



## **e-con Systems India Pvt Ltd**

RISHABH INFO PARK,  
RR Tower – IV, 7th Floor,  
Super A-16 & A-17, Thiru-Vi-Ka Industrial Estate,  
Guindy, Chennai - 600 032.  
[www.e-consystems.com](http://www.e-consystems.com)

## **See3CAM\_CX3RDK**

## **API Guide**

Revision 1.2  
25-March-2014



1	Revision History .....	3
2	Introduction .....	4
3	Scope .....	4
4	Intended Audience .....	5
4.1	Why Should You Read This Document? .....	5
5	Supported API's: .....	5
5.1	Initialization APIs .....	6
5.1.1	Void esOV5640_Base_Config (void); .....	6
5.1.2	Void esOV5640_VGA_config (void); .....	6
5.1.3	Void esOV5640_VGA_HS_config (void); .....	6
5.1.4	Void esOV5640_720P_config (void); .....	6
5.1.5	Void esOV5640_1080P_config (void); .....	7
5.1.6	Void esOV5640_5MP_config (void); .....	7
5.1.7	Void esCamera_Power_Down (void); .....	7
5.1.8	Void esCamera_Power_Up (void); .....	7
5.1.9	Void esOV5640_Auto_Focus_Config (void); .....	7
5.2	UVC Controls .....	8
5.2.1	Int8_t esOV5640_GetBrightness (uint8_t option); .....	8
5.2.2	Void esOV5640_SetBrightness (int8_t Brightness); .....	8
5.2.3	Int8_t esOV5640_GetContrast (uint8_t option); .....	8
5.2.4	Void esOV5640_SetContrast (int8_t Contrast); .....	8
5.2.5	Int32_t esOV5640_GetHue (uint8_t option); .....	9
5.2.6	Void esOV5640_SetHue (int32_t Hue); .....	9
5.2.7	Int8_t esOV5640_GetSaturation (uint8_t option); .....	9
5.2.8	Void esOV5640_SetSaturation (uint32_t Saturation); .....	9
5.2.9	Int8_t esOV5640_GetSharpness (uint8_t option); .....	10
5.2.10	Void esOV5640_SetSharpness (uint8_t Sharp); .....	10
5.2.11	UInt8_t esOV5640_GetWhiteBalance (uint8_t option); .....	10
5.2.12	Void esOV5640_SetWhiteBalance (uint8_t WhiteBalance); .....	10
5.2.13	UInt8_t esOV5640_GetAutoWhiteBalance (uint8_t option); .....	11
5.2.14	Void esOV5640_SetAutoWhiteBalance (uint8_t AutoWhiteBalance); .....	11
5.2.15	Int32_t esOV5640_GetExposure (uint8_t option); .....	11
5.2.16	Void esOV5640_SetExposure (int32_t Exposure); .....	12
5.2.17	UInt8_t esOV5640_GetAutoExposure (uint8_t option); .....	12
5.2.18	Void esOV5640_SetAutoExposure (uint8_t AutoExp); .....	12
5.2.19	UInt8_t esOV5640_GetAutofocus (uint8_t option); .....	12
5.2.20	Void esOV5640_SetAutofocus (uint8_t Is_Enable); .....	13
5.2.21	UInt16_t esOV5640_GetManualfocus (uint8_t option); .....	13
5.2.22	Void esOV5640_SetManualfocus (uint16_t manualfocus); .....	13
5.2.23	Void esOV5640_Nightmode (Cybool_t Enable, uint8_t Nightmode_option); .....	13
6	Conclusion .....	14



## See3CAM\_CX3RDK\_e-CAM59CX3

### 1 Revision History

Rev No	Date	Major Changes	Author
Initial Draft	10-March-2014	Initial Draft	Shanthakumar
1.1	17-March-2014	Reviewed and modified	Subbarao
1.2	25-March-2014	Added detailed description of each API. Grouped the APIs	Subbarao



