

ICD Requirements

For

Neptune

H264

Revision : 4.58
Date : 20/03/13
By : Itamar Levit

Version	Date	Change Description	Name
3.30	09/10/13		Rivka Levit
3.31	20/11/13	Mux instead of channel	Michal Tvito
3.32	27/1/14	Improve Playback	Rivka Levit
3.35	12/03/14	Fix some worn items	Itamar Levit
3.36	31/08/14	Add 4 channels case of 2 tvp5158	Rivka Levit
3.37	13/01/15	Split - mode	Rivka Levit
3.38	10/02/15	Add parameters to split	Michal Tvito
3.39	17/02/15	Add Set Stabilizer Search	Itamar Levit
3.40	22/02/15	Add Set Stabilizer Params	Michal Tvito
3.41	15/03/14	Add set Deinterlace Params	Michal Tvito
3.42	18/03/15	Add IP to uart setup	Michal Tvito
3.43	30/3/15	Change a value in deinterlace	Michal Tvito
3.70	21/4/15	New parameters for uart - parity and data size	Michal Tvito
3.80	03/05/15	Add Staring option to stabilizer	Michal Tvito
3.90	04/05/15	Add Input parameter to audio	Michal Tvito
4.00	07/05/15	Add Player API Commands	Michal Tvito
4.01	10/05/15	Add JPEG option to codec(video)	Michal Tvito
4.02	11/05/15	Add Display for streaming setup	Michal Tvito
4.03	13/05/15	Fix a mistake	Michal Tvito
4.04	18/05/15	Add uart disable	Michal Tvito
4.05	21/05/15	Add oat command params Add stabilizer operational	Michal Tvito
4.06	08/06/15	File_id->file_name	Michal Tvito
4.07	10/06/15	Update text overlay command	Itamar
4.08	29/06/15	Update text overlay command	Rivka Cohen
4.09	07/01/2015	Add note to get storage_list	Michal Tvito
4.10	01/07/15	Change overlay color type	Rivka Cohen
4.11	01/07/15	1. Add demux# to play_file 2. Add function get tag count() 3. Api is using mux instead of file_name	Michal Tvito
4.12	06/07/15	1. Correct setup mux length 2. Changing the formulation of Message length to Data length	Rivka Cohen
4.13	07/07/15	Update overlay setup	Rivka Cohen
4.14	08/07/15	Create display config command	Rivka Cohen
4.15	09/07/15	Add demux parameter to info	Michal Tvito
4.16	13/07/15	Edit Display config	Rivka Cohen
4.17	13/07/15	Play - status : demux->type	Michal Tvito
4.18	14/07/15	Remove open and close file command. Add parameter to tag commands	Michal Tvito
4.19	21/07/15	Add Device No to Play file	Itamar
4.19	21/07/15	Add Speed 0 to Play file	Itamar

ICD requirement

4.19	21/07/15	Add device Number to Delete File	Itamar
4.20	23/07/15	Edit display driver commands (118-123)	Rivka Cohen
4.20	28/07/15	Update display driver commands (118-123)	Rivka Cohen
4.20	29/07/15	1. Edit playback commands (74-75) 2. Add new commands audio ptt(68-69)	Rivka Cohen
4.21	30/07/15	Get file status: second instead of position	Michal Tvito
4.22	03/08/15	Add command Update Player	Michal Tvito
4.23	04/08/15	Play file with no device no	Michal Tvito
4.24	16/08/15	Edit setup overlay command	Rivka Cohen
4.25	16/08/15	Play status- add End & start of file	Michal Tvito
4.25	16/08/15	Erase device with name	Michal Tvito
4.26	18/08/15	Change speed in player	Michal Tvito
4.27	23/08/15	Add get HW state commands(94-95)	Rivka Cohen
4.27	27/08/15	Add new command HW-reset errors(96)	Rivka Cohen
4.28	16/09/15	Add Play Mux command(153)	Michal Tvito
4.28	16/09/15	Add full path name to get play status(158)	Michal Tvito
4.28	16/09/15	Edit time & date function (9-10)	Rivka Cohen
4.28	16/09/15	Add enable index command (87-88)	Michal Tvito
4.28	17/09/15	Add new command - setup tw9910 camera (124-126)	Rivka Cohen
4.29	24/09/15	Add new commands – setup snapshot (127-129)	Rivka Cohen
4.29	06/10/15	Add new commands – setup external sync (130-131)	Rivka Cohen
4.29	28/10/15	Edit set/get stabilizer command (64-65)	Rivka Cohen
4.30	04/11/15	Edit get play status command(158)	Rivka Cohen
4.30	17/11/15	Add new commands	Rivka Cohen
4.31	10/12/15	Sdx play change API	Michal Tvito
4.32	14/12/15	Edit uart setup command(21-22)	Rivka Cohen
4.32	21/12/15	Edit get mux status command(59)	Rivka Cohen
4.32	23/12/15	Add Mux State in command (158)	Michal Tvito
4.32	29/12/15	Edit play state command(158)	Rivka Cohen
4.32	31/12/15	Change value of setup mux auto param command (41-42)	Rivka Cohen
4.32	19/01/16	Add: 1. maris commands(170-171) Format commands()	Rivka Cohen
4.32	25/02/16	Change “set bitrate” to “set jitter buffer” and add parameter to jitter buffer (78-79)	Rivka Cohen
4.32	03/03/16	Add options to uart params(to parity and to flow control)	Rivka Cohen
4.32	09/03/16	Change IP and port to source address and add destination IP and port (102-103)	Rivka Cohen
4.32	21/03/16	*Correct mistake in ack/nack format *add parameters to pixel format(104-105) * (51-52)	Rivka Cohen
4.32	29/03/16	Add new command: set/get web status (25-26)	Rivka Cohen
4.33	18/05/16	Convert setup index commands(87-88) to setup record and add parameters.	Rivka Cohen

Prepared for:

ICD requirement

4.33	29/05/16	Add new commands (30-31) of set/get record auto delete and removing these parameters from record setup(87-88)	
4.33	05/06/16	Content and spelling corrections	Rivka Cohen
4.33	07/06/16	Add new commands: (35-37) of overlay display and zixi module	Rivka Cohen
4.33	22/06/16	Add new command: get overlay items count(36)	Rivka Cohen
4.34	26/07/16	Remove speed(x2,x3,x4,x5)	Michal Tvito
4.34	01/08/16	Add new commands :set network interface(27-28)	Rivka Cohen
4.34	04/08/16	Add new commands get active interface network (55)	Rivka Cohen
4.34	09/08/16	Add new commands get current mount (86)	Rivka Cohen
4.34	08/09/16	Add new commands of cellular network (138-140)	Rivka Cohen
4.35	22/09/16	Add option of GPS in the overlay type (53) Add new commands of time source(141-142)	Rivka Cohen
4.35	27/09/16	Add functions of set/get GPS device params (143-144)	Rivka Cohen
4.35	28/09/16	Change ack/nak ID to 0xFF only in document	Rivka Cohen
4.36	2/11/2016	Usb to disk mode(145,146)	Michal Tvito
4.37	22/11/2016	Gps OSD gps status(147,148,149)	Michal Tvito
4.38	29/11/2016	Add record parameters to gps device	Michal Tvito
4.39	20/12/2016	Add cdc client and cdc ost	Michal Tvito
4.39	10/01/2017	Add row bitmap to stream protocol	Michal Tvito
4.39	13/02/2017	Edit overlay commands (32-34)	Rivka Cohen
4.39	23/03/2017	Add parameters to extended codec (82-83)	Rivka Cohen
4.40	20/04/2017	Remove parameters of mux from GetFileCount GetFileInfo (151,152)	Itamar Levit
4.41	05/03/2017	Add CSI –Type	Michal Tvito
4.42	29/05/2017	Add new GPIO commands(174-180)	Michal Tvito
4.43	08/07/2017	Choos Application	Michal Tvito
4.44	30/08/2017	Change CMD 45,46 Input value ,and Mono Stereo value	Itamar Levit
4.45	31/08/2017	Set and get HDMI params (181,182)	Michal Tvito
4.45	31/08/2017	Set and get Onvif (183,184)	Michal Tvito
4.46	09/05/2017	Integration between hdmi and external setup (remove 182,182)	Michal Tvito
4.47	19/09/2017	Enable USB Camera(remove in 4.48)	Michal Tvito
4.48	03/10/2017	Usb camera modes	Michal Tvito
4.49	02/05/2018	Add spi interface for xstream command 41 3 – SPI (Xstream View)	Itamar Levit
4.49	09/10/2018	Remove Set Recording MUX	Itamar Levit
4.50	03/01/2019	Change Test -> virtual video (41) Change PIP commands params (77-78) Add set and get virtual video parama(195,196)	Michal Tvito
4.51	07/07/2019	Multi Stream commands 200-203	Michal Tvito

ICD requirement

4.52	19/09/2019	record : Sync\nno split	Michal Tvito
4.53	11/11/2019	Format Encryption Encoder fast	Michal Tvito
4.54	28/11/2019	Key_board and led	Michal Tvito
4.55	14/01/2020	Add VMD option(Command 41,42,205,206)	Rivka Cohen
4.56	20/01/2020	Update multi stream and usb(186,187,188,200,201)	Michal Tvito
4.57	1/28/2020	Passes Trough(207,208)	Michal Tvito
4.58	06/02/2020	Tcp/udp for uart->eth	Michal Tvito

Table of Contents

Contents

1	General.....	10
1.1	Communication	10
1.2	LAN multicast message that the board register itself with the multicast IP / port.	10
1.3	A LAN remote protocol that control and operate the LDVC board	10
1.4	RS232 Interface	10
1.5	Definitions	11
1.6	Message Encoding.....	11
1.7	Message Format	11
1.8	ACK/NAK Message Format	12
1.9	Data Flow	12
1.10	Auto recognition	13
2	API Messages.....	14
2.1	Power Off	14
2.2	Set LDVC Configuration Mode	14
2.3	Get LDVC Configuration Mode.....	14
2.4	Set Operation Mode.....	15
2.5	Get Operation Mode.....	15
2.6	Get Test Results.....	16
2.7	Set LDVC Networking Mode.....	16
2.8	Get LDVC Networking Mode	17
2.9	Set Time & Date.....	18
2.10	Get Time & Date.....	18
2.11	Get Version	19
2.12	Set Factory Default	19
2.13	Set Configuration Number	19
2.14	Get Configuration Number	20
2.15	Reboot System	20
2.16	Set CSI Config	20
2.17	Get CSI Config.....	21
2.18	Get Camera Config	21
2.19	Set UTC Time	22
2.20	Get UTC Time	22
2.21	Set DATA Source	22
2.22	Get DATA Source.....	23
2.23	Set UART cmd mode.....	24
2.24	Get UART cmd mode	25
2.25	Set Web Status	26
2.26	Get Web Status.....	26
2.27	Set Network Interface	26
2.28	Get Network Interface.....	27
2.29	Set Record Auto Delete.....	27
2.30	Get Record Auto Delete	28
2.31	Set Display Overlay	28
2.32	Get Display Overlay.....	29
2.33	Update Display Overlay.....	31
2.34	Overlay Operations	31
2.35	Get Overlay Items Count	31
2.36	Overlay Operations	32
2.37	Set ZIXI Params.....	32

2.38	Get ZIXI Params	33
2.39	Get ZIXI Status	34
2.40	Get Capture Input Status	34
2.41	Setup Streaming Mux.....	35
2.42	Get Streaming Mux.....	37
2.43	Set Streaming Operation	38
2.44	Get Streaming State	39
2.45	Set Audio Source mode.....	39
2.46	Get Audio Source mode.....	40
2.47	Set Video Encoding Quality.....	40
2.48	Get Video Encoding Quality.....	41
2.49	Set Video Frame Rate	42
2.50	Get Video Frame Rate.....	42
2.51	Set ROI.....	43
2.52	Get ROI.....	43
2.53	Set Capture Overlay	44
2.54	Get Capture Overlay	45
2.55	Get Active Interface Network	46
2.56	Set Automatic Mode	46
2.57	Get Automatic Mode.....	47
2.58	Set Recording Operation	47
2.59	Get Active Mux State.....	48
2.60	Get Audio Codec.....	48
2.61	Set Stabilizer Search Direction	49
2.62	Get Stabilizer Search Direction.....	49
2.63	Set Stabilizer Params	50
2.64	Get Stabilizer params	50
2.65	Set Deinterlace Params	51
2.66	Get Deinterlace params	51
2.67	Set Audio PTT	52
2.68	Get Audio PTT.....	52
2.69	Stream Player Operation	52
2.70	Stop Stream Player (Not used).....	53
2.71	Get Player State.....	53
2.72	Get Active Status	54
2.51	Set Playback Parameters	54
2.52	Get Playback Parameters.....	55
2.53	Set PIP.....	57
2.54	Get PIP	57
2.55	Set Jitter Buffer	58
2.56	Get Jitter Buffer	58
2.57	Get Playback Statistic	59
2.58	Get Video Codec.....	60
2.59	Set Extended Video Codec.....	60
2.60	Get Extended Video Codec.....	61
2.61	Set Display Parameters	61
2.62	Get Display Parameters.....	62
2.63	Get Current Mount.....	62
2.64	Set Record Mode.....	63
2.65	Get Record Mode	63
2.66	Set Split Params	64
2.67	Get Split Params.....	65
2.68	Set Encoder Tune	65
2.69	Get Encoder Tune	67
2.70	Reset Encoder Tune	69

2.71	Get HW Error Count	69
2.72	Get HW Error Info	69
2.73	Reset HW Errors	70
2.74	Set TTL	70
2.75	Get TTL.....	70
2.76	Erase Device	71
2.77	Get Storage Information.....	71
2.78	Get UART Device.....	72
2.79	Set UART Parameters for uart to eth	72
2.80	Get UART Parameters	73
2.81	Setup Camera Driver.....	74
2.82	Get Camera Driver	75
2.83	Set GPIO (see new command for gpio (174)).....	76
2.84	Get GPIO (see new command for gpio (175)).....	76
2.85	Delete GPIO (see new command for gpio (177))	76
2.86	Get GPIO Function	77
2.87	Get Temperature	77
2.88	Set RTSP Server.....	77
2.89	Get RTSP Server.....	78
2.90	Get System Info	78
2.91	Test Display	79
2.92	Get Display Driver Config	79
2.93	Setup Display Config	80
2.94	Get Display Config	80
2.95	Get Display modes	81
2.96	Get Current Display mode	81
2.97	Set Display Mode.....	82
2.98	Set Analog Camera Params.....	82
2.99	Get Analog Camera Params	82
2.100	Reset Analog Camera.....	83
2.101	Set Snapshot.....	83
2.102	Get Snapshot	84
2.103	Snapshot.....	84
2.104	Set Camera Ext.....	84
2.105	Get Camera Ext.....	85
2.106	Set Emergency Boot Params	86
2.107	Get Emergency Boot Params	86
2.108	Set Display Operation	86
2.109	Get Display Operation	87
2.110	Set FPGA	87
2.111	Get FPGA.....	87
2.112	Set Cellular Network.....	88
2.113	Get Cellular Network	88
2.114	Get Cellular Network Status	89
2.115	Set Time Source	89
2.116	Get Time Source	90
2.117	Set GPS Device.....	90
2.118	Get GPS Device	91
2.119	Get Usb To Disk Mode	92
2.120	Set Usb To Disk Mode.....	92
2.121	Get Gps Statust.....	92
2.122	Get Gps OSD	93
2.123	Set GPS OSD	94
2.124	Get Device Count.....	94
2.125	Get File Count.....	94

