

Table of Contents

1	Scope	2
2	BICOM in RCP+	2
2.1	CONF_BICOM_COMMAND	2
2.1.1	Payload Structure	2
2.1.2	Example	4
3	Features.....	5
3.1	Wiper	5
3.1.1	Wiper object (0x024.1).....	5
3.2	Washer	5
3.2.1	WiperWasher object (0x024.0).....	5
3.2.2	Washer object (0x024.2).....	6
3.3	Defog.....	6
3.3.1	Defog object (0x057.0)	6
3.4	Illumination mode	7
3.4.1	Illuminator.Setup object (0x041.2).....	7
3.4.2	IR on/off: Illuminator object (0x041.0).....	7
3.4.3	IR intensity: IlluminatorIntensity object (0x041.F).....	8
3.4.4	VL on/off: VisibleLightEnable object (0x041.5)	8
3.4.5	VL intensity: VisibleLightIntensity object (0x041.6)	9
3.4.6	VL timeout mode: VisibleLightTimeOutMode object (0x041.D)	9
3.4.7	VL timeout value: VisibleLightTimeOutValue object (0x041.E).....	9
3.5	Visual mode	10
3.5.1	VisualMode object (0x04C.1)	10
4	Operational considerations	10

 How to integrate Bosch MIC features

1 Scope

The scope of this document is to help through the RCP+ based integration of MIC specific features. However to be more precise the integration mode for this feature-set is to use BICOM (camera frontend controls) over a RCP+ transmission layer since the FW does not yet (current FW status 5.90) include dedicated RCP+ commands for the features.

The reader of this documentation should be acquainted with the RCP+ and the BICOM protocols, since there is no detailed explanation of the protocols to be found below. For further information the RCP+ descriptions (e.g. "Advanced integration package") and the BICOM documentation should be considered. Both documents can be found on the Bosch IPP web-portal (<http://ipp.boschsecurity.com>).

The special MIC features that are taken into account in this document are:

- Wiper on/on
- Washer on/off
- Illumination state (IR- / white light, on/off)
- Visual mode (thermal- / visible mode)
- Thermal mode selection

2 BICOM in RCP+

The following section gives a rough overview on the usage of BICOM commands in RCP+. As already mentioned before, the document is not intended to give an 'in-depth' introduction to the RCP+ protocol.

2.1 CONF_BICOM_COMMAND

	Tag code	NumDes	Message	SNMP Support
	0x09a5	no	yes	no
	Datatype	Access Level	Description	
Read	p_octet	noprot	not supported	
Write	p_octet	l_user	Sends a BICOM read or write command to the local camera frontend, see detailed description.	
			Note: for some BICOM commands an access level higher than 'l_user' is needed.	

2.1.1 Payload Structure

Bits					
0	8	16	24	32	40
Flags	Bicom Server ID	Object ID	Opera-		

 How to integrate Bosch MIC features

1 Byte	2 Byte	2 Byte	tion 1 Byte
--------	--------	--------	----------------

 Bicom Payload - n Bytes

2.1.1.1 Flags (Transmission-Flags)

Values:

Bit 0: Return_Payload
(Must be set to 1 if return payload is expected)

Bit 1: Best_Effort (Set to 1 to transmitt as best effort frame)

Bit 2: Native_Errors (Set to 1 to receive the native BICOM errors)

Bit 3: Lease_Time_Available
(Set to 1 if a lease time is included in the request)

Bit 4: unused set to 0

Bit 5: unused set to 0

Bit 6: unused set to 0

Bit 7: Flags_Available (Must be always set to 1)

2.1.1.2 Bicom Server ID

For a complete list of the Bicom Server IDs see the "BICOM application documentation". The Server ID that is needed for the MIC integration features is the 'Camera Server ID' (0x0004).

