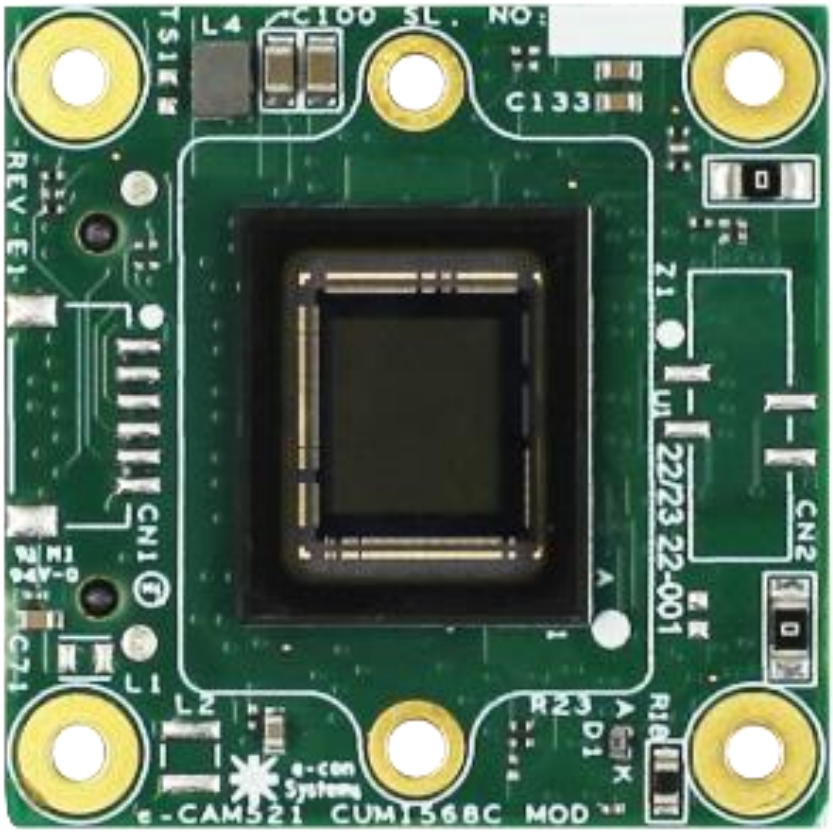
e-CAM521_ CUMI568C_MOD

MCU Protocol Application Note



e-con Systems

Think Camera. Think e-con.

Version 1.5

e-con Systems

3/22/2024

Disclaimer

The specifications and features of e-CAM521_CUMI568C_MOD_H01R1 camera board are provided here as reference only and e-con Systems reserves the right to edit/modify this document without any prior intimation of whatsoever.

Contents

Introduction to e-CAM521_CUMI568C _MOD_H01R1

e-CAM521_CUMI568C_MOD_H01R1 is a high performance, 5 MP auto low light camera module from e-con Systems, a company with over two decades of experience in designing, developing, and manufacturing OEM cameras. It is based on IMX568 CMOS image sensor from SONY® PregiusS™ family. e-CAM521_CUMI568C_MOD_H01R1 is designed to connect with any application processor that has MIPI interface.

This document is intended for developers who may try to integrate e-CAM521_CUMI568C_MOD_H01R1 with any host system other than what is supported by e-con Systems directly. This also provides a detailed understanding of I2C based protocol used by the host application processor, for communicating with the microcontroller provided as part of the e-con Systems e-CAM521_CUMI568C_MOD_H01R1 camera module.

Description

e-CAM521_CUMI568C_MOD_H01R1 is a high performance, 5 MP auto low light camera module. It is based on IMX568 CMOS image sensor from SONY® PregiusS™ family. e-CAM521_CUMI568C_MOD_H01R1 is designed to connect with any application processor that has MIPI interface. The MCU protocol is explained corresponding to the 32-byte MCU Firmware Version ID **1_2_IMX5_6_1_1_0_7a3b199**.

The following figure shows the top view of e-CAM521_CUMI568C_MOD_H01R1 camera module.

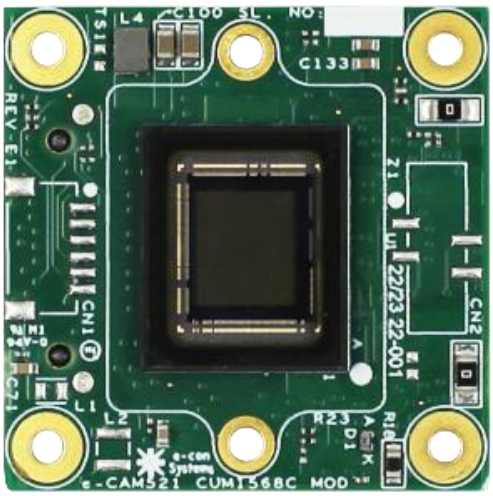


Figure 1: Top View of e-CAM521_CUMI568C_MOD_H01R1

Camera Module Features

The camera module features are as follows:

- 1/1.8" optical form-factor, 5 MP camera module
- Manual focus/Auto focus lens *
- Low light camera sensor
- External ISP
- On-board micro-controller to communicate to sensor through I2C interface
- MIPI CSI-2 video output
- Capable of high frame rate video
- Small form factor enclosure with lens
- PCB Size: 30 mm x 30 mm
- Restriction of Hazardous Substances (RoHS) compliant
- One 26-pin SMT connectors
- Operating Voltage: 3.3V ±5%, Power consumption: ~1W

CMOS Image Sensor Features

The features of the CMOS image sensor are as follows:

- IMX568 – 5 MP RAW 10/12-bit CMOS image sensor from SONY®
- RGB RAW 10-bit/12-bit Data per pixel
- 1/1.8" optical form-factor
- Unit cell size: 2.74 µm x 2.74 µm

Maximum Frame Rate Supported

The maximum frame rate supported in the e-CAM521_CUMI568C_MOD is listed in the following table.

Table 1: Maximum Frame Rate Supported

Lane	Resolution	Frame Rate in 10-bit	Frame Rate in 12-bit
4	2432 x 2048	79	67
4	1920 x 1080	142	121
4	1280 x 720	202	172
4	640 x 480	280	240

Note: The frame rates listed in the above table varies based on platform capability.

Camera Operation Sequence

This section describes the basic operation for initiating the stream sequence and changing the control values using MCU.

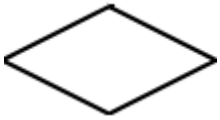
The basic operation for initiating the stream sequence is explained below:

- Host processor refers to the application processor, for example, TX1, TX2, Jetson Nano™, Xavier™ NX, Xavier AGX™, AGX ORIN™ or FX3 processors will act as I^2C master throughout this protocol.
- MCU acts as I^2C Slave in this entire protocol.
- The I^2C Master always initializes every transaction.
- Length of the byte sequence between the MCU and host processor is either constant or pre-negotiated for each transaction.
- If a transaction is from host processor to MCU, the host processor will perform the following I^2C sequence:
  - I^2C start condition.
  - 7-bit slave address of MCU.
  - Write bit.
  - Host processor provides data according to the byte sequence defined for that specific command. For more details, please refer to the *MCU Command Description* section.
  - I^2C stop condition.
- If a transaction is from MCU to host processor, the host processor will perform the following I^2C sequence:
  - I^2C start condition.
  - 7-bit slave address of MCU.
  - Read bit.
  - MCU will provide data according to the byte sequence specified in the command.
  - I^2C stop condition.
- Checksum is calculated by performing bitwise XOR of the payload data which is not same as the traditional checksum.

The legend and its description used in flowchart are listed in the following table.

Table 2: Legend and its Description

Legend	Description
	Start or Stop
	MCU Command Block

	Process Block
	Decision Block

Streaming Start Sequence

To start streaming, the sequence to be performed are as follows:

1. Send **Init Camera** command through I^2C interface.
2. Send **Get Command Status** command sequence through I^2C interface repeatedly, until the return status code is 0x0000.
3. Send **Stream Configure** command with desired frame format, width, height and frame rate.
4. Send **Get Command Status** command sequence through I^2C interface repeatedly, until the return status code is 0x0000.
5. Send **Stream ON** command to initiate streaming with updated stream configurations.
6. Send **Get Command Status** command sequence through I^2C interface repeatedly, until the return status code is 0x0000.

The flowchart of streaming start sequence is shown below.