2.126	Get File Info	95
2.127	Play Mux	96
2.128	Seek File.....	96
2.129	Play File	96
2.130	Stop Play File	97
2.131	Get Play Status	97
2.132	Delete File	98
2.133	Set Mux Tag.....	98
2.134	Get Tag Count.....	98
2.135	Get Tag Seconds	99
2.136	Update Player	99
2.137	Format Device.....	100
2.138	Get Format Device Status	100
2.139	Maris Messages.....	101
2.140	Set Clip Mode	101
2.141	Get Clip Mode.....	101
2.142	Set GPIO (new).....	102
2.143	Get GPIO by ID (new)	103
2.144	Get GPIO by Number (new)	104
2.145	Set GPIO Feedback Params	106
2.146	Get GPIO Feedback param	107
2.147	Delete GPIO (new).....	108
2.148	Get GPIO value.....	108
2.149	Set ONVIF Enable Mode	108
2.150	Get ONVIF Enable Mode	109
2.151	Get USB Camera Modes Count	109
2.152	Set USB Camera Mode By ID	110
2.153	Get USB Camera Mode By ID.....	110
2.154	Send Data by serial	110
2.155	Set Virtual Video	111
2.156	Get Virtual Video.....	111
2.157	Set Encryption Mode	112
2.158	Get Encryption Mode.....	112
2.159	Set multi stream params	113
2.160	Get multi stream params	113
2.161	Active multi stream.....	114
2.162	Get Multi Stream status.....	114
2.163	Set Fast Encode Params	115
2.164	Set VMD Params	115
2.165	Get VMD Params.....	116
2.166	Get Update Status.....	117
2.167	Set Passes Through Params.....	117
2.168	Get Passes Through Params	117
3	<i>LDVC Config File</i>	<i>119</i>

1 General

1.1 Communication

1.2 LAN multicast message that the board register itself with the multicast IP / port.

The connection between the board and the PC will be initiated by multicast: all the boards that send multicast address, give their IP address and to the desired board will be communicated in unicast.

Board multicast address **225.10.2.2:30003**

How this is done?:

When the board is powered-up it start sending multicast message repeated on every 1 second. The PC will register on this multicast so it can get the IP address of the board. In case of a PC connected to many MC boards the application may list the IP address and give the user to select the MC board to communicate with.

Board Device ID Table:

Device Name	ID	Description	
HDSDI Input 1	1	HDSDI camera receiver	
HDSDI Input 2	2	HDSDI camera receiver	
HDSDI Output 1	3	HDSDI Display device	
HDSDI Output 2	4	HDSDI Display device	
TVOut 1	5	TVOut device	
TVOut 2	6	TVOut device	
TVOut 3	7	TVOut device	
TVOut 4	8	TVOut device	
TVIn	9	Four-Channel NTSC/PAL Video Decoders	
Audio Codec	10	Audio Codec Device	

1.3 A LAN remote protocol that control and operate the LDVC board

Is a LAN protocol that controls the LDVC board. On both we use UDP protocol port 3100.

The followings describe the protocol details.

1.4 RS232 Interface

- Bits per second: 115200
- Data bits: 8
- Parity: None
- Stop Bit: 1

Prepared for:

- Flow Control: None

1.5 Definitions

- **Media** – SD, NAND
- **Primary/Secondary Media** – the currently selected primary and secondary media for recording (SD, NAND or any combination)
- **File Name** – a directory including channel 1,2,3,4 Stream Files, Data File and Stills Images
- **Stream File** – transport stream file including video, audio and metadata
- **Data File** – data file in the User's configuration
- **Streaming Channel** – the selected channel for streaming

1.6 Message Encoding

Hexadecimal values will be preceded by a '0x' and the hex-digits (such as 0xAB) and decimal values will appear normally without any header.

The API Protocol is Encoded Binary, Little Endian (least significant byte is stored first).

1.7 Message Format

The message format is comprised of a fixed length Message Header and a variable length Message Data as follows:

Message Section	Field Name	Data Type	Value
<i>Message Header</i>	Message Sync	Byte	0xA5
	Message ID	Byte	
	Checksum	Byte	
	Data Length	Byte	
<i>Message Data</i>	Parameter 1	According to Message	
	...		
	Parameter N	According to Message	

Message Fields:

- **Message Sync** – A Sync byte (0xA5)
- **Message ID** – A unique message identifier
- **Checksum** – A Sum of all the Message (not include itself)
- **Data Length** – Total number of bytes in Message Data
- **Parameters 1 to N** – Message parameters

1.8 ACK/NAK Message Format

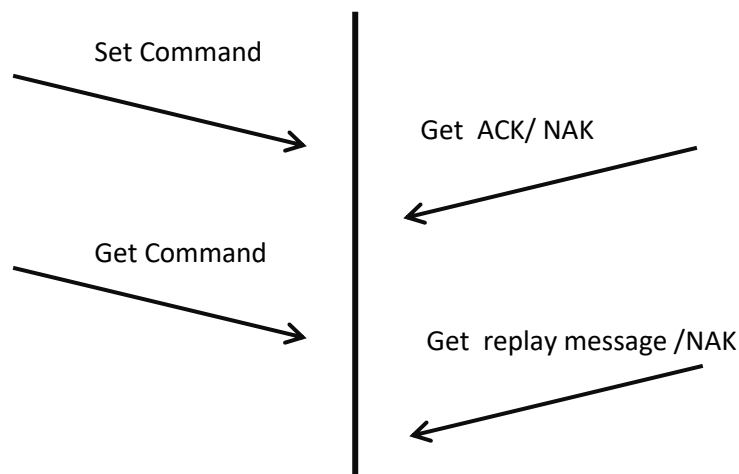
The message format is comprised of a fixed length Message Header and data as follows:

Message Section	Field Name	Data Type	Value
<i>Message Header</i>	Message Sync	Byte	0xA5
	ACK/NAK	Byte	0xFF
	Message ID		
	Checksum	Byte	
	Data Length	Byte	3
	ACK/NAK	Byte	ACK - 0xBB NAK - 0xCC
	Message ID	Byte	
	Error Number	Byte	Not implemented now

Message Fields:

- **Message Sync** – A Sync byte (0xA5)
- **Data Number** – A unique message identifier for ack/nack message
- **Checksum** – A Sum of all the Message (not include itself)
- **Data Length** – Total number of bytes in Message Data
- **Data 0** - ACK (0xBB) or NACK (0xCC)
- **Data 1** –Message ID we ACK/NAK
- **Data 2** – A unique error identifier

1.9 Data Flow



A message sequence:

- The user sends a message to the board.
- In the case of “set command” the program returns the acknowledge (ACK) for success or not acknowledge (NAK) for failure.
- In the case of “get command” the program returns replay message for success or not acknowledge (NAK) for failure.

1.10 Auto recognition

The default of the input camera resolution will define automatically by the LDVC.

2 API Messages

Note: Each command that config the system, write in the config file.

2.1 Power Off

Note: Power off the system: after sending ACK the system go to sleep mode.

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	01	
...	
Data Length	Byte	4	
MAGIC ID1	Byte	0xD	
MAGIC ID2	Byte	0xE	
MAGIC ID3	Byte	0xA	
MAGIC ID4	Byte	0xD	

Reply Message Structure: ACK/NAK

2.2 Set LDVC Configuration Mode

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	02	
...	
Data Length	Byte	1	
LDVC Mode	Byte	0 – Idle 1 – Transmitter 2 – Receiver	

Reply Message Structure: ACK/NAK

2.3 Get LDVC Configuration Mode

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	03	
...	
Data Length	Byte	0	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	

Message ID	Byte	03	
...	
Data Length	Byte	1	
LDVC Mode	Byte	0 – Idle 1 – Transmitter 2 – Receiver	

2.4 Set Operation Mode

Note: Operational mode is the normal operation of the system. This is the default. The test mode, tests all the channels, and to know when the test finish, the user sends repeatedly the command “Get Operational Mode” (See the following command), until he get “idle mode”.

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	04	
...	
Data Length	Byte	1	
Operation Mode	Byte	0 – Idle 1 – Operational 2 – Test mode	

Reply Message Structure: ACK/NAK or operation mode or system status

2.5 Get Operation Mode

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	05	
...	
Data Length	Byte	0	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	05	
...	
Data Length	Byte	1	
Operation Mode	Byte	0 – Idle 1 – Operational 2 – Test mode	

2.6 Get Test Results

Note: The result is a pair of bytes. First is the device ID as list in the table (device ID table).

And the device test status: 0 – not identified, 1 – identified, 2 – connected, 3 – failuer.

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	06	
...	
Data Length	Byte	0	

Reply Message Structure: returns list all device status

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	06	
...	
Data Length	Byte	Number of device x 2	
Device ID	Byte	# Device ID	
Device Status	Byte	0 –Failed 1 – Pass	
.....			

2.7 Set LDVC Networking Mode

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	07	
...	
Data Length	Byte	19	
Mac Address #1	Byte	1 Value	
Mac Address #2	Byte	2 Value	
Mac Address #3	Byte	3 Value	
Mac Address #4	Byte	4 Value	
Mac Address #5	Byte	5 Value	
Mac Address #6	Byte	6 Value	
Mode	Byte	0 – DHCP (default) 1 – Manual	
IP Address #1	Byte	1st Value	Manual
IP Address #2	Byte	2nd Value	Manual
IP Address #3	Byte	3rd Value	Manual
IP Address #4	Byte	4th Value	Manual
Mask Address #1	Byte	1st Value	Manual
Mask Address #2	Byte	2nd Value	Manual
Mask Address #3	Byte	3rd Value	Manual

Mask Address #4	Byte	4th Value	Manual
Gateway Address #1	Byte	1st Value	Manual
Gateway Address #2	Byte	2nd Value	Manual
Gateway Address #3	Byte	3rd Value	Manual
Gateway Address #4	Byte	4th Value	Manual

Reply Message Structure: ACK/NAK

2.8 Get LDVC Networking Mode

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	08	
...	
Data Length	Byte	0	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	08	
...	
Data Length	Byte	19	
Mac Address #1	Byte	1 Value	
Mac Address #2	Byte	2 Value	
Mac Address #3	Byte	3 Value	
Mac Address #4	Byte	4 Value	
Mac Address #5	Byte	5 Value	
Mac Address #6	Byte	6 Value	
Mode	Byte	0 – DHCP 1 – Manual 2 – None	
IP Address #1	Byte	1st Value	Manual
IP Address #2	Byte	2nd Value	Manual
IP Address #3	Byte	3rd Value	Manual
IP Address #4	Byte	4th Value	Manual
Mask Address #1	Byte	1st Value	Manual
Mask Address #2	Byte	2nd Value	Manual
Mask Address #3	Byte	3rd Value	Manual
Mask Address #4	Byte	4th Value	Manual
Gateway Address #1	Byte	1st Value	Manual
Gateway Address #2	Byte	2nd Value	Manual

Gateway Address #3	Byte	3rd Value	Manual
Gateway Address #4	Byte	4th Value	Manual

2.9 Set Time & Date

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	09	
...	
Data Length	Byte	6	
Hour	Byte	0 – 23	
Minute	Byte	0 – 59	
Second	Byte	0 – 59	
Day	Byte	1 – 31	
Month	Byte	1 – 12	
Year	Byte	2 last digits of year, start from 2000 (13,..)	

Reply Message Structure: ACK/NAK

2.10 Get Time & Date

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	10	
...	
Data Length	Byte	0	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	10	
...	
Data Length	Byte	6	
Hour	Byte	0 – 23	
Minute	Byte	0 – 59	
Second	Byte	0 – 59	
Day	Byte	1 – 31	
Month	Byte	1 – 12	
Year	Byte	Number minus 1900	

2.11 Get Version

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	11	
...	
Data Length	Byte	0	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	11	
...	
Data Length	Byte	2	
Version	Byte		
Subversion	Byte		

2.12 Set Factory Default

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	12	
...	
Data Length	Byte	4	
MAGIC ID1	Byte	0xD	
MAGIC ID2	Byte	0xE	
MAGIC ID3	Byte	0xA	
MAGIC ID1	Byte	0xD	

Reply Message Structure: ACK/NAK

2.13 Set Configuration Number

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	13	
...	
Data Length	Byte	1	
CFG Number	Byte	0 – CFG 1(default) 1 – CFG 2 2 – CFG 3 3 – CFG 4	

Reply Message Structure: ACK/NAK

2.14 Get Configuration Number

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	14	
...	
Data Length	Byte	0	

Reply Message Structure:

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	14	
...	
Data Length	Byte	1	
CFG Number	Byte	0 – CFG 1(default) 1 – CFG 2 2 – CFG 3 3 – CFG 4	

Reply Message Structure: ACK/NAK

2.15 Reboot System

Note: Reboot system after sending ACK the system go to sleep mode.

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	15	
...	
Data Length	Byte	4	
MAGIC ID1	Byte	0xD	
MAGIC ID1	Byte	0xE	
MAGIC ID1	Byte	0xA	
MAGIC ID1	Byte	0xD	

Reply Message Structure: ACK/NAK

2.16 Set CSI Config

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	16	
...	

