

Official Publication Date: March 2021
Official Expiration Date: Until next release



Document Number: 102-2020-42
Rev400

102-2020-42, Neutrino LC Software Interface Description Document, Rev 400
Information on this page is subject to change without notice.

EAR Controlled: EAR99
These commodities, technology or software are subject to the U.S. Export Administration Regulations (EAR). Diversion contrary to U.S. law is prohibited.

FLIR NEUTRINO LC SOFTWARE IDD

1	NeutrinoLC SDK Description	8
1.1	Global Objects	8
1.1.1	Basic Data Types	8
1.1.2	Enums.....	8
1.1.3	Structs	8
1.1.4	Functions.....	9
1.2	Module: AGC.....	9
1.2.1	Enums.....	9
1.2.2	Structs	10
1.2.3	Functions.....	10
1.3	Module: BOSON.....	23
1.3.1	Enums.....	23
1.3.2	Structs	24
1.3.3	Functions.....	25
1.4	Module: BPR.....	59
1.4.1	Enums.....	59
1.4.2	Structs	59
1.4.3	Functions.....	59
1.5	Module: CAPTURE.....	63
1.5.1	Enums.....	63
1.5.2	Structs	63
1.5.3	Functions.....	64
1.6	Module: COLORLUT.....	65
1.6.1	Enums.....	65
1.6.2	Structs	66
1.6.3	Functions.....	66
1.7	Module: COOLER	68

102-2020-42, Neutrino LC Software Interface Description Document, Rev 400
Information on this page is subject to change without notice.

EAR Controlled – See Cover Page



The World's Sixth Sense™

FLIR NEUTRINO LC SOFTWARE IDD

1.7.1	Enums.....	68
1.7.2	Structs	68
1.7.3	Functions.....	68
1.8	Module: CPE.....	73
1.8.1	Enums.....	73
1.8.2	Structs	73
1.8.3	Functions.....	73
1.9	Module: DBMF.....	77
1.9.1	Enums.....	77
1.9.2	Structs	77
1.9.3	Functions.....	77
1.10	Module: DUMMY.....	82
1.10.1	Enums.....	82
1.10.2	Structs	82
1.10.3	Functions.....	82
1.11	Module: DVO.....	82
1.11.1	Enums.....	82
1.11.2	Structs	84
1.11.3	Functions.....	85
1.12	Module: EXTLENSCONTROL.....	97
1.12.1	Enums.....	97
1.12.2	Structs	98
1.12.3	Functions.....	98
1.13	Module: FILEOPS.....	105
1.13.1	Enums.....	105
1.13.2	Structs	105
1.13.3	Functions.....	106

102-2020-42, Neutrino LC Software Interface Description Document, Rev 400
Information on this page is subject to change without notice.

EAR Controlled – See Cover Page



The World's Sixth Sense™

FLIR NEUTRINO LC SOFTWARE IDD

1.14	Module: FLASHIO	110
1.14.1	Enums.....	110
1.14.2	Structs	110
1.14.3	Functions.....	110
1.15	Module: GAO.....	111
1.15.1	Enums.....	111
1.15.2	Structs	111
1.15.3	Functions.....	112
1.16	Module: I2CBUS	116
1.16.1	Enums.....	117
1.16.2	Structs	117
1.16.3	Functions.....	118
1.17	Module: I2CCPE.....	119
1.17.1	Enums.....	119
1.17.2	Structs	119
1.17.3	Functions.....	119
1.18	Module: I2CTOSERIAL.....	122
1.18.1	Enums.....	122
1.18.2	Structs	123
1.18.3	Functions.....	123
1.19	Module: IMAGESTATS.....	128
1.19.1	Enums.....	129
1.19.2	Structs	129
1.19.3	Functions.....	129
1.20	Module: JFFS2	133
1.20.1	Enums.....	133
1.20.2	Structs	133

102-2020-42, Neutrino LC Software Interface Description Document, Rev 400
Information on this page is subject to change without notice.



FLIR NEUTRINO LC SOFTWARE IDD

1.20.3	Functions.....	133
1.21	Module: MEM.....	134
1.21.1	Enums.....	134
1.21.2	Structs.....	135
1.21.3	Functions.....	135
1.22	Module: ROIC.....	139
1.22.1	Enums.....	139
1.22.2	Structs.....	140
1.22.3	Functions.....	140
1.23	Module: SCALER.....	144
1.23.1	Enums.....	144
1.23.2	Structs.....	145
1.23.3	Functions.....	145
1.24	Module: SCNR.....	147
1.24.1	Enums.....	147
1.24.2	Structs.....	147
1.24.3	Functions.....	147
1.25	Module: SPLASHSCREEN.....	156
1.25.1	Enums.....	156
1.25.2	Structs.....	156
1.25.3	Functions.....	156
1.26	Module: SYMBOLOGY.....	158
1.26.1	Enums.....	159
1.26.2	Structs.....	159
1.26.3	Functions.....	159
1.27	Module: SYSCTRL.....	173
1.27.1	Enums.....	173

102-2020-42, Neutrino LC Software Interface Description Document, Rev 400
Information on this page is subject to change without notice.

EAR Controlled – See Cover Page



The World's Sixth Sense™

FLIR NEUTRINO LC SOFTWARE IDD

1.27.2	Structs	173
1.27.3	Functions.....	173
1.28	Module: SYSINFO.....	177
1.28.1	Enums.....	177
1.28.2	Structs	177
1.28.3	Functions.....	178
1.29	Module: SYSTEMSYMBOLS.....	181
1.29.1	Enums.....	181
1.29.2	Structs	181
1.29.3	Functions.....	181
1.30	Module: TELEMETRY	183
1.30.1	Enums.....	183
1.30.2	Structs	183
1.30.3	Functions.....	183
1.31	Module: TESTRAMP	186
1.31.1	Enums.....	186
1.31.2	Structs	186
1.31.3	Functions.....	187
1.32	Module: TF	189
1.32.1	Enums.....	190
1.32.2	Structs	190
1.32.3	Functions.....	190
1.33	Module: UART.....	194
1.33.1	Enums.....	194
1.33.2	Structs	194
1.33.3	Functions.....	194



102-2020-42, Neutrino LC Software Interface Description Document, Rev 400
Information on this page is subject to change without notice.

EAR Controlled – See Cover Page



The World's Sixth Sense™

1 NeutrinoLC SDK Description

The SDK describes the command and control API for the NeutrinoLC camera. Description of video interfaces and image capture are not included in this SDK.

1.1 Global Objects

1.1.1 Basic Data Types

SDK Name	C	C#	Python
CHAR	"int8_t"	"SByte"	"int"
UCHAR	"uint8_t"	"Byte"	"int"
INT_16	"int16_t"	"Int16"	"int"
UINT_16	"uint16_t"	"UInt16"	"int"
INT_32	"int32_t"	"Int32"	"int"
UINT_32	"uint32_t"	"UInt32"	"int"
FLOAT	"float"	"Double"	"float"
DOUBLE	"double"	"Double"	"float"

Some functions will use fixed size arrays of these data types. These arrays will be represented by "<DataType>*<size_of_array>". Other functions will use dynamically sized arrays (limited to the UCHAR or BYTE type). This will be noted as a BYTEARRAY. Note that functions using the "BYTEARRAY" type are limited in maximum size. Each SDK has a property, definition, or static variable that declares the maximum transfer size (typically 256 bytes).

1.1.2 Enums

Global Enumerations are available to all modules.

1.1.2.1 FLR_ENABLE_E — <INT_32>

```
FLR_DISABLE = 0
FLR_ENABLE = 1
FLR_ENABLE_END = 2
```

1.1.3 Structs

Global Structure types available to all modules.

1.1.3.1 FLR_ROI_T

Field Name	DataType	Bytes
------------	----------	-------

FLIR NEUTRINO LC SOFTWARE IDD

rowStart	UINT_16	2
rowStop	UINT_16	2
colStart	UINT_16	2
colStop	UINT_16	2

1.1.4 Functions

1.1.4.1 *Initialize(port, baudrate=921600)*

Starts communications and returns handle. Argument port may be "COM<N>" or <N-1> depending on platform. Baudrate is 921600 for Boson.

1.1.4.2 *Close(handle)*

Stops communications and releases handle.

1.2 Module: AGC

The Automatic Gain Control module provides API's to control and interrogate the automatic gain control algorithm.

1.2.1 Enums

1.2.1.1 *FLR_AGC_MODE_E — <INT_32>*

FLR_AGC_MODE_NORMAL = 0
FLR_AGC_MODE_HOLD = 1
FLR_AGC_MODE_THRESHOLD = 2
FLR_AGC_MODE_AUTO_BRIGHT = 3
FLR_AGC_MODE_AUTO_LINEAR = 4
FLR_AGC_MODE_MANUAL = 5
FLR_AGC_MODE_END = 6

1.2.1.2 *FLR_AGC_REGION_E — <INT_32>*

FLR_AGC_REGION_DISABLED = 0
FLR_AGC_REGION_CORRELATED = 1
FLR_AGC_REGION_NON_CORRELATED = 2
FLR_AGC_REGION_COLORIZE = 3
FLR_AGC_REGION_BLEND = 4
FLR_AGC_REGION_CORRELATED_HSV = 5
FLR_AGC_REGION_NON_CORRELATED_HSV = 6
FLR_AGC_REGION_LAST = 7

FLIR NEUTRINO LC SOFTWARE IDD

1.2.2 Structs

1.2.2.1 FLR_AGC_ISOOTHERM_COLOR_T

Field Name	DataType	Bytes
r	UINT_16	2
g	UINT_16	2
b	UINT_16	2

1.2.2.2 FLR_AGC_ISOOTHERM_COLORS_T

Field Name	DataType	Bytes
range1	FLR_AGC_ISOOTHERM_COLOR_T	6
range2	FLR_AGC_ISOOTHERM_COLOR_T	6
range3	FLR_AGC_ISOOTHERM_COLOR_T	6
num	UINT_16	2

1.2.3 Functions

1.2.3.1 *agcSetPercentPerBin()*

Defines that maximum percentage of pixels allowed in a bin in relation to the total number of pixels accumulated.

Input/Send parameters:

Name	DataType	Bytes	Notes
FunctionID	0x00090001	N/A	N/A
data	FLOAT	0:4	

No output parameters.