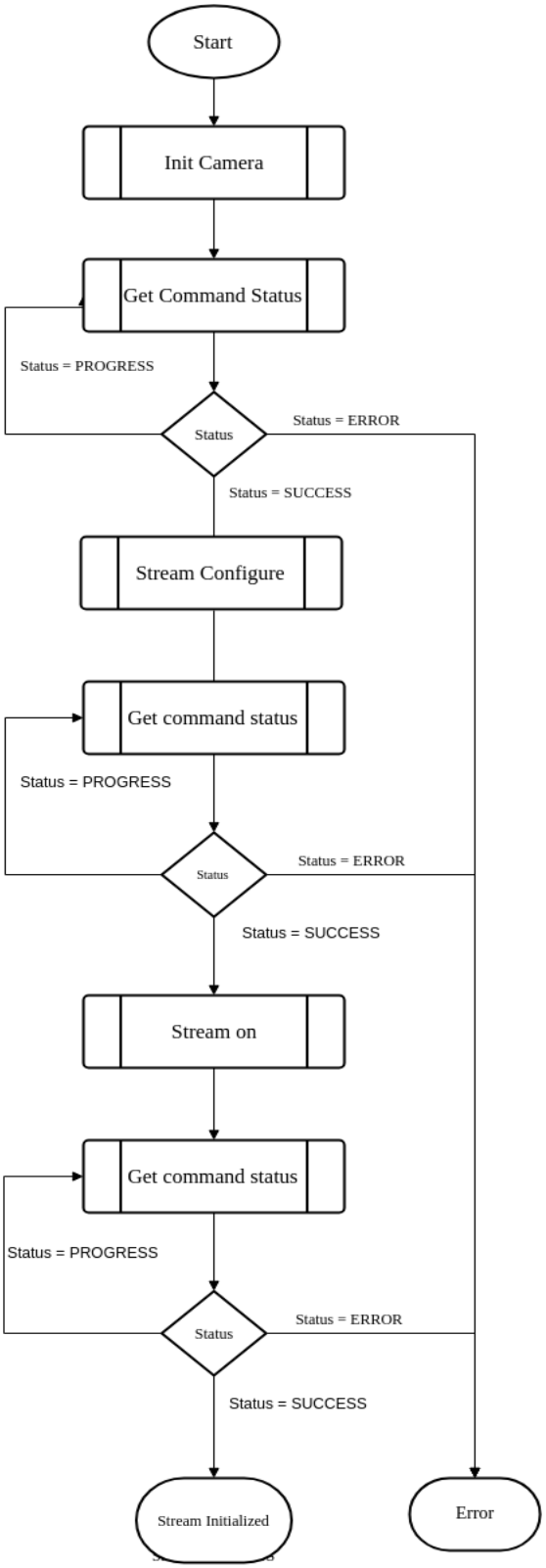


Figure 2: Flowchart of Streaming Start Sequence

Streaming Stop Sequence

To stop streaming, the sequence to be performed are as follows:

1. Send **Stream OFF** command through I²C interface.
2. Send **Get Command Status** command sequence through I²C interface repeatedly, until the return status code is 0x0000.
3. Send **De-Init Camera** command through I²C interface.
4. Send **Get Command Status** command sequence through I²C interface repeatedly, until the return status code is 0x0000.

The flowchart of streaming stop sequence is shown below.

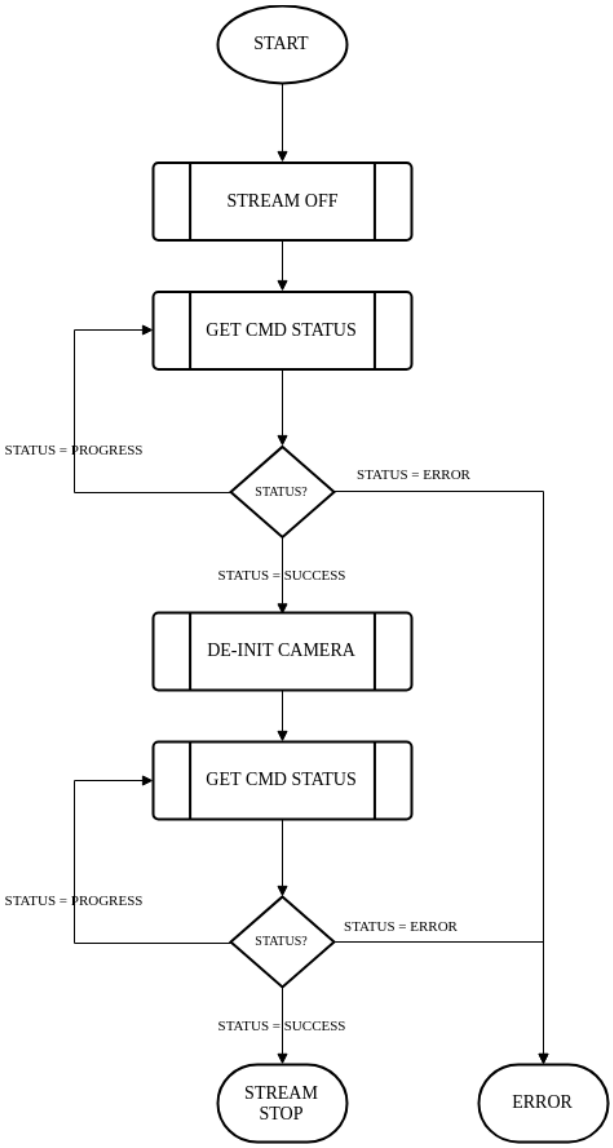


Figure 3: Flowchart of Streaming Stop Sequence

Changing Control Values

To change the control values, the sequence to be followed are as follows:

1. Ensure whether the camera is in streaming state, by performing the sequence of *Streaming Start Sequence* section.
2. Send **Set Control Value** command with desired control index, and ID value.
3. Send **Get Command Status** command sequence through I^2C interface repeatedly, until the return status code is 0x0000.

The flowchart of changing control values is shown below.