Server Type: 0x0004	Camera Server
----------------------------	---------------

2.1.1.3 Object ID

The Object ID defines the command that is addressed in the BICOM realm and can be directly mapped to MIC integration features. The following sections are going to provide a detailed overview on the Object IDs that are needed to accomplish the specific feature integration. For a complete list of the Object IDs see the "BICOM application documentation".

2.1.1.4 Operation

The Operation values define the mode of operation of the command that is sent to the BICOM server.

Values:

0x01: GET

0x02: SET

0x03: SET_GET

0x04: INC

0x05: INC_GET

From
ST-VS/MKA-NuOur Reference
spc5nu3Tel
+49 911 93456-0

Fax

Nuremberg
2015-01-12How to integrate Bosch MIC features

0x06:	DEC
0x07:	DEC_GET
0x08:	SET_DFLT
0x09:	SET_GET_DFLT

2.1.1.5 Bicom Payload

The Bicom payload as defined in the "BICOM application documentation".

2.1.2 Example

A complete RCP+ request for setting the camera's heater to ,auto' mode would then consist of the following RCP+ payload:

81	Return payload expected
00 04	Server ID = 4 („Camera server“)
02 30	Object „Heater“
02	Operation „SET“
00 01	Unsigned short value = 00 01 („Auto“)

→ The looks like: 0x8100040230020001, which results in a RCP+ over CGI call as follows:

http://<ip>/rcp.xml?command=0x09A5&type=P_OCTET&direction=WRITE&num=1&payload=0x8100040230020001

3 Features

The following sections are describing the integration features with their BICOM Server ID and the Object IDs that are linked to the features. Moreover the mode of operation and the datatype is specified for each Server object. Additionally each feature description is providing an example to show the correct usage.

3.1 Wiper

3.1.1 Wiper object (0x024.1)

Server ID	Camera	0x0004
Object	Wiper	0x024.1
Operation	GET, SET	0x01, 0x02
Values	unsigned short	0x0000: Off 0x0001: Start (start wiper only) 0x0002: <i>Intermittent</i> 0x0003: One Shot

The modes of operation for the Wiper can be described as follows:

- **Off:** Wiper is not operating
- **Start:** The wiper is operating permanently
- **Intermittent:** The wiper is operating with delays in between the wiper 'shots' (roughly one shot every 30 seconds).
- **One Shot:** The wiper is doing a single wipe

Example: "Do a 'One Shot' wipe"

http://<ip>/rcp.xml?command=0x09A5&type=P_OCTET&direction=WRITE&num=1&payload=0x8100040241020003

3.2 Washer

A washer unit is available as an extension to the MIC. There are basically 2 BICOM objects that can be used to control the washer. However one object is 'just' a sub-set of the other.

In any case the wiper is automatically affected when the washer is activated.

3.2.1 WiperWasher object (0x024.0)

Server ID	Camera	0x0004
Object	WiperWasher	0x024.0
Operation	GET, SET	0x01, 0x02

How to integrate Bosch MIC features

Values	unsigned short	0x0000 = Off 0x0001 = Start (start wiper/washer sequence)
--------	----------------	--

The procedure after setting the WiperWasher object to '1' is as follows:

1. Camera goes to preset 62 (washer preset)
2. Washer relay is turned ON (doesn't matter if the washer is connected and the tank has liquid or not)
3. Delay is executed
4. Wiper does few wipes
5. Camera returns to wherever it came from (before step 1)

This object is recommended to be used for using the washer.

Example: "Start a wiper/washer run"

http://<ip>/rcp.xml?command=0x09A5&type=P_OCTET&direction=WRITE&num=1&payload=0x8100040240020001

3.2.2 Washer object (0x024.2)

Server ID	Camera	0x0004
Object	Washer	0x024.2
Operation	GET, SET	0x01, 0x02
Values	unsigned short	0x0000 = Off 0x0001 = Start (start washer only)

The procedure after setting the Washer object to 1 is the same as for the 'WiperWasher' object (0x024.0) with the only difference that the sequence from above starts with step 2.

