- **49.BE.55.{ID - 8 bytes}:**
Set an existing playlist to start automatically at boot. If the playlist ID specified in the last 8 bytes doesn't refer to an existing playlist in the system, no playlist will be used at boot.

Clear Auto Start: 41.BE.55, use to clear the auto start playlist.

- **41.BE.55:**
This command does not require an ID, it will clear the playlist auto start.

Start Playlist: 49.BE.56, use to start a playlist immediately.

- **49.BE.56.{ID - 8 bytes}:**
Set an existing playlist to start immediately. If the playlist ID specified in the last 8 bytes doesn't refer to an existing playlist in the system, no playlist will start.

Delete Playlist: 49.BE.57, use to delete a playlist from the system.

- **49.BE.57.{ID - 8 bytes}:**
Delete an existing playlist specified by the last 8 bytes.

Browse First ID: 61.BE.50, use to browse the first playlist on the system.

- **61.BE.50:**
Similar to the file browse command, this command will return:

7D.BE.50.{index - 4 bytes}.{ID - 8 bytes}. ID is the first playlist in the system. The index (big endian) is used for list browsing.

Or

71.be.50 to indicate that there are no playlists in the system

Browse Next ID: 65.BE.51, use to browse the first playlist on the system.

- **65.BE.51.{index - 4 bytes}:**
This command should be followed by a 4 byte index.
Similar to the file browse command, this command will return:

7D.BE.51.{index2 - 4 bytes}.{OID - 8 bytes}
returns the OID of the next playlist in the system after the playlist entry index.

Or

71.BE.51 to indicate that the playlist specified in {index} is the last one in the system.

Sense Auto Start: 61.BE.55, use to sense the index of the auto start playlist.

- **61.BE.55:**
This command does not require an ID, it will return:
79.BE.55.{ID - 8 bytes}, where ID is the auto start playlist. If ID is all zeros, there is no auto start playlist specified.

2 Nugget Administration File

All Nugget units have administration files that define default parameters and certain behaviors.

- [RW] means Read / Write
- [RO] means Read only

All these files can be modified or created using an ftp client software. The login should be “admin” password is “1234” port must be set to “2121”. ONLY FOR ADVANCED USERS.

2.1 Global Machine Settings

[RW] /doremi/etc/odeticsd.auto_start:

- This file is empty. It's presence forces the Nugget to run with the odetics layer EXCEPT if the odetics.no_auto_start file is present (see below)

[RW] /doremi/etc/odeticsd.no_auto_start

- If this file is present it will deactivate the odetics layer

[RO] /doremi/etc/current_ip_address

- Contains current nugget IP address

[RO] /doremi/etc/current_ip_netmask

- Contains current nugget IP netmask

[RW] /doremi/etc/next_ip_address

- New IP settings that will be used for next reboot
- Format: IP/Netmask
- Example: 192.168.100.191/255.255.255.0

[RW] /doremi/etc/debug_log

- Log selection:
 - 0 for quiet production
 - 1 for all debugging info

[RW] /doremi/etc/debug_level

- Log verbosity -
 - 0 for production
 - 500 for debugging/evaluation session

[RW] /doremi/etc/cinelink

- Startup HD/SDI encryption
 - 0: no encryption
 - 1: encryption
- Default: 0
- Note: the cinelink global option must be enabled to set encryption on.

[RW] /doremi/etc/sync

- Set the startup sync method
- Format: Integer that represent sync method
 - SYNC_METHOD_AUTO = 0
 - SYNC_METHOD_HOUSE = 1,
 - SYNC_METHOD_INTERNAL = any value different from 0 or 1,

[RW] /doremi/etc/genlock_horizontal_offset

- Set the genlock horizontal offset
- Format: Signed Integer
- default: 0
-

[RW] /doremi/etc/genlock_vertical_offset

- Set the genlock vertical offset
- Format: Signed Integer
- default: 0

2.2 Specific Channel Settings

- Only one channel for now labeled 0.