Data Length	Byte	2	
Csi1 #	Byte	-1 – None Values between 0 – 9	
Csi2#	Byte	-1 – None Values between 0 – 9	

Reply Message Structure: ACK/NAK

2.17 Get CSI Config

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	17	
...	
Data Length	Byte	0	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	17	
...	
Data Length	Byte	2	
Csi1 #	Byte	-1 None Camera Driver Values between 0 – 9	
Csi2#	Byte	-1 None Camera Driver Values between 0 – 9	

2.18 Get Camera Config

If camera driver is not exist return NAK

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	18	
...	
Data Length	Byte	1	
Camera #	Byte	Values between 0 – 9	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	18	
...	
Data Length	Byte	22	

Camera #	Byte	Values between 0– 9	
Type	Byte	0 – SD 1 – HD 2 – multiple	
Camera Name	String	20 Bytes	

2.19 Set UTC Time

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	19	
...	
Data Length	Byte	9	
Time	8 X Byte	UTC time in microseconds since 1/1/1970 00:00	
Validity	1 Bytes	0 – Valid 1 – Invalid	

2.20 Get UTC Time

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	20	
...	
Data Length	Byte	0	

Replay Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	20	
...	
Data Length	Byte	9	
Time	8 Bytes	UTC time in microseconds since 1/1/1970 00:00	
Validity	Byte	0 – Valid 1 – Invalid	

2.21 Set DATA Source

Message Structure:

Field Name	Data Type	Value	Depended
------------	-----------	-------	----------

...	
Message ID	Byte	21	
...	
Data Length	Byte	14	
Source ID	Byte	Source ID number 0	
Source Type	Byte	0 – Unicast 1 – Multicast 2 – UART 3 – CSI-1 4 – CSI-2	
IP	4 Bytes		Multicast
Port	2 Bytes		Multicast /Unicast
UART #	Byte	Values between 0– 9	UART
Buad Rate	Byte	0 – 2400 1 – 4800 2 – 9600 3 – 19200 4 – 38400 5 – 57600 6 – 115200 (default) 7 – 230400	UART
Flow Control	Byte	0 – None (default) 1 – Hardware	UART
CSI-TYPE	Byte	0 – Type1 (default) 1 – Type2	CSI-1/CSI-2
DID	Byte	0 – 255 (0 - default)	CSI-1/CSI-2
SDID	Byte	0 – 255 (0 - default)	CSI-1/CSI-2
Net Protocol	Byte	0 – UDP 1 – TCP	

2.22 Get DATA Source

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	22	
...	
Data Length	Byte	1	
Source ID	Byte	Source ID number 1..2	

Replay Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	22	
...	
Data Length	Byte	15	
Source ID	Byte	Source ID number 1..2	
Source Type	1 Bytes	0 – Unicast	

		1 – Multicast 2 – UART 3 – CSI-1 4 – CSI-2	
IP	4 Bytes		Multicast
Port	2 Bytes		Multicast /Unicast
UART #	Byte	Values between 0– 9	UART
Buad Rate	Byte	0 – 2400 1 – 4800 2 – 9600 3 – 19200 4 – 38400 5 – 57600 6 – 115200 (default) 7 – 230400	UART
Flow Control	Byte	0 – None (default) 1 – Hardware	UART
CSI-TYPE	Byte	0 – Type1 (default) 1 – Type2	CSI-1/CSI-2
DID	Byte	0 – 255 (0 - default)	CSI-1/CSI-2
SDID	Byte	0 – 255 (0 - default)	CSI-1/CSI-2
Net Protocol	Byte	0 – UDP 1 – TCP	Multicast /Unicast

2.23 Set UART cmd mode

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	23	
...	
Data Length	Byte	12	
UART #	Byte	Values between 0– 9	
State	Byte	0 – Off 1 – On (default)	
Baud Rate	Byte	0 – 2400 1 – 4800 2 – 9600 3 – 19200 4 – 38400 5 – 57600 6 – 115200 (default) 7 – 230400	
Flow Control	Byte	0 – None (default) 1 – Hardware	
Parity	Byte	0 – None 1 – Odd 2 – Even	

		3 – Mark 4 – Space	
Size	Byte	7- 7 bits 8 -8 bits	
Rezerved	6 Bytes		

Reply Message Structure: ACK/NAK

2.24 Get UART cmd mode

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	24	
...	
Data Length	Byte	0	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	24	
...	
Data Length	Byte	12	
UART #	Byte	Values between 0– 9	
State	Byte	0 – Off 1 – On	
Buad Rate	Byte	0 – 2400 1 – 4800 2 – 9600 (default) 3 – 19200 4 – 38400 5 – 57600 6 – 115200 7 – 230400	
Flow Control	Byte	0 – None (default) 1 – Hardware	
Parity	Byte	0 – None 1 – Odd 2 – Even 3 – Mark 4 – Space	
Size	Byte	7- 7 bits 8 -8 bits	
Rezerved	6 Bytes		

2.25 Set Web Status

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	25	
...	
Data Length	Byte	1	
Web Status	Byte	0 – Off 1 – On	

Reply Message Structure: ACK/NAK

2.26 Get Web Status

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	26	
...	
Data Length	Byte	0	

Replay Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	26	
...	
Data Length	Byte	1	
Web Status	Byte	0 – Off 1 – On	

2.27 Set Network Interface

Message Structure:

D-BYTE	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	27	
	
	Data Length	Byte	22	
0	Network Name	Byte	0 – eth0 1 – usb0 2 – usb1	
1	Type	Byte	0 – Ethernet 1 – CDC Client 2 – CDC OST	
2	Enabled	Byte	0 – Off 1 – On	

3	Mac Address	6 Bytes		
9	Mode	Byte	0 – DHCP 1 – Manual	
10	IP Address	4 Bytes		
14	Mask Address	4 Bytes		
18	Gateway Address	4 Bytes		

Reply Message Structure: ACK/NAK

2.28 Get Network Interface

Message Structure:

D-BYTE	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	28	
	
	Data Length	Byte	1	
0	Network Name	Byte	0 – eth0 1 – usb0 2 – usb1	

Replay Message Structure:

D-BYTE	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	28	
	
	Data Length	Byte	21	
0	Network Name	Byte	0 – eth0 1 – usb0 2 – usb1	
1	Type	Byte	0 – Ethernet 1 – CDC Ethernet	
2	Enabled	Byte	0 – Off 1 – On	
3	Mac Address	6 Bytes		
9	Mode	Byte	0 – DHCP 1 – Manual	
10	IP Address	4 Bytes		
14	Mask Address	4 Bytes		
18	Gateway Address	4 Bytes		

2.29 Set Record Auto Delete

Message Structure:

Field Name	Data Type	Value	Depended
------------	-----------	-------	----------

...	
Message ID	Byte	30	
...	
Data Length	Byte	5	
Auto Delete Files	Byte	0 – Off 1 – On	
Disk Limit Size	4 Bytes	In MB	

Reply Message Structure: ACK/NAK

2.30 Get Record Auto Delete

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	31	
...	
Data Length	Byte	0	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	31	
...	
Data Length	Byte	5	
Auto Delete Files	Byte	0 – Off 1 – On	
Disk Limit Size	4 Bytes	In MB	

2.31 Set Display Overlay

Message Structure:

D-BYTE	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	32	
	
	Data Length	Byte	56	
0	Frame Buffer#	Byte		
1	Slave Frame Buffer#	Byte	Values between 0-3, if there is no slave, use -1	
2	Overlay ID#	Byte		
3	Mode	Byte	0 – Hide (default) 1 – Show	

4	Location	Byte	0 – Top-R 1 – Top-L 2 – Bottom-R 3 – Bottom-L 4 – Defined	
5	Type	Byte	0 – user text 1 – time + frame number (modulo 30)	
6	Text Color	Byte	0 – Black (default) 1 – White 2 – Red 3 – Green 4 – Blue 5 – Yellow 6 – Orange	
7	Background Color	Byte	0 – None (default) 1 – White 2 – Red 3 – Green 4 – Blue 5 – Yellow 6 – Orange	
8	Font	Byte	0 – Font 1(default) 1 – Font 2 2 – Font 3 3 – Font 4	
9	Font Size	Byte		
10	X Pixel Position	2 Bytes		Location- Defined
12	Y Pixel Position	2 Bytes		Location- Defined
14	Order by	Byte	0 – One Line 1 – Lines (default)	Location- Defined
15	Field Length	Byte		
16	Label Text	40 Bytes max	ASCII String	

Reply Message Structure: ACK/NAK

2.32 Get Display Overlay

Message Structure:

D-BYTE	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	33	
	
	Data Length	Byte	2	

0	Frame Buffer#	Byte		
1	Overlay ID#	Byte		

Reply Message Structure:

D-BYTE	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	33	
	
	Data Length	Byte	56	
0	Frame Buffer#	Byte		
1	Slave Frame Buffer#	Byte	Values between 0-3, if there is no slave, use -1	
2	Overlay ID#	Byte		
3	Mode	Byte	0 – Hide (default) 1 – Show	
4	Location	Byte	0 – Top-R 1 – Top-L 2 – Bottom-R 3 – Bottom-L 4 – Defined	
5	Type	Byte	0 – user text 1 – time + frame no	
6	Text Color	Byte	0 – Black (default) 1 – White 2 – Red 3 – Green 4 – Blue 5 – Yellow 6 – Orange	
7	Background Color	Byte	0 – None (default) 1 – White 2 – Red 3 – Green 4 – Blue 5 – Yellow 6 – Orange	
8	Font	Byte	0 – Font 1 (default) 1 – Font 2 2 – Font 3 3 – Font 4	
9	Font Size	Byte		
10	H Pixel Position	2 Bytes		Location-Defined
12	V Pixel Position	2 Bytes		Location-Defined
14	Order by	Byte	0 – One Line 1 – Lines (default)	Location-Defined

15	Field Length	Byte		
16	Label Text	40 Bytes max	ASCII String	

2.33 Update Display Overlay

Message Structure:

D-BYTE	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	34	
	
	Data Length	Byte	42	
0	Frame Buffer#	Byte		
1	Overlay ID#	Byte		
2	Text	40 Bytes max	ASCII String	

Reply Message Structure: ACK/NAK

2.34 Overlay Operations

Note: This command makes operations in all fields:
It allows to hide/show all the fields at once, or delete the database of all the fields.

Message Structure:

D-BYTE	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	35	
	
	Data Length	Byte		
0	Frame Buffer#	Byte		
1	Operation	Byte	0 – Hide 1 – Show 2 – Delete	

Reply Message Structure: ACK/NAK

2.35 Get Overlay Items Count

Message Structure:

D-BYTE	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	36	
	

	Data Length	Byte	1	
0	Frame Buffer#	Byte		

Reply Message Structure:

D-BYTE	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	36	
	
	Data Length	Byte	2	
0	Frame Buffer#	Byte		
1	Overlay Items Count	Byte		

2.36 Overlay Operations

Message Structure:

D-BYTE	Field Name	Data Type	Value	Depended	Notes
		
	Message Number	Byte	35		
		
	Data Length	Byte	2		
0	Reserve	Byte	0		
1	Operation	Byte	0 – Hide 1 – Show 2 – Delete		

Reply Message Structure: ACK/NAK

Note: This command makes operations in all fields:
It allows to hide/show all the fields at once, or delete the database of all the fields.

2.37 Set ZIXI Params

Message Structure:

D-BYTE	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	37	
	
	Data Length	Byte	98	
0	Mode	Byte	0 – Off 1 - On	
1	In Port	2 Bytes		

3	Host	40 Bytes		
43	Out Port	2 Bytes		
45	Latency	4 Bytes	MS	
49	Bitrate	4 Bytes	KBPS	
53	Enforce Bitrate	4 Bytes	KBPS	
57	Channel Name	40 Bytes	ASCII String	
97	Log Level	Byte	-1 - None 0 - All 1 - Debug 2 - Info 3 - Warnings 4 - Errors 5 - Fatal	

Reply Message Structure: ACK/NAK

2.38 Get ZIXI Params

Message Structure:

D-BYTE	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	38	
	
	Data Length	Byte	0	

Reply Message Structure:

D-BYTE	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	38	
	
	Data Length	Byte	98	
0	Mode	Byte	0 - On 1 - Off	
1	In Port	2 Bytes		
3	Host	40 Bytes		
43	Out Port	2 Bytes		
45	Latency	4 Bytes	MS	
49	Bitrate	4 Bytes	KBPS	
53	Enforce Bitrate	4 Bytes	KBPS	
57	Channel Name	40 Bytes	ASCII String	
97	Log Level	Byte	-1 - None 0 - All 1 - Debug 2 - Info 3 - Warnings 4 - Errors	

			5 – Fatal	
--	--	--	-----------	--

2.39 Get ZIXI Status

Message Structure:

D-BYTE	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	39	
	
	Data Length	Byte	0	

Reply Message Structure:

D-BYTE	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	39	
	
	Data Length	Byte	9	
0	Status	Byte	0 – Not Active 1 – Connecting 2 – Connected 3 – Failure 4 – Disconnected	
1	Total Packets	4 Bytes		
2	Lost Packets	4 Bytes		

2.40 Get Capture Input Status

Note: This command can be done only after config the board as the transmitter.

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	40	
...	
Data Length	Byte	1	
Channel #	Byte	0 – SD Channel 1 1 – SD Channel 2 2 – SD Channel 3 3 – SD Channel 4 4 – HD Channel 1 5 – HD Channel 2 6 – SD Channel 5 7 – SD Channel 6	

		8 – SD Channel 7 9 – SD Channel 8	
--	--	--------------------------------------	--

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	40	
...	
Data Length	Byte	9	
Channel #	Byte	0 – SD Channel 1 1 – SD Channel 2 2 – SD Channel 3 3 – SD Channel 4 4 – HD Channel 1 5 – HD Channel 2 6 – SD Channel 5 7 – SD Channel 6 8 – SD Channel 7 9 – SD Channel 8	
Status	Byte	0 – Not Exist 1 – Unlock 2 – Lock	
Resolution - std	Byte	0 – Unknown 1 – PAL 2 – NTSC ... 0x10 – 1920X1080 0x11 - 1280X720 0x12 - 1440X487 0x13 - 1440X507 0x14 – 6600X4096	
Interlaced	Byte	0 – Unknown 1 – Noninterlaced 2 – Interlaced	
Fps	Byte	0 – Unknown 30,60,25,50	
Resolution - Width	2 Bytes		
Resolution - Heigth	2 Bytes		

2.41 Setup Streaming Mux

1. When the user select unicast the system can transmit, broadcast if the IP destination is broadcast IP, for example (192.168.0.255)
2. In RTSP protocol the user needs to enable the RTSP server because the transition IP and ports and are selected by the client.

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	41	
...	
Data Length	Byte	16	
Mux #	Byte	Mux Number 0-N	
Auto	Byte	BIT 0 – Auto Streaming BIT 1 – Auto Recording BIT 2 – Auto Display BIT 3 - VMD 0 – Off 1 – On	
Video Channel	Byte	-1 - None 0 – SD1 1 – SD2 2 – SD3 3 – SD4 4 – HD1 5 – HD2 6 – SD5 7 – SD6 8 – SD7 9 – SD8 10 – Virtual Video	
Audio Channel	Byte	-1 – None 0 – Channel1 1 – Channel2	
Data Channel	Byte	-1 – None 0 – Data1 1 – Data2	
Interface	Byte	-1 - None 0 – Network 1 – Data Clock 2 – UART 3 – SPI (Xstream View)	
Protocol	Byte	0 – Private 1 – TS 2 – RTP 3 – RTSP 4 – Split 5 – ROW BMP	Interface – Network
Net Mode	Byte	0 – Unicast 1 – Multicast	Interface – Network
IP Address #1	Byte	1 st Value	Interface – Network
IP Address #2	Byte	2 nd Value	Interface – Network

IP Address #3	Byte	3 rd Value	Interface Network -
IP Address #4	Byte	4 th Value	Interface Network -
Port Address #1	Byte	1st Value	Interface Network -
Port Address #2	Byte	2nd Value	Interface Network -
UART Port	Byte		Interface UART -
FB Number	Byte		

Reply Message Structure: ACK/NAK

2.42 Get Streaming Mux

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	42	
...	
Data Length	Byte	1	
Mux #	Byte	Mux Number 0-N	

Replay Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	42	
...	
Data Length	Byte	16	
Mux #	Byte	Mux Number 0-N	
Auto	Byte	BIT 0 – Auto Streaming BIT 1 – Auto Recording BIT 2 – Auto Display BIT 3 - VMD 0 – Off 1 – On	
Video Channel	Byte	-1 - None 0 – SD1 1 – SD2 2 – SD3 3 – SD4 4 – HD1 5 – HD2 6 – SD5 7 – SD6 8 – SD7	

		9 – SD8 10 – Virtual video	
Audio Channel	Byte	-1 – None 1 – Channel1 2 – Channel2	
Data Channel	Byte	-1 – None 1 – Data1 2 – Data2	
Interface	Byte	-1 – None 0 – Network 1 – Data Clock 2 – UART	
Protocol	Byte	0 – Private 1 – TS 2 – RTP 3 – RTSP 4 – Split	Interface – Network
Net Mode	Byte	0 – Unicast 1 – Multicast	Interface – Network
IP Address #1	Byte	1 st Value	Interface – Network
IP Address #2	Byte	2 nd Value	Interface – Network
IP Address #3	Byte	3 rd Value	Interface – Network
IP Address #4	Byte	4 th Value	Interface – Network
Port Address #1	Byte	1st Value	Interface – Network
Port Address #2	Byte	2nd Value	Interface – Network
UART Port	Byte		Interface – UART
FB Number	Byte		

2.43 Set Streaming Operation

After setup of the mux and all other parameters the user can start / stop each stream

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	43	
...	
Data Length	Byte	2	
Mux #	Byte	Mux Number 0-N	

Operation	Byte	0 – Stop 1 – Start	
-----------	------	-----------------------	--

Reply Message Structure: ACK/NAK

2.44 Get Streaming State

Returns byte of status bits for each max stats (ON/OFF).

