ROS API

This is the official driver for the ESPROS TOF>cam 635. The annotation follows the rules of ROS.org.

Start of the node

If you use in terminal mode the APIs only, without GUI:
Start the ROS operating system in a Terminal1 with the command: **roscore**
Start the TOF>cam 635 in a Terminal2 with the command: **rosrun espros_tof_cam635 tof_cam635_node**

Published topics

Topic name	ROS msgs file	ROS message type	Function
camera/image_raw1	sensor_msgs	Image	Sends the grayscale or amplitude image according the selected image type parameter
camera/image_raw2	sensor_msgs	Image	Sends the distance image for image type parameters which include distance
camera/imageHeader	std_msg	Int32MultiArray	Sends the image header. Refer to Table 2.
camera/points	sensor_msgs	PointCloud2	Sends the point cloud image for image type parameters which include distance

Table 1: ESPROS ROS topics

Entry	Index
Header version	1
Frame counter	2
Timestamp	3
TOFCOS version	4
Hardware version	5
Chip ID	6
Image width (x-axis)	7
Image height (y-axis)	8
Image origin X	9

Entry	Index
reserved2	22
AmplitudeLimit0 WFOV	23
AmplitudeLimit1 WFOV	24
AmplitudeLimit2 WFOV	25
AmplitudeLimit3 WFOV	26
AmplitudeLimit4 NFOV	27
Offset	28
Binning type	29
DistanceTemporalFilter-Factor WFOV	30

Entry	Index	Entry	Index
Image origin Y	10	DistanceTemporalFilter-Threshold WFOV	31
CurrentIntegrationTime3D WFOV	11	SingleValueTemporalFilter-Factor NFOV	32
CurrentIntegrationTime3D NFOV	12	SingleValueTemporalFilter-Threshold NFOV	33
CurrentIntegrationTimeGrayscale	13	Modulation frequency	34
IntegrationTimeGrayscale	14	Modulation channel	35
IntegrationTime0 WFOV	15	Flags	36
IntegrationTime1 WFOV	16	Temperature	37
IntegrationTime2 WFOV	17	Illumination beam	38
IntegrationTime3 WFOV	18	NFOV distance	39
IntegrationTime4 NFOV	19	NFOV amplitude	40
Integration time5 NFOV	20	reserved3	41
reserved1	21	n/a	n/a

Table 2: Header parameters (see also notes below)

Notes:

- ROS header bytes: In total 164 bytes. Each parameter is transmitted in corresponding 32 bit data format (Int32MultiArray).

Dynamically reconfigurable parameters

Refer for details on the dynamically reconfigurable parameters to the enclosed "dynamic_reconfigure package" or to http://wiki.ros.org/dynamic_reconfigure.

Parameter	Function	Data format	Default
~image_type	Sets the image acquisition type 0: Grayscale 1: Distance 2: Distance and amplitude 3: Distance and grayscale	int	1
~mode	Sets the operation mode 0: Mode - WFOV 1: Mode - NFOV 2: Mode - NFOV result 3: Mode - NFOV result and image 4: Mode - WFOV and NFOV spot 5. Mode - WFOV and NFOV auto 6: Mode - Either WFOV, if object is inside range, else NFOV spot	int	0
~frame_rate	Sets frame acquisition speed Hz	double	30
~start_stream	Enables image streaming	bool	False
~trigger_single_shot	Starts single measurement after change from false to true	bool	False
~hdr	Sets HDR mode: 0: HDR OFF 1: HDR spatial 2: HDR temporal	int	0
~automatic_integration_time	Automatic mode: Integration time is set automatically for WFOV and NFOV between 1 and 1'000 µs.	bool	False
~integration_time_0	Sets the WFOV integration time for distance measurements in microseconds. Range: 1 … 4'000 µs	int	200
~integration_time_1		int	0
~integration_time_2		int	0
~integration_time_3		int	0
~integration_time_4	Sets the NFOV integration time for distance measurements in microseconds. Range: 1 … 4'000 µs	int	200
~integration_time_5		int	0
~integration_time_gray	Sets the integration time for grayscale measurements in microseconds. Range: 0 … 50'000 µs	int	0
~temporal_filter_factor_WFOV	Sets the factor 'k' of the WFOV temporal filter (Kalman)	int	10
~temporal_filter_threshold_WFOV	Sets the threshold 'T' of the WFOV temporal filter (Kalman)	int	300
~temporal_filter_factor_NFOV	Sets the factor 'k' of the NFOV temporal filter (Kalman)	int	10

Parameter	Function	Data format	Default
~temporal_filter_threshold_NFOV	Sets the threshold 'T' of the NFOV temporal filter (Kalman)	int	300
~spatial_average_filter	Enables the spatial average filter for distance filtering	bool	False
~spatial_median_filter	Enables the spatial median filter for distance filtering	bool	False
~min_amplitude_0	Sets the amplitude limits for WFOV. Range 0 … 2'047 LSB	int	50
~min_amplitude_1		int	100
~min_amplitude_2		int	200
~min_amplitude_3		int	500
~min_amplitude_4	Sets the amplitude limits for NFOV. Range 0 … 2'047 LSB	int	200
~roi_left_x	Sets the left edge of the ROI	int	0
~roi_right_x	Sets the right edge of the ROI	int	159
~roi_top_y	Sets the top edge of the ROI	int	0
~roi_bottom_y	Sets the top edge of the ROI	int	59

Table 3: ROS parameter table

Parameter	Function	Data format	Default	Reference
~lens_angle	FOV horizontally (x-axis) in degree, for linear lenses. It is used for the transformation from polar to cartesian coordinate system. Range 0 ... 180 degree. Note: Use for non-linear lenses the undistortion function.	int	50	n/a
~enable_undistortion	Enables for image and point cloud the lens undistortion based on calibrated parameters	bool	False	n/a
~enable_cartesian	Enables point cloud cartesian transformation (false = spheric)	bool	False	n/a
~enable_images	Activates imagePublisher1 and imagePublisher2 nodes to send information (camera/image_raw1/2)	bool	True	Table 1
~enable_point_cloud	Activates pointCloud2Publisher node to send information (camera/points)	bool	True	Table 1
~enable_image_header	Activates imageHeader node to send information (camera/imageHeader)	bool	True	Table 1

Table 3 cont: ROS parameter table