## 2 Introduction

Denebola (See3CAM\_CX3RDK) is a USB3.0 UVC Reference Design Kit (RDK) developed by e-con Systems using the EZ-USB® CX3 USB3.0 Peripheral controller from Cypress Semiconductors. The Cypress EZ-USB® CX3 is a USB 3.0 peripheral controller that enables developers to add USB 3.0 connectivity to any image sensors compliant with Mobile Industry Processor Interface (MIPI) Camera Serial Interface Type 2 (CSI-2) standard. The Denebola Reference Design Kit developed using EZ-USB® CX3 is a complete Reference Design Kit and has OmniVision OV5640 CMOS image sensor interfaced to it through 2-lane MIPI CSI-2 interface. This is a fully functional camera reference design kit that can stream uncompressed 720p60, 1080p30 and full 5MP@15fps. e-con Systems, a Silver level partner of Cypress Partner program, has developed the reference design kit for EZ-USB® CX3 from Cypress Semiconductors and this kit can be purchased directly from our web store.

Based on the proven EZ-USB FX3 Platform, CX3 comes with an ARM9 CPU and 512KB SRAM that provides 200 MIPS of computational power. The CX3 supports MIPI CSI-2 version 1.01, up to 4 data lanes with data speed up to 1Gbps per lane, for a total bandwidth of 4 Gbps. CX3 is ideally suited for high-definition or high-speed image-capturing applications. CX3 supports multiple peripheral interfaces such as I2C, SPI, and UART, which can be programmed to support Pan, Tilt and Zoom or other camera control functions.

Denebola RDK is a two-board solution containing base board designed around Cypress CX3 USB3.0 Peripheral controller and the Camera daughter board (part number: e-CAM59CX3) designed using OmniVision OV5640 CMOS image sensor based Autofocus camera module. e-con also plans to support multiple camera sensor daughter boards in the future that can be used to evaluate CX3 performance and also other CMOS Image sensors.

See3CAM\_CX3RDK with e-CAM59CX3 is a UVC compliant device and does not require any drivers to be installed on the PC. The native UVC drivers of Windows and Linux Operating Systems shall be compatible with this camera. E-con also provides the sample application that demonstrates some of the features of this camera. However, this camera can be utilized any DirectShow application such as Skype etc.

## 3 Scope

e-con Systems provide the complete firmware source code that is running on CX3 performing a full-featured USB Video Class (UVC) camera functionality. The 5MP Autofocus camera is exposed as a UVC device to the PC and the exposes some of the common UVC controls. This UVC stack firmware is provided to the customers in source code form allowing customers to extend the features or use the same as the base to develop their own UVC cameras.

However, due to the NDA obligations, e-con does not publish the OV5640 sensor initialization or configuration settings that has been used in this CX3 RDK. The e-CAM59CX3 board contains the OmniVision OV5640 image sensor and this sensor requires its registers to be programmed for various configuration and controls that are supported in the UVC controls. The UVC firmware provided by e-con Systems supports these UVC controls such as various resolution formats, brightness, contrast, sharpness, auto/manual exposure, auto/manual focus etc. These controls are implemented by programming certain registers of OV5640 and these settings are not disclosed. These settings are provided as a library file named, libCX3OV5640.a (exact path: "..\..\CX3OV5640LIB\LIB\Release\libCX3OV5640.a"). This library has few APIs which are called by the main UVC stack to implement the UVC controls.



The scope of this document is to explain about how to use the API's available in the sensor library files provided by e-con Systems as part of the Denebola RDK. This document explains the functionality of each API and how this can be extended for any other sensors.

## 4 Intended Audience

This document is intended for developers who want to

1. Integrate any new sensor to the CX3 RDK (instead of the supplied OV5640 image sensor daughter board)
2. Understand or enhance the CX3 RDK and how the UVC controls are implemented in the supplied CX3 RDK firmware.

### 4.1 Why Should You Read This Document?

e-con has received multiple questions from our customers about the contents of the library file libCX3OV5640.a and how they can interface their own sensors. To support these customers, e-con has prepared this application note that explains the APIs provided by this library and what each API performs.

It is important to note the following:

1. The libCX3OV5640.a does not contain any executable code. No C/C++ source code built inside this library.
2. The libCX3OV5640.a just contains the sensor register settings for every video configuration and camera settings (UVC controls). These register settings are programmed to OV5640 image sensor in this library code.

So, it is easy for any developer to understand the functionality performed by each of these APIs and then implement their own code to program their own image sensors. For example, if any of the customer is interested in using this CX3 RDK with an Aptina's CMOS image sensor, they just need to implement these APIs to program the Aptina's image sensor to perform the respective functions.