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	44	
...	
Data Length	Byte	0	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	44	
...	
Data Length	Byte	1	
Mux #	Byte	BIT 0 – Mux1 . . BIT N – MuxN 0 – Off 1 – On	

2.45 Set Audio Source mode

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	45	
...	
Data Length	Byte	10	
Volume	Byte	(Default 20)	
Channel	Byte	0 – Channel1 1 – Channel2	
Codec	Byte	Values between 0– 9	
Sample Rate	Byte	0 – 8000 1 – 44100(default) 2 – 48000	
Bitrate	3 Bytes		
Bit Per Sample	Byte	One of these: 8, 16 (default)	

Channels	Byte	1 – Mono (default) 2 – Stereo	
Input	Byte	0– Line In 1 – Mic	

Reply Message Structure: ACK/NAK

2.46 Get Audio Source mode

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	46	
...	
Data Length	Byte	1	
Channel	Byte	0 – Channel1 1 – Channel2	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	46	
...	
Data Length	Byte	10	
Volume	Byte	(Default 20)	
Channel	Byte	0 – Channel1 1 – Channel2	
Codec	Byte	Values between 0– 9	
Sample Rate	Byte	0 – 8000 1 – 44100(default) 2 – 48000	
Bitrate	3 Bytes		
Bit per sample	Byte	One of these: 8, 16 (default)	
Channels	Byte	1 – Mono (default) 2 – Stereo	
Input	Byte	0– Line In 1 – Mic	

2.47 Set Video Encoding Quality

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	47	
...	
Data Length	Byte	6	

Mux #	Byte	0 – Mux 1 1 – Mux 2 2 – Mux 3 3 – Mux 4	
Encode Mode	Byte	0 – VBR 1 – CBR (default) 2 – VBR Block	
Bitrate	Byte	N*100Kbs (default 30)	CBR
Gop	Byte	(default 30)	
IQ	Byte	-1 – auto 0 – 51 - values: 0 – Highest, 51 – Lowest (default -1)	CBR
QL	Byte	0 – 50	VBR

Reply Message Structure: ACK/NAK

2.48 Get Video Encoding Quality

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	48	
...	
Data Length	Byte	1	
Mux #	Byte	See video channels	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	48	
...	
Data Length	Byte	6	
Mux #	Byte	0 – Mux 1 1 – Mux 2 2 – Mux 3 3 – Mux 4	
Encode Mode	Byte	0 – VBR 1 – CBR	
Bitrate	Byte	N*100Kbs (default 30)	CBR
Gop	Byte	30 (default)	CBR
IQ	Byte	-1 – auto 0 – 51 – values (default -1)	CBR
QL	Byte	0 – 50	VBR

2.49 Set Video Frame Rate

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	49	
...	
Data Length	Byte	4	
Mux #	Byte	0 – Mux 1 1 – Mux 2 2 – Mux 3 3 – Mux 4	
Frame Rate	Byte	0 – Full (default) 1 – Time lapse	
Time Laps	Byte	30(default) FPS	Time lapse
Reserved	Byte		

Reply Message Structure: ACK/NAK

2.50 Get Video Frame Rate

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	50	
...	
Data Length	Byte	1	
Mux #	Byte	See video channels	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	50	
...	
Data Length	Byte	4	
Mux #	Byte	0 – Mux 1 1 – Mux 2 2 – Mux 3 3 – Mux 4	
Frame Rate	Byte	0 – Full 1 – Time lapse	
Time Laps	Byte	Fps	Time lapse
Reserved	Byte		

2.51 Set ROI

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	51	
...	
Data Length	Byte	8	
Mux #	Byte	0 – Mux 1 1 – Mux 2 2 – Mux 3 3 – Mux 4	
Mode	Byte	0 – Disabled(default) 1 – ROI Zoom 2 – Auto Full D1 3 – Auto 4CIF 4 – Auto CIF 5 - Scale	
Source Width	Byte	[1-255] 1= 8,2-16,..	
Source Height	Byte	[1-255] 1= 8,2-16,..	
Source X	Byte	[1-255] 1= 8,2-16,..	
Source Y	Byte	[1-255] 1= 8,2-16,..	
Dist Width	Byte	[1-255] 1= 8,2-16,..	
Dist Height	Byte	[1-255] 1= 8,2-16,..	

Reply Message Structure: ACK/NAK

2.52 Get ROI

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	52	
...	
Data Length	Byte	1	
Mux #	Byte	0 – Mux 1 1 – Mux 2 2 – Mux 3 3 – Mux 4	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	52	
...	
Data Length	Byte	8	
Mux #	Byte	0 – Mux 1 1 – Mux 2	

		2 – Mux 3 3 – Mux 4	
Mode	Byte	0 – Disabled(default) 1 – ROI Zoom 2 – Auto Full D1 3 – Auto 4CIF 4 – Auto CIF 5 – Scale	
Source Widgh	Byte	[1-255] 1= 8,2-16,.	
Source Height	Byte	[1-255] 1= 8,2-16,.	
Source X	Byte	[1-255] 1= 8,2-16,.	
Source Y	Byte	[1-255] 1= 8,2-16,.	
Dist Width	Byte	[1-255] 1= 8,2-16,.	
Dist Height	Byte	[1-255] 1= 8,2-16,.	

2.53 Set Capture Overlay

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	53	
...	
Data Length	Byte	9 + text length	
Mux ID#	Byte		
Overlay ID#	Byte	Number from 0 – 3	
Mode	Byte	0 – Disabled(default) 1 – Enabled	
Location	Byte	0 – Top-R 1 – Top-L 2 – Bottom-R (default) 3 – Bottom-L 4 – Defined	
Type	Byte	0 – user text 1 – time + frame no 2 – GPS	
Color	Byte	0 – Black 1 – White 2 – Red 3 – Green 4 – Blue 5 – Yellow 6 – Orange	
H Pixel Position	Byte	N/8	Defined
V Pixel Position	Byte	N/8	Defined
Order by	Byte	0 – One Line 1 – Lines	

Text	80 Bytes	ASCII String	
------	----------	--------------	--

Reply Message Structure: ACK/NAK

2.54 Get Capture Overlay

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	54	
...	
Data Length	Byte	2	
Mux ID#	Byte		
Overlay ID#	Byte	Number from 0 – 3	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	54	
...	
Data Length	Byte	9 + text length	
Mux ID#	Byte		
Overlay ID#	Byte	Number from 0 – 3	
Mode	Byte	0 – Disabled(default) 1 – Enabled	
Location	Byte	0 – Top-R 1 – Top-L 2 – Bottom-R (default) 3 – Bottom-L 4 – Defined	
Type	Byte	0 – user text 1 – time + frame no 2 – GPS	
Color	Byte	0 – Black 1 – White 2 – Red 3 – Green 4 – Blue 5 – Yellow 6 – Orange	
H Pixel Position	Byte	N/8	Defined
V Pixel Position	Byte	N/8	Defined
Order by	Byte	0 – One Line 1 – Lines	
Text	80 Bytes	ASCII String	

2.55 Get Active Interface Network

Message Structure:

D-BYTE	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	55	
	
	Data Length	Byte	1	
0	Network Name	Byte	0 – eth0 1 – usb0 2 – usb1	

Reply Message Structure:

D-BYTE	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	55	
	
	Data Length	Byte	12	
0	Network Name	Byte	0 – eth0 1 – usb0 2 – usb1	
1	Is Exist	Byte	0 – No 1 – Yes	
2	MAC Address	6 Bytes		
8	IP Address	4 Bytes		

2.56 Set Automatic Mode

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	56	
...	
Data Length	Byte	4	
Mux #	Byte	0 – Mux 1 1 – Mux 2 2 – Mux 3 3 – Mux 4	
Mode	Byte	0 – None (default) 1 – Auto Frame Rate 2 – Auto Bitrate	
Frame Rate	Byte		
Bitrate	Byte		

Reply Message Structure: ACK/NAK

2.57 Get Automatic Mode

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	57	
...	
Data Length	Byte	1	
Mux #	Byte	0 – Mux 1 1 – Mux 2 2 – Mux 3 3 – Mux 4	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	57	
...	
Data Length	Byte	4	
Mux #	Byte	0 – Mux 1 1 – Mux 2 2 – Mux 3 3 – Mux 4	
Mode	Byte	0 – None 1 – Auto Frame Rate 2 – Auto Bitrate	
Frame Rate	Byte		
Bitrate	Byte	X (100 Kbit/Sec)	

2.58 Set Recording Operation

After setup of the mux and all other parameters the user can start / stop each stream

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	58	
...	
Data Length	Byte	2	
Mux #	Byte	Mux Number 0-N	
Operation	Byte	0 – Stop 1 – Start	

Reply Message Structure: ACK/NAK

2.59 Get Active Mux State

Returns byte of status bits for each mux state (ON/OFF).

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	59	
...	
Data Length	Byte	0	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	59	
...	
Data Length	Byte	4	
Mux 1	Byte	BIT 0 – Streaming BIT 1 – Recording BIT 2 – JPEG BIT 3 – Display 0 – Off 1 – On	
Mux 2	Byte	BIT 0 – Streaming BIT 1 – Recording BIT 2 – JPEG BIT 3 – Display 0 – Off 1 – On	
Mux 3	Byte	BIT 0 – Streaming BIT 1 – Recording BIT 2 – JPEG BIT 3 – Display 0 – Off 1 – On	
Mux 4	Byte	BIT 0 – Streaming BIT 1 – Recording BIT 2 – JPEG BIT 3 – Display 0 – Off 1 – On	

2.60 Get Audio Codec

If Codec is not exist return NAK

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	61	
...	
Data Length	Byte	1	
Codec #	Byte	Values between 0 – 9	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	61	
...	
Data Length	Byte	21	
Codec #	Byte	Values between 0– 9	
Codec Name	String	20 Bytes	

2.61 Set Stabilizer Search Direction

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	62	
...	
Data Length	Byte	8	
Mux	Byte	Mux Number 0-N	
Mode	Byte	0- Disable 1- Horizontal 2- Vertical 3- Both sides	
Reserved	Byte	6	

Reply Message Structure: ACK/NAK

2.62 Get Stabilizer Search Direction

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	63	
...	
Data Length	Byte	1	
MUX	Byte	0-N	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	63	
...	
Data Length	Byte	8	
Mux	Byte	Mux Number 0-N	
Mode	Byte	0 – Disable 1 – Horizontal 2 – Vertical 3 – Both sides	
Reserved	Byte	6	

2.63 Set Stabilizer Params

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	64	
...	
Data Length	Byte	4	
Mux	Byte	Mux Number 0-N	
Mode	Byte	0 – Disable 1 – Staring 2 – Movie	
Center	2 Bytes	Minimum value 60	

Reply Message Structure: ACK/NAK

2.64 Get Stabilizer params

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	65	
...	
Data Length	Byte	1	
MUX	Byte	0-N	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	65	
...	
Data Length	Byte	4	

Mux	Byte	Mux Number 0-N	
Mode	Byte	0 – Disable 1 – Staring 2 – Movie	
Center	2 Bytes	Minimum value 60	

2.65 Set Deinterlace Params

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	66	
...	
Data Length	Byte	3	
Mux	Byte	Mux Number 0-N	
Mode	Byte	0 – Disable 1 – Law 2 – Medium 3 – High	
Operational	Byte	0 – Hardware 1 – Software	

Reply Message Structure: ACK/NAK

2.66 Get Deinterlace params

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	67	
...	
Data Length	Byte	1	
MUX	Byte	0-N	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	67	
...	
Data Length	Byte	3	
Mux	Byte	Mux Number 0-N	
Mode	Byte	0 – Disable 1 – Law	

		2 – Medium 3 – High	
Operational	Byte	0 – Hardware 1 – Software	

2.67 Set Audio PTT

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	68	
...	
Data Length	Byte	3	
Channel#	Byte	Channel Number 0-1	
Mode	Byte	0 – Disable 1 – Enable	
GPIO Number	Byte		

Reply Message Structure: ACK/NAK

2.68 Get Audio PTT

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	69	
...	
Data Length	Byte	1	
Channel#	Byte	Channel Number 0-1	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	69	
...	
Data Length	Byte	3	
Channel#	Byte	Channel Number 0-1	
Mode	Byte	0 – Disable 1 – Enable	
GPIO Number	Byte		

2.69 Stream Player Operation

Note: The player can display each channel on several displays.

