RS194 - BC-200 Command Set

No	Issue Date	Description	Apply Firmware
1	2025/05/12	First version.	VUO105

***Notice:**

1. **The command list is for BC-200.**
2. **The yellow highlight**  **means the latest update.**
3. **The blue highlight**  **means the deleted item.**

Table of Contents

1 ACK & Completion message

	Reply Packet	Note
Ack	X0 4Y FF	Y = socket number
Completion (commands)	X0 5Y FF	Y = socket number
Completion (Inquiries)	X0 50 ... FF	
X = 9 to F==>camera address + 8 , Y=1 to 2		

2 Error message

Error Packet	Description
X0 60 02 FF	Syntax Error
X0 60 03 FF	Command buffer full
X0 6Y 04 FF	Command cancelled
X0 6Y 05 FF	No socket (to be cancelled)
X0 6Y 41 FF	Command not executable
X = 9 to F==>camera address + 8, Y = socket number, Y=0 to 2, 0: Inquiry not execution	

3 Command execution cancel

	Cancel Packet	Note
Cancel	8X 2Y FF	Y = socket number
X = 1 to 7==>camera address, Y = socket number, Y=1 to 2		

4 Network Change

	Packet	Note
Address set	88 30 01 FF	Always broadcasted (Reply:88 30 0w FF w:1+Address)
Network Change	X0 38 FF	
X = 9 to F==>camera address + 8		

5 IF_Clear

	Command	Reply Packet Note
IF_Clear	8X 01 00 01 FF	X0 50 FF
IF_Clear (broadcast)	88 01 00 01 FF	88 01 00 01 FF
X = 1 to 7==>camera address (For inquiry packet)		
X = 9 to F==>camera address +8 (For reply packet)		

6 Command List

Command set	Command	Command packet	Comments
Dig-Effect	Rotation	8x 01 04 67 0p FF	p: 2=On, 3=Off (Mirror+Flip)
Ethernet	DHCP	8x 01 7C 01 0p FF	p: 2=On, 3=Off
	IP Address	8x 01 7C 02 0p 0q 0r 0s 0t 0u 0v 0x FF	address : pq.rs.tu.vx (HEX), pq = 0~255, rs = 0~255, tu = 0~255, vx = 0~255, e.g. 192.168.100.150 => 81 01 7C 02 0C 00 0A 08 06 04 09 06 FF
	Subnet Mask	8x 01 7C 03 0p 0q 0r 0s 0t 0u 0v 0x FF	address : pq.rs.tu.vx (HEX), pq = 0~255, rs = 0~255, tu = 0~255, vx = 0~255, e.g. 255.255.255.0 => 81 01 7C 03 0F 0F 0F 0F 0F 0F 00 00 FF
	Gateway	8x 01 7C 04 0p 0q 0r 0s 0t 0u 0v 0x FF	address : pq.rs.tu.vx (HEX), pq = 0~255, rs = 0~255, tu = 0~255, vx = 0~255, e.g. 192.168.100.254 => 81 01 7C 04 0C 00 0A 08 06 04 0F 0E FF
	DNS	8x 01 7C 05 0p 0q 0r 0s 0t 0u 0v 0x FF	address : pq.rs.tu.vx (HEX), pq = 0~255, rs = 0~255, tu = 0~255, vx = 0~255, e.g. 8.8.8.8 => 81 01 7C 05 00 08 00 08 00 08 00 08 FF
Power	On/Standby	8x 01 04 00 0p FF	p: 2=On, 3=Standby
System	Factory Reset (Soft)	8x 01 04 3F 03 00 FF	Reset camera setting
	Factory Reset (Hard)	8x 01 04 3F 03 01 FF	Reset camera and network setting
	Indicator LED	8x 01 7E 01 0A 05 0p FF	p: 2=On, 3=Off
	Reboot	8x 01 DE 01 FF	Set to reboot

7 Inquiry Command List

Inquiry Command	Command Packet	Inquiry Packet	Comments
Rotation Inq	8x 09 04 67 FF	y0 50 0p FF	p: 2=On, 3=Off (Mirror+Flip)
MAC Address Inq	8x 09 04 78 FF	y0 50 0a 0b 0c 0d 0e 0f 0g 0h 0i 0j 0k 0l FF	MAC Address = ab: cd: ef: gh: ij: kl
DHCP Inq	8x 09 7C 01 FF	y0 50 0p FF	p: 2=On, 3=Off
IP Address Inq	8x 09 7C 02 FF	y0 50 0p 0q 0r 0s 0t 0u 0v 0x FF	address : pq.rs.tu.vx (HEX)
Subnet Mask Inq	8x 09 7C 03 FF		
Gateway Inq	8x 09 7C 04 FF		
DNS Inq	8x 09 7C 05 FF		
Power Mode Inq	8x 09 04 00 FF	y0 50 0p FF	p: 2=On, 3=Standby
FW version Inq - Linux	8x 09 00 02 00 03 FF	y0 50 mm nn oo pp qq rr ss FF	mmnnooppqqrrss : XXXxxxx(Ascii, data range = 0x00 ~ 0x7F)
SERIAL INQ	8x 09 02 18 FF	y0 50 aa bb cc dd ee ff gg hh ii FF	aa bb cc dd ee ff gg hh ii : Serial Number (ASCII)
Model Name	8x 09 06 A0 FF	y0 50 ii jj kk ll mm nn oo pp qq rr ss tt uu FF	ii : 0x00 jj kk ll mm nn oo pp qq rr ss tt : model name (ASCII)
Indicator LED Inq	8x 09 7E 01 0A 05 FF	y0 50 0p FF	p: 2=On, 3=Off
Camera ID Inq	8x 09 7E CE FF	y0 50 aa bb cc dd ee ff gg hh ii jj kk ll FF	aa bb cc dd ee ff gg hh ii jj kk ll : Camera ID (ASCII)
Extern Camera ID Inq	8x 09 7E DE FF		aa bb cc dd ee ff gg hh ii jj kk ll : Extern Camera ID (ASCII)

8 Communication method of VISCA over IP

8.1 Communication method

VISCA over IP can process the VISCA communication between the controllers and peripheral devices using the messages that can be identified on the LAN, and sends/receives them. Because of this, VISCA over IP is not concerned about the contents of the communication between the controllers and peripheral devices. However, the VISCA communication sequence is different, depending on the types, as follows.