## 5 Supported API's:

The Sensor Library file contains the API functions that are used to write the camera settings for a particular resolution & it also contains the camera control functions.

This section is split in to two groups: The Initialization APIs and UVC Controls APIs.

The Initialization APIs initialize the OV5640 image sensor for various resolutions as requested from the host PC. The host PC may require the sensor to stream at any of the supported resolutions and the UVC firmware should respond to these commands from host PC by configuring the sensor registers accordingly. This set of APIs also include the power-up/power-down modes and Autofocus firmware initialization for OV5640 sensor. For your image sensor you may support all these APIs or support only part of these APIs as per your requirement.

The UVC Controls are the controls exposed to the host PC. The PC side user can configure various camera settings such as brightness, contrast, exposure, focus position, white balance, sharpness etc. All these are standard UVC controls as specified in the UVC Standard. These controls usually ask for the minimum value (GET\_MIN), maximum value (GET\_MAX), currently set value (GET\_CUR) and resolution (GET\_RES). These commands are sent by the host PC and



based on the sensor support, the UVC firmware has to send the response to these commands. This set of APIs support handling these UVC control requests from host PC.

Please refer USB Video Class Standard for more details about these commands.

## 5.1 Initialization APIs

### 5.1.1 Void esOV5640\_Base\_Config (void);

**Parameters:** None.

**Return Values:** None.

**Description:** This function will write the base settings for the Image sensor through I2C. This is the basic initialization of the OV5640 image sensor that initializes the OV5640 sensor.

### 5.1.2 Void esOV5640\_VGA\_config (void);

**Parameters:** None.

**Return Values:** None.

**Description:** This function will write the VGA (640x480) resolution settings for the frame rate of 60fps to the Image sensor through I2C. Once this function is executed, the OV5640 image sensor starts outputting VGA resolution video at 60fps.

### 5.1.3 Void esOV5640\_VGA\_HS\_config (void);

**Parameters:** None.

**Return Values:** None.

**Description:** This function will write the VGA (640x480) resolution settings for the frame rate of 30fps to the Image sensor through I2C. Once this function is executed, the OV5640 image sensor starts outputting VGA resolution video at 30fps.

### 5.1.4 Void esOV5640\_720P\_config (void);

**Parameters:** None.

**Return Values:** None.

**Description:** This function will write the 720p (1280x720) resolution settings for the frame rate of 60fps to the Image sensor through I2C. Once this function is executed, the OV5640 image sensor starts outputting 1280x720 video at 60fps.



### 5.1.5 Void esOV5640\_1080P\_config (void);

**Parameters:** None.

**Return Values:** None.

**Description:** This function will write the 1080p (1920x1080) resolution settings for the frame rate of 30fps to the Image sensor through I2C. Once this function is executed, the OV5640 image sensor starts outputting 1920x1080 video at 30fps.

### 5.1.6 Void esOV5640\_5MP\_config (void);

**Parameters:** None.

**Return Values:** None.

**Description:** This function will write the 5MP (2592x1944) resolution settings for the frame rate of 15fps to the Image sensor through I2C. Once this function is executed, the OV5640 image sensor starts outputting 5MP resolution video at 15fps.

### 5.1.7 Void esCamera\_Power\_Down (void);

**Parameters:** None.

**Return Values:** None.

**Description:** This function will make the sensor to go into the power down mode. Once this function is executed, the sensor is programmed for power down mode. After entering this mode, the next API must be exCamera\_Power\_Up so that the sensor is powered up.

### 5.1.8 Void esCamera\_Power\_Up (void);

**Parameters:** None.

**Return Values:** None.

**Description:** This function will bring the sensor from power down mode to power up mode.

### 5.1.9 Void esOV5640\_Auto\_Focus\_Config (void);

**Parameters:** None.

**Return Values:** None.

**Description:** This function will write the sensor settings for the autofocus firmware. The OV5640 sensor has a special Autofocus firmware that performs the autofocus inside the camera module. This autofocus firmware is provided by OmniVision for OV5640 sensor with a VCM Autofocus



module. This firmware is used to initialize the autofocus algorithm inside the OV5640 based camera module.

