

Tara Stereo Vision USB 3.0 Camera

Linux API Manual



Version 1.4

e-con Systems

6/9/2018

Disclaimer

e-con Systems reserves the right to edit/modify this document without any prior intimation of whatsoever.

Contents

INTRODUCTION TO TARA - SEE3CAM STEREO	3
DESCRIPTION	3
BLOCK DIAGRAM OF TARA SDK IN WINDOWS	4
SUPPORTED APIS	5
TARACAMPARAMETERS APIS:	5
BOOL INIT()	5
BOOL REMAPSTEREOIMAGE(MAT LFRAME, MAT RFRAME, MAT *RIMAGE, MAT *LIMAGE)	5
BOOL LOADCAMERAMATRIX()	5
BOOL COMPUTERECTIFYPRAMS()	5
DISPARITY APIS:	6
BOOL INITCAMERA(BOOL GENERATEDISPARITY, BOOL FILTEREDDISPARITYMAP)	6
BOOL GRABFRAME(MAT *LEFTIMAGE, MAT *RIGHTIMAGE)	6
BOOL GETDISPARITY(MAT LIMAGE, MAT RIMAGE, MAT *DISPARITYMAP, MAT *DISP_FILTERED)	6
BOOL ESTIMATEDEPTH(POINT PT, FLOAT *DEPTHVALUE)	7
BOOL SETEXPOSURE(INT EXPOSUREVAL)	7
BOOL GETEXPOSURE(INT *EXPOSUREVAL)	7
BOOL SETAUTOEXPOSURE()	7
BOOL SETALGORITHMPARAM()	8
BOOL INIT(BOOL GENERATEDISPARITY)	8
BOOL SETBRIGHTNESS(INT BRIGHTNESSVAL)	8
BOOL SETSTREAMMODE(UINT STREAMMODE)	8
BOOL GETSTREAMMODE(UINT *STREAMMODE)	9
CAMERAENUMERATION:	9
INT GETLISTOFDEVICESECON()	9
BOOL GETDEVICEIDCON(INT *DEVICEID, SIZE *RESOLUTIONSELECTED)	9
BOOL ISSTEREODEVICEAVAIL(CHAR *PID)	9
CAMERAENUMERATION(INT *DEVICEID, SIZE *RESOLUTIONSELECTED)	10
VOID QUERY_RESOLUTION(INT DEVICEID)	10
VOID FREEDEVICES(VOID)	10
SUPPORT	11

Introduction to Tara - See3CAM_Stereo

The Tara - See3CAM_Stereo, hereafter called as Tara, is a UVC compliant USB 3.0 SuperSpeed Stereo vision camera from e-con Systems, a leading embedded Product Design Services company which specializes in the advanced camera solutions. Tara is based on MT9V024 stereo sensor from OnSemi and it supports a maximum resolution of WVGA at 60fps over USB 3.0 in uncompressed format. Tara is the latest member of the Stereo Vision family of USB3.0 SuperSpeed camera products launched by e-con Systems.

Tara is a monochrome camera with the S-mount (also known as M12 board lens) lens holder. The S-mount is one of the most commonly used small form-factor lens mounts for board cameras. Tara has two OnSemi's 1/3-inch MT9V024 image sensors separated by an inter-ocular distance or base line of 60 mm. With USB 3.0 interface to the host PC, Tara can stream WVGA (2 x 752 x 480) resolution at 60 fps, VGA (2 x 640 x 480) resolution at 60 fps and QVGA (2 x 320 x 240) resolution at 60 fps in uncompressed Y16 (8 Bit pixel data from each camera) and RGB24 (10 Bit pixel data from each camera). It also has ability to capture still images. Tara is also backward compatible with USB 2.0 host ports and does not require any special camera drivers to be installed in the host PC. In USB 2.0, the camera is capable of streaming in lower frame rates.