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	70	
...	
Data Length	Byte	2	
Mux #	Byte	Mux Number 0-N	
Operation	Byte	0 – Stop 1 - Start	

Reply Message Structure: ACK/NAK

2.70 Stop Stream Player (Not used)

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	71	
...	
Data Length	Byte	1	
Mux #	Byte	Mux Number 0-N	

Reply Message Structure: ACK/NAK

2.71 Get Player State

Note: Each bit from 0 to 5 represents a channel. Bit 0 stops the activity of the channel. Bit 1 starts the channel activity.
(Start/Stop Network streaming switch for each channel)

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	72	
...	
Data Length	Byte	0	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	72	
...	
Data Length	Byte	1	
Dmux #	Byte	Dmux Number 0-N	

		0 – Off 1 – On	
--	--	-------------------	--

2.72 Get Active Status

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	73	
...	
Data Length	Byte	0	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	73	
...	
Data Length	Byte	2	
Playback	Byte	0 – Off 1 – On	
Streaming	Byte	0 – Off 1 – On	

2.51 Set Playback Parameters

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	74	
...	
Data Length	Byte	25	
Demux #	Byte	0 – Demux 1 1 – Demux 2 2 – Demux 3 3 – Demux 4	
Auto	Byte	0 – Off 1 – On	
Interface	Byte	0 – Network 1 – Data Clock 2 – UART 3 – DISK	
Protocol	Byte	0 – Private 1 – TS 2 – RTP 3 – RTSP 4 – Split	Interface Network

Net Mode	Byte	0 – Unicast 1 – Multicast	Interface – Networkk
Port Address #1	Byte	1st Value	Interface – Networkk
Port Address #2	Byte	2nd Value	Interface – Networkk
IP Address #1	Byte	1 st Value	Interface – Networkk
IP Address #2	Byte	2 nd Value	Interface – Networkk
IP Address #3	Byte	3 rd Value	Interface – Networkk
IP Address #4	Byte	4 th Value	Interface –Networkk
UART Port	Byte		Interface – UART
Destination Base Port #1	Byte	1st Value	
Destination Base Port #2	Byte	2nd Value	
Destination IP #1	Byte	1 st Value	
Destination IP #2	Byte	2 nd Value	
Destination IP #3	Byte	3 rd Value	
Destination IP #4	Byte	4 th Value	
Demux Destination Mode	Byte	0 – Seperated 1 – Combined	
Video Flags	Byte	BIT 0 – FB 0 BIT 1 – FB 1 BIT 2 – FB 2 BIT 7 – Network	
Audio Flags	Byte	BIT 0 – Channel 1 BIT 1 – Channel 2 BIT 7 – Network	
Data Flags	Byte	BIT 0 – Data 1 BIT 1 – Data 2 BIT 7 – Network	
Volume	Byte	Values between 0 - 100	
Delay	2 Bytes	Ms value	

Reply Message Structure: ACK/NAK

2.52 Get Playback Parameters

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	75	
...	
Data Length	Byte	1	
Demux #	Byte	0 – Demux 1	

		1 – Demux 2 2 – Demux 3 3 – Demux 4	
--	--	---	--

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	75	
...	
Data Length	Byte	25	
Demux #	Byte	0 – Demux 1 1 – Demux 2 2 – Demux 3 3 – Demux 4	
Auto	Byte	0 – Off 1 – On	
Interface	Byte	0 – Network 1 - Data Clock 2 – UART 3 - DISK	
Protocol	Byte	0 – Private 1 – TS 2 – RTP 3 - RTSP 4 - Split	Interface – Networkk
Net Mode	Byte	0 – Unicast 1 – Multicast	Interface – Networkk
Port Address #1	Byte	1st Value	Interface – Networkk
Port Address #2	Byte	2nd Value	Interface – Networkk
IP Address #1	Byte	1 st Value	Net Mode – Multicast
IP Address #2	Byte	2 nd Value	Net Mode – Multicast
IP Address #3	Byte	3 rd Value	Net Mode – Multicast
IP Address #4	Byte	4 th Value	Net Mode – Multicast
UART Port	Byte		Interface – UART
Destination Base Port #1	Byte	1st Value	
Destination Base Port #2	Byte	2nd Value	
Destination IP #1	Byte	1 st Value	
Destination IP #2	Byte	2 nd Value	
Destination IP #3	Byte	3 rd Value	
Destination IP #4	Byte	4 th Value	

Demux Destination Mode	Byte	0 – Seperated 1 – Combined	
Video Flags	Byte	BIT 0 – FB 0 BIT 1 – FB 1 BIT 2 – FB 2 BIT 7 – Network	
Audio Flags	Byte	BIT 0 – Channel 1 BIT 1 – Channel 2 BIT 7 – Network	
Data Flags	Byte	BIT 0 – Data 1 BIT 1 – Data 2 BIT 7 – Network	
Volume	Byte	Values between 0 - 100	
Delay	2 Bytes	Ms value	

2.53 Set PIP

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	76	
...	
Data Length	Byte	4	
Mux#	Byte	Mux number 0-N	
Mode	Byte	0 – Disable 1 – Source 2 – Destination	
Size	Byte	0 – 720X576 1 – 1280X720 2 – 1920X1080	
Location	Byte	0 – Top left 1 – Top right 2 – Bottom left 3 – Bottom right 4 – Center	

Reply Message Structure: ACK/NAK

2.54 Get PIP

If uart driver is not exist return NAK

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	77	
...	
Data Length	Byte	1	

Mux#	Byte	Mux number 0-N	
------	------	----------------	--

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	77	
...	
Data Length	Byte	4	
Mux#	Byte	Mux number 0-N	
Mode	Byte	0 – Disable 1 – Source 2 – Destination	
Size	Byte	0 – 720X576 1 – 1280X720 2 – 1920X1080	
Location	Byte	0 – Top left 1 – Top right 2 – Bottom left 3 – Bottom right 4 – Center	

2.55 Set Jitter Buffer

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	78	
...	
Data Length	Byte	4	
Mux #	Byte	0 – Mux 1 1 – Mux 2 2 – Mux 3 3 – Mux 4	
Const Bitrate	Byte	0 – On 1 – Off	
Delay(ms)	Byte		
Jitter	Byte	in percentage	

Reply Message Structure: ACK/NAK

2.56 Get Jitter Buffer

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	79	

...	
Data Length	Byte	1	
Mux #	Byte	0 – Mux 1 1 – Mux 2 2 – Mux 3 3 – Mux 4	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	79	
...	
Data Length	Byte	4	
Mux #	Byte	0 – Mux 1 1 – Mux 2 2 – Mux 3 3 – Mux 4	
Const Bitrate	Byte	0 – On 1 – Off	
Delay(ms)	Byte		
Jitter	Byte	in percentage	

2.57 Get Playback Statistic

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	80	
...	
Data Length	Byte	2	
Demux #	Byte	0 – Demux 1 1 – Demux 2 2 – Demux 3 3 – Demux 4	
Reset After	Byte	0 – Not 1-Reset	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	80	
...	
Data Length	Byte	50	
Demux #	Byte		
Width	2 Bytes	Video Res width	
Height	2 Bytes	Video Res Height	
Fps	Byte	Frame Per second	

Decode Frame	Byte	255 modulo	
Decode Error	Byte	255 modulo	
Packet Error	Byte	255 modulo	
Standard Deviation Frame arrival time	Byte	Ms	

2.58 Get Video Codec

If Codec is not exist return NAK

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	81	
...	
Data Length	Byte	1	
Codec #	Byte	Values between 0 – 9	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	81	
...	
Data Length	Byte	21	
Codec #	Byte	Values between 0– 9	
Codec Name	String	20 Bytes	

2.59 Set Extended Video Codec

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	82	
...	
Data Length	Byte	52	
Mux#	Byte	Mux Number 0-N	
Codec	Byte	Values between 0– 9	
File Format	Byte	0 – TS 1 – MP4 2 - JPEG	
Color	Byte	0 – Colorful (default) 1 – Gray-Level	
TS Flush	Byte	0 – Off 1 – On	

Mirror	Byte	0 - None 1 - Vertical 2 - Horizontal 3 - Vertical & Horizontal	
Reserved	46 Bytes		

Reply Message Structure: ACK/NAK

2.60 Get Extended Video Codec

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	83	
...	
Data Length	Byte	1	
Mux#	Byte	Mux Number 0-N	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	83	
...	
Data Length	Byte	52	
Mux#	Byte	Mux Number 0-N	
Codec	Byte	Values between 0- 9	
File Format	Byte	0 - TS 1 - MP4 2 - JPEG	
Color		0 - Colorful (default) 1 - Gray-Level	
TS Flush	Byte	0 - Off 1 - On	
Mirror	Byte	0 - None 1 - Vertical 2 - Horizontal 3 - Vertical & Horizontal	
Reserved	46 Bytes		

2.61 Set Display Parameters

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	84	

...	
Data Length	Byte	20	
TV#	Byte	0 – TV1 1 – TV2	
Mode	Byte	0 – None 1 – Off 2 – PAL 3 – NTSC	
Reserved	18 Bytes		

Reply Message Structure: ACK/NAK

2.62 Get Display Parameters

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	85	
...	
Data Length	Byte	1	
TV#	Byte	0 – TV1 1 – TV2	

Replay Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	85	
...	
Data Length	Byte	20	
TV#	Byte	0 – TV1 1 – TV2	
Mode	Byte	0 – None 1 – Off 2 – PAL 3 – NTSC	
Rezerved	18 Bytes		

2.63 Get Current Mount

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	86	
...	
Data Length	Byte	0	

Replay Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	86	
...	
Data Length	Byte	1	
Device	Byte	0 – No Device 1 – sd1 2 – sda1	

2.64 Set Record Mode

Message Structure:

D-BYTE	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	87	
	
	Data Length	Byte	11	
0	Mux#	Byte	Mux Number 0-N	
1	Index Mode	Byte	0 – Off 1 – On	
2	Split Mode	Byte	0 – Auto 1 – User 2 – None	
3	Split Time	4 Bytes	In Seconds	
7	Split Naming	Byte	0 – Time+Index 1 – New Time	
8	Disc Sync	Byte	0 – Off 1 – On	
9	Sync Cycle ms	2 bytes		

Reply Message Structure: ACK/NAK

2.65 Get Record Mode

Message Structure:

D-BYTE	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	88	
	
	Data Length	Byte	1	
0	Mux#	Byte	Mux Number 0-N	

Reply Message Structure:

D-BYTE	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	88	
	
	Data Length	Byte	11	
0	Mux#	Byte	Mux Number 0-N	
1	Mode	Byte	0 – Off 1 – On	
2	Split Mode	Byte	0 – Auto 1 – User 2 – None	
3	Split Time	4 Bytes		
7	Split Naming	Byte	0 – Time+Index 1 – New Time	
8	Disc Sync	Byte	0 – Off 1 – On	
9	Sync Cycle ms	2 bytes		

2.66 Set Split Params

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	89	
...	
Data Length	Byte	25	
Mux#	Byte	Mux Number 0-N	
X Overlay	Byte	0,8,16(default),32	
Y Overlay	Byte	0,8,16(default),32	
Size Mode	Byte	0 – split size(default) 1 – split count	
Split W	Byte	(128 – 1920) / 8 Default = 1920/8	
Split H	Byte	(64 – 1080) / 8 Default = 1080/8	
Count W	Byte	1 – N	
Count H	Byte	1 – N	
Buff Pool Size	Byte		
Split Task Count	Byte		
Net Protocol	Byte		
Feedback Port #1	Byte	1st Value	
Feedback Port #2	Byte	2nd Value	
Packet Size	2 Bytes		
Mdata Delay	Byte	(0 – 6)	
Start Skip	Byte	(0 – 5)	
Rezerved	8 Bytes		

Reply Message Structure: ACK/NAK

2.67 Get Split Params

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	90	
...	
Data Length	Byte	1	
Mux#	Byte	Mux Number 0-N	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	90	
...	
Data Length	Byte	25	
Mux#	Byte	Mux Number 0-N	
X Overlay	Byte	0,8,16(default),32	
Y Overlay	Byte	0,8,16(default),32	
Size Mode	Byte	0 – split size(default) 1 – split count	
Split W	Byte	(128 – 1920) /8 Default = 1920/8	
Split H	Byte	(64 – 1080) /8 Default = 1080/8	
Count W	Byte	1 – N	
Count H	Byte	1 – N	
Buff Pool Size	Byte		
Split Task Count	Byte		
Net Protocol	Byte		
Feedback Port #1	Byte	1st Value	
Feedback Port #2	Byte	2nd Value	
Packet Size	2 Bytes		
Mdata Delay	Byte	(0-6)	
Start Skip	Byte	(0-5)	
Reserved	8 Bytes		

2.68 Set Encoder Tune

Message Structure:

Field Name	Data Type	Value	Depended
...	

Message ID	Byte	91	
...	
Data Length	Byte	43	
Mux#	Byte	Mux Number 0-N	
Constrained Intra Pred	Byte	0 – Disabled(default) 1 – Enabled	
Disabled Deblock	Byte	0 – Enabled(default) 1 – Disabled 2 – Disabled at slice	
Filter Offset Alpha	Byte	-6 to 6 (0 default)	
Filter Offset Beta	Byte	-6 to 6 (0 default)	
Chroma Qp Offset	Byte	-12 to 12 (10 default)	
AUD RBSP	Byte	0 – Disabled(default) 1 – Enabled	
Interview	Byte	0 – Disabled(default) 1 – Enabled	
SPS/PPS	Byte	0 – Disabled(default) 1 – Enabled	
Intra Refresh	Byte	0 – Disabled 1 – Enabled(default)	
Intra Refresh Size	Byte	0 – 16k	
Initial Delay	Byte	0 – 30 (0 default)	
VBV Buffer Size	Byte	0 – 16k (0 default)	
User QP Min Enable	Byte	0 – Disabled(default) 1 – Enabled	
User QP Min	Byte	0 – 51 (20 default)	
User QP Max Enable	Byte	0 – Disabled(default) 1 – Enabled	
User QP Max	Byte	0 – 51 (33 default)	
User Gamma	Byte	(0-100) -> (0 – 32768) (default 75 (75/100))	
RC Interval Mode	Byte	0 – Normal Mode 1 – Frame Level (default) 2 – Slice Level 3 - User Defined	
MB Interval	Byte	Number Of Rows (default 2 (W /16 *2))	
Intra16x16 Enable	Byte	0 – Disabled(default) 1 – Enabled	
ME Search Range	Byte	0 – Horizontal (-128 ~ 127), Vertical(-64 ~ 63) 1 - Horizontal (-64 ~ 63), Vertical(-32 ~ 31) 2 - Horizontal (-32 ~ 31), Vertical(-16 ~ 15)	

		3 - Horizontal (-16 ~ 15), Vertical(-16 ~ 15) (default)	
ME Use Zero PMV	Byte	0 – Neighbor PMV 1 - Zero PMV	
Intra Cost Weight	Byte	0 – 51 (default 0)	
Enable Auto Skip	Byte	0 – Disabled 1 – Enabled(default)	
Slice Mode	Byte	0 – Slice per picture(default) 1 – Multiple slice per picture	
Slice Size Mode	Byte	0 – By bits 1 – By MB number(default)	
Slice Size	Byte	In bits(kb) or MB(lines)	
Rezerved	15 Bytes		

Reply Message Structure: ACK/NAK

2.69 Get Encoder Tune

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	92	
...	
Data Length	Byte	1	
Mux#	Byte	Mux Number 0-N	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	92	
...	
Data Length	Byte	43	
Mux#	Byte	Mux Number 0-N	
Constrained Intra Pred	Byte	0 – Disabled(default) 1 – Enabled	
Disabled Deblock	Byte	0 – Enabled(default) 1 – Disabled 2 – Disabled at slice	
Filter Offset Alpha	Byte	-6 to 6 (0 default)	
Filter Offset Beta	Byte	-6 to 6 (0 default)	
Chroma Qp Offset	Byte	-12 to 12 (10 default)	
AUD RBSP	Byte	0 – Disabled(default) 1 – Enabled	
Interview	Byte	0 – Disabled(default) 1 – Enabled	

SPS/PPS	Byte	0 – Disabled(default) 1 – Enabled	
Intra Refresh	Byte	0 – Disabled 1 – Enabled(default)	
Intra Refresh Size	2 Bytes	0 – 16k	
Initial Delay	Byte	0 – 30 (0 default)	
VBV Buffer Size	2 Bytes	0 – 16k (0 default)	
User QP Min Enable	Byte	0 – Disabled(default) 1 – Enabled	
User QP Min	Byte	0 – 51 (20 default)	
User QP Max Enable	Byte	0 – Disabled(default) 1 – Enabled	
User QP Max	Byte	0 – 51 (33 default)	
User Gamma	Byte	(0-100) -> (0 – 32768) (default 75 (75/100))	
RC Interval Mode	Byte	0 – Normal Mode 1 – Frame Level default) 2 – Slice Level 3 - User Defined	
MB Interval	Byte	Number Of Rows (default 2 (W /16 *2))	
Intra16x16 Enable	Byte	0 – Disabled(default) 1 – Enabled	
ME Search Range	Byte	0 – Horizontal (-128 ~ 127), Vertical(-64 ~ 63) 1 - Horizontal (-64 ~ 63), Vertical(-32 ~ 31) 2 - Horizontal (-32 ~ 31), Vertical(-16 ~ 15) 3 - Horizontal (-16 ~ 15), Vertical(-16 ~ 15) (default)	
ME Use Zero PMV	Byte	0 – Neighbor PMV 1 - Zero PMV	
Intra Cost Weight	Byte	0 – 51 (default 0)	
Enable Auto Skip	Byte	0 – Disabled 1 – Enabled(default)	
Slice Mode	Byte	0 – Slice per picture(default) 1 – Multiple slice per picture	
Slice Size Mode	Byte	0 – By bits 1 – By MB number(default)	
Slice Size	Byte	In bits or MB	
Rezerved	15 Bytes		

2.70 Reset Encoder Tune

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	93	
...	
Data Length	Byte	1	
Mux#	Byte	Mux Number 0-N	

Reply Message Structure: ACK/NAK

2.71 Get HW Error Count

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	94	
...	
Data Length	Byte	0	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	94	
...	
Data Length	Byte	1	
Count	Byte		

2.72 Get HW Error Info

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	95	
...	
Data Length	Byte	1	
Index	Byte		

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	95	
...	