## 5.2 UVC Controls

### 5.2.1 `Int8_t esOV5640_GetBrightness (uint8_t option);`

**Parameters:** `uint8_t option` – GET\_DEF or GET\_MIN or GET\_MAX or GET\_CUR or GET\_RES.

**Return Values:** `int8_t` – Returns the read back brightness value.

**Description:** This function is used to get the brightness value of the image sensor according to the option parameter passed. This is a UVC control feature implemented in our CX3 RDK firmware.

If the Option parameter is passed as GET\_DEF means it will return the default brightness value of the image sensor. For every request as mentioned in the parameter (GET\_MIN, GET\_MAX, GET\_CUR, GET\_RES), the corresponding values as support by OV5640 sensor shall be returned.

### 5.2.2 `Void esOV5640_SetBrightness (int8_t Brightness);`

**Parameters:** `int8_t Brightness` – The Brightness value need to be set on the image sensor.

**Return Values:** None.

**Description:** This function is used to Set the brightness value of the image sensor according to the Brightness parameter passed. This is a UVC control feature implemented in our CX3 RDK firmware.

### 5.2.3 `Int8_t esOV5640_GetContrast (uint8_t option);`

**Parameters:** `uint8_t option` – GET\_DEF or GET\_MIN or GET\_MAX or GET\_CUR or GET\_RES.

**Return Values:** `int8_t` – Returns the read back contrast value.

**Description:** This function is used to get the contrast value of the image sensor according to the option parameter passed. This is a UVC control feature implemented in our CX3 RDK firmware.

If the Option parameter is passed as GET\_DEF means it will return the default contrast value of the image sensor. For every request as mentioned in the parameter (GET\_MIN, GET\_MAX, GET\_CUR, GET\_RES), the corresponding values as support by OV5640 sensor shall be returned.

### 5.2.4 `Void esOV5640_SetContrast (int8_t Contrast);`

**Parameters:** `int8_t Contrast` – The Contrast value need to be set on the image sensor.





**Return Values:** None.

**Description:** This function is used to Set the contrast value of the image sensor according to the Contrast parameter passed. This is a UVC control feature implemented in our CX3 RDK firmware.

### 5.2.5 Int32\_t esOV5640\_GetHue (uint8\_t option);

**Parameters:** uint8\_t option – GET\_DEF or GET\_MIN or GET\_MAX or GET\_CUR or GET\_RES.

**Return Values:** int32\_t – Returns the read back hue value.

**Description:** This function is used to get the hue value of the image sensor according to the option parameter passed. This is a UVC control feature implemented in our CX3 RDK firmware.

If the Option parameter is passed as GET\_DEF means it will return the default hue value of the image sensor. For every request as mentioned in the parameter (GET\_MIN, GET\_MAX, GET\_CUR, GET\_RES), the corresponding values as support by OV5640 sensor shall be returned.

### 5.2.6 Void esOV5640\_SetHue (int32\_t Hue);

**Parameters:** int32\_t Hue – The Hue value need to be set on the image sensor.

**Return Values:** None.

**Description:** This function is used to Set the hue value of the image sensor according to the Hue parameter passed. This is a UVC control feature implemented in our CX3 RDK firmware.

### 5.2.7 Int8\_t esOV5640\_GetSaturation (uint8\_t option);

**Parameters:** uint8\_t option – GET\_DEF or GET\_MIN or GET\_MAX or GET\_CUR or GET\_RES.

**Return Values:** int8\_t – Returns the read back saturation value.

**Description:** This function is used to get the saturation value of the image sensor according to the option parameter passed. This is a UVC control feature implemented in our CX3 RDK firmware.

If the Option parameter is passed as GET\_DEF means it will return the default saturation value of the image sensor. For every request as mentioned in the parameter (GET\_MIN, GET\_MAX, GET\_CUR, GET\_RES), the corresponding values as support by OV5640 sensor shall be returned.

### 5.2.8 Void esOV5640\_SetSaturation (uint32\_t Saturation);

**Parameters:** uint32\_t Saturation – The Saturation value need to be set on the image sensor.

**Return Values:** None.



**Description:** This function is used to Set the Saturation value of the image sensor according to the Saturation parameter passed. This is a UVC control feature implemented in our CX3 RDK firmware.