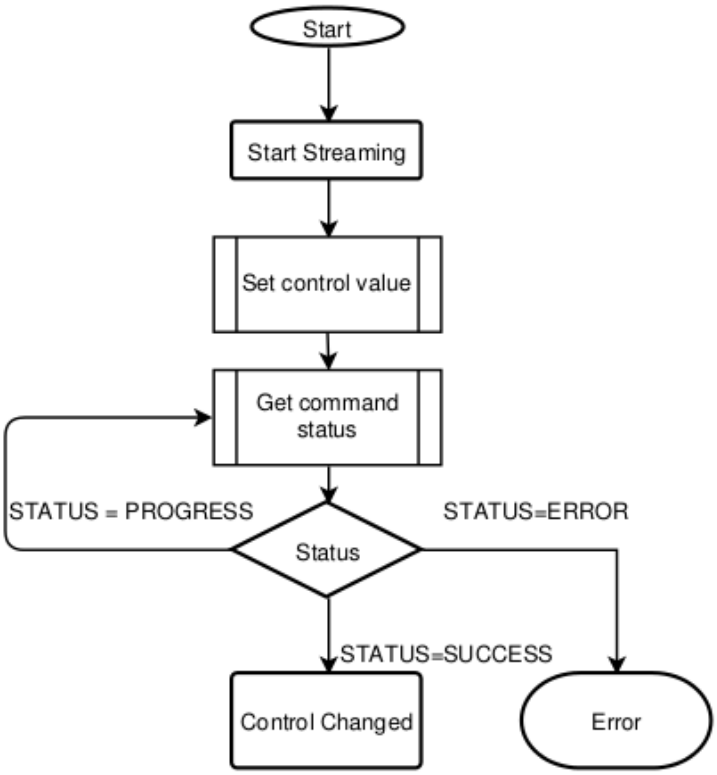


Figure 4: Flowchart of Changing Control Values

Changing Streaming Resolution

To change the streaming resolution, the sequence to be performed are as follows:

1. Follow the steps of *Streaming Stop Sequence* section if the streaming is already started.
2. Follow the steps of *Streaming Start Sequence* section with the desired stream configuration.

MCU Command Overview

This section describes all the commands transferred between the host processor and the MCU, and the return values from the MCU to the Host processor.

MCU I²C Slave Address

MCU has a I²C Slave address of 0x42, which is 7-bit addressing mode.

Note: The above address is required for all the transactions.

Types of Camera Commands

The different types of camera commands are as follows:

- Status command
- Configure command
- Query-Reply command

The types of MCU commands are listed in the following table.

Table 3: Types of MCU Commands

Types	MCU Commands
Status	Get Command Status
Configure	Configure Lane
	Configure MIPI Clock
	Init Camera
	De-Init Camera
	Stream ON
	Stream OFF
	Configure Stream
	Set Control Value
Query-Reply	Get Firmware Version
	Get Stream Info
	Get Control Info
	Get Control Value

List of Camera Commands

The camera commands that are supported by MCU are listed and described in the following table.

Table 4: List of Camera Commands

S.NO	Command	Description	Query Command ID (Hex)
1	Get Firmware Version	This command is used to get the actual firmware version in the MCU.	0x00
2	Init Camera	This command is used to initialize the camera by changing state of hardware pins and writing appropriate settings to the camera. This command returns immediately.	0x04
3	Get Command Status	This command is used to query the status of any executed command. Additionally, it will return the current status of camera (Idle, Busy, and so on) and the hardware specific errors with respect to MCU.	0x05
4	De-Init Camera	This command is used to de-initialize the camera by changing the state of hardware pins and writing the necessary configuration settings.	0x06
5	Stream On	This command is used to start the camera streaming process.	0x07
6	Stream Off	This command is used to stop the camera streaming process.	0x08
7	Configure Stream	This command is used to set the format, width, height and frame rate in the camera. The valid values are specified by Get Stream Info command.	0x09
8	Get Control value	This command is used to get the value of any control enumerated by Get Control Info command.	0x10
9	Set Control Value	This command is used to set the value of any control enumerated by Get Control Info command.	0x11
10	Configure lane	This command is used to configure the number of lanes.	0x17
11	Configure MIPI Clock	This command is used to configure the maximum clock supported by the platform.	0x18

List of Camera Formats

The camera format codes that are returned from MCU to host processor are listed in the following table.

Table 5: List of Camera Formats

Format Code	Description
0x30314752	RGGB Bayer 10bit
0x32314752	RGGB Bayer 12bit

List of Return Codes

The return codes that are transmitted from MCU to host processor are listed and described in the following table.

Table 6: List of Return Codes

Return Code	Description
0x00	Success or Command Completed
0x01	Busy or Command in Progress
0x02	Invalid Argument
0x03	Permission Denied
0x04	Device Not Found
0x05	I/O Error between ISP and MCU
0x06	Hardware Specific Error
0x07	Try Again
0x08	Already in Effect
0x09	Not Implemented
0x0A	Out of Range
0x0B - 0xFE	Reserved
0xFF	Unknown Failure

Note: For more details, please refer to the ***Error! Reference source not found.***

List of Command Status Codes

The length of command status is 2-bytes. The command status code is returned by the MCU to the host processor. The command status code and its description are listed in the following table.