The commonly used functions are put into a single namespace for modularity, in-order to reuse them in the examples. Tara is the namespace that contains the following three classes:

- i. TaraCam Parameters
- ii. Disparity
- iii. CameraEnumeration

Description

Tara namespace contains many commonly used functions. Loading the calibrated files from the camera flash, computing disparity map, rectification of frames, camera enumeration and initialization, etc. are some of them.

Note:

For Sample usage of the API's, refer source code.

Block Diagram of Tara SDK in Windows

The block diagram of Tara SDK in Windows is shown below.

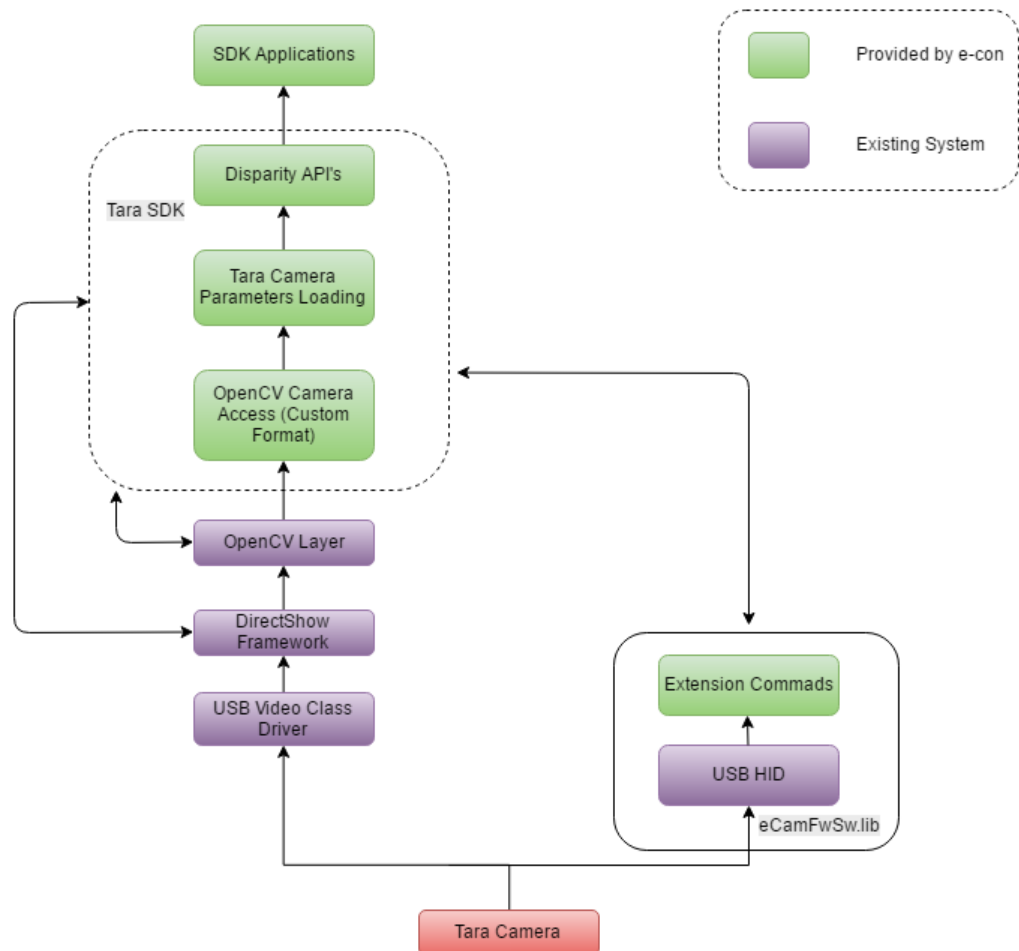


Figure 1: Block Diagram of Tara SDK in Windows

Supported APIs

The details regarding the supported APIs are explained below.

TaraCamParameters APIs:

BOOL Init()

This function invokes the method LoadCameraMatrix.