### 5.2.9 Int8\_t esOV5640\_GetSharpness (uint8\_t option);

**Parameters:** uint8\_t option – GET\_DEF or GET\_MIN or GET\_MAX or GET\_CUR or GET\_RES.

**Return Values:** int8\_t – Returns the read back sharpness value.

**Description:** This function is used to get the sharpness value of the image sensor according to the option parameter passed. This is a UVC control feature implemented in our CX3 RDK firmware.

If the Option parameter is passed as GET\_DEF means it will return the default sharpness value of the image sensor. For every request as mentioned in the parameter (GET\_MIN, GET\_MAX, GET\_CUR, GET\_RES), the corresponding values as support by OV5640 sensor shall be returned.

### 5.2.10 Void esOV5640\_SetSharpness (uint8\_t Sharp);

**Parameters:** uint8\_t Sharp – The Sharpness value need to be set on the image sensor.

**Return Values:** None.

**Description:** This function is used to Set the Sharpness value of the image sensor according to the Sharp parameter passed. This is a UVC control feature implemented in our CX3 RDK firmware.

### 5.2.11 Uint8\_t esOV5640\_GetWhiteBalance (uint8\_t option);

**Parameters:** uint8\_t option – GET\_DEF or GET\_MIN or GET\_MAX or GET\_CUR or GET\_RES.

**Return Values:** uint8\_t – Returns the read back Manual White Balance value.

**Description:** This function is used to get the Manual white balance value of the image sensor according to the option parameter passed. This is a UVC control feature implemented in our CX3 RDK firmware.

If the Option parameter is passed as GET\_DEF means it will return the default manual white balance value of the image sensor. For every request as mentioned in the parameter (GET\_MIN, GET\_MAX, GET\_CUR, GET\_RES), the corresponding values as support by OV5640 sensor shall be returned.

### 5.2.12 Void esOV5640\_SetWhiteBalance (uint8\_t WhiteBalance);



**Parameters:** uint8\_t WhiteBalance – The manual white balance value need to be set on the image sensor.

**Return Values:** None.

**Description:** This function is used to Set the manual white balance value of the image sensor according to the WhiteBalance parameter passed. This is a UVC control feature implemented in our CX3 RDK firmware.

### 5.2.13 Uint8\_t esOV5640\_GetAutoWhiteBalance (uint8\_t option);

**Parameters:** uint8\_t option – GET\_DEF or GET\_MIN or GET\_MAX or GET\_CUR or GET\_RES.

**Return Values:** uint8\_t – Returns the read back Auto white balance value.

**Description:** This function is used to get the auto white balance value of the image sensor according to the option parameter passed. This is a UVC control feature implemented in our CX3 RDK firmware.

If the Option parameter is passed as GET\_DEF means it will return the default auto white balance value of the image sensor. For every request as mentioned in the parameter (GET\_MIN, GET\_MAX, GET\_CUR, GET\_RES), the corresponding values as support by OV5640 sensor shall be returned.

### 5.2.14 Void esOV5640\_SetAutoWhiteBalance (uint8\_t AutoWhiteBalance);

**Parameters:** uint8\_t AutoWhiteBalance – The Auto white balance value need to be set on the image sensor.

**Return Values:** None.

**Description:** This function is used to Set the Auto white Balance of the image sensor according to the AutoWhiteBalance parameter passed. This is a UVC control feature implemented in our CX3 RDK firmware.

### 5.2.15 Int32\_t esOV5640\_GetExposure (uint8\_t option);

**Parameters:** uint8\_t option – GET\_DEF or GET\_MIN or GET\_MAX or GET\_CUR or GET\_RES.

**Return Values:** int32\_t – Returns the read back Manual exposure value.

**Description:** This function is used to get the Manual exposure value of the image sensor according to the option parameter passed. This is a UVC control feature implemented in our CX3 RDK firmware.

If the Option parameter is passed as GET\_DEF means it will return the default manual exposure value of the image sensor. For every request as mentioned in the parameter (GET\_MIN, GET\_MAX, GET\_CUR, GET\_RES), the corresponding values as support by OV5640 sensor shall be returned.



### 5.2.16 Void esOV5640\_SetExposure (int32\_t Exposure);

**Parameters:** int32\_t Exposure – The manual exposure value need to be set on the image sensor.

**Return Values:** None.

**Description:** This function is used to Set the manual exposure value of the image sensor according to the Exposure parameter passed. This is a UVC control feature implemented in our CX3 RDK firmware.