Note that the object 0x024.0 should be used instead of the object 0x024.2.

Example: "Start a washer run" (do not go to preset 62)

http://<ip>/rcp.xml?command=0x09A5&type=P_OCTET&direction=WRITE&num=1&payload=0x8100040242020001

3.3 Defog

3.3.1 Defog object (0x057.0)

Server ID	Camera	0x0004
Object	Anti Fog	0x057.0
Operation	GET, SET	0x01, 0x02
Values	unsigned short	0x0000: Off 0x0001: On 0x0002: Auto

From ST-VS/MKA-Nu	Our Reference spc5nu3	Tel +49 911 93456-0	Fax	Nuremberg 2015-01-12
----------------------	--------------------------	------------------------	-----	-------------------------

How to integrate Bosch MIC features

The value '0' switches the Defog off, the value '1' switches it on. In 'Auto' mode the Defog is turned on automatically when conditions demand it.

Example: "Set Defog to Auto"

http://<IP>/rcp.xml?command=0x09A5&type=P_OCTET&direction=WRITE&num=1&payload=0x8100040570020001

3.4 Illumination mode

There are basically 2 illuminators available for the MIC devices. A MIC can be equipped with infra-red lighting as well as it is possibly equipped with a 'visual-light' battery. A combination of the setup is also possible which clearly indicates the need for illumination-setup retrieval from the device in question. Such a command is available and the 'Illuminator.Setup' object can be used to retrieve the current lighting status.

It is worth mentioning that there are separate Bicom objects for the 2 lighting paths, which means that triggering and controlling the illuminators is accomplished by separate objects. Both ways will be explained in the following sections.

3.4.1 Illuminator.Setup object (0x041.2)

The Illuminator.Setup object gives an overview on the physically available illuminators on a specific MIC.

Server ID	Camera	0x0004
Object	Setup	0x041.2
Operation	GET, SET	0x01, 0x02
Values	unsigned short	0x0000 = Unknown (cannot be used in Set method) 0x0001 = no IR available 0x0002 = IR available 0x0003 = Visible White Light available 0x0004 = IR and Visible White Light available

Once the availability of a certain illuminator has been identified the following objects can be used appropriately.

Example: "Get the illuminator setup"

http://<ip>/rcp.xml?command=0x09A5&type=P_OCTET&direction=WRITE&num=1&payload=0x810004024201

3.4.2 IR on/off: Illuminator object (0x041.0)

Server ID	Camera	0x0004
-----------	--------	--------

From ST-VS/MKA-Nu	Our Reference spc5nu3	Tel +49 911 93456-0	Fax	Nuremberg 2015-01-12
----------------------	--------------------------	------------------------	-----	-------------------------

How to integrate Bosch MIC features

Object	Illuminator	0x041.0
Operation	GET, SET	0x01, 0x02
Values	unsigned short	0x0000 = Off 0x0001 = On 0x0002 = Auto

The value '0' switches the IR illuminator off, the value '1' switches it on. In 'Auto' mode the IR illuminator is turned on automatically when the lighting conditions demand for it.

Example: "Set the IR illuminator to auto mode"

http://<ip>/rcp.xml?command=0x09A5&type=P_OCTET&direction=WRITE&num=1&payload=0x8100040410020002

3.4.3 IR intensity: IlluminatorIntensity object (0x041.F)

Server ID	Camera	0x0004
Object	IlluminatorIntensity	0x041.F
Operation	GET, SET, INC_GET, INC_SET	0x01, 0x02, 0x05, 0x07
Values	unsigned short	0 ... 100

This value defines the intensity of IR illuminator in percent.

Example: "Set the IR illuminator intensity to 0x50"

http://<ip>/rcp.xml?command=0x09A5&type=P_OCTET&direction=WRITE&num=1&payload=0x810004041F020050

3.4.4 VL on/off: VisibleLightEnable object (0x041.5)

Server ID	Camera	0x0004
Object	VisibleLightEnable	0x041.5
Operation	GET, SET	0x01, 0x02
Values	unsigned short	0 = Off 1 = On

The value '0' switches the visible light illuminator off, the value '1' switches it on. In 'Auto' mode the IR illuminator is turned on automatically when the lighting conditions demand it.