Parameters	Description	Return Values
None	N/A.	TRUE on Success FALSE on Failure

BOOL RemapStereoImage(Mat LFrame, Mat RFrame, Mat *RImage, Mat *LImage)

This function performs rectification of right and left image with the computed matrix values such that the image differs only in the x-axis.

Parameters	Description	Return Values
Mat LFrame	Left Frame of the Camera	TRUE on Success FALSE on Failure
Mat RFrame	Right Frame of the Camera	
Mat *RImage	Right Image of the Camera	
Mat *LImage	Left Image of the Camera	

BOOL LoadCameraMatrix()

This function is used to load the camera matrix from the flash of the camera using the HID commands. Loads all the data from the intrinsic and extrinsic file to the Mat object created.

Parameters	Description	Return Values
None	N/A	TRUE on Success FALSE on Failure

BOOL ComputeRectifyPrms()

Computes the Q Mat used for the transformation of 2D points to 3D points. The maps to rectify the image is also created.

Parameters	Description	Return Values
None	N/A	TRUE on Success FALSE on Failure

Disparity APIs:

BOOL InitCamera(bool GenerateDisparity, bool FilteredDisparityMap)

This function creates an object for the CameraEnumeration class where the Device ID and Resolution selected by the user are returned. Opens the camera device with the following inputs. Invokes the InitExtensionUnit method of the HID. The Init function is called based on the input parameters of the Disparity map are set.

Parameters	Description	Return Values
Bool GenerateDisparity	Generates Disparity Map	TRUE on Success FALSE on Failure
Bool FilteredDisparityMap	Generates Filtered Disparity Map	

BOOL GrabFrame(Mat *LeftImage, Mat *RightImage)

This function grabs the frame the camera device initialized. Converts the Y16 data and splits the right image and left image respectively. Rectifies the frame using the RemapStereoImage method and pass the image as the reference.

Parameters	Description	Return Values
Mat *LeftImage	Left Image of the camera	TRUE on Success FALSE on Failure
Mat *RightImage	Right Image of the camera	

Note:

For the camera to stream the custom format, OpenCV has to be rebuild. Refer to the Tara-SDK_User_Manual in the documents folder of the SDK.

BOOL GetDisparity(Mat LImage, Mat RImage, Mat *DisparityMap, Mat *disp_filtered)

This function computes the disparity map for the left and right images passed with the parameters set in the SetAlgorithmParam method.

Parameters	Description	Return Values
Mat LImage	Left Image of the camera	TRUE on Success FALSE on Failure

Mat RImage	Right Image of the Camera
Mat *DispartiyMap	Disparity Map computed
Mat *disp_filtered	Disparity map for visualization is passed.

BOOL EstimateDepth(Point Pt, float *DepthValue)

This function estimates the depth of the point passed from the disparity map computed and returns the depth of the point.

Parameters	Description	Return Values
Point Pt	Point of which the depth is found	TRUE on Success FALSE on Failure
Float *DepthValue	Pointer to return the depth of the point passed	

BOOL SetExposure(int ExposureVal)

This function invokes the HID function SetManualExposureStereo to set the exposure of the device initialized. The range of the exposure is 10 to 1000000 micro seconds.

Parameters	Description	Return Values
Int ExposureVal	Exposure value to be set to the camera.	TRUE on Success FALSE on Failure

BOOL GetExposure(int *ExposureVal)

This function invokes the HID function GetManualExposureStereo to read the exposure of the device. The range of the manual exposure is 10 to 1000000 micro seconds. When the camera is in Auto Exposure, ExposureVal will be read as 1.

Parameters	Description	Return Values
Int *ExposureVal	Pointer to store the current value of the exposure of the camera.	TRUE on Success FALSE on Failure

BOOL SetAutoExposure()

This function invokes the HID function SetAutoExposureStereo to enable the Auto Exposure in the Camera. Exposure value is set to 1, to enable Auto Exposure.