Table 7: List of Command Status Codes (General)

Command Status Code	Description
0x0000	No error or Command Completed
0xF000	Command in Progress

Note: For more details, please refer to the ***Error! Reference source not found.***

Camera Status Codes

The error codes returned by camera to the host processor are listed in the following table.

Table 8: List of Command Status Codes (ISP)

Command Status Code	Description
0x0FF0	Camera is Powered Down
0x0FF1	Camera is Uninitialized

MCU Status Codes

The error codes which describe the enumerations of errors specific to MCU are listed in the following table.

Table 9: List of Command Status Codes (MCU)

Command Status Code	Description
0x2001	Master I^2C Init Error
0x2002	Master I^2C Timeout
0x2003	Master I^2C I/O Error
0x2004	SPI Init Error
0x2005	SPI Timeout Error
0x2006	SPI I/O Error
0x2007	USART Init Error
0x2008	Framework Error
0x2009	Slave I^2C I/O Error
0x200A	CRC Error

Note: The return values help in querying the current state of MCU.

MCU Command Description

This section explains the transactions handled from the MCU to the host processor while processing the basic MCU Commands.

Status Command

The status command is used to query the status of MCU using the **Get Command Status** command. This command involves three transactions where the reply length from MCU is always constant. The transaction of status command is shown below.

Transaction 1 (Host Processor to MCU) (Write)	0x43	0x05	0x00, 0x01	0x01		
	Communication ID (1 Byte)	Command ID (1 Byte)	Payload Length (2 Bytes)	Check Sum of Payload (1Byte)		
Transaction 2 (Host Processor to MCU) (Write)	0x43	0x05	0xFF			
	Communication ID (1 Byte)	Command ID (1 Byte)	Query Command ID (Payload length)			
Transaction 3 (MCU to Host Processor) (Read)	0x43	0x05	0xXX	0xXX, 0xXX	0xXX	0xXX
	Communication ID (1 Byte)	Command ID (1 Byte)	Command ID of Issued Command (1 Byte)	Command Status Code (2 Bytes)	Check Sum (1 Byte)	Return code (1 Byte)

Figure 5: Transaction of Status Command

Note: The command ID returned by the MCU in Transaction 3 corresponds to the Command ID that was used prior to **Get Command Status** command. Please refer to the *List of Command Status Codes* section to know the various command status codes returned by MCU.

Configure Command

The configure command is used for starting an operation in the sensor through MCU. For example, Lane Config, Stream Config, Set Control Config, Init Cam and so on. This command always returns immediately, while MCU executes the request in the background. The status of the last issued command can be queried through the **Get Command Status** command. The configure command involves two transactions as shown below.

Transaction 1 (Write)	0x43	0xXX	0xXX, 0xXX	0xXX
	Communication ID (1 Byte)	Command ID (1 Byte)	Payload Length (2 Bytes)	Checksum (1 Byte)
Transaction 2 (Write)	0x43	0xXX	0xXX, 0xXX	0xXX
	Communication ID (1 Byte	Command ID (1 Byte)	Payload Data (Payload Length bytes)	Checksum (1 Byte)

Figure 6: Transaction of Configure Command

The transaction values of configure command are listed in the following table.

Table 10: Configure Command Transaction Values

Transaction	Packet		Configure Lane	Configure MIPI Clock	Init Camera	De Init Camera	Stream ON	Stream OFF	Configure Stream	Set Control Value
Transaction 1 (Host Processor to MCU)	Communication ID		0x43	0x43	0x43	0x43	0x43	0x43	0x43	0x43
	Command ID		0x17	0x18	0x04	0x06	0x07	0x08	0x09	0x11
	Payload Length	Byte 1	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0xXX, 0xXX (Based on Control Index)
		Byte 2	0x02	0x02	0x00	0x00	0x00	0x00	0x0E	
	Checksum		0x02	0x02	0x00	0x00	0x00	0x00	0x0E	0xXX (Based on Control Index)
Transaction 2 (Host Processor to MCU)	Communication ID		0x43	0x43	0x43	0x43	0x43	0x43	0x43	0x43
	Command ID		0x17	0x18	0x04	0x06	0x07	0x08	0x09	0x11
	Payload		2-byte data (Based on Number of lanes). For more details, please refer to *Table 11*	2-byte data (Based on Max MIPI Clock). For more details, please refer to *Table 12*	-	-	-	-	14-byte data (Based on Format Index). For more details, please refer to *Table 13*	0xXX (Based on Control Index). For more details, please refer to *Table 17*
	Checksum		0xXX (Based on Number of lanes)	0xXX (Based on Max MIPI clock specified)	-	-	-	-	0xXX (Based on Format Index)	0xXX (Based on Control Index)

Note: Please traverse the above table from top to bottom.