[RW] /doremi/etc/0/startup_playlist:

If this file is present and contain a valid playlist filename present in the /data directory then the playlist is executed. To run a playlist automatically everytime you turn the Nugget unit ON:

- Use ListMaker to create a playlist on the Nugget and save the playlist file as list_1.vcf (for example)
- Use an ftp client software to transfer the list_1.vcf file to the Nugget's /data directory
- Modify the startup_playlist file with a text editor to show only list_1.vcf
- At this point you can reboot the Nugget for automatic playback of the list, but if you want to test before you reboot, connect to the Nugget using telnet and login as root.
- Enter /doremi/bin/drmplaylistd.out /data/list_1.vcf . There will be a 10 seconds delay before the playlist actually starts. This is done on purpose to make sure the nugget had time to start and display the colorbar before running the playlist.

If the software can not detect a running odetics layer, it will just abort
The playlist can be stopped at any time by issuing a Stop command.

[RW] /doremi/etc/0/show_colorbar

- Show colorbar at startup
 - 0: off
 - 1: on
- Default: 1

[RW] /doremi/etc/0/loop_mode

- Set the loop_mode on/off on startup
 - 0: off
 - 1: on
- Default: 0

[RW] /doremi/etc/0/timer_mode

- Set the initial timer mode
 - TIMER_TIME_CODE = 0
 - TIMER_ATIME = 1
- Default: 0 (Timecode)

[RW] /doremi/etc/0/screen_saver

- Set the time elapsed in second before turning the screen saver on
- Default: 0 (Off)
- **NOT IMPLEMENTED**

[RW] /doremi/etc/0/control_mode

- Set the initial control_mode remote/local
- Default: 0
- **NOT IMPLEMENTED**

[RW] /doremi/etc/0/video_format

- Set the initial video format
 - VIDEO_STD_PAL 0, 0x00
 - VIDEO_STD_NTSC 1, 0x01
 - VIDEO_STD_1080_59_94i 2, 0x02
 - VIDEO_STD_1080_60i 3, 0x03
 - VIDEO_STD_1080_25p 4, 0x04
 - VIDEO_STD_1080_50i 5, 0x05
 - VIDEO_STD_1080_23_98p 6, 0x06
 - VIDEO_STD_1080_24p 0x07
 - VIDEO_STD_720_59_94p 8, 0x08
 - VIDEO_STD_720_60p 9, 0x09
 - VIDEO_STD_1080_29_97p 10, 0x0a
 - VIDEO_STD_1080_30p 11, 0x0b
 - VIDEO_STD_1080_47_95i 12, 0x0c
 - VIDEO_STD_1080_48i 13, 0x0d
 - VIDEO_STD_480_30 14, 0x0e
 - VIDEO_STD_480_29_97p 15, 0x0f
 - VIDEO_STD_576_25p 16, 0x10

- o VIDEO_STD_480_24 17, 0x11
- o VIDEO_STD_480_23_98p 18, 0x12
- o VIDEO_STD_576_24 19, 0x13
- o VIDEO_STD_720_29_97p 20, 0x14
- o VIDEO_STD_720_30 21, 0x15
- o VIDEO_STD_720_25p 22, 0x16
- o VIDEO_STD_720_23_98p 23, 0x17
- o VIDEO_STD_720_24p 24, 0x18
- o VIDEO_STD_720_59_94i 25, 0x19
- o VIDEO_STD_720_60i 26, 0x1A
- o VIDEO_STD_720_50i 27, 0x1B
- o VIDEO_STD_720_47_95i 28, 0x1C
- o VIDEO_STD_720_48i 29, 0x1D
- o VIDEO_STD_480_59_94p 30, 0x1E
- o VIDEO_STD_480_60p 31, 0x1F
- o VIDEO_STD_2K_23_98p 32, 0x20
- o VIDEO_STD_2K_24p 33, 0x21
- o VIDEO_STD_2K_47_95i 34, 0x22
- o VIDEO_STD_2K_48i 35, 0x23

- Default: 2 (1080i59.97 for USA) or 5 (1080i50 for Europe)