1.2.3.2 *agcGetPercentPerBin()*

Get the current PercentPerBin

Input/Send parameters:

Name	DataType	Bytes	Notes
FunctionID	0x00090002	N/A	N/A

Output/Receive parameters:

Name	DataType	Bytes	Notes
------	----------	-------	-------

102-2020-42, Neutrino LC Software Interface Description Document, Rev400

Information on this page is subject to change without notice

This document does not contain any export-controlled information.



FLIR NEUTRINO LC SOFTWARE IDD

data	FLOAT	0:4		
-------------	-------	-----	--	--

1.2.3.3 *agcSetLinearPercent()*

Defines how linear the mapping from the input to output dynamic range will be. The valid range of the variable is [0 100] where a value of 0 (%) means the transfer function shape will be based entirely on the input histogram and a value of 100 (%) means the transfer function will be a straight line (linear).

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x00090003	N/A	N/A	
data	FLOAT	0:4		

No output parameters.

1.2.3.4 *agcGetLinearPercent()*

Get the current setting for transfer function linearity.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x00090004	N/A	N/A	

Output/Receive parameters:

Name	Data Type	Bytes	Notes	
data	FLOAT	0:4		

1.2.3.5 *agcSetOutlierCut()*

Defines the amount of histogram "outliers" (beginning and end) to ignore as a percentage of histSum. A non-zero value for this parameter will limit the effect of outlier pixel values such as non-operational pixels or small areas with extremely high values (high irradiance) or low values (low irradiance).

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x00090005	N/A	N/A	
data	FLOAT	0:4		

FLIR NEUTRINO LC SOFTWARE IDD

No output parameters.

1.2.3.6 *agcGetOutlierCut()*

Get the outlier cut

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00090006	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLOAT	0:4	

1.2.3.7 *agcGetDrOut()*

Get the current dynamic range setting.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00090008	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLOAT	0:4	

1.2.3.8 *agcSetMaxGain()*

Set the maximum transfer function gain. This gain limit is applied on a per bin basis such that locally the transfer function slope never exceeds the limit defined by the maxGain parameter.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00090009	N/A	N/A
data	FLOAT	0:4	

No output parameters.

FLIR NEUTRINO LC SOFTWARE IDD

1.2.3.9 *agcGetMaxGain()*

Get the current maximum gain of the transfer function.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0009000A	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLOAT	0:4	

1.2.3.10 *agcSetdf()*

Set the damping factor. This controls the update rate of the transfer function per function call. The damping factor has a valid range of [0 1] where a value of 0 means there is no damping and the latest calculated transfer function will be the output transfer function and a value of 1.0 for df will freeze the update of the transfer function (i.e. 100% damped).

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0009000B	N/A	N/A
data	FLOAT	0:4	

No output parameters.

1.2.3.11 *agcGetdf()*

Get the current damping factor.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0009000C	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLOAT	0:4	

FLIR NEUTRINO LC SOFTWARE IDD

1.2.3.12 *agcSetGamma()*

Set the gamma correction value. This parameter can be used to compensate for the gamma of the display. In this implementation $\text{gamma} < 1$ will generate a transfer function that has more contrast in the high irradiance range. Negative values for gamma are not allowed.

Input/Send parameters:

Name	DataType	Bytes	Notes
FunctionID	0x0009000D	N/A	N/A
data	FLOAT	0:4	

No output parameters.

1.2.3.13 *agcGetGamma()*

Get the current gamma correction value.

Input/Send parameters:

Name	DataType	Bytes	Notes
FunctionID	0x0009000E	N/A	N/A

Output/Receive parameters:

Name	DataType	Bytes	Notes
data	FLOAT	0:4	

1.2.3.14 *agcGetFirstBin()*

Get the index of the first populated bin in the histogram (starting from bin 0). If *outlierCut* is set to a value greater than zero *firstBin* may not be the first bin containing non-zero value (see *outlierCut*).

Input/Send parameters:

Name	DataType	Bytes	Notes
FunctionID	0x00090010	N/A	N/A

Output/Receive parameters:

Name	DataType	Bytes	Notes
data	UINT_32	0:4	

FLIR NEUTRINO LC SOFTWARE IDD

1.2.3.15 *agcGetLastBin()*

Get the index of the last populated bin in the histogram (starting from bin 0). If outlierCut is set to a value greater than zero lastBin may not be the last bin containing a non-zero value (see outlierCut).

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00090012	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	UINT_32	0:4	

1.2.3.16 *agcSetDetailHeadroom()*

Set the amount of headroom to be given to the detail component when DDE is enabled and has a non-zero gain. If this parameter is set to zero: positive valued detail signals (HP signal) in the highest irradiance regions of the image may saturate at the drOut level and negative valued detail signals in the lowest irradiance regions may saturate at 0. The allowed range for this parameter is [0 drOut-1] with a typical value of 10.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00090013	N/A	N/A
data	FLOAT	0:4	

No output parameters.

1.2.3.17 *agcGetDetailHeadroom()*

Get the current Detail Headroom

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00090014	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLOAT	0:4	

FLIR NEUTRINO LC SOFTWARE IDD

1.2.3.18 *agcSetd2br()*

Set the detail-to-background-ratio (d2br). This defines the ratio of the detail (HP) gain and the maximum slope/gain of the background (LP). The allowable range for this parameter is [0 inf] with a typical setting of 1.3.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00090015	N/A	N/A
data	FLOAT	0:4	

No output parameters.

1.2.3.19 *agcGetd2br()*

Get the current detail-to-background-ratio (d2br).

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00090016	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLOAT	0:4	

1.2.3.20 *agcSetSigmaR()*

Set the smoothing factor. This defines the properties of the edge-preserving low pass filter used for the DDE functionality. Higher values for this parameter will result in more aggressive low pass filtering which will cause higher amplitude signals to be present in the detail (HP) component. Allowable range is [0 inf] with a typical setting of 2000. Value should be proportional to imager responsivity.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00090017	N/A	N/A
data	FLOAT	0:4	

FLIR NEUTRINO LC SOFTWARE IDD

No output parameters.

1.2.3.21 *agcGetSigmaR()*

Get the current smoothing factor.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00090018	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLOAT	0:4	

1.2.3.22 *agcSetUseEntropy()*

Switches from Plateau Equalization to Entropy Equalization

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0009001E	N/A	N/A
data	FLR_ENABLE_E	0:4	

No output parameters.

1.2.3.23 *agcGetUseEntropy()*

Get the Entropy State of AGC

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0009001F	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLR_ENABLE_E	0:4	

FLIR NEUTRINO LC SOFTWARE IDD

1.2.3.24 *agcSetROI()*

Define the current region of interest. Set the start and stop columns and rows, starting with column=0, row=0 in the upper left corner.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00090020	N/A	N/A
roi	FLR_ROI_T	0:8	

No output parameters.

1.2.3.25 *agcGetROI()*

Get the current boundaries of the ROI. The returned ROI is a datatype that describes row start, row stop, column start, and column stop.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00090021	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
roi	FLR_ROI_T	0:8	

1.2.3.26 *agcGetMaxGainApplied()*

Gets the scaled value of the max gain

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00090025	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLOAT	0:4	

1.2.3.27 *agcGetSigmaRApplied()*

Gets the scaled value of sigma R

FLIR NEUTRINO LC SOFTWARE IDD

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00090026	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLOAT	0:4	

1.2.3.28 *agcSetOutlierCutBalance()*

Adjust the outlier cut to be asymmetrical at the top and bottom of histogram.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00090027	N/A	N/A
data	FLOAT	0:4	

No output parameters.

1.2.3.29 *agcGetOutlierCutBalance()*

Get the current outlier cut ratio between top and bottom of histogram.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00090028	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLOAT	0:4	

1.2.3.30 *agcGetOutlierCutApplied()*

Get the applied outlier cut percentages at the top and bottom of histogram.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00090029	N/A	N/A

FLIR NEUTRINO LC SOFTWARE IDD

Output/Receive parameters:

Name	Data Type	Bytes	Notes
percentHigh	FLOAT	0:4	
percentLow	FLOAT	4:8	

1.2.3.31 *agcGetTfThresholds()*

Get the threshold values for AGC.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00090030	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
tf_thresholddMin	UINT_16	0:2	
tf_thresholddMax	UINT_16	2:4	

1.2.3.32 *agcSetTfThresholds()*

Set the threshold values used when overriding AGC (using *agcSetMode*).

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00090031	N/A	N/A
tf_thresholddMin	UINT_16	0:2	
tf_thresholddMax	UINT_16	2:4	

No output parameters.

1.2.3.33 *agcGetMode()*

The AGC mode. Normal - automatic processing based on the image. Hold - take what was calculated in Normal mode and make it fixed. Threshold - analyze frame but set first and last bin values based on thresholds.

FLIR NEUTRINO LC SOFTWARE IDD

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00090032	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
mode	FLR_AGC_MODE_E	0:4	

1.2.3.34 *agcSetMode()*

The AGC mode. Normal - automatic processing based on the image. Hold - take what was calculated in Normal mode and make it fixed. Threshold - analyze frame but set first and last bin values based on thresholds.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00090033	N/A	N/A
mode	FLR_AGC_MODE_E	0:4	

No output parameters.

1.2.3.35 *agcGetContrast()*

The user adjustable contrast of the scene

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00090040	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
contrast	INT_32	0:4	

1.2.3.36 *agcSetContrast()*

The user adjustable contrast of the scene

Input/Send parameters:

Name	Data Type	Bytes	Notes
------	-----------	-------	-------

FLIR NEUTRINO LC SOFTWARE IDD

FunctionID	0x00090041	N/A	N/A	
contrast	INT_32	0:4		

No output parameters.

1.2.3.37 *agcGetBrightnessBias()*

The user adjustable brightness bias of the scene

Input/Send parameters:

Name	DataType	Bytes	Notes	
FunctionID	0x00090042	N/A	N/A	

Output/Receive parameters:

Name	DataType	Bytes	Notes	
brightness Bias	INT_32	0:4		

1.2.3.38 *agcSetBrightnessBias()*

The user adjustable brightness bias of the scene

Input/Send parameters:

Name	DataType	Bytes	Notes	
FunctionID	0x00090043	N/A	N/A	
brightness Bias	INT_32	0:4		

No output parameters.