### 5.2.17 Uint8\_t esOV5640\_GetAutoExposure (uint8\_t option);

**Parameters:** uint8\_t option – GET\_DEF or GET\_MIN or GET\_MAX or GET\_CUR or GET\_RES.

**Return Values:** uint8\_t – Returns the read back Auto exposure value.

**Description:** This function is used to get the Auto exposure value of the image sensor according to the option parameter passed. This is a UVC control feature implemented in our CX3 RDK firmware.

If the Option parameter is passed as GET\_DEF means it will return the default Auto exposure value of the image sensor. For every request as mentioned in the parameter (GET\_MIN, GET\_MAX, GET\_CUR, GET\_RES), the corresponding values as support by OV5640 sensor shall be returned.

### 5.2.18 Void esOV5640\_SetAutoExposure (uint8\_t AutoExp);

**Parameters:** uint8\_t AutoExp – The Auto exposure value need to be set on the image sensor.

**Return Values:** None.

**Description:** This function is used to Set the Auto exposure value of the image sensor according to the AutoExp parameter passed. This is a UVC control feature implemented in our CX3 RDK firmware.

### 5.2.19 Uint8\_t esOV5640\_GetAutofocus (uint8\_t option);

**Parameters:** uint8\_t option – GET\_DEF or GET\_MIN or GET\_MAX or GET\_CUR or GET\_RES.

**Return Values:** uint8\_t – Returns the read back Auto focus value.

**Description:** This function is used to get the Auto focus value of the image sensor according to the option parameter passed. This is a UVC control feature implemented in our CX3 RDK firmware.

If the Option parameter is passed as GET\_DEF means it will return the default Auto focus value of the image sensor. For every request as mentioned in the parameter (GET\_MIN, GET\_MAX, GET\_CUR, GET\_RES), the corresponding values as support by OV5640 sensor shall be returned.



### 5.2.20 Void esOV5640\_SetAutofocus (uint8\_t Is\_Enable);

**Parameters:** uint8\_t Is\_Enable – The Auto focus value need to be set on the image sensor.

**Return Values:** None.

**Description:** This function is used to Set the Auto focus value of the image sensor according to the Is\_Enable parameter passed. This is a UVC control feature implemented in our CX3 RDK firmware.

### 5.2.21 Uint16\_t esOV5640\_GetManualfocus (uint8\_t option);

**Parameters:** uint8\_t option – GET\_DEF or GET\_MIN or GET\_MAX or GET\_CUR or GET\_RES.

**Return Values:** uint16\_t – Returns the read back Manual focus value.

**Description:** This function is used to get the Manual focus value of the image sensor according to the option parameter passed. This is a UVC control feature implemented in our CX3 RDK firmware.

If the Option parameter is passed as GET\_DEF means it will return the default manual focus value of the image sensor. For every request as mentioned in the parameter (GET\_MIN, GET\_MAX, GET\_CUR, GET\_RES), the corresponding values as support by OV5640 sensor shall be returned.

### 5.2.22 Void esOV5640\_SetManualfocus (uint16\_t manualfocus);

**Parameters:** uint16\_t manualfocus – The manual focus value need to be set on the image sensor.

**Return Values:** None.

**Description:** This function is used to Set the manual focus value of the image sensor according to the manualfocus parameter passed. This is a UVC control feature implemented in our CX3 RDK firmware.

### 5.2.23 Void esOV5640\_Nightmode (Cybool\_t Enable, uint8\_t Nightmode\_option);

**Parameters:** Cybool\_t Enable – Enable or Disable Night mode.

uint8\_t Nightmode\_option – The Night mode value need to be set on the image sensor.

**Return Values:** None.

**Description:** This function is used to enable or disable the Night mode of the image sensor. If the Enable parameter is given as TRUE means it will send the night mode with the value given in the Nightmode\_option parameter that is passed. If the Enable parameter is given as FALSE it will disable the night mode in the image sensor. This is a special mode supported by OV5640 image



sensor. This is not a standard UVC control. If your sensor does not support this, you can ignore this API.

## 6 Conclusion

Using this document, developers can support any image sensor they want with the e-con's CX3 RDK. The developers can extend or add more features to the UVC stack provided by e-con for supporting various image sensors.