The details about the communication to MCU from host, for lane configuration are listed in the following table.

Table 11: Configure Lane Payload Data

Number of Lanes	Payload Data (2B)	Payload Checksum (1B)
4	0x00, 0x04	0x04

Note: Lanes must be configured before Camera Init.

The details about the communication to MCU from host, for configuring MIPI Clock are listed in the following table.

Table 12: Configure MIPI Clock Payload Data

MIPI Clock (MHz)	Payload Data (2B)	Payload Checksum (1B)
1188	0x05, 0xA0	0xA5

Note: MIPI clock must be configured before Camera Init.

The details about the communication to MCU from host, for configure stream are listed in the following table.

Table 13: Configure Stream Payload Data (MIPI Clock - 1188MHz, MIPI Lanes - 4)

Formats	Payload Data in Bytes						Payload Checksum (1 B)
	Stream Index (2 B)	FourCC Format (4 B)	Width (2 B)	Height (2 B)	Frame Rate (Numerator) (2 B)	Frame Rate (Denominator) (2 B)	
2048p 12bit at 67 fps	0x00, 0x00	0x32, 0x31, 0x47, 0x52	0x09, 0x80	0x08, 0x00	0x00, 0x43	0x00, 0x01	0xD5
2048p 10bit at 79 fps	0x00, 0x01	0x30, 0x31, 0x47, 0x52	0x09, 0x80	0x08, 0x00	0x00, 0x4F	0x00, 0x01	0xDA

Note: Please traverse the above table from left to right.

Query-Reply Command

The query-reply command is used to query information such as Streaming formats, Controls and so on, from the MCU. This command uses four transactions between the host and MCU as shown below.

Transaction 1 (Write)	0x43	0xXX	0xXX	0xXX
	Communication ID (1 Byte)	Command ID (1 Byte)	Payload Length (2 Bytes)	Checksum (1 Byte)
Transaction 2 (Write)	0x43	0xXX	0xXX, 0xXX	0xXX
	Communication	Command ID	Payload Data (Payload	Checksum

	ID (1 Byte)	(1 Byte)	Length)		(1 Byte)
	0x43	0xXX	0xXX, 0xXX	0xXX	0xXX
Transaction 3 (Read)	Communication ID (1 Byte)	Command ID (1 Byte)	Reply Length (2 Bytes)	Checksum (1 Byte)	Return Code (1 Byte)
	0x43	0x00	0XX, 0XX	0XX	0XX
Transaction 4 (Read)	Communication ID (1 Byte)	Command ID (1 Byte)	Reply Data (Reply Length)	Check sum (1 Byte)	Return code (1 Byte)

Figure 7: Transaction of Query-Reply Command

The transaction values of query-reply command are listed in the following table.

Table 14: Query-Reply Command Transaction Values

Transaction	Packet		Get Firmware Version	Get Control Value
Transaction 1 (Host Processor to MCU)	Communication ID		0x43	0x43
	Command ID		0x00	0x10
	Payload Length	Byte 1	0x00	0x00
		Byte 2	0x00	0x02
	Checksum		0x00	0x02
Transaction 2 (Host Processor to MCU)	Communication ID		0x43	0x43
	Command ID		0x00	0x10
	Payload Data		-	0xXX, 0xXX (Control Index)
	Checksum		-	0xXX (Based on Control Index)
Transaction 3 (MCU to Host Processor)	Communication ID		0x43	0x43
	Command ID		0x00	0x10
	Reply Length	Byte 1	0x00	0x00
		Byte 2	0x20	0x09
	Checksum		0x20	0x09
	Return Code		0x00	0x00
Transaction 4 (MCU to Host Processor)	Communication ID		0x43	0x43
	Command ID		0x00	0x10
	Reply Data		Firmware version - 32-bytes	Based on Control Index. For more details, please refer to *Table 17*
	Checksum		Based on reply data	Based on reply data
	Return Code		0x00	0x00

Note: Please traverse the above table from top to bottom.

Controls of e-CAM521_CUMI568C_MOD_H01R1

This section describes the controls available in e-CAM84_CUMI485_MOD, which can be set or get through the MCU.

The available controls of e-CAM84_CUMI485_MOD_H01R1 are as follows:

- Sensor Gain (64-bit)
- Exposure (64-bit)
- Frame Rate (64-bit)
- Sensor Mode Index (32-bit)
- Trigger (bool)
- Trigger_Frequency (intmenu)

The get or set control payload data of e-CAM521_CUMI568C_MOD controls are listed in the following table.