1.2.3.39 *agcGetBrightness()*

The user adjustable brightness of the scene

Input/Send parameters:

Name	DataType	Bytes	Notes	
FunctionID	0x00090044	N/A	N/A	

Output/Receive parameters:

Name	DataType	Bytes	Notes	
------	----------	-------	-------	--

FLIR NEUTRINO LC SOFTWARE IDD

brightness	INT_32	0:4		
-------------------	--------	-----	--	--

1.2.3.40 *agcSetBrightness()*

The user adjustable brightness of the scene

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x00090045	N/A	N/A	
brightness	INT_32	0:4		

No output parameters.

1.3 Module: BOSON

Functions to control the general operation of the Boson camera.

1.3.1 Enums

1.3.1.1 *FLR_BOSON_GAINMODE_E* — <INT_32>

FLR_BOSON_HIGH_GAIN = 0
FLR_BOSON_LOW_GAIN = 1
FLR_BOSON_AUTO_GAIN = 2
FLR_BOSON_DUAL_GAIN = 3
FLR_BOSON_MANUAL_GAIN = 4
FLR_BOSON_GAINMODE_END = 5

1.3.1.2 *FLR_BOSON_FFCMODE_E* — <INT_32>

FLR_BOSON_MANUAL_FFC = 0
FLR_BOSON_AUTO_FFC = 1
FLR_BOSON_EXTERNAL_FFC = 2
FLR_BOSON_SHUTTER_TEST_FFC = 3
FLR_BOSON_FFCMODE_END = 4

1.3.1.3 *FLR_BOSON_TIMESTAMPTYPE_E* — <INT_32>

FLR_BOSON_UARTINIT = 0
FLR_BOSON_PIXELCLOCKINIT = 1
FLR_BOSON_AUTHEVENT = 2
FLR_BOSON_FIRSTVALIDIMAGE = 3
FLR_BOSON_TIMESTAMPTYPE_END = 4

102-2020-42, Neutrino LC Software Interface Description Document, Rev400

Information on this page is subject to change without notice

This document does not contain any export-controlled information.



FLIR NEUTRINO LC SOFTWARE IDD

1.3.1.4 FLR_BOSON_FFCSTATUS_E — <INT_32>

FLR_BOSON_NO_FFC_PERFORMED = 0
FLR_BOSON_FFC_IMMINENT = 1
FLR_BOSON_FFC_IN_PROGRESS = 2
FLR_BOSON_FFC_COMPLETE = 3
FLR_BOSON_FFCSTATUS_END = 4

1.3.1.5 FLR_BOSON_MYRIADTEMPMODE_E — <INT_32>

FLR_BOSON_NORMAL_MYRIADTEMP_MODE = 0
FLR_BOSON_STATIC_MYRIADTEMP_MODE = 1

1.3.1.6 FLR_BOSON_EXT_SYNC_MODE_E — <INT_32>

FLR_BOSON_EXT_SYNC_DISABLE_MODE = 0
FLR_BOSON_EXT_SYNC_MASTER_MODE = 1
FLR_BOSON_EXT_SYNC_SLAVE_MODE = 2
FLR_BOSON_EXT_SYNC_END = 3

1.3.1.7 FLR_BOSON_TEMP_DIODE_STATUS_E — <INT_32>

FLR_BOSON_TEMP_DIODE_NORMAL = 0
FLR_BOSON_TEMP_DIODE_FAULT = 1
FLR_BOSON_TEMP_DIODE_END = 2

1.3.1.8 FLR_BOSON_TABLETYPE_E — <INT_32>

FLR_BOSON_LOWGAIN_TABLE = 0
FLR_BOSON_HIGHGAIN_TABLE = 1
FLR_BOSON_TABLETYPE_END = 2

1.3.2 Structs

1.3.2.1 FLR_BOSON_PARTNUMBER_T

Field Name	Data Type	Bytes
value	UCHAR*20	20

1.3.2.2 FLR_BOSON_SENSOR_PARTNUMBER_T

Field Name	Data Type	Bytes
value	UCHAR*32	32

1.3.2.3 FLR_BOSON_GAIN_SWITCH_PARAMS_T

Field Name	Data Type	Bytes
------------	-----------	-------

102-2020-42, Neutrino LC Software Interface Description Document, Rev400

Information on this page is subject to change without notice

This document does not contain any export-controlled information.



FLIR NEUTRINO LC SOFTWARE IDD

pHighToLowPercent	UINT_32	4
cHighToLowPercent	UINT_32	4
pLowToHighPercent	UINT_32	4
hysteresisPercent	UINT_32	4

1.3.3 Functions

1.3.3.1 *bosonGetCameraSN()*

Returns the camera's serial number.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050002	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	UINT_32	0:4	

1.3.3.2 *bosonGetCameraPN()*

Returns the camera's part number.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050004	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLR_BOSON_PARTNUMBER_T	0:20	

1.3.3.3 *bosonGetSensorSN()*

Returns the sensor's serial number.

Input/Send parameters:

102-2020-42, Neutrino LC Software Interface Description Document, Rev400

Information on this page is subject to change without notice

This document does not contain any export-controlled information.

FLIR NEUTRINO LC SOFTWARE IDD

Name	Data Type	Bytes	Notes
FunctionID	0x00050006	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	UINT_32	0:4	

1.3.3.4 *bosonRunFFC()*

Performs an FFC operation.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050007	N/A	N/A

No output parameters.

1.3.3.5 *bosonSetFFCTempThreshold()*

Sets the temperature threshold (in degC*10) for the FFC desired flag. If the camera is in Auto FFC mode, an FFC desired flag will result in an automatic FFC event.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050008	N/A	N/A
data	UINT_16	0:2	

No output parameters.

1.3.3.6 *bosonGetFFCTempThreshold()*

Gets the temperature threshold (in degC*10) for the FFC desired flag.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050009	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	UINT_16	0:2	

FLIR NEUTRINO LC SOFTWARE IDD

1.3.3.7 *bosonSetFFCFrameThreshold()*

Sets the time threshold (in seconds) for the FFC desired flag. If the camera is in Auto FFC mode, an FFC desired flag will result in an automatic FFC event.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0005000A	N/A	N/A
data	UINT_32	0:4	

No output parameters.

1.3.3.8 *bosonGetFFCFrameThreshold()*

Gets the time threshold (in seconds) for the FFC desired flag.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0005000B	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	UINT_32	0:4	

1.3.3.9 *bosonGetFFCInProgress()*

Gets the mode of the FFC state machine.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0005000C	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	INT_16	0:2	

1.3.3.10 *bosonReboot()*

Tells the camera to perform a reboot.

102-2020-42, Neutrino LC Software Interface Description Document, Rev400

Information on this page is subject to change without notice

This document does not contain any export-controlled information.



FLIR NEUTRINO LC SOFTWARE IDD

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050010	N/A	N/A

No output parameters.

1.3.3.11 *bosonSetFFCMode()*

Sets the mode of the camera's FFC operation.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050012	N/A	N/A
ffcMode	FLR_BOSON_FFCMODE_E	0:4	

No output parameters.

1.3.3.12 *bosonGetFFCMode()*

Gets the mode of the camera's FFC operation.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050013	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
ffcMode	FLR_BOSON_FFCMODE_E	0:4	

1.3.3.13 *bosonSetGainMode()*

The mode of the camera's temperature compensation operation.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050014	N/A	N/A
gainMode	FLR_BOSON_GAINMODE_E	0:4	

No output parameters.

FLIR NEUTRINO LC SOFTWARE IDD

1.3.3.14 *bosonGetGainMode()*

The mode of the camera's temperature compensation operation.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050015	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
gainMode	FLR_BOSON_GAINMODE_E	0:4	

1.3.3.15 *bosonWriteDynamicHeaderToFlash()*

Takes the current settings of the camera and stores them to the Dynamic header, part of the non-volatile flash for User settings, to be used at start-up.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050018	N/A	N/A

No output parameters.

1.3.3.16 *bosonReadDynamicHeaderFromFlash()*

Reads the settings stored in Dynamic header and writes them over the current values in use.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050019	N/A	N/A

No output parameters.

1.3.3.17 *bosonRestoreFactoryDefaultsFromFlash()*

Reads the settings stored in Factory header and writes them over the current values in use.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0005001B	N/A	N/A

No output parameters.

FLIR NEUTRINO LC SOFTWARE IDD

1.3.3.18 *bosonRestoreFactoryBadPixelsFromFlash()*

Reads the bad pixels stores in the Factory Bad Pixel map and writes them over the current bap pixel map in use.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050020	N/A	N/A

No output parameters.

1.3.3.19 *bosonWriteBadPixelsToFlash()*

Writes the current bad pixel and vector offsets in use to the User Bad Pixel portion of the non-volatile flash.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050021	N/A	N/A

No output parameters.

1.3.3.20 *bosonGetSoftwareRev()*

Returns the version of the Camera Software.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050022	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
major	UINT_32	0:4	
minor	UINT_32	4:8	
patch	UINT_32	8:12	

1.3.3.21 *bosonSetBadPixelLocation()*

Mark a pixel location as bad, for replacement by the Bad Pixel Replacement module.

Input/Send parameters:

FLIR NEUTRINO LC SOFTWARE IDD

Name	Data Type	Bytes	Notes
FunctionID	0x0005002D	N/A	N/A
row	UINT_32	0:4	
col	UINT_32	4:8	

No output parameters.

1.3.3.22 *bosonlookupFPATempDegCx10()*

Returns the camera's sensor temp in degrees Celcius x10.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050030	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	INT_16	0:2	

1.3.3.23 *bosonlookupFPATempDegKx10()*

Returns the camera's sensor temp in degrees Kelvin x10.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050031	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	UINT_16	0:2	

1.3.3.24 *bosonSetLensNumber()*

Sets the desired lens number

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050038	N/A	N/A
lensNumbe	UINT_32	0:4	

FLIR NEUTRINO LC SOFTWARE IDD

r				
----------	--	--	--	--

No output parameters.

1.3.3.25 *bosonGetLensNumber()*

Returns the current lens number.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x00050039	N/A	N/A	

Output/Receive parameters:

Name	Data Type	Bytes	Notes	
lensNumber r	UINT_32	0:4		

1.3.3.26 *bosonSetTableNumber()*

Sets the desired table number.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x0005003A	N/A	N/A	
tableNumber	UINT_32	0:4		

No output parameters.