Data Length	Byte	63	
Device#	Byte		
Type	Byte	0 – Analog Camera 1 – Digital Camera 2 – Display 3 – Flash 4 – Memory 5 – Disk 6 – SD 7 – UART 8 – Ethernet	
Error	Byte	0 – None 1 – Chip IO 2 – Memory	
Name	40 bytes		
Rezerved	20 bytes		

Note: If the device does not exist Replay Message is NAK

2.73 Reset HW Errors

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	96	
...	
Data Length	Byte	0	

Reply Message Structure: ACK/NAK

2.74 Set TTL

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	97	
...	
Data Length	Byte	1	
TTL	Byte		

Reply Message Structure: ACK/NAK

2.75 Get TTL

Message Structure:

Field Name	Data Type	Value	Depended
------------	-----------	-------	----------

...	
Message ID	Byte	98	
...	
Data Length	Byte	0	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	98	
...	
Data Length	Byte	1	
TTL	Byte		

2.76 Erase Device

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	99	
...	
Data Length	Byte	10	
Device Name	10 Byte		

Reply Message Structure: ACK/NAK

2.77 Get Storage Information

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	100	
...	
Data Length	Byte	1	
Device#	Byte		

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	100	
...	
Data Length	Byte	50	
Device#	Byte		
Size	4 bytes	in MB	
Free Size	4 bytes	in MB	
Format	Byte	0 – FAT	

		1 – EXT(x)	
Name	20 bytes		
Reserved	20 bytes		

Note: If the device does not exist Replay Message is NAK

2.78 Get UART Device

If uart driver is not exist return NAK

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	101	
...	
Data Length	Byte	1	
UART #	Byte	Values between 0 – 9	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	101	
...	
Data Length	Byte	21	
UART #	Byte	Values between 0 – 9	
UART Name	String	20 Bytes	

2.79 Set UART Parameters for uart to eth

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	102	
...	
Data Length	Byte	23	
UART #	Byte	Values between 0– 9	
State	Byte	0 – Off(default) 1 – On	
Source IP #1	Byte	1 st Value	
Source IP #2	Byte	2 st Value	
Source IP #3	Byte	3 st Value	
Source IP #4	Byte	4 st Value	
Source Port #1	Byte	1st Value	
Source Port #2	Byte	2nd Value	
Delay	Byte	ms	
Buad Rate	Byte	0 – 2400 1 – 4800 2 – 9600 (default)	

		3 – 19200 4 – 38400 5 – 57600 6 – 115200 7 – 230400	
Flow Control	Byte	0 – None(default) 1 – HW	
Parity	Byte	0 – None (default) 1 – Odd 2 – Even 3 – Mark 4 – Space	
Size	Byte	7 - 7 bits 8 - 8 bits (default)	
Destination IP	4 Bytes		
Destination Port	2 Bytes		
Net Protocol	Byte	0 – UDP 1 – TCP	
Rezerved	3 Bytes		

Reply Message Structure: ACK/NAK

2.80 Get UART Parameters

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	103	
...	
Data Length	Byte	1	
UART #	Byte	Values between 0– 9	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	103	
...	
Data Length	Byte	23	
UART #	Byte	Values between 0– 9	
State	Byte	0 – Off 1 – On	
Source IP #1	Byte	1 st Value	
Source IP #2	Byte	2 st Value	
Source IP #3	Byte	3 st Value	
Source IP #4	Byte	4 st Value	
Source Port #1	Byte	1st Value	
Source Port #2	Byte	2nd Value	

Delay	Byte		
Baud Rate	Byte	0 – 2400 1 – 4800 2 – 9600 (default) 3 – 19200 4 – 38400 5 – 57600 6 – 115200 7 – 230400	
Flow Control	Byte	0 – None 1 – HW	
Parity	Byte	0 – None 1 – Odd 2 – Even 3 – Mark 4 – Space	
Size	Byte	7- 7 bits 8 -8 bits	
Destination IP	4 Bytes		
Destination Port	2 Bytes		
Net Protocol	Byte	0 – UDP 1 – TCP	
Rezerved	3 Bytes		

2.81 Setup Camera Driver

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	104	
...	
Data Length	Byte	13	
Camera #	Byte	Values between 0 – 9	
CSI	Byte	0 / 1	
X Offset	Byte		
Y Offset	Byte		
Picture Width	2 bytes		
Picture Height	2 bytes		
Filed Mode	Byte	0 – Progressive 1 – Filed 0 first 2 – Field 1 first	
Interface Type	Byte	0 – bt.656 1 – bt.1120 2 – external sync	
Bus Withdh	Byte	8 – 8 bits 16 – 16 bits	
Pixel Format	Byte	0 – YUYV	

		1 – YVYU 2 – UYVY 3 – VYUY 4 – RGB24 5 – BGR24	
FPS	Byte		

Reply Message Structure: ACK/NAK

2.82 Get Camera Driver

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	105	
...	
Data Length	Byte	2	
Camera #	Byte	Values between 0– 9	
CSI	Byte	0 / 1	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	105	
...	
Data Length	Byte	13	
Camera #	Byte	Values between 0 – 9	
CSI	Byte	0 / 1	
X Offset	Byte		
Y Offset	Byte		
Picture Width	2 bytes		
Picture Height	2 bytes		
Filed Mode	Byte	0 – Progressive 1 – Filed 0 first 2 – Field 1 first	
Interface Type	Byte	0 – bt.656 1 – bt.1120 2 – external sync	
Bus Witdh	Byte	8 – 8 bits 16 – 16 bits	
Pixel Format	Byte	0 – YUYV 1 – YVYU 2 – UYVY 3 – VYUY 4 – RGB24 5 – BGR24	
FPS	Byte		

2.83 Set GPIO (see new command for gpio (174))**Message Structure:**

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	106	
...	
Data Length	Byte	4	
GPIO Number	Byte		
GPIO Time	2 bytes		
Function ID	Byte		

Reply Message Structure: ACK/NAK**2.84 Get GPIO (see new command for gpio (175))****Message Structure:**

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	107	
...	
Data Length	Byte	1	
GPIO Index	Byte		

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	107	
...	
Data Length	Byte	5	
GPIO Index	Byte		
GPIO Number	Byte		
GPIO Time	2 bytes		
Function ID	Byte		

2.85 Delete GPIO (see new command for gpio (177))**Message Structure:**

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	108	
...	
Data Length	Byte	1	

GPIO Index	Byte		
------------	------	--	--

Reply Message Structure: ACK/NAK**2.86 Get GPIO Function**

If GPIO function is not exist return NAK

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	109	
...	
Data Length	Byte	1	
Function ID	Byte		

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	109	
...	
Data Length	Byte	31	
Function ID	Byte		
Function Name	30 Bytes		

2.87 Get Temperature**Message Structure:**

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	110	
...	
Data Length	Byte	0	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	110	
...	
Data Length	Byte	1	
Temperture	Byte	C	

2.88 Set RTSP Server**Message Structure:**

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	111	
...	
Data Length	Byte	3	
Mode	Byte	0 – Off 1 – On	
Port	2 Bytes		

Reply Message Structure: ACK/NAK

2.89 Get RTSP Server

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	112	
...	
Data Length	Byte	0	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	112	
...	
Data Length	Byte	3	
Mode	Byte	0 – Off 1 – On	
Port	2 Bytes		

2.90 Get System Info

If Codec is not exist return NAK

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	113	
...	
Data Length	Byte	1	
Codec #	Byte	Values between 0 – 9	

Reply Message Structure:

Field Name	Data Type	Value	Index
...	

Message ID	Byte	113	
...	
Data Length	Byte	166	
Board Version	20 Bytes		0
CPU Type	10 Bytes		20
CPU Number	Byte		30
CPU Speed	2 Bytes		31
Linux date	30 Bytes		33
libudvp Version	10 Bytes		63
LDVC Version	2 Bytes		73
FPGA	Byte		75
IP Address	20 Bytes		76
MAC Address	20 Bytes		96
Camera Interface	Byte		116
Rezerved	50 Bytes		

2.91 Test Display

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	114	
...	
Data Length	Byte	1	
Operation	Byte	0 – Stop 1 – Start	

Reply Message Structure: ACK/NAK

2.92 Get Display Driver Config

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	118	
...	
Data Length	Byte	2	
Frame Buffer #	Byte	Values between 0 – 1	
Display Driver#	Byte	Values between 0 – 4	

Reply Message Structure:

Field Name	Data Type	Value	Index
...	
Message ID	Byte	118	
...	

Data Length	Byte	12(+20)	
Frame Buffer #	Byte	Values between 0 – 1	
Display Driver#	Byte	Values between 0 – 4	
Driver Name	10 Bytes	String	
Rezerved	20 Bytes		

2.93 Setup Display Config

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	119	
...	
Data Length	Byte	2(+50)	
FB 0 Driver#	Byte	-1 = None Values between 0 – 9	
FB 1 Driver#	Byte	-1 = None Values between 0 – 9	
Rezerved	50 Bytes	String	

Reply Message Structure: ACK/NAK

2.94 Get Display Config

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	120	
...	
Data Length	Byte	0	
Frame Buffer #	Byte	Values between 0 – 2	

Reply Message Structure:

Field Name	Data Type	Value	Index
...	
Message ID	Byte	120	
...	
Data Length	Byte	2(+50)	
FB 0 Driver#	Byte	-1 = None Values between 0 – 9	
FB 1 Driver#	Byte	-1 = None Values between 0 – 9	
Rezerved	50 Bytes		

2.95 Get Display modes

If Codec is not exist return NAK

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	121	
...	
Data Length	Byte	2	
Frame Buffer #	Byte	Values between 0 – 2	
Mode#	Byte	Values between 0 – 9	

Reply Message Structure:

Field Name	Data Type	Value	Index
...	
Message ID	Byte	121	
...	
Data Length	Byte	43(+50)	
Frame Buffer #	Byte	Values between 0 – 2	
Mode#	Byte	Values between 0 – 9	
Mode Name	40 Bytes		
Rezerved	50 Bytes		

2.96 Get Current Display mode

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	122	
...	
Data Length	Byte	1	
Frame Buffer #	Byte	Values between 0 – 2	

Reply Message Structure:

Field Name	Data Type	Value	Index
...	
Message ID	Byte	122	
...	
Data Length	Byte	41(+50)	
Frame Buffer #	Byte	Values between 0 – 2	
Mode State	Byte	0 – None 1 – Off 2 – Active	
Mode Name	40 Bytes		Active
Rezerved	50 Bytes		

2.97 Set Display Mode

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	123	
...	
Data Length	Byte	42(+50)	
Frame Buffer #	Byte	Values between 0 – 2	
Mode State	Byte	0 – None 1 – Off 2 – Active	
Mode Name	40 Bytes		Active
Rezerved	50 Bytes		

Reply Message Structure: ACK/NAK

2.98 Set Analog Camera Params

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	124	
...	
Data Length	Byte	10(+20)	
CSI#	Byte		
Mode	Byte	0 – Off 1 – On	
X Delay	2 Bytes		
X Active	2 Bytes		
Y Delay	2 Bytes		
Y Active	2 Bytes		
Rezerved	20 Bytes		

Reply Message Structure: ACK/NAK

2.99 Get Analog Camera Params

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	125	
...	
Data Length	Byte	1	

CSI#	Byte		
------	------	--	--

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	125	
...	
Data Length	Byte	10(+20)	
CSI#	Byte		
Mode	Byte	0 – Off 1 – On	
X Delay	2 Bytes		
X Active	2 Bytes		
Y Delay	2 Bytes		
Y Active	2 Bytes		
Rezerved	20 Bytes		

2.100 Reset Analog Camera

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	126	
...	
Data Length	Byte	1	
CSI#	Byte		

Reply Message Structure: ACK/NAK

2.101 Set Snapshot

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	127	
...	
Data Length	Byte	4	
Mux #	Byte	0 – Mux 1 1 – Mux 2 2 – Mux 3 3 – Mux 4	
Quality	Byte	Values between 1-10	
Rezerved	10 Bytes		

Reply Message Structure: ACK/NAK

2.102 Get Snapshot

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	128	
...	
Data Length	Byte	1	
Mux #	Byte	0 – Mux 1 1 – Mux 2 2 – Mux 3 3 – Mux 4	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	128	
...	
Data Length	Byte	4	
Mux #	Byte	0 – Mux 1 1 – Mux 2 2 – Mux 3 3 – Mux 4	
Quality	Byte	Values between 1-10	
Rezerved	10 Bytes		

2.103 Snapshot

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	129	
...	
Data Length	Byte	1	
Mux #	Byte	0 – Mux 1 1 – Mux 2 2 – Mux 3 3 – Mux 4	

Reply Message Structure: ACK/NAK

2.104 Set Camera Ext

Message Structure:

Field Name	Data Type	Value	Depended
...	