8.2 VISCA command

This is a command from the controller to the peripheral device. When the peripheral device receives this command, Acknowledge is returned. After completing command processing, a completion notice is returned. This command uses the socket of VISCA. The order of completion notices may be changed if the multiple commands are sent to the same peripheral device.

8.3 VISCA inquiry

This is an inquiry from the controller to the peripheral device. When the peripheral device receives this type of command, the reply for the inquiry is returned. This command does not use the socket of VISCA. The order of the replies is not changed if a multiple commands are sent.

8.4 VISCA reply

This is an Acknowledge, completion notice, reply, or error reply from the peripheral device to the controller. The classification for sending messages from the peripheral device to the controller is common.

8.5 VISCA device setting command

This is the device setting command from the controller to the peripheral device. When the peripheral device receives this classifications command, the peripheral device performs the function depend on the command.

8.6 Address

Sets the address of the peripheral device, and does not return a reply to the controller. While using VISCA over IP, the address command is not sent from the controller because a Network Change command from the peripheral device that triggers sending command is not issued.

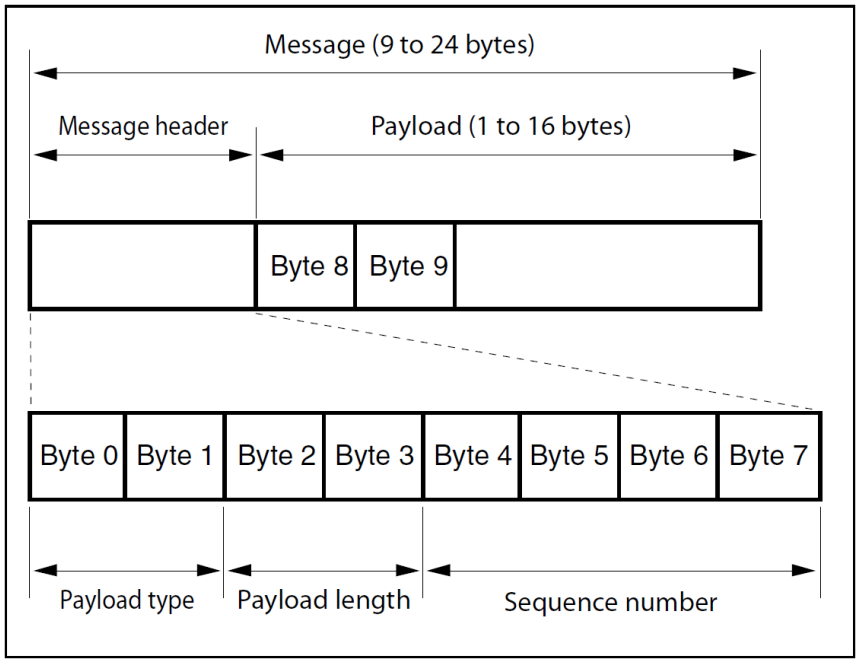
8.7 IF_Clear

Sends the reply message to the controller after clearing, without using VISCA socket.

8.8 CAM_VerslonInq

Sends the reply message to the controller, without using VISCA socket.

8.9 Format



8.10 Payload type

Stores the value (Byte 0 and Byte 1) of the following table on the payload division.

Name	Value (Byte 0)	Value (Byte 1)	Description
VISCA command	01h	00h	Stores the VISCA command.
VISCA inquiry	01h	10h	Stores the VISCA inquiry.
VISCA reply	01h	11h	Stores the reply for the VISCA command and VISCA inquiry, or VISCA device setting command.
VISCA device setting command	01h	20h	Stores the VISCA device setting command.
Control command	02h	00h	Stores the control command.
Control reply	02h	01h	Stores the reply for the control command.

8.11 Payload Length

Stores the number of bytes (1 to 16) of data is stored on the payload.
Example: when the payload length is 16 bytes.
Byte 2:00h
Byte 3:10h

8.12 Sequence number

The controller stores the sequence number that is added every time a message is sent. If the sequence number reaches the limit, next values will be 0. The peripheral device saves the sequence number in the message from the controller, and stores the sequence number of the received message corresponding to the message sent to the controller.

8.13 Payload

Depending on the payload type, the following are stored.

1. VISCA command
   - Stores the packet of the VISCA command.
2. VISCA inquiry
   - Stores the packet of VISCA message.
3. VISCA reply
   - Stores the reply for the command or inquiry (Acknowledge message, completion message, or error message).
4. VISCA device setting command
   - Stores the packet of the VISCA device setting command.
5. Control command
   - The following are stored on the payload division of the control command.

Name	Value	Description
RESET	01h	Resets the sequence number to 0. The value that was set as the sequence number is ignored.
ERROR	0Fyyh	yy=01: Abnormality in the sequence number.
		yy=02: Abnormality in the message (message type)

6. Controlled reply
   - The following are stored on the payload division of the reply for the control command.

Message	Value	Description
Acknowledge	01h	Reply for RESET.