1.3.3.27 *bosonGetTableNumber()*

Returns the current table number.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x0005003B	N/A	N/A	

Output/Receive parameters:

Name	Data Type	Bytes	Notes	
tableNumber r	UINT_32	0:4		

FLIR NEUTRINO LC SOFTWARE IDD

1.3.3.28 *bosonGetSensorPN()*

Returns the sensor's part number.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0005003F	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
sensorPN	FLR_BOSON_SENSOR_PARTNUMBE R_T	0:32	

1.3.3.29 *bosonGetCLOWToHighPercent()*

The calculated percent counts for the transition from low gain to high gain state.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050044	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
cLOWToHighPercent	UINT_32	0:4	

1.3.3.30 *bosonGetMaxNUCTables()*

Returns the number of the highest Gain table.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050045	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
maxNUCTables	UINT_32	0:4	

FLIR NEUTRINO LC SOFTWARE IDD

1.3.3.31 *bosonGetMaxLensTables()*

Returns the number of Lens tables the camera supports.

Input/Send parameters:

Name	DataType	Bytes	Notes
FunctionID	0x00050046	N/A	N/A

Output/Receive parameters:

Name	DataType	Bytes	Notes
maxLensTables	UINT_32	0:4	

1.3.3.32 *bosonGetFfcWaitCloseFrames()*

Gets number of frames to wait for the shutter to close during an Auto or Manual FFC.

Input/Send parameters:

Name	DataType	Bytes	Notes
FunctionID	0x0005004E	N/A	N/A

Output/Receive parameters:

Name	DataType	Bytes	Notes
data	UINT_16	0:2	

1.3.3.33 *bosonSetFfcWaitCloseFrames()*

Sets number of frames to wait for the shutter to close during an Auto or Manual FFC.

Input/Send parameters:

Name	DataType	Bytes	Notes
FunctionID	0x0005004F	N/A	N/A
data	UINT_16	0:2	

No output parameters.

1.3.3.34 *bosonCheckForTableSwitch()*

Performs table switch if camera's "Table Switch Desired" flag has been set.

FLIR NEUTRINO LC SOFTWARE IDD

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050050	N/A	N/A

No output parameters.

1.3.3.35 *bosonGetDesiredTableNumber()*

Gets the table number that the camera wants to switch to.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050052	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
desiredTableNumber	UINT_32	0:4	

1.3.3.36 *bosonGetFfcStatus()*

Gets the status of the FFC function - FLR_BOSON_NO_FFC_PERFORMED, FLR_BOSON_FFC_IMMINENT, FLR_BOSON_FFC_IN_PROGRESS, FLR_BOSON_FFC_COMPLETE

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050054	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
ffcStatus	FLR_BOSON_FFCSTATUS_E	0:4	

1.3.3.37 *bosonGetFfcDesired()*

Gets the state of the FFC desired flag.

Input/Send parameters:

Name	Data Type	Bytes	Notes
------	-----------	-------	-------

FLIR NEUTRINO LC SOFTWARE IDD

FunctionID	0x00050055	N/A	N/A	
-------------------	------------	-----	-----	--

Output/Receive parameters:

Name	Data Type	Bytes	Notes	
ffcDesired	UINT_32	0:4		

1.3.3.38 *bosonGetSwRevInHeader()*

Gets the version of the software that the header was written with. It could be different than the current software version.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x00050056	N/A	N/A	

Output/Receive parameters:

Name	Data Type	Bytes	Notes	
major	UINT_32	0:4		
minor	UINT_32	4:8		
patch	UINT_32	8:12		

1.3.3.39 *bosonWriteCurrentNUCHeaderToFlash()*

Write current NUC header parameters to flash.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x00050058	N/A	N/A	

No output parameters.

1.3.3.40 *bosonGetLastFFCFrameCount()*

Gets the frame counter value at the time of the last FFC

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x0005005D	N/A	N/A	

FLIR NEUTRINO LC SOFTWARE IDD

Output/Receive parameters:

Name	Data Type	Bytes	Notes
frameCount	UINT_32	0:4	

1.3.3.41 *bosonGetLastFFCTempDegKx10()*

Gets the FPA temperature value at the time of the last FFC

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0005005E	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
temp	UINT_16	0:2	

1.3.3.42 *bosonGetTableSwitchDesired()*

Determine if camera gain wants to select a different table.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0005005F	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
tableSwitch Desired	UINT_16	0:2	

1.3.3.43 *bosonGetOverTempThreshold()*

Gets the temperature threshold in deg C above with the camera will go into low power state.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050061	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
------	-----------	-------	-------

FLIR NEUTRINO LC SOFTWARE IDD

temperatureInC	FLOAT	0:4		
-----------------------	-------	-----	--	--

1.3.3.44 bosonGetLowPowerMode()

Gets the state of low power flag. If enabled, the camera is in low power state and not fully functional.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050062	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
lowPowerMode	UINT_16	0:2	

1.3.3.45 bosonGetOverTempEventOccurred()

Gets the state of the overTemp event occurred flag. If enabled, it means that an overTemp event has occurred. It continues to be set in the low power state. It gets cleared when the core temperature goes below or is equal to Threshold - 6°C.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050063	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
overTempEventOccurred	UINT_16	0:2	

1.3.3.46 bosonSetPermitThermalShutdownOverride()

Sets the flag to ignore the overTemp event.

Input/Send parameters:

Name	Data Type	Bytes	Notes
------	-----------	-------	-------

FLIR NEUTRINO LC SOFTWARE IDD

FunctionID	0x00050064	N/A	N/A	
permitThermalShutdownOverride	FLR_ENABLE_E	0:4		

No output parameters.

1.3.3.47 *bosonGetPermitThermalShutdownOverride()*

Gets the state of the flag to ignore the overTemp event.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x00050065	N/A	N/A	

Output/Receive parameters:

Name	Data Type	Bytes	Notes	
permitThermalShutdownOverride	FLR_ENABLE_E	0:4		

1.3.3.48 *bosonGetMyriadTemp()*

Gets the core temperature in °C.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x00050068	N/A	N/A	

Output/Receive parameters:

Name	Data Type	Bytes	Notes	
myriadTemp	FLOAT	0:4		

1.3.3.49 *bosonWriteNucTableToFlash()*

Write current NUC table to flash.

FLIR NEUTRINO LC SOFTWARE IDD

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0005006B	N/A	N/A

No output parameters.

1.3.3.50 *bosonSetFFCWarnTimeInSecx10()*

Sets the amount of time in 10ths of a second before the occurrence of FFC that the warn time symbol should be displayed and the ffc state is set to imminent.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050074	N/A	N/A
ffcWarnTime	UINT_16	0:2	

No output parameters.

1.3.3.51 *bosonGetFFCWarnTimeInSecx10()*

Gets the ffc warn time in 10ths of a second.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050075	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
ffcWarnTime	UINT_16	0:2	

1.3.3.52 *bosonGetOverTempEventCounter()*

Gets the counter value that counts the number of times the overTemp event occurred.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050076	N/A	N/A

Output/Receive parameters:

102-2020-42, Neutrino LC Software Interface Description Document, Rev400

Information on this page is subject to change without notice

This document does not contain any export-controlled information.

FLIR NEUTRINO LC SOFTWARE IDD

Name	Data Type	Bytes	Notes
overTempEventCounter	UINT_32	0:4	

1.3.3.53 *bosonSetOverTempTimerInSec()*

Sets the time in seconds that we want to wait before setting the camera in low power state after an overTemp event has occurred.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050077	N/A	N/A
overTempTimerInSec	UINT_16	0:2	

No output parameters.

1.3.3.54 *bosonGetOverTempTimerInSec()*

Gets the overTemp countdown timer in seconds.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050078	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
overTempTimerInSec	UINT_16	0:2	

1.3.3.55 *bosonUnloadCurrentLensCorrections()*

Set current lens maps to unity. Lens gain map will remain until next reboot, SFFC/NVFFC will remain until next lens switch (or reboot). See also: *bosonReloadCurrentLensCorrections*.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050079	N/A	N/A

FLIR NEUTRINO LC SOFTWARE IDD

No output parameters.

1.3.3.56 *bosonSetTimeForQuickFFCsInSecs()*

Sets the number of seconds after startup that FFC trigger params are 'reduced' to produce FFC events more often.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0005007A	N/A	N/A
timeForQuickFFCsInSecs	UINT_32	0:4	

No output parameters.

1.3.3.57 *bosonGetTimeForQuickFFCsInSecs()*

Gets the number of seconds after startup that FFC trigger params are 'reduced' to produce FFC events more often.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0005007B	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
timeForQuickFFCsInSecs	UINT_32	0:4	

1.3.3.58 *bosonReloadCurrentLensCorrections()*

Reload current lens maps from non-volatile flash.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0005007C	N/A	N/A

No output parameters.

FLIR NEUTRINO LC SOFTWARE IDD

1.3.3.59 *bosonGetBootTimestamps()*

Get several hard coded timestamp values.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0005007F	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
FirstLight	FLOAT	0:4	
StartInit	FLOAT	4:8	
BosonExec Done	FLOAT	8:12	
Timestamp 4	FLOAT	12:16	

1.3.3.60 *bosonSet2ptResponsivityHighLimit()*

Sets the response high threshold for bad pixel identification occurring with a 2-point calibration.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050081	N/A	N/A
responsivityHighLimit	FLOAT	0:4	

No output parameters.

1.3.3.61 *bosonGet2ptResponsivityHighLimit()*

Gets the response high threshold for bad pixel identification occurring with a 2-point calibration.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050082	N/A	N/A

Output/Receive parameters:

FLIR NEUTRINO LC SOFTWARE IDD

Name	Data Type	Bytes	Notes
responsivityHighLimit	FLOAT	0:4	

1.3.3.62 *bosonSet2ptResponsivityLowLimit()*

Sets the response low threshold for bad pixel identification occurring with a 2-point calibration.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050083	N/A	N/A
responsivityLowLimit	FLOAT	0:4	

No output parameters.

1.3.3.63 *bosonGet2ptResponsivityLowLimit()*

Gets the response low threshold for bad pixel identification occurring with a 2-point calibration.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050084	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
responsivityLowLimit	FLOAT	0:4	

1.3.3.64 *bosonGet2ptResponsivityHighLimitErrorCount()*

Gets the count of pixels identified as inoperable during a 2-point calibration due to the response high threshold.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050085	N/A	N/A

FLIR NEUTRINO LC SOFTWARE IDD

Output/Receive parameters:

Name	Data Type	Bytes	Notes
responsivityHighLimitErrorCount	UINT_32	0:4	

1.3.3.65 *bosonGet2ptResponsivityLowLimitErrorCount()*

Gets the count of pixels identified as inoperable during a 2-point calibration due to the response low threshold.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050086	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
responsivityLowLimitErrorCount	UINT_32	0:4	

1.3.3.66 *bosonGet2ptPixelHighLimit()*

Gets the offset high threshold for bad pixel identification occurring with a 1-point or 2-point calibration.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050087	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
pixelHighLimit	UINT_32	0:4	

1.3.3.67 *bosonSet2ptPixelHighLimit()*

Sets the offset high threshold for bad pixel identification occurring with a 1-point or 2-point calibration.