Message ID	Byte	130	
...	
Data Length	Byte	6(+6)	
CSI#	Byte		
Enable	Byte	0 – Off 1 – On	
External Mode	Byte	0 – Embedded (default) 1 – External Sync	
Frame mode	Byte	0 – Non Interlaced(default) 1 – Interlaced	
Frame Type	Byte	0 – Frame (default) 1 – Field	
FPS	Byte		
Reserved	7 Bytes		

Reply Message Structure: ACK/NAK

2.105 Get Camera Ext

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	131	
...	
Data Length	Byte	1	
CSI#	Byte		

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	131	
...	
Data Length	Byte	6(+6)	
CSI#	Byte		
Enable	Byte	0 – Off 1 – On	
External Mode	Byte	0 – External Sync 1 – Embedded	
Frame mode	Byte	0 – Non Interlaced(default) 1 – Interlaced	
Frame Type	Byte	0 – Frame (default) 1 – Field	Interlaced
FPS	Byte		
Reserved	6 Bytes		

2.106 Set Emergency Boot Params**Message Structure:**

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	132	
...	
Data Length	Byte	44(+10)	
Board IP Address	22 bytes		
Server Address	22 bytes		
Rezerved	10 Bytes		

Reply Message Structure: ACK/NAK**2.107 Get Emergency Boot Params**

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	133	
...	
Data Length	Byte	0	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	133	
...	
Data Length	Byte	66(+10)	
Ethernet Address	22 bytes	Ethernet ip	
Board IP Address	22 bytes	IP Address	
Server Address	22 bytes	Server Address	
Rezerved	10 Bytes		

2.108 Set Display Operation**Message Structure:**

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	134	
...	
Data Length	Byte	2	
Mux #	Byte	0 – Mux 1 1 – Mux 2 2 – Mux 3 3 – Mux 4	
Mode	Byte	0 – Stop	

		1 – Start	
--	--	-----------	--

Reply Message Structure: ACK/NAK

2.109 Get Display Operation

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	135	
...	
Data Length	Byte	1	
Mux #	Byte	0 – Mux 1 1 – Mux 2 2 – Mux 3 3 – Mux 4	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	135	
...	
Data Length	Byte	2	
Mux #	Byte	0 – Mux 1 1 – Mux 2 2 – Mux 3 3 – Mux 4	
Mode	Byte	0 – Stop 1 – Start	

2.110 Set FPGA

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	136	
...	
Data Length	Byte	1	
Mode	Byte	0 – Off 1 – On	

Reply Message Structure: ACK/NAK

2.111 Get FPGA

Field Name	Data Type	Value	Depended
------------	-----------	-------	----------

...	
Message ID	Byte	137	
...	
Data Length	Byte	0	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	137	
...	
Data Length	Byte	1	
Mode	Byte	0 – Off 1 – On	

2.112 Set Cellular Network

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	138	
...	
Data Length	Byte	42	
Modem Number	Byte		
Enabled	Byte	0 – Off 1 – On	
APN Name	40 Bytes		

Reply Message Structure: ACK/NAK

2.113 Get Cellular Network

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	139	
...	
Data Length	Byte	0	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	139	
...	
Data Length	Byte	42	
Modem Number	Byte		
Enabled	Byte	0 – Off	

		1 – On	
APN Name	40 Bytes		

2.114 Get Cellular Network Status

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	140	
...	
Data Length	Byte	1	
Modem Number	Byte		

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	140	
...	
Data Length	Byte	4	
Modem Number	Byte		
State	Byte	0 – Disconnected 1 – Connected	
Signal Strength	Byte	0 – Weak (-96 and lower) 1 – Low (-86 to -95) 2 – Medium (-71 to -85) 3 – High (-10 to -70)	
EC/IO	Byte	0 – Weak (-9.0 +) 1 – Low (-6.0 to -9.0) 2 – Medium (-1.5 to -5.9) 3 – High (0 to -1.5)	

2.115 Set Time Source

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	141	
...	
Data Length	Byte	2	
Source	Byte	0 – User (default) 1 – GPS	
Offset	Byte		

Reply Message Structure: ACK/NAK

2.116 Get Time Source

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	142	
...	
Data Length	Byte	0	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	142	
...	
Data Length	Byte	1	
Source	Byte	0 – User (default) 1 – GPS	
Offset	Byte		

2.117 Set GPS Device**Message Structure:**

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	143	
...	
Data Length	Byte	12	
UART #	Byte		
State	Byte	0 – Off 1 – On	
Delay	Byte	ms	
Buad Rate	Byte	0 – 2400 1 – 4800 2 – 9600 (default) 3 – 19200 4 – 38400 5 – 57600 6 – 115200 7 – 230400	
Flow Control	Byte	0 – None(default) 1 – HW	
Parity	Byte	0 – None (default) 1 – Odd 2 – Even 3 – Mark 4 – Space	
Size	Byte	7 - 7 bits	

		8 - 8 bits (default)	
UTC ON	Byte	0 – Off (default) 1 – On	
UTC Time	Byte	-12 – 12 (0 default)	
GPX File	Byte	0 – Off (default) 1 – On	
KML File	Byte	0 – Off (default) 1 – On	
Sample Rate	Byte	0 – 1 hz (default) 1 – 5 hz	

Reply Message Structure: ACK/NAK

2.118 Get GPS Device

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	144	
...	
Data Length	Byte	0	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	144	
...	
Data Length	Byte	12	
UART #	Byte		
State	Byte	0 – Off 1 – On	
Delay	Byte	ms	
Buad Rate	Byte	0 – 2400 1 – 4800 2 – 9600 (default) 3 – 19200 4 – 38400 5 – 57600 6 – 115200 7 – 230400	
Flow Control	Byte	0 – None(default) 1 – HW	
Parity	Byte	0 – None (default) 1 – Odd 2 – Even 3 – Mark 4 – Space	
Size	Byte	7 - 7 bits 8 - 8 bits (default)	

UTC ON	Byte	0 – Off (default) 1 – On	
UTC Time	Byte	-12 – 12 (0 default)	
GPX File	Byte	0 – Off (default) 1 – On	
KML File	Byte	0 – Off (default) 1 – On	
Sample Rate	Byte	0 – 1 hz (default) 1 – 5 hz	

2.119 Get Usb To Disk Mode

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	145	
...	
Data Length	Byte	0	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	145	
...	
Data Length	Byte	1	
Mode	Byte	Value	

2.120 Set Usb To Disk Mode

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	146	
...	
Data Length	Byte	1	
Mode	Byte	Value	

Reply Message Structure: ACK/NAK

2.121 Get Gps Statust

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	147	

...	
Data Length	Byte	0	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	147	
...	
Data Length	Byte	112	
Signal	Byte	0 – No Data 1 – Bad Data 2 – Good Data	
Sattellites	Byte	Value	
System time	30 Bytes		
GPS time	30 Bytes		
Speed	10 Bytes		
Coordinates	40 bytes		

2.122 Get Gps OSD

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	148	
...	
Data Length	Byte	0	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	148	
...	
Data Length	Byte	4	
Speed OSD	Byte	0 – Off 1 – On	
Speed Format	Byte	0 – KT 1 – MPH 2 – KPH	
Coordinates OSD	Byte	0 – Off 1 – On	
Coordinates Format	Byte	0 – Degrees,Min,Sec 1 – Degrass,Min 2 – Degrass	

2.123 Set GPS OSD

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	148	
...	
Data Length	Byte	4	
Speed Format	Byte	0 – Off 1 – On	
Speed OSD	Byte	0 – KT 1 – MPH 2 – KPH	
Coordinates OSD	Byte	0 – Off 1 – On	
Coordinates Format	Byte	0 – Degrees,Min,Sec 1 – Degress,Min 2 – Degress	

Reply Message Structure: ACK/NAK

2.124 Get Device Count

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	150	
...	
Data Length	Byte	0	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	150	
...	
Data Length	Byte	1	
Device Count	Byte	Value	

2.125 Get File Count

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	151	

...	
Data Length	Byte	1	
Device No	Byte	Values between 0 – N	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	151	
...	
Data Length	Byte	1	
File Count	Byte	Value	

2.126 Get File Info

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	152	
...	
Data Length	Byte	2	
Device No	Byte	Values between 0 – N	
File Index	Byte		

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	152	
...	
Data Length	Byte	54	
File Name	40 Bytes		
Type	Byte	0 – Transport 1 – MPEG4 2 – JPEG	
Size KB	4 Bytes	Size in KB	
Seconds	4 Bytes	Size in seconds	
Flags	4 Bytes	BIT 0 – Has index BIT 1 – In producion BIT 2 – Cyclic BIT 3 – Busy	
Demux	Byte	Values between 0 – N	Flags&Busy

Reply Message Structure: ACK/NAK

2.127 Play Mux

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	153	
...	
Data Length	Byte	5	
Device	Byte	Values between 0 – N	
Mux	Byte	BIT 0 – Mux 0 BIT 1 – Mux 1 ... BIT N – Mux N	
Demux	Byte	Values between 0 – N	
Direction	Byte	0 – Forward 1 – Previous	
Speed	Byte	0 – Pause 1 - Normal 2 – X2 3 – X3 4 – X4 5 – X5 100 – Slow	

Reply Message Structure: ACK/NAK

2.128 Seek File

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	155	
...	
Data Length	Byte	54	
Full File Name	50 Bytes		
Seconds	4 Bytes	Size in seconds	

Reply Message Structure: ACK/NAK

2.129 Play File

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	156	
...	
Data Length	Byte	53	

Full Path For File Name (See Note)	50 Bytes		
Demux #	Byte	Mux Number 0-N	
Direction	Byte	0 – Forward 1 – Previous	
Speed	Byte	1 – Normal 6 – Fast	

Reply Message Structure: ACK/NAK

2.130 Stop Play File

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	157	
...	
Data Length	Byte	1	
Demux	Byte	Values between 0 – N	

Reply Message Structure: ACK/NAK

2.131 Get Play Status

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	158	
...	
Data Length	Byte	1	
Demux	Byte	Values between 0 – N	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	158	
...	
Data Length	Byte	64	
State	Byte	0 – Off 1 – Playing 2 – Pausing 3 – End of file 4 – Start of file	
Full Path For File Name (see note in 156 Message ID)	50 Bytes		
Current Seconds	4 Bytes		
Direction	Byte	0 – Forward	

		1 – Previous	
Type	Byte	0 – Stream 1 – File	
Total Seconds	4 Bytes		
Mux Flags	Byte	BIT 0 – Mux 1 BIT 1 – Mux 2 BIT 2 – Mux 3 BIT 3 – Mux4 0 – Off 1 – On	
Speed	Byte	0 – Pause 1 – Normal 6 – Fast	
Is Recording	Byte	0 – Off 1 – On	

2.132 Delete File

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	159	
...	
Data Length	Byte	50	
File Name	50 Bytes		

Reply Message Structure: ACK/NAK

2.133 Set Mux Tag

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	160	
...	
Data Length	Byte	2	
Mux/Demux	Byte	Values between 0 – N	
Type	Byte	0 – Mux 1 – Demux	

Reply Message Structure: ACK/NAK

2.134 Get Tag Count

Field Name	Data Type	Value	Depended
------------	-----------	-------	----------

...	
Message ID	Byte	161	
...	
Data Length	Byte	50	
Full Name	50 Bytes		

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	162	
...	
Data Length	Byte	54	
Full File Name	50 Byte	Values between 0 – N	
Tag No	4 Bytes		

2.135 Get Tag Seconds

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	162	
...	
Data Length	Byte	54	
Full File Name	50 Byte	Values between 0 – N	
Tag No	4 Bytes		

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	162	
...	
Data Length	Byte	4	
Second	4 Bytes		

2.136 Update Player

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	163	
...	
Data Length	Byte	3	
Demux	Byte	Values between 0 – N	
Direction	Byte	0 – Forward 1 – Previous	

Speed	Byte	0 – Pause 1 – Normal 2 – X2 3 – X3 4 – X4 5 – X5 6 – Fast	
-------	------	---	--

Reply Message Structure: ACK/NAK

2.137 Format Device

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	164	
...	
Data Length	Byte	2	
Device Number	Byte	Values between 0 – N	
FS	Byte	1 – FAT32 2 – EXT4	

Reply Message Structure: ACK/NAK

2.138 Get Format Device Status

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	165	
...	
Data Length	Byte	0	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	165	
...	
Data Length	Byte	1	
Status	Byte	0 – Off 1 – Busy	

2.139 Maris Messages

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	168	

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	169	

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	170	

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	171	

Reply Message Structure: ACK/NAK

2.140 Set Clip Mode

D-BYTE	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	172	
	
	Data Length	Byte	10	
0	FB Number	Byte	0/ 1	
1	Type	Byte	0 – Off 1 – Relative	
2	X	2 Bytes		
4	Y	2 Bytes		
6	Crop - W	2 Bytes		
8	Crop - H	2 Bytes		

Reply Message Structure: ACK/NAK

2.141 Get Clip Mode

D-BYTE	Field Name	Data Type	Value	Depended
	

	Message ID	Byte	173	
	
	Data Length	Byte	1	
0	FB Number	Byte	0/ 1	

Reply Message Structure:

D-BYTE	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	173	
	
	Data Length	Byte	1	
0	FB Number	Byte	0/ 1	
1	Type	Byte	0 – Off 1 – Relative	
2	X	2 Bytes		
4	Y	2 Bytes		
6	Crop - W	2 Bytes		
8	Crop - H	2 Bytes		

2.142 Set GPIO (new)

D-BYTE	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	174	
	
	Data Length	Byte	38	
0	Number	Byte		
1	Name	20 Bytes		
21	IO-Mode	Byte	0 – Input 1 – Output	
22	Action	Byte	0 – Toggle 1 – Start 2 – Stop 3 – Reboot	Input
23	Mux event	Byte	0 – N Mux	
24	Demux event	Byte	0 – N Mux	
25	Events	2 Bytes	Bit 0 – Record Bit 1 – Stream Bit 2 – Play Stream Bit 3 – Feedback	Input
27	Click	Byte	0 – Press 1 – Click (default) 2 – Long Press 3 – Long Click	Input

28	Click Value	Byte	0/1	Input
29	Click-During(ms)	2 Bytes		Long Press/ Long Click
31	Trigger	Byte	0 – Keep a live 1 – Camera 2 – Task	Output
32	Trigger ID	Byte	Camera# Or 0 – Record task 1 – Stream task 2 – Play stream task 3 – Playback task 4 – Display task	Camera Task
33	Pulse Mode	Byte	0 – Continues Pulse 1 – Blink Pulse	Output
34	Pulse value	Byte	0 /1	Continues Pulse
35	Blink On(ms)	Byte		Blink Pulse
36	Blink Off(ms)	Byte		Blink Pulse
37	During (ms)	Byte	0 - Not allotted time 1-255	Output

Reply Message Structure: ACK/NAK

2.143 Get GPIO by ID (new)

D-BYTE	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	175	
	
	Data Length	Byte	1	
0	ID	Byte		

Reply Message Structure:

D-BYTE	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	175	
	
	Data Length	Byte	42	
0	Number	Byte		
1	Name	20 Bytes		

21	IO-Mode	Byte	0 – Input 1 – Output	
22	Action	Byte	0 – Toggle 1 – Start 2 – Stop 3 – Shut Down	Input
23	Mux event	Byte	0 – N Mux	
24	Demux event	Byte	0 – N Mux	
25	Events	2 Bytes	Bit 0 – Record Bit 1 – Stream Bit 2 – Play Stream Bit 3 – Feedback	Input
27	Click	Byte	0 – Press 1 – Click (default) 2 – Long Press 3 – Long Click	Input
28	Click Value	Byte	0/1	Input
29	Click-During(ms)	2 Bytes		Long Press/ Long Click
31	Trigger	Byte	0 – Keep a live 1 – Camera 2 – Task	Output
32	Trigger ID	Byte	Camera# Or 0 – Record task 1 – Stream task 2 – Play stream task 3 – Playback task 4 – Display task	Camera Task
33	Activation Mode	Byte	0 – Continues Pulse 1 – Blink Pulse	Output
34	Activation on-value	Byte	0 /1	Output
35	Blink On(ms)	2 Bytes		Blink Pulse
37	Blink Off(ms)	2 Bytes		Blink Pulse
39	Long	Byte	0 – Forever 1 – Ms	Output
40	During (ms)	2 Bytes	Value	Output