Table 15: Get or Set Control Payload Data

Control Name	Control Index (2 B)		Payload Data in Bytes					
			Control ID (4 B)				Control Type (1 B)	Current Value (8 B)
Sensor Mode Index	0x00	0x05	0x00	0x9A	0x20	0x08	0x01	0xXX,0xXX,0xXX,0xXX, 0xXX,0xXX,0xXX,0xXX
Gain	0x00	0x00	0x00	0x9A	0x20	0x09	0x02	0xXX,0xXX,0xXX,0xXX, 0xXX,0xXX,0xXX,0xXX
Exposure	0x00	0x01	0x00	0x9A	0x20	0x0A	0x02	0xXX,0xXX,0xXX,0xXX, 0xXX,0xXX,0xXX,0xXX
Frame Rate	0x00	0x03	0x00	0x9A	0x20	0x0B	0x02	0xXX,0xXX,0xXX,0xXX, 0xXX,0xXX,0xXX,0xXX
Trigger	0x00	0x06	0x00	0x9A	0x20	0x0F	0x02	0xXX,0xXX,0xXX,0xXX, 0xXX,0xXX,0xXX,0xXX
Trigger_Frequency	0x00	0x07	0x00	0x9A	0x20	0x10	0x02	0xXX,0xXX,0xXX,0xXX, 0xXX,0xXX,0xXX,0xXX

Note: Please traverse the above table from left to right.

Sensor Gain

The range of gain supported natively by IMX568 sensor varies from 1 to 24 dB. The subset of analog sensor gain supported in e-CAM521_CUMI568C_MOD varies from 1 to 24 dB. The minimum step value for sensor gain is 0.3 dB.

The sensor gain details are as follows:

- Format of value: In dB.
- Datatype: 64-bit unsigned integer.
- Mapping of values to be passed to actual gain value: Actual gain value in dB << 22.

  For example, if a gain value is 1.3 dB, then the mapping value can be 27262976. Hence, 1.3 << 22.

- Control ID: 0x009A2009.

Note: The formula used for converting gain in terms of multiplication factor to dB.

Gain in dB = 20 $\log_{10}$ (Gain in multiplication factor).

And gain in dB to multiplication factor.

Gain in multiplication factor = 10 ^ (Gain in dB/20).

Setting (Write) Sensor Gain Value

MCU command type: Configuration command. Please refer to the *Set Control Value* of *Table 10*.

Getting (Read) Current Sensor Gain Value

MCU command type: Query-Reply command. Please refer to the *Get Control Value* of *Table 16*.

Exposure Control

The exposure control is handled by the exposure feature provided by the e-CAM521_CUMI568C_MOD. The range for programming exposure time in e-CAM521_CUMI568C_MOD is 450 – 400000us. The steps in which exposure will be calculated is 1.

The exposure details are as follows:

- Data Type: 64-bit unsigned integer.
- Mapping of values to be passed to actual exposure value: (Actual exposure value << 22).
- For example, to set an exposure of 0.033s, the value that needs to be set is 138412 (0.033 << 22).
- Control ID: 0x009A200A.

Setting (Write) Exposure Value

MCU command type: Configuration command. Please refer to the *Set Control Value* of *Table 10*.

Getting (Read) Current Exposure Value

MCU command type: Query-Reply command. Please refer to the *Get Control Value* of *Table 16.*

Frame Rate Range Control

The frame rate feature is also known as the long exposure operation and can be used to program the frame rate at which the camera streams.

The range of frame rate supported by e-CAM521_CUMI568C_MOD is 8388608 - 377487360. The frame length can be calculated in one step.

The frame rate details are as follows:

- Data Type: 64-bit unsigned integer.
- Mapping of values to be passed to actual Frame Rate value: (Actual frame rate value << 22).
- For example, to set a frame rate of 15 fps, the value that needs to be set is 62914560 (15 << 22).
- Control ID: 0x009A200B.

Setting (Write) Frame Rate Value

MCU command type: Configuration command. Please refer to the *Set Control Value* of *Table 10*.

Getting (Read) Current Frame Rate Value

MCU command type: Query-Reply command. Please refer to the *Get Control Value* of *Table 16*.

Sensor Mode Index

Sensor mode index refers to a configuration of the IMX568 sensor, to a pre-determined combination of image height, image width, fps, bits per pixel (bpp). The control ID is 0x009A2008.

The various sensor mode indices supported are mentioned below:

MIPI Clock: 1188MHz MIPI Lanes: 4

- 2048p_67fps_12bpp
- 2048p_79fps_10bpp
- 1080p_121fps_12bpp
- 1080p_142fps_10bpp
- 720p_172fps_12bpp
- 720p_202fps_10bpp
- 480p_240fps_12bpp

- 480p_280fps_10bpp

Note: Ensure the camera stream is OFF before changing the sensor mode index.

Changing Sensor Mode from Host to MCU

To change the sensor mode index from the host to the MCU, follow these steps:

Note: If the camera module is already streaming, send Stream OFF command as mentioned in *Configure Command* section.

1. Change the sensor mode ID using the *Set Control Value* command mentioned in *Configure Command* section.
2. Check the status of the MCU using Get Command Status command mentioned in *Status Command*

   Note: Ensure that the status of the MCU is success.