FLIR NEUTRINO LC SOFTWARE IDD

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050088	N/A	N/A
pixelHighLimit	UINT_32	0:4	

No output parameters.

1.3.3.68 *bosonGet2ptPixelLowLimit()*

Gets the offset low threshold for bad pixel identification occurring with a 1-point or 2-point calibration.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050089	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
pixelLowLimit	UINT_32	0:4	

1.3.3.69 *bosonSet2ptPixelLowLimit()*

Sets the offset low threshold for bad pixel identification occurring with a 1-point or 2-point calibration.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0005008A	N/A	N/A
pixelLowLimit	UINT_32	0:4	

No output parameters.

1.3.3.70 *bosonGet2ptPixelHighLimitErrorCount()*

Gets the count of pixels identified as inoperable during a 1-point or 2-point calibration due to the offset high threshold.

Input/Send parameters:

102-2020-42, Neutrino LC Software Interface Description Document, Rev400

Information on this page is subject to change without notice

This document does not contain any export-controlled information.

FLIR NEUTRINO LC SOFTWARE IDD

Name	Data Type	Bytes	Notes
FunctionID	0x0005008B	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
pixelHighLimitErrorCount	UINT_32	0:4	

1.3.3.71 *bosonGet2ptPixelLowLimitErrorCount()*

Gets the count of pixels identified as inoperable during a 1-point or 2-point calibration due to the offset low threshold.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0005008C	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
pixelLowLimitErrorCount	UINT_32	0:4	

1.3.3.72 *bosonGet2ptTotalBadPixelErrorCount()*

Gets the total count of pixels identified as inoperable during a 1-point or 2-point calibration.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0005008D	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
totalBadPixelErrorCount	UINT_32	0:4	

FLIR NEUTRINO LC SOFTWARE IDD

1.3.3.73 *bosonGet2ptNucStatusState()*

Gets the current state or status of the 1-point or 2-point NUC calibration.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0005008E	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
statusState	UINT_32	0:4	
statusStringLength	UINT_32	4:8	

1.3.3.74 *bosonSet2ptNucStatusState()*

Sets the current state or status of the 1-point or 2-point NUC calibration.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0005008F	N/A	N/A
statusState	UINT_32	0:4	

No output parameters.

1.3.3.75 *bosonReset2ptNucStatusState()*

Resets the current state or status of the 1-point or 2-point NUC calibration.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050090	N/A	N/A

No output parameters.

1.3.3.76 *bosonGet2ptNucStatusStateString()*

Gets the current state or status string of the 1-point or 2-point NUC calibration.

Input/Send parameters:

Name	Data Type	Bytes	Notes
------	-----------	-------	-------

FLIR NEUTRINO LC SOFTWARE IDD

FunctionID	0x00050091	N/A	N/A	
statusState	UINT_32	0:4		
sizeInBytes	UINT_16	4:6		

Output/Receive parameters:

Name	DataType	Bytes	Notes	
data	BYTEARRAY	0:512		

1.3.3.77 *bosonGet2ptNucResultCode()*

Gets the code result of the 1-point or 2-point NUC calibration process.

Input/Send parameters:

Name	DataType	Bytes	Notes	
FunctionID	0x00050092	N/A	N/A	

Output/Receive parameters:

Name	DataType	Bytes	Notes	
resultCode	UINT_32	0:4		
resultString Length	UINT_32	4:8		

1.3.3.78 *bosonGet2ptNucResultString()*

Gets the string result of the 1-point or 2-point NUC calibration process.

Input/Send parameters:

Name	DataType	Bytes	Notes	
FunctionID	0x00050093	N/A	N/A	
resultNumber	UINT_32	0:4		
sizeInBytes	UINT_16	4:6		

Output/Receive parameters:

Name	DataType	Bytes	Notes	
data	BYTEARRAY	0:512		

FLIR NEUTRINO LC SOFTWARE IDD

1.3.3.79 *boson2ptNucStart()*

Starts the 1-point or 2-point NUC state machine.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050094	N/A	N/A

No output parameters.

1.3.3.80 *boson2ptNucNext()*

Transitions the 1-point or 2-point NUC state machine to the next state.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050095	N/A	N/A

No output parameters.

1.3.3.81 *boson2ptNucAbort()*

Aborts the 1-point or 2-point calibration process by exiting the state machine.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050096	N/A	N/A

No output parameters.

1.3.3.82 *bosonSetExtSyncMode()*

Sets the External Sync mode (master, slave, disabled)

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050098	N/A	N/A
mode	FLR_BOSON_EXT_SYNC_MODE_E	0:4	

No output parameters.

1.3.3.83 *bosonGetExtSyncMode()*

Gets the External Sync mode (master, slave, disabled)

FLIR NEUTRINO LC SOFTWARE IDD

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00050099	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
mode	FLR_BOSON_EXT_SYNC_MODE_E	0:4	

1.3.3.84 *bosonGetLastCommand()*

Get sequence number and command ID of last command the camera received.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0005009A	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
sequenceNumber	UINT_32	0:4	
cmdID	UINT_32	4:8	

1.3.3.85 *bosonGetSensorHostCalVersion()*

The sensor calibration version code.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000500A0	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
version	UINT_32	0:4	

1.3.3.86 *bosonSetDesiredStartupTableNumber()*

Sets the Start-up NUC table number, the NUC table loaded by default at start-up

Input/Send parameters:

102-2020-42, Neutrino LC Software Interface Description Document, Rev400

Information on this page is subject to change without notice

This document does not contain any export-controlled information.

FLIR NEUTRINO LC SOFTWARE IDD

Name	Data Type	Bytes	Notes
FunctionID	0x000500A1	N/A	N/A
table	INT_32	0:4	

No output parameters.

1.3.3.87 *bosonGetDesiredStartupTableNumber()*

Gets the Start-up NUC table number, the NUC table loaded by default at start-up

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000500A2	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
table	INT_32	0:4	

1.3.3.88 *bosonSetNvFFCMeanValueLens0()*

The mean FFC value for NvFFC lens 0.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000500A3	N/A	N/A
meanValue	FLOAT	0:4	

No output parameters.

1.3.3.89 *bosonGetNvFFCMeanValueLens0()*

The mean FFC value for NvFFC lens 0.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000500A4	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
meanValue	FLOAT	0:4	

FLIR NEUTRINO LC SOFTWARE IDD

1.3.3.90 *bosonSetNvFFCMeanValueLens1()*

The mean FFC value for NvFFC lens 1.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000500A5	N/A	N/A
meanValue	FLOAT	0:4	

No output parameters.

1.3.3.91 *bosonGetNvFFCMeanValueLens1()*

The mean FFC value for NvFFC lens 1.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000500A6	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
meanValue	FLOAT	0:4	

1.3.3.92 *bosonSetInvertImage()*

Sets the invert (horizontal flip) image orientation state

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000500A7	N/A	N/A
invertImage	FLR_ENABLE_E	0:4	

No output parameters.

1.3.3.93 *bosonGetInvertImage()*

Gets the invert (horizontal flip) image orientation state

Input/Send parameters:

FLIR NEUTRINO LC SOFTWARE IDD

Name	Data Type	Bytes	Notes
FunctionID	0x000500A8	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
invertImage	FLR_ENABLE_E	0:4	

1.3.3.94 *bosonSetRevertImage()*

Sets the revert (vertical flip) image orientation state

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000500A9	N/A	N/A
revertImage	FLR_ENABLE_E	0:4	

No output parameters.

1.3.3.95 *bosonGetRevertImage()*

Gets the revert (vertical flip) image orientation state

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000500AA	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
revertImage	FLR_ENABLE_E	0:4	

1.3.3.96 *bosonGetTimeStamp()*

Get the value for a specified timestamp enumeration.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000500AB	N/A	N/A

FLIR NEUTRINO LC SOFTWARE IDD

timeStamp Type	FLR_BOSON_TIMESTAMPTYPE_E	0:4		
-----------------------	---------------------------	-----	--	--

Output/Receive parameters:

Name	Data Type	Bytes	Notes	
timeStamp	FLOAT	0:4		

1.3.3.97 *bosonGetISPFrameCount()*

Get the current ISP frame counter.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x000500AC	N/A	N/A	

Output/Receive parameters:

Name	Data Type	Bytes	Notes	
ispFrameCount	UINT_32	0:4		

1.3.3.98 *bosonWriteUserBadPixelsToAllTables()*

Writes the user bad pixels in DRAM to all NUC tables in Flash.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x000500AD	N/A	N/A	

No output parameters.

1.3.3.99 *bosonWriteFactoryBadPixelsToAllTables()*

Writes the factory bad pixels in DRAM to all NUC tables in Flash.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x000500AE	N/A	N/A	

No output parameters.

FLIR NEUTRINO LC SOFTWARE IDD

1.3.3.100 *bosonGetTempDiodeStatus()*

The status of temperature diode.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000500B1	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
status	FLR_BOSON_TEMP_DIODE_STATUS_E	0:4	

1.3.3.101 *bosonClearFactoryBadPixelsInDDR()*

Empty the current bad pixel map in DDR.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000500B2	N/A	N/A

No output parameters.

1.3.3.102 *bosonGetFfcWaitOpenFrames()*

Number of frames to wait for the shutter to open during an Auto or Manual FFC.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000500B3	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	UINT_16	0:2	

1.3.3.103 *bosonSetFfcWaitOpenFrames()*

Number of frames to wait for the shutter to open during an Auto or Manual FFC.

Input/Send parameters:

Name	Data Type	Bytes	Notes
------	-----------	-------	-------

FLIR NEUTRINO LC SOFTWARE IDD

FunctionID	0x000500B4	N/A	N/A	
data	UINT_16	0:2		

No output parameters.

1.3.3.104 *bosonGetFfcWaitOpenFlagSettleFrames()*

Number of frames to wait for the shutter and sensor to settle after opening the shutter during an Auto or Manual FFC.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x000500B5	N/A	N/A	

Output/Receive parameters:

Name	Data Type	Bytes	Notes	
data	UINT_16	0:2		

1.3.3.105 *bosonSetFfcWaitOpenFlagSettleFrames()*

Number of frames to wait for the shutter and sensor to settle after opening the shutter during an Auto or Manual FFC.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x000500B6	N/A	N/A	
data	UINT_16	0:2		

No output parameters.