[RW] /doremi/etc/0/video_aspect_ratio

- Set the initial aspect_ratio
 - o CORRECT_FIT = 0, Source = screen aspect ratio
 - o LETTERBOX = 1, Includes sidebar
 - o SIDEBAR_OVERSCAN = 2, Reduced sidebar
 - o PAN_AND_SCAN = 3, Includes tilt & scan
 - o ANAMORPHIC = 4, Unequal x/y conversion - Use display_width, display_height
 - o PARTIAL_LETTERBOX = 5, Combination of letterbox and pan & scan
 - o NONUNIFORM_SCALE = 6, Scale edges more than center
 - o GLASS_PRISM = 7, Stretch outside pixels to edges of screen
 - o PRE_LETTERBOXED = 8, Source image is narrow screen letterbox or wide screen sidebar
 - o PRE_PARTIAL_LETTERBOXED = 9, Source image is narrow screen letterbox or wide screen sidebar
 - o CINEMASCOPE = 10, 2.35:1
 - o WATERGLASS = 11, Scale edges more than center smoothly
 - o BML_HD_16_9 = 12, ARIB BML High definition wide
 - o BML_HD_4_3 = 13, ARIB BML High definition narrow
 - o BML_SD_16_9 = 14, ARIB BML Standard definition wide
 - o BML_SD_4_3 = 15, ARIB BML Standard definition narrow
 - o ANAMORPHIC_CROP = 16, Similar to ANAMORPHIC
 - o ANAMORPHIC_NO_CROP = 17, Anaorphic, ignore display_width, display_height in most cases
 - o PRE_SIDEBARRED_WATERGLASS = 18,
 - o PRE_SIDEBARRED_TILT_AND_SCAN = 19,
 - o PRE_SIDEBARRED_SIDEBAR = 20
- Default: 1 (letterbox)

[RW] /doremi/etc/0/video_color_space

- Set the initial video color space
 - COLOR_SPACE_NO_FORCE = 0
 - COLOR_SPACE_RGB = 1
 - COLOR_SPACE_YPBPR = 2
 - COLOR_SPACE_PC_MONITOR = 3
- Default: 1 (YPbPr)

[RW] /doremi/etc/0/video_background_color

- Set the initial video background color shown in eject mode
- **default:**

[RW] /doremi/etc/0/video_aux_settings

- Set the initial aux settings that control composite & SDI output
 - noAuxVideo = 0, Disable the auxiliary video port
 - showAuxVideoPlaneOnlyNarrow = 1, Show video plane only in aux port, narrow display
 - showAuxVideoPlaneOnlyWide = 2, Show video plane only in aux port, wide display
 - showAuxVideoAndGraphicsNarrow = 3, Show video + graphics in aux port, narrow display
 - showAuxVideoAndGraphicsWide = 4, Show video + graphics in aux port, wide display
 - Extended aux modes do not crop the top, bottom 10 lines if main outputs 1080i and enable letterbox
 - ExtAuxVideoPlaneOnly modes still display full screen (or letterbox/sidebar) even if the main video window size is reduced by specifying a small WinRect in SetSingleVideoDisplay.
 - showExtAuxVideoPlaneOnlyWide = 5,
 - showExtAuxVideoAndGraphicsWide = 6,
 - showExtAuxVideoPlaneOnlyNarrowLetterbox = 7,
 - showExtAuxVideoAndGraphicsNarrowLetterbox = 8,
 - showExtAuxVideoPlaneOnlyNarrowPanAndScan = 9,
 - showExtAuxVideoAndGraphicsNarrowPanAndScan = 10,
 - showExtAuxVideoPlaneOnlyAnamorphic = 11.
- Default: 0 Off

[RW] /doremi/etc/0/audio_spdif

- Set the initial SPDIF audio control value
 - 0: off
 - 1: auto detect
 - 2: pcm
- Default: 1

[RW] /doremi/etc/0/audio_rendering

- Enable audio output
 - 0: off
 - 1: on
- Default: 1

[RW] /doremi/etc/0/audio_delay

- Set the audio/video delay in frame
- Format: signed integer
- Default: 0

[RW] /doremi/etc/0/input_channel

- Set the ID for ASI input
- **Default: Not Implemented**

[RW] /doremi/etc/0/no_input_channel

- Disable ASI input
- Format: ASI IN is not active when this file is present.