Parameters	Description	Return Values
None	N/A	TRUE on Success FALSE on Failure

BOOL SetAlgorithmParam()

This function sets the disparity algorithm(Stereo BM / Stereo 3 way) to run based on the macro DISPARITY_OPTION.

DISPARITY_OPTION:

When set to 1 – Best quality Depth Map and Lower Frame Rate – Stereo_SGBM 3 way generic left to right.

When set to 0 – Low Quality Depth Map and Higher Frame Rate – Stereo_BM generic left to right.

Parameters	Description	Return Values
None	N/A	TRUE on Success FALSE on Failure

BOOL Init(bool GenerateDisparity)

This function invokes the init method of the TaraCamParameters. Initialises all the disparity variables and invokes the SetAlgorithmParam.

Parameters	Description	Return Values
Bool GenerateDisparity	Generates the disparity map	TRUE on Success FALSE on Failure

BOOL SetBrightness(int BrightnessVal)

This function uses the set property of VideoCapture in OpenCV to set the brightness. Value ranges from 1 to 7.

Parameters	Description	Return Values
Int BrightnessVal	Brightness value to be set to the camera.	TRUE on Success FALSE on Failure

BOOL SetStreamMode(UINT StreamMode)

This function sets the camera to the stream mode passed. Tara supports two models i) Master mode ii) Trigger mode. 1 is passed to set Master Mode and 0 is passed to set Trigger Mode.

Parameters	Description	Return Values
UINT StreamMode	Sets the Stream mode passed to the camera	TRUE on Success FALSE on Failure

BOOL GetStreamMode(UINT *StreamMode)

This function returns the stream mode in which the camera is streamed. Tara supports two modes i) Master mode ii) Trigger Mode. 1 is returned when the camera is in master mode and 0 if it is in Trigger mode.

Parameters	Description	Return Values
UINT *StreamMode	Pointer to the Stream Mode is passed.	TRUE on Success FALSE on Failure

CameraEnumeration:

int GetListofDevicesCon()

This function enumerates the devices connected to the computer using Directshow for windows and V4L2 for linux.

Parameters	Description	Return Values
None	N/A	Returns the number of connected devices to system.

BOOL GetDeviceIDeCon(int *DeviceID, Size *ResolutionSelected)

This function gets the user input for the device to be streamed. Device ID and Resolution to be set for the camera to stream.

Parameters	Description	Return Values
Int *DeviceId	Device ID of the camera to be streamed	TRUE on Success FALSE on Failure
Size *ResolutionSelected	Resolution to be set to the camera device	

BOOL IsStereoDeviceAvail(char *pid)

This function checks whether the enumerated device is eCon's Stereo Camera.

Parameters	Description	Return Values
Char *pid	Pointer holding the Product ID of the device selected.	TRUE on Success FALSE on Failure

CameraEnumeration(int *deviceId, Size *ResolutionSelected)

This function is the constructor of the class which invokes the GetDeviceIDeCon method.

Parameters	Description	Return Values
Int *deviceId	Pointer to read the MSB value of the temperature.	TRUE on Success FALSE on Failure
Size *ResolutionSelected	Pointer to read the LSB value of the temperature.	

void query_resolution(int DeviceID)

This function query's the resolutions supported by the selected camera in Y16 format and saves it in a vector declared in the CameraEnumeration Class.

Parameters	Description	Return Values
int DeviceID	Device id of the camera to be streamed.	N/A

Void freeDevices(void)

This function destroys the memory allocated while creating the device list of enumerated cameras.

Parameters	Description	Return Values
None	N/A	N/A

Contact Us

If you need any support on Tara 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
1.1	21-May-2016	Initial Draft	Karthikeyan A
1.2	14-September-2016	Added Auto Exposure and Trigger support	Karthikeyan A
1.3	15-December-2016	Added Auto Exposure support	Karthikeyan A
1.4	09-Jun-2018	Added new IMU Sensor and OpenCV version 3.4.1 support	Chandra Sekar V