1.3.3.106 *bosonGetTauExtFfcCompatibilityMode()*

When enabled, this flag changes the table switching behavior when in External FFC mode so that table switches will happen automatically.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x000500BA	N/A	N/A	

Output/Receive parameters:

FLIR NEUTRINO LC SOFTWARE IDD

Name	Data Type	Bytes	Notes
data	FLR_ENABLE_E	0:4	

1.3.3.107 *bosonSetTauExtFfcCompatibilityMode()*

When enabled, this flag changes the table switching behavior when in External FFC mode so that table switches will happen automatically.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000500BB	N/A	N/A
data	FLR_ENABLE_E	0:4	

No output parameters.

1.3.3.108 *bosonGetImageValid()*

Indicates when the output image becomes valid

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000500C9	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	INT_16	0:2	

1.3.3.109 *bosonGetCurrentTableType()*

The (gain) type of the current table.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000500CA	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLR_BOSON_TABLETYPE_E	0:4	

1.4 Module: BPR

The BPR module exposes functions to control the Bad Pixel Replace algorithm

1.4.1 Enums

1.4.1.1 FLR_BPR_DISPLAY_MODE_E — <INT_32>

FLR_BPR_NORMAL_DISPLAY_MODE = 0
FLR_BPR_MIN_VALUE_ONLY_MODE = 1
FLR_BPR_MAX_VALUE_ONLY_MODE = 2
FLR_BPR_MIN_MAX_TOGGLE_MODE = 3
FLR_BPR_BPR_DISPLAY_MODE_END = 4

1.4.2 Structs

No struct types in module bpr.

1.4.3 Functions

1.4.3.1 *bprGetState()*

Reads the state (enabled/disabled) of the bad-pixel replace (BPR) algorithm.

Input/Send parameters:

Name	DataType	Bytes	Notes
FunctionID	0x00030001	N/A	N/A

Output/Receive parameters:

Name	DataType	Bytes	Notes
data	FLR_ENABLE_E	0:4	

1.4.3.2 *bprSetState()*

Enables / disables the bad-pixel replace (BPR) algorithm.

Input/Send parameters:

Name	DataType	Bytes	Notes
FunctionID	0x00030002	N/A	N/A
data	FLR_ENABLE_E	0:4	

No output parameters.

FLIR NEUTRINO LC SOFTWARE IDD

1.4.3.3 *bprGetStats()*

Get frame statistics about pixel replacement.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00030003	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
threeby	UINT_32	0:4	
fiveby	UINT_32	4:8	
rows	UINT_32	8:12	
budget	UINT_32	12:16	
used	UINT_32	16:20	

1.4.3.4 *bprGetDisplayMode()*

Get current display mode for BPR.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00030005	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLR_BPR_DISPLAY_MODE_E	0:4	

1.4.3.5 *bprSetDisplayMode()*

Set the display mode for BPR.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00030006	N/A	N/A
data	FLR_BPR_DISPLAY_MODE_E	0:4	

No output parameters.

FLIR NEUTRINO LC SOFTWARE IDD

1.4.3.6 *bprGetDisplayModeMinValue()*

Get the displayed minimum value.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00030007	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	UINT_16	0:2	

1.4.3.7 *bprSetDisplayModeMinValue()*

Set the displayed minimum value.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00030008	N/A	N/A
data	UINT_16	0:2	

No output parameters.

1.4.3.8 *bprGetDisplayModeMaxValue()*

Get the displayed maximum value.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00030009	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	UINT_16	0:2	

1.4.3.9 *bprSetDisplayModeMaxValue()*

Set the displayed maximum value.

Input/Send parameters:

FLIR NEUTRINO LC SOFTWARE IDD

Name	Data Type	Bytes	Notes
FunctionID	0x0003000A	N/A	N/A
data	UINT_16	0:2	

No output parameters.

1.4.3.10 *bprGetWorkBufIndex()*

Current work buffer index for getting stats.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0003000B	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	UINT_32	0:4	

1.4.3.11 *bprSetWorkBufIndex()*

Current work buffer index for getting stats.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0003000C	N/A	N/A
data	UINT_32	0:4	

No output parameters.

1.4.3.12 *bprGetWorkBufStats()*

Get single work-buffer statistics about pixel replacement.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0003000D	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
threeby	UINT_32	0:4	

FLIR NEUTRINO LC SOFTWARE IDD

fiveby	UINT_32	4:8		
rows	UINT_32	8:12		
budget	UINT_32	12:16		
used	UINT_32	16:20		

1.5 Module: CAPTURE

This set of controls is used to capture one or more video frames to Boson's internal memory. These captured frames are stored in DRAM and will be erased when the camera reboots or loses power.

1.5.1 Enums

1.5.1.1 *FLR_CAPTURE_SRC_E* — <INT_32>

FLR_CAPTURE_SRC_NUC = 1
FLR_CAPTURE_SRC_RESERVED = 2
FLR_CAPTURE_SRC_TNF = 3
FLR_CAPTURE_SRC_BLEND = 4
FLR_CAPTURE_SRC_VIS = 5
FLR_CAPTURE_SRC_MSX = 6
FLR_CAPTURE_SRC_RAW = 7
FLR_CAPTURE_SRC_TLINEAR = 8
FLR_CAPTURE_SRC_END = 9

1.5.1.2 *FLR_CAPTURE_FILE_TYPE_E* — <INT_32>

FLR_CAPTURE_NONE = 0
FLR_CAPTURE_JPEG = 1
FLR_CAPTURE_PNG = 2

1.5.2 Structs

1.5.2.1 *FLR_CAPTURE_SETTINGS_T*

Field Name	Data Type	Bytes
dataSrc	FLR_CAPTURE_SRC_E	4
numFrames	UINT_32	4
bufferIndex	UINT_16	2

FLIR NEUTRINO LC SOFTWARE IDD

1.5.2.2 FLR_CAPTURE_FILE_SETTINGS_T

Field Name	DataType	Bytes
captureFileType	FLR_CAPTURE_FILE_TYPE_E	4
filePath	UCHAR*128	128

1.5.3 Functions

1.5.3.1 *captureSingleFrame()*

Deprecated/Legacy

Input/Send parameters:

Name	DataType	Bytes	Notes
FunctionID	0x00070001	N/A	N/A

No output parameters.

1.5.3.2 *captureFrames()*

Capture up to 16 frames from specified source to buffer(s). You can select which slot the capture begins in, though you cannot capture more frames than slots. Ex: Capturing 16 frames can only be accomplished if slot 0 is selected as the start point.

Input/Send parameters:

Name	DataType	Bytes	Notes
FunctionID	0x00070002	N/A	N/A
data	FLR_CAPTURE_SETTINGS_T	0:10	

No output parameters.

1.5.3.3 *captureSingleFrameWithSrc()*

Capture a single frame from the specified source. Always uses capture slot 0.

Input/Send parameters:

Name	DataType	Bytes	Notes
FunctionID	0x00070003	N/A	N/A
data	FLR_CAPTURE_SRC_E	0:4	

No output parameters.

FLIR NEUTRINO LC SOFTWARE IDD

1.5.3.4 *captureSingleFrameToFile()*

Capture a single frame to file. Not available in Boson

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00070004	N/A	N/A

No output parameters.

1.6 Module: COLORLUT

This module is used to control which (if any) false color mode is applied to the 8-bit video output.

1.6.1 Enums

1.6.1.1 *FLR_COLORLUT_ID_E* — <INT_32>

FLR_COLORLUT_WHITEHOT = 0
FLR_COLORLUT_DEFAULT = 0
FLR_COLORLUT_0 = 0
FLR_COLORLUT_1 = 1
FLR_COLORLUT_BLACKHOT = 1
FLR_COLORLUT_RAINBOW = 2
FLR_COLORLUT_2 = 2
FLR_COLORLUT_RAINBOW_HC = 3
FLR_COLORLUT_3 = 3
FLR_COLORLUT_4 = 4
FLR_COLORLUT_IRONBOW = 4
FLR_COLORLUT_LAVA = 5
FLR_COLORLUT_5 = 5
FLR_COLORLUT_6 = 6
FLR_COLORLUT_ARCTIC = 6
FLR_COLORLUT_7 = 7
FLR_COLORLUT_GLOBOW = 7
FLR_COLORLUT_GRADEDFIRE = 8
FLR_COLORLUT_8 = 8
FLR_COLORLUT_HOTTEST = 9
FLR_COLORLUT_9 = 9
FLR_COLORLUT_ID_END = 10

FLIR NEUTRINO LC SOFTWARE IDD

1.6.2 Structs

No struct types in module colorLut.

1.6.3 Functions

1.6.3.1 *colorLutSetControl()*

Set the current enable state of the colorize module.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000B0001	N/A	N/A
data	FLR_ENABLE_E	0:4	

No output parameters.

1.6.3.2 *colorLutGetControl()*

Get the current enable state of the colorize module.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000B0002	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLR_ENABLE_E	0:4	

1.6.3.3 *colorLutSetId()*

Set the current color palette, by ID.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000B0003	N/A	N/A
data	FLR_COLORLUT_ID_E	0:4	

No output parameters.

1.6.3.4 *colorLutGetId()*

Get the current color palette ID.

FLIR NEUTRINO LC SOFTWARE IDD

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000B0004	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLR_COLORLUT_ID_E	0:4	

1.6.3.5 *colorLutSetOutlineColor()*

Set the R,G,B value used for display of outline edges.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000B0005	N/A	N/A
red	UCHAR	0:1	
green	UCHAR	1:2	
blue	UCHAR	2:3	

No output parameters.

1.6.3.6 *colorLutGetOutlineColor()*

Get the R,G,B value used for display of outline edges.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000B0006	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
red	UCHAR	0:1	
green	UCHAR	1:2	
blue	UCHAR	2:3	

1.7 Module: COOLER

This module exposes the camera parameters that are used to control the beginning of imaging and the display of the cooler metrics log.

1.7.1 Enums

1.7.1.1 FLR_COOLER_STARTUPMODE_E — <INT_32>

FLR_COOLER_NORMAL = 0
FLR_COOLER_FAST = 1

1.7.1.2 FLR_COOLER_LOG_CSV_FORMAT_E — <INT_32>

FLR_COOLER_RAW = 0
FLR_COOLER_PRETTY_PRINT = 1

1.7.2 Structs

No struct types in module cooler.