Example: "Enable the visual while light"

http://<ip>/rcp.xml?command=0x09A5&type=P_OCTET&direction=WRITE&num=1&payload=0x8100040415020001

How to integrate Bosch MIC features

3.4.5 VL intensity: VisibleLightIntensity object (0x041.6)

Server ID	Camera	0x0004
Object	VisibleLightIntensity	0x041.6
Operation	GET, SET, INC_GET, INC_SET	0x01, 0x02, 0x05, 0x07
Values	unsigned short	0 ... 100

This value defines the intensity of the visible light illuminator in percent.

Example: "Set the IR illuminator intensity to 0x30"

http://<ip>/rcp.xml?command=0x09A5&type=P_OCTET&direction=WRITE&num=1&payload=0x8100040416020030

3.4.6 VL timeout mode: VisibleLightTimeOutMode object (0x041.D)

Server ID	Camera	0x0004
Object	VisibleLightTimeOutMode	0x041.D
Operation	GET, SET	0x01, 0x02
Values	unsigned short	0x0000 = Off 0x0001 = On (default)

The visual white light illuminator comes with a built-in timer that switches the lighting off after a certain timeout. For a permanent operation of the visible light this timeout mechanism can be switched off with this object.

Example: "Disable the timeout mechanism for the visible light illuminator"

http://<ip>/rcp.xml?command=0x09A5&type=P_OCTET&direction=WRITE&num=1&payload=0x810004041D020000

3.4.7 VL timeout value: VisibleLightTimeOutValue object (0x041.E)

Server ID	Camera	0x0004
Object	VisibleLightTimeOutValue	0x041.E
Operation	GET, SET	0x01, 0x02
Values	unsigned short	0 ... 30 (minutes, default 30 min)

This object defined the duration (in minutes) for the visual light timeout to happen. Of course this object is only evaluated in case the visual white light illuminator is set to use the timeout mechanism.

Example: "Set the timeout to 10 minutes"

http://<ip>/rcp.xml?command=0x09A5&type=P_OCTET&direction=WRITE&num=1&payload=0x810004041E02000A

3.5 Visual mode

The MIC 612 model can operate in 2 different visual modes such as 'regular visual mode' and in 'thermal mode'. The following object is to be used to switch between these modes:

3.5.1 VisualMode object (0x04C.1)

Server ID	Camera	0x0004
Object	VisualMode	0x04C.1
Operation	GET, SET	0x01, 0x02
Values	unsigned short	0x0000 = Thermal mode 0x0001 = Visual mode

This object switches between the thermal visual mode and the regular visual mode.

Example: "Set the visual mode to thermal"

http://<ip>/rcp.xml?command=0x09A5&type=P_OCTET&direction=WRITE&num=1&payload=0x81000404C1030000

4 Operational considerations

From a general point of view the integration features may be controlled and used in different use cases. Some of them are clearly 'operator parameters' whereas some of them might be considered to be 'administrator parameters'.

Operator features are considered to be 'on/off' switches whereas the administrator features seem to be more setup related and tend to use value related parameters. Therefore the following grouping seems to be viable for the different use cases.

Operator features	Administrator features
Wiper object (0x024.1)	
WiperWasher object (0x024.0)	
Illuminator.Setup object (0x041.2)	Illuminator.Setup object (0x041.2)
Illuminator object (0x041.0)	
	IlluminatorIntensity object (0x041.F)
VisibleLightEnable object (0x041.5)	
	VisibleLightIntensity object
	VisibleLightTimeOutMode object (0x041.D)
	VisibleLightTimeOutValue object (0x041.E)
VisualMode object (0x04C.1)	
AntiFog object 0x057.0	