2.144 Get GPIO by Number (new)

D-BYTE	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	176	
	

	Data Length	Byte	1	
0	Numnber	Byte		

Reply Message Structure:

D-BYTE	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	176	
	
	Data Length	Byte	42	
0	Number	Byte		
1	Name	20 Bytes		
21	IO-Mode	Byte	0 – Input 1 – Output	
22	Action	Byte	0 – Toggle 1 – Start 2 – Stop 3 – Reboot	Input
23	Mux event	Byte	0 – N Mux	
24	Demux event	Byte	0 – N Mux	
25	Events	2 Bytes	Bit 0 – Record Bit 1 – Stream Bit 2 – Play Stream Bit 3 – Feedback	Input
27	Click	Byte	0 – Press 1 – Click (default) 2 – Long Press 3 – Long Click	Input
28	Click Value	Byte	0/1	Input
29	Click-During(ms)	2 Bytes		Long Press/ Long Click
31	Trigger	Byte	0 – Keep a live 1 – Camera 2 – Task	Output
32	Trigger ID	Byte	Camera# Or 0 – Record task 1 – Stream task 2 – Play stream task 3 – Playback task 4 – Display task	Camera Task
33	Pulse Mode	Byte	0 – Continues Pulse 1 – Blink Pulse	Output
34	Pulse value	Byte	0 /1	Output
35	Blink On(ms)	2 Bytes		Blink Pulse

37	Blink Off(ms)	2 Bytes		Blink Pulse
39	Long	Byte	0 – Forever 1 – Ms	Output
40	During (ms)	2 Bytes	Value	Output

Reply Message Structure: ACK/NAK

2.145 Set GPIO Feedback Params

D-BYTE	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	177	
	
	Data Length	Byte	13	
0	Mode	Byte	0 – No feedback 1 – Ethernet 2 – UART	
1	IP #1	Byte	1 st Value	Ethernet
2	IP #2	Byte	2 st Value	
3	IP #3	Byte	3 st Value	
4	IP #4	Byte	4 st Value	
5	Port	2 Bytes		Ethernet
7	Uart Number	Byte		UART
8	Delay(ms)	Byte		UART
9	Baud Rate	Byte	0 – 2400 1 – 4800 2 – 9600 (default) 3 – 19200 4 – 38400 5 – 57600 6 – 115200 7 – 230400	UART
10	Flow Control	Byte	0 – None 1 – HW	UART
11	Parity	Byte	0 – None (default) 1 – Odd 2 – Even 3 – Mark 4 – Space	UART
12	Size	Byte	0 – 7 bits 1 – 8 bits	UART

Reply Message Structure: ACK/NAK

2.146 Get GPIO Feedback param

D-BYTE	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	178	
	
	Data Length	Byte	0	

Reply Message Structure:

D-BYTE	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	178	
	
	Data Length	Byte	13	
0	Mode	Byte	0 – No feedback 1 – Ethernet 2 – UART	
1	IP #1	Byte	1 st Value	Ethernet
2	IP #2	Byte	2 st Value	
3	IP #3	Byte	3 st Value	
4	IP #4	Byte	4 st Value	
5	Port	2 Bytes		Ethernet
7	Uart Number	Byte		UART
8	Delay(ms)	Byte		UART
9	Baud Rate	Byte	0 – 2400 1 – 4800 2 – 9600 (default) 3 – 19200 4 – 38400 5 – 57600 6 – 115200 7 – 230400	UART
10	Flow Control	Byte	0 – None 1 – HW	UART
11	Parity	Byte	0 – None (default) 1 – Odd 2 – Even 3 – Mark 4 – Space	UART
12	Size	Byte	0 – 7 bits 1 – 8 bits	UART

2.147 Delete GPIO (new)

D-BYTE	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	179	
	
	Data Length	Byte	1	
0	GPIO Number	Byte		

Reply Message Structure: ACK/NAK

2.148 Get GPIO value

D-BYTE	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	180	
	
	Data Length	Byte	1	
0	GPIO Number	Byte		

Reply Message Structure:

D-BYTE	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	180	
	
	Data Length	Byte	2	
0	GPIO Number	Byte		
1	GPIO value	Byte		

2.149 Set ONVIF Enable Mode

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	183	
...	

Data Length	Byte	1	
Mode	Byte	0 – Disable 1 – Enable	

Reply Message Structure: ACK/NAK

2.150 Get ONVIF Enable Mode

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	184	
...	
Data Length	Byte	0	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	184	
...	
Data Length	Byte	1	
Mode	Byte	0 – Disable 1 – Enable	

Reply Message Structure: ACK/NAK

2.151 Get USB Camera Modes Count

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	186	
...	
Data Length	Byte	1	
USB #	Byte	Value between 0– 1	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	186	
...	
Data Length	Byte	2	
USB #	Byte	Value between 0– 1	
Count	Byte		

2.152 Set USB Camera Mode By ID

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	187	
...	
Data Length	Byte	2	
USB #	Byte	Value between 0– 1	
Mode ID	Byte		

Reply Message Structure: ACK/NAK

2.153 Get USB Camera Mode By ID

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	188	
...	
Data Length	Byte	2	
USB #	Byte	Value between 0– 1	
Mode ID	Byte		

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	188	
...	
Data Length	Byte	9	
USB #	Byte	Value between 0– 1	
Mode ID	Byte		
Is Active	Byte	0 – false 1 – true	
Width	2 bytes		
Height	2 bytes		
Fps	Byte		
Frame type	Byte	0 – MJPEG 1 – YUV	

2.154 Send Data by serial

Field Name	Data Type	Value	Depended
------------	-----------	-------	----------

...	
Message ID	Byte	191	
...	
Data Length	Byte	1+n bytes	
UART #	Byte	Values between 0– 9	
Data	N bytes		

Reply Message Structure: ACK/NAK

2.155 Set Virtual Video

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	195	
...	
Data Length	Byte	3	
Mux#	Byte	Mux number 0-N	
Color	Byte	0 – Current pattern 1 – White 2 – Black 3 – Gray 4 – Yellow 5 – Green 6 – Red 7 – Blue	
FPS	Byte	0 – 10 1 – 30 2 – 40	

Reply Message Structure: ACK/NAK

2.156 Get Virtual Video

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	196	
...	
Data Length	Byte	1	
Mux#	Byte	Mux number 0-N	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	

Message ID	Byte	196	
...	
Data Length	Byte	3	
Mux#	Byte	Mux number 0-N	
Color	Byte	0 – Current pattern 1 – White 2 – Black 3 – Gray 4 – Yellow 5 – Green 6 – Red 7 – Blue	
FPS	Byte	0 – 10 1 – 30 2 – 40	

2.157 Set Encryption Mode

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	197	
...	
Data Length	Byte	2	
Mux-i	Byte		
Mode	Byte	0 – disable 1 – enable	

Reply Message Structure: ACK/NAK

2.158 Get Encryption Mode

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	198	
...	
Data Length	Byte	1	
Mux-i	Byte		

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	198	
...	
Data Length	Byte	2	

Mux-i	Byte		
Mode	Byte	0 – disable 1 – enable	

2.159 Set multi stream params

D-byte	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	200	
	
	Data Length	Byte	22	
0	Mux count	Byte	0-max	
1	First mux-i	Byte		
2	Auto mode	Byte	0 – Disable 1 – Enable	
3	Base Port	2 bytes		
5	Multicast Enable	Byte	0 – No 1 – Yes	
6	Capture IP	4 bytes		Multicast Enable=Yes
10	Is record	Byte	0 – Off	
11	Is routing	Byte	1 – On	
12	Routing ip	4 bytes		Is routing=ON
16	Routing port	2 bytes		
18	Split Mode	Byte	0 – Auto 1 – Time 2 – None	
19	Seconds	2 bytes		
21	Name Mode	Byte	0 – Time+Index 1 – New Time	

Reply Message Structure: ACK/NAK

2.160 Get multi stream params

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	201	
...	
Data Length	Byte	0	

Reply Message Structure:

D-Byte	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	201	
	
	Data Length	Byte	22	
0	Mux count	Byte	0-max	
1	Starrt mux	Byte	17	
2	Auto mode	Byte	0 – Disable 1 – Enable	
3	Base Port	2 bytes		
5	Multicast enable	Byte	0 – No 1 – Yes	
6	Capture IP	4 bytes		Multicast Enable=Yes
10	Is record	Byte	0 – Off	
11	Is routing	Byte	1 – On	
12	Routing ip	4 bytes		Is routing=ON
16	Routing port	2 bytes		
18	Split Mode	Byte	0 – Auto 1 – Time 2 – None	
19	Seconds	2 bytes		Split Mode=Time
21	Name Mode	Byte	0 – Time+Index 1 – New Time	Split Mode!=None

2.161 Active multi stream

D-byte	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	202	
	
	Data Length	Byte	1	
0	Mux count	Byte	0 – Start 1 – Stop	

Reply Message Structure: ACK/NAK

2.162 Get Multi Stream status

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	203	
...	
Data Length	Byte	0	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	203	
...	
Data Length	Byte	34	
Mux-i	Byte		
Mux Count	Byte		
Mux 0 – status	Byte	Bit 0 – Active	
Mux 1 – status	Byte	Bit 1 – Capture	
...	Byte	Bit 2 – Record	
Mux-N status	Byte	Bit 3 – Routing	

2.163 Set Fast Encode Params

D-byte	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	204	
	
	Data Length	Byte	6	
0	Mux#	Byte	0 – mux 0 1 – mux 1 ... N-1 mux N-1	
1	Active flag	Byte	Bit 0 – kbps Bit 1 – fps Bit 2 – scale	
2	Bitrate	Byte	Out of 100 (for example- 70 for 7000 kbps)	
3	Fps	Byte		
4	Width	Byte	Out of 8 (for example- 60 for 480)	
5	Height	Byte		

Reply Message Structure: ACK/NAK

2.164 Set VMD Params

D-byte	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	205	

	
	Data Length	Byte	8	
0	Mux #	Byte	Mux Number 0-N	
1	Size Sensitivity	Byte		
2	Motion Sensitivity	Byte		
3	Frame Distance	Byte		
4	Activity(ms)	Byte		
5	Stop Action After(sec)	Byte		
6	Event Flags	Byte	BIT 0 – Start action BIT 1 – Stop action	
7	Show rectangle	Byte	Disabled – 0 Enabled – 1	

Reply Message Structure: ACK/NAK

2.165 Get VMD Params

D-Byte	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	206	
	
	Data Length	Byte	1	
0	Mux #	Byte	Mux Number 0-N	

Reply Message Structure:

D-Byte	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	206	
	
	Data Length	Byte	8	
0	Mux #	Byte	Mux Number 0-N	
1	Size Sensitivity	Byte		
2	Motion Sensitivity	Byte		
3	Frame Distance	Byte		
4	Activity(ms)	Byte		
5	Stop Action After(sec)	Byte		
6	Event Flags	Byte	BIT 0 – Start action BIT 1 – Stop action	
7	Show rectangle	Byte	Disabled – 0 Enabled – 1	

2.166 Get Update Status

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	253	
...	
Data Length	Byte	0	

Reply Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	253	
...	
Data Length	Byte	1	
Status	Byte	0 – Off 1 – Busy	

2.167 Set Passes Through Params

D-byte	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	207	
	
	Data Length	Byte	3	
0	Mode	Byte	0 – Off 1 – On	
1	Vin to -	Byte	0 – Vin->Vout 1 – Vin->Vselect	On
2	Vselect	Byte	0 – 1.8V 1 – 3.3V 2 – 5V 3 – 12V	Vin->Vselect

Reply Message Structure: ACK/NAK

2.168 Get Passes Through Params

Message Structure:

Field Name	Data Type	Value	Depended
...	
Message ID	Byte	208	
...	
Data Length	Byte	0	

Reply Message Structure:

D-byte	Field Name	Data Type	Value	Depended
	
	Message ID	Byte	208	
	
	Data Length	Byte	3	
0	Mode	Byte	0 – Off 1 – On	
1	Vin to -	Byte	0 – Vin->Vout 1 – Vin->Vselect	On
2	Vselect	Byte	0 – 1.8V 1 – 3.3V 2 – 5V 3 – 12V	Vin->Vselect

3 LDVC Config File

The ldvc.cfg file save the parameters that the user sets by command or by self-editing and the system uses them.

If the user doesn't put values the system uses the default values we set.

That is example of ldvc.cfg file:

```
ldvc :
{
  network_mode :
  {
    mode = 1;
    ip = "192.168.0.40";
    mask = "255.255.255.0";
    gateway = "192.168.0.8";
    mac_addr = "0a:3a:4e:db:47:38";
  };
  camera_setup :
  {
    csi1 = -1;
    csi2 = -1;
  };
  mux_0 :
  {
    mux_setup :
    {
      auto_activ = 0;
      video_src = 1;
      audio_src = 1;
      data_src = -1;
      interface = 0;
      netmode = 0;
      protocol = 1;
      ip = "192.168.0.8";
      port = 1235;
      uart_no = 0;
      pretend_auto = 0;
    };
    video_frame_rate :
    {
      frame_rate = 0;
      time_laps = 30;
      deinterlace = 0;
    };
    video_encode_quality :
    {
      encodemode = 1;
      bitrate = 7000;
      gop = 30;
    };
  };
}
```

```
iq = 255;
ql = 23;
};
bitrate_setup :
{
  bitrate_pres = 0;
  jitter_flag = 0;
};
roi :
{
  mode = 0;
  source_width = 720;
  source_height = 480;
  source_x = 0;
  source_y = 0;
  dist_width = 720;
  dist_height = 480;
};
video_stabilizer :
{
  mode = 0;
  margin_w = 10;
  margin_h = 10;
};
};
mux_1 :
{
  mux_setup :
  {
    auto_activ = 0;
    video_src = 1;
    audio_src = 1;
    data_src = -1;
    interface = 0;
    netmode = 0;
    protocol = 1;
    ip = "192.168.0.8";
    port = 1238;
    uart_no = 0;
    pretend_auto = 0;
  };
};
mux_2 :
{
  mux_setup :
  {
    auto_activ = 0;
    video_src = 1;
    audio_src = -1;
    data_src = -1;
```

```
interface = 0;
netmode = 0;
protocol = 1;
ip = "192.168.0.8";
port = 1240;
uart_no = 0;
pretend_auto = 0;
};
};
rtsp_setup :
{
mode = 0;
port = 9999;
};
data_source :
{
data_0 :
{
type = 1;
ip = "225.0.0.1";
port = 0;
buad_rate = 6;
flow_ctl = 0;
};
};
serial_port :
{
uart_0 :
{
state = 0;
s_ip = "0.0.0.0";
port = 0;
delay = 10;
speed = 9600;
flow_control = 0;
};
};
ch_0 :
{
audio_source :
{
volume = 30;
quality = 1;
codec = 2;
sample_rate = 1;
bitrate = 32000;
channels = 1;
bits_per_sample = 16;
};
};
};
```

```
tv_1 :  
{  
  display_setup :  
  {  
    mode = 1;  
  };  
};  
tv_0 :  
{  
  display_setup :  
  {  
    mode = 1;  
  };  
};  
};
```