1.7.3 Functions

1.7.3.1 coolerGetLogInfo()

Get the number of elements and the size in bytes of the saved log file

Input/Send parameters:

Name	DataType	Bytes	Notes
FunctionID	0x002E0004	N/A	N/A

Output/Receive parameters:

Name	DataType	Bytes	Notes
numberOfElements	UINT_32	0:4	
logSizeInBytes	UINT_32	4:8	

1.7.3.2 coolerShowLog()

Show a comma separated variable display of the current log file. The log displayed depends on the formatting selection chosen using the prepare log function.

Input/Send parameters:

FLIR NEUTRINO LC SOFTWARE IDD

Name	Data Type	Bytes	Notes
FunctionID	0x002E0005	N/A	N/A
logByteOffset	UINT_32	0:4	
sizeInBytes	UINT_16	4:6	

Output/Receive parameters:

Name	Data Type	Bytes	Notes
outLogByteOffset	UINT_32	0:4	
data	BYTEARRAY	4:516	

1.7.3.3 coolerGetUptime()

Get the current uptime in seconds

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x002E0006	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	UINT_32	0:4	

1.7.3.4 coolerGetMaxTemp()

Return the current value of the maximum temperature reached by the FPA during this power cycle.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x002E0007	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	UINT_16	0:2	

FLIR NEUTRINO LC SOFTWARE IDD

1.7.3.5 coolerGetCooldownTime()

Get the cooldown time. This will show -1 if the camera has not cooled down.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x002E0008	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	UINT_16	0:2	

1.7.3.6 coolerGetTotalNumberOfCycles()

Returns the number of power cycles or resets the camera has gone through.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x002E0009	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
totalNumberOfCycles	UINT_16	0:2	

1.7.3.7 coolerGetStartingTemp()

Returns the first temperature measured when the current power cycle was initiated.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x002E000A	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
startingTemp	UINT_16	0:2	

FLIR NEUTRINO LC SOFTWARE IDD

1.7.3.8 coolerSetStartupMode()

Selects either NORMAL or FAST mode.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x002E000B	N/A	N/A
startupMode	FLR_COOLER_STARTUPMODE_E	0:4	

No output parameters.

1.7.3.9 coolerGetStartupMode()

Returns the current value of the startup mode.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x002E000C	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
startupMode	FLR_COOLER_STARTUPMODE_E	0:4	

1.7.3.10 coolerGetNormalStartupTemp()

Return the value of the NORMAL startup mode temperature.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x002E000E	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
normalStartupTemp	UINT_16	0:2	

1.7.3.11 coolerGetFastStartupTemp()

Return the value of the FAST startup mode temperature.

FLIR NEUTRINO LC SOFTWARE IDD

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x002E0010	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
fastStartupTemp	UINT_16	0:2	

1.7.3.12 coolerGetLogEntry()

From the selected log entry return the address of the log offset, the uptime, cooldown time, and maximum temperature.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x002E0011	N/A	N/A
logOffset	UINT_16	0:2	

Output/Receive parameters:

Name	Data Type	Bytes	Notes
returnedLogOffset	INT_32	0:4	
uptime	INT_32	4:8	
cooldownTime	INT_32	8:12	
maxTemp	INT_32	12:16	

1.7.3.13 coolerPrepareLog()

Use this function to prepare the log file for reading from the camera. There are two formats: RAW - just an unformatted csv list of entries, and Pretty Print - a spaced formatted csv list of entries. The returned values are the number of log elements that make up the log, and the size of the log in bytes.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x002E0012	N/A	N/A

FLIR NEUTRINO LC SOFTWARE IDD

formatLog	FLR_COOLER_LOG_CSV_FORMAT_E	0:4		
------------------	-----------------------------	-----	--	--

Output/Receive parameters:

Name	Data Type	Bytes	Notes	
numberOfElements	UINT_32	0:4		
logSizeInBytes	UINT_32	4:8		

1.8 Module: CPE

These APIs are used to access the registers of the CPE

1.8.1 Enums

No enumerations in module cpe.

1.8.2 Structs

No struct types in module cpe.

1.8.3 Functions

1.8.3.1 *cpeIntegrationTimeTicksGet()*

Returns the integration time in clock ticks

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x00200004	N/A	N/A	

Output/Receive parameters:

Name	Data Type	Bytes	Notes	
data	INT_32	0:4		

1.8.3.2 *cpeIntegrationTimeMsGet()*

Returns the integration time in microseconds

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x00200005	N/A	N/A	

102-2020-42, Neutrino LC Software Interface Description Document, Rev400

Information on this page is subject to change without notice

This document does not contain any export-controlled information.

73



The World's Sixth Sense™

FLIR NEUTRINO LC SOFTWARE IDD

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	INT_32	0:4	

1.8.3.3 *cpeIntegrationTimeTicksSet()*

Sets the integration time in clock ticks

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00200006	N/A	N/A
intTime	INT_32	0:4	

No output parameters.

1.8.3.4 *cpeIntegrationTimeMsSet()*

Sets the integration time in microseconds

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00200007	N/A	N/A
data	INT_32	0:4	

No output parameters.

1.8.3.5 *cpeGetFirmwareVersion()*

No description available

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0020000A	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
version	UINT_32	0:4	

FLIR NEUTRINO LC SOFTWARE IDD

1.8.3.6 *cpeSetIntegrationMode()*

Sets the integration mode to ITR, IWR, or URESTRICTED

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0020001F	N/A	N/A
data	UINT_32	0:4	

No output parameters.

1.8.3.7 *cpeGetIntegrationMode()*

Gets the current value of the integration mode

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00200020	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	UINT_32	0:4	

1.8.3.8 *cpeSetRunSelect()*

This changes the mode of the External Sync from Disabled, to Slave, or Master.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00200033	N/A	N/A
data	UINT_32	0:4	

No output parameters.

1.8.3.9 *cpeGetRunSelect()*

Returns the current value of the External Sync setting

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00200034	N/A	N/A

FLIR NEUTRINO LC SOFTWARE IDD

Output/Receive parameters:

Name	DataType	Bytes	Notes
data	UINT_32	0:4	

1.8.3.10 *cpeGetFpaClkFreq()*

Return the clock frequency of the FPA.

Input/Send parameters:

Name	DataType	Bytes	Notes
FunctionID	0x00200038	N/A	N/A

Output/Receive parameters:

Name	DataType	Bytes	Notes
data	UINT_32	0:4	

1.8.3.11 *cpeUserIntegrationTimeMsSet()*

Sets the integration time given in microseconds and then does an automatic cpe latch to cause the new integration time to take effect in the CPE.

Input/Send parameters:

Name	DataType	Bytes	Notes
FunctionID	0x00200039	N/A	N/A
data	INT_32	0:4	

No output parameters.

1.8.3.12 *cpeUserGetFramerate()*

Get the framerate of the camera in Hz * 100 frames per second (e.g. 6000).

Input/Send parameters:

Name	DataType	Bytes	Notes
FunctionID	0x00200046	N/A	N/A

Output/Receive parameters:

Name	DataType	Bytes	Notes
------	----------	-------	-------

FLIR NEUTRINO LC SOFTWARE IDD

data	UINT_32	0:4		
-------------	---------	-----	--	--

1.8.3.13 *cpeUserSetFramerate()*

Set the framerate of the camera in Hz * 100 frames per second (e.g. 6000).

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x00200047	N/A	N/A	
data	UINT_32	0:4		

No output parameters.

1.8.3.14 *cpeUserIntegrationTimeMsGet()*

Get the integration time. Returned value is in microseconds.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x0020004A	N/A	N/A	

Output/Receive parameters:

Name	Data Type	Bytes	Notes	
data	UINT_32	0:4		

1.9 Module: DBMF

Decision Based Median correction filter.

1.9.1 Enums

No enumerations in module dbmf.

1.9.2 Structs

No struct types in module dbmf.

1.9.3 Functions

1.9.3.1 *dbmfSetEnableState()*

DBMF correction state.

FLIR NEUTRINO LC SOFTWARE IDD

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00410000	N/A	N/A
data	FLR_ENABLE_E	0:4	

No output parameters.

1.9.3.2 *dbmfGetEnableState()*

DBMF correction state.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00410001	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLR_ENABLE_E	0:4	

1.9.3.3 *dbmfSetThreshInit()*

DBMF threshold initial.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00410002	N/A	N/A
data	UINT_16	0:2	

No output parameters.

1.9.3.4 *dbmfGetThreshInit()*

DBMF threshold initial.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00410003	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
------	-----------	-------	-------

FLIR NEUTRINO LC SOFTWARE IDD

data	UINT_16	0:2		
-------------	---------	-----	--	--

1.9.3.5 *dbmfSetIdealReplacementCt()*

DBMF ideal replacement count.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x00410004	N/A	N/A	
data	UINT_32	0:4		

No output parameters.

1.9.3.6 *dbmfGetIdealReplacementCt()*

DBMF ideal replacement count.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x00410005	N/A	N/A	

Output/Receive parameters:

Name	Data Type	Bytes	Notes	
data	UINT_32	0:4		

1.9.3.7 *dbmfSetDF()*

DBMF Damping factor.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x00410006	N/A	N/A	
data	FLOAT	0:4		

No output parameters.

1.9.3.8 *dbmfGetDF()*

DBMF Damping factor.

Input/Send parameters:

102-2020-42, Neutrino LC Software Interface Description Document, Rev400

Information on this page is subject to change without notice

This document does not contain any export-controlled information.

FLIR NEUTRINO LC SOFTWARE IDD

Name	Data Type	Bytes	Notes
FunctionID	0x00410007	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLOAT	0:4	

1.9.3.9 *dbmfSetThreshClampMin()*

DBMF minimum clamp.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00410008	N/A	N/A
data	UINT_16	0:2	

No output parameters.

1.9.3.10 *dbmfGetThreshClampMin()*

DBMF minimum clamp.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00410009	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	UINT_16	0:2	

1.9.3.11 *dbmfGetThresh()*

DBMF threshold.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0041000A	N/A	N/A

Output/Receive parameters:

FLIR NEUTRINO LC SOFTWARE IDD

Name	Data Type	Bytes	Notes
data	UINT_16	0:2	

1.9.3.12 *dbmfGetTotalReplacementCt()*

DBMF total replacement count.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0041000B	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	UINT_32	0:4	

1.9.3.13 *dbmfSetThreshClampMax()*

DBMF maximum clamp.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0041000C	N/A	N/A
data	UINT_16	0:2	

No output parameters.