3. Configure the stream based on the sensor mode ID using the *Configure Stream* command mentioned in *Configure Command* section.
4. Check the status of the MCU using Get Command Status command mentioned in *Status Command*

   Note: Ensure that the status of the MCU is success.

5. Stream the camera using Stream On command followed by the Get Command Status command mentioned in *Configure Command* and *Status Command*

   The MCU tends to retain the gain, exposure and frame rate values and it is recommended to call these controls after the stream is configured to ensure proper streaming in camera.

Trigger

Trigger refers to a configuration of the IMX568 sensor, to enable stream using trigger signal. The control ID is 0x009A200F.

If stream to be verified using external trigger, make set up using “e-CAM56_CUOAGX_External_Trigger_Setup_Guide_Rev_<Ver>.pdf”

To check the trigger control value, run the below command.

```
$ v4l2-ctl -L | grep "trigger"
```

```
trigger 0x009a200f (bool)   : default=0 value=0 flags=execute-on-write
```

Figure 7: v4l2 control for trigger

The trigger mode can be enabled by running v4l2-ctl command with the **trigger control** value to 1 using the below command.

```
$ v4l2-ctl -c trigger=1
```

Note: By default, trigger mode is disabled.

Value = 0 – trigger mode is disabled

1 – trigger mode is enabled

The same can be enabled using Trigger control in eCAM_argus_camera application. Please refer "e-CAM56_CUOAGX_eCAM_Argus_Camera_App_User_Manual_Rev_<Ver>.pdf"

Trigger_Frequency

Trigger_Frequency control is used to configure the PWM frequency when the camera is streaming using internal trigger from hex cam base board in trigger mode. The control ID is 0x009A2010.

Note: To enable trigger mode, set trigger control to 1 and make sure that the switch (SW1) position is in INT_TRIG on e-CAM30_HEXCUXVR_BASE_BRD to verify stream using internal trigger.

To check the trigger_frequency control value, run the below command.

```
$ v4l2-ctl -L | grep "trigger_frequency"
```

```
trigger_frequency 0x009a2010 (intmenu): min=0 max=1 default=0 value=0
                  0: 30 (0x1e)
                  1: 60 (0x3c)
```

Figure 8: v4l2 control for trigger_frequency

By default, in internal trigger mode the PWM frequency is set 30Hz and can be configured to 60Hz by setting the **trigger_frequency** control value to 1 using the below command.

```
$ v4l2-ctl -c trigger_frequency=1
```

Value = 0 – configure with 30Hz PWM frequency

1 – configure with 60Hz PWM frequency

The same can be enabled using Trigger Frequency control in eCAM_argus_camera application. Please refer "e-CAM56_CUOAGX_eCAM_Argus_Camera_App_User_Manual_Rev_<Ver>.pdf"

Output Image Data

The captured images from the sensor, in RAW Bayer RGGB 10-bit/12-bit per pixel, come over the MIPI bus.

What's Next?

After understanding the detailed information of I^2C based protocol used by the host application processor for communicating with the microcontroller, you can refer to the *e-CAM521_CUMI568C_MOD Datasheet* to understand more about e-CAM521_CUMI568C_MOD_H01R1.

Glossary

CMOS: Complementary Metal Oxide Semiconductor.

CSI: Camera Serial Interface.

FHD: Full HD (Industry name for 1920 x 1080P resolution).

MCU: Microcontroller unit.

MIPI: Mobile Industry Processor Interface.

RoHS: Restriction of Hazardous Substances.

Contact Us

If you need any support on e-CAM521_CUMI568C_MOD product, please contact us using the Live Chat option available on our website - https://www.e-consystems.com/

Creating a Ticket

If you need to create a ticket for any type of issue, please visit the ticketing page on our website - https://www.e-consystems.com/create-ticket.asp

RMA

To know about our Return Material Authorization (RMA) policy, please visit the RMA Policy page on our website - https://www.e-consystems.com/RMA-Policy.asp

General Product Warranty Terms

To know about our General Product Warranty Terms, please visit the General Warranty Terms page on our website - https://www.e-consystems.com/warranty.asp

Revision History

Rev	Date	Description	Author
1.0	22-Aug-2023	Initial draft	Camera Dev Team
1.1	20-Oct-2023	Updated to 2-lane configuration with six cameras and resolution to 2432x2048	Camera Dev Team
1.2	31-Oct-2023	Updated the firmware version as per latest release and trigger control details are added	Camera Dev Team
1.3	16-Nov-2023	Updated the firmware version as per latest release and trigger_frequency control details are added	Camera Dev Team
1.4	11-Mar-2024	Removed 2-lane support and added the document version in the first page	Camera Dev Team
1.5	22-Mar-2024	Updated Get and Set control payload data and Configure Stream Payload Data tables	Camera Dev Team