1.9.3.14 *dbmfGetThreshClampMax()*

DBMF maximum clamp.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0041000D	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	UINT_16	0:2	

1.10 Module: DUMMY

Please provide description.

1.10.1 Enums

No enumerations in module dummy.

1.10.2 Structs

No struct types in module dummy.

1.10.3 Functions

1.10.3.1 *dummyBadCommand()*

Deliberately invalid Command Id for negative testing.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0xDEADBEEF	N/A	N/A

No output parameters.

1.11 Module: DVO

The Digital Video Out module provides functions to control and interrogate the digital video display if connected to the digital output interface.

1.11.1 Enums

1.11.1.1 *FLR_DVO_OUTPUT_FORMAT_E* — <INT_32>

FLR_DVO_RGB = 0
FLR_DVO_YCBCR = 1
FLR_DVO_DEFAULT_FORMAT = 2
FLR_DVO_IR16 = 3
FLR_DVO_OUTPUT_FORMAT_END = 4

1.11.1.2 *FLR_DVO_OUTPUT_RGB_FORMAT_E* — <INT_32>

FLR_DVO_RGB888 = 0
FLR_DVO_MRGB888 = 1
FLR_DVO_OUTPUT_RGB_FORMAT_END = 2

1.11.1.3 *FLR_DVO_OUTPUT_YCBCR_FORMAT_E* — <INT_32>

FLR_DVO_YCBCR422_8B = 0

102-2020-42, Neutrino LC Software Interface Description Document, Rev400

Information on this page is subject to change without notice

This document does not contain any export-controlled information.

FLR_DVO_MYCBCR422_8B = 1
FLR_DVO_OUTPUT_YCBCR_FORMAT_END = 2

1.11.1.4 FLR_DVO_OUTPUT_IR16_FORMAT_E — <INT_32>

FLR_DVO_IR16_16B = 0
FLR_DVO_MIR16_8B = 1
FLR_DVO_OUTPUT_IR16_FORMAT_END = 2

1.11.1.5 FLR_DVO_OUTPUT_CBCR_ORDER_E — <INT_32>

FLR_DVO_CRCB = 0
FLR_DVO_CBCR = 1
FLR_DVO_OUTPUT_CBCR_ORDER_END = 2

1.11.1.6 FLR_DVO_OUTPUT_Y_ORDER_E — <INT_32>

FLR_DVO_YFIRST = 0
FLR_DVO_YLAST = 1
FLR_DVO_OUTPUT_Y_ORDER_END = 2

1.11.1.7 FLR_DVO_OUTPUT_RGB_ORDER_E — <INT_32>

FLR_DVO_ORDER_RGB = 0
FLR_DVO_ORDER_BGR = 1
FLR_DVO_OUTPUT_RGB_ORDER_END = 2

1.11.1.8 FLR_DVO_TYPE_E — <INT_32>

FLR_DVO_TYPE_MONO16 = 0
FLR_DVO_TYPE_MONO8 = 1
FLR_DVO_TYPE_COLOR = 2
FLR_DVO_TYPE_ANALOG = 3
FLR_DVO_TYPE_RAW = 4
FLR_DVO_TYPE_MONO14 = 5
FLR_DVO_TYPE_TLINEAR = 6
FLR_DVO_TYPE_MONO12 = 7
FLR_DVO_TYPE_END = 8

1.11.1.9 FLR_DVO_DISPLAY_MODE_E — <INT_32>

FLR_DVO_CONTINUOUS = 0
FLR_DVO_ONE_SHOT = 1
FLR_DVO_DISPLAY_MODE_END = 2

1.11.1.10 FLR_DVO_VIDEO_STANDARD_E — <INT_32>

FLR_DVO_NTSC = 0

FLIR NEUTRINO LC SOFTWARE IDD

FLR_DVO_PAL = 1
FLR_DVO_VIDEO_STANDARD_END = 2

1.11.1.11 FLR_DVO_LCD_CONFIG_ID_E — <INT_32>

FLR_DVO_DEFAULT = 0
FLR_DVO_CUSTOM1 = 1
FLR_DVO_CUSTOM2 = 2
FLR_DVO_CONFIG1 = 3
FLR_DVO_CONFIG2 = 4

1.11.1.12 FLR_DVO_LCD_CLOCK_RATE_E — <INT_32>

FLR_DVO_LCD_CLOCK_DEFAULT = 0
FLR_DVO_LCD_CLOCK_27MHZ = 1
FLR_DVO_LCD_CLOCK_13_5MHZ = 2
FLR_DVO_LCD_CLOCK_48MHZ = 3
FLR_DVO_LCD_CLOCK_60MHZ = 4
FLR_DVO_LCD_CLOCK_END = 5

1.11.1.13 FLR_DVO_MIPI_STATE_E — <INT_32>

FLR_DVO_MIPI_STATE_OFF = 0
FLR_DVO_MIPI_STATE_PAUSED = 1
FLR_DVO_MIPI_STATE_ACTIVE = 2
FLR_DVO_MIPI_STATE_END = 3

1.11.1.14 FLR_DVO_MIPI_CLOCK_LANE_MODE_E — <INT_32>

FLR_DVO_MIPI_CLOCK_LANE_MODE_NON_CONTINUOUS = 0
FLR_DVO_MIPI_CLOCK_LANE_MODE_CONTINUOUS = 1
FLR_DVO_MIPI_CLOCK_LANE_MODE_END = 2

1.11.2 Structs

1.11.2.1 FLR_DVO_YCBCR_SETTINGS_T

Field Name	Data Type	Bytes
ycbcrFormat	FLR_DVO_OUTPUT_YCBCR_FORMAT_E	4
cbcrOrder	FLR_DVO_OUTPUT_CBCR_ORDER_E	4
yOrder	FLR_DVO_OUTPUT_Y_ORDER_E	4

1.11.2.2 FLR_DVO_RGB_SETTINGS_T

Field Name	Data Type	Bytes
------------	-----------	-------

102-2020-42, Neutrino LC Software Interface Description Document, Rev400

Information on this page is subject to change without notice

This document does not contain any export-controlled information.



FLIR NEUTRINO LC SOFTWARE IDD

rgbFormat	FLR_DVO_OUTPUT_RGB_FORMAT_E	4
rgbOrder	FLR_DVO_OUTPUT_RGB_ORDER_E	4

1.11.2.3 FLR_DVO_LCD_CONFIG_T

Field Name	Data Type	Bytes
width	UINT_32	4
hPulseWidth	UINT_32	4
hBackP	UINT_32	4
hFrontP	UINT_32	4
height	UINT_32	4
vPulseWidth	UINT_32	4
vBackP	UINT_32	4
vFrontP	UINT_32	4
outputFormat	UINT_32	4
control	UINT_32	4
rotation	UINT_32	4
pixelClockkHz	UINT_32	4

1.11.3 Functions

1.11.3.1 dvoSetAnalogVideoState()

Sets the state of analog video

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00060004	N/A	N/A
analogVideoState	FLR_ENABLE_E	0:4	

No output parameters.

1.11.3.2 dvoGetAnalogVideoState()

Gets the state of analog video

Input/Send parameters:

Name	Data Type	Bytes	Notes
------	-----------	-------	-------

FLIR NEUTRINO LC SOFTWARE IDD

FunctionID	0x00060005	N/A	N/A	
-------------------	------------	-----	-----	--

Output/Receive parameters:

Name	Data Type	Bytes	Notes	
analogVideoState	FLR_ENABLE_E	0:4		

1.11.3.3 *dvoSetOutputFormat()*

Sets the output format for the lcd output. In order to apply these settings, dvoApplyCustomSettings function needs to be called.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x00060006	N/A	N/A	
format	FLR_DVO_OUTPUT_FORMAT_E	0:4		

No output parameters.

1.11.3.4 *dvoGetOutputFormat()*

Gets the output format for the lcd output

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x00060007	N/A	N/A	

Output/Receive parameters:

Name	Data Type	Bytes	Notes	
format	FLR_DVO_OUTPUT_FORMAT_E	0:4		

1.11.3.5 *dvoSetOutputYCbCrSettings()*

Sets the YCBCR mode, Y order and CB/CR order. In order to apply these settings, dvoApplyCustomSettings function needs to be called.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x00060008	N/A	N/A	

FLIR NEUTRINO LC SOFTWARE IDD

settings	FLR_DVO_YCBCR_SETTINGS_T	0:12		
-----------------	--------------------------	------	--	--

No output parameters.

1.11.3.6 *dvoGetOutputYCbCrSettings()*

Gets the YCBCR settings

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x00060009	N/A	N/A	

Output/Receive parameters:

Name	Data Type	Bytes	Notes	
settings	FLR_DVO_YCBCR_SETTINGS_T	0:12		

1.11.3.7 *dvoSetOutputRGBSettings()*

Sets the RGB mode and RGB order. In order to apply these settings, dvoApplyCustomSettings function needs to be called.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x0006000A	N/A	N/A	
settings	FLR_DVO_RGB_SETTINGS_T	0:8		

No output parameters.

1.11.3.8 *dvoGetOutputRGBSettings()*

Gets the RGB settings

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x0006000B	N/A	N/A	

Output/Receive parameters:

Name	Data Type	Bytes	Notes	
settings	FLR_DVO_RGB_SETTINGS_T	0:8		

FLIR NEUTRINO LC SOFTWARE IDD

1.11.3.9 *dvoApplyCustomSettings()*

Applies the settings set by `dvoSetOutputFormat`, `dvoSetOutputYCbCrSettings` and `dvoSetOutputRGBSettings`. If `FLR_DVO_DEFAULT_FORMAT` is chosen, this function applies the default lcd settings for the selected source. If `FLR_DVO_YCBCR` is set then the settings set by `dvoSetOutputYCbCrSettings` are applied and if `FLR_DVO_RGB` is selected, the settings set by `dvoSetOutputRGBSettings` are applied to the lcd output format.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0006000C	N/A	N/A

No output parameters.

1.11.3.10 *dvoSetDisplayMode()*

Sets the display mode to continuous or one shot

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0006000D	N/A	N/A
displayMode	FLR_DVO_DISPLAY_MODE_E	0:4	

No output parameters.

1.11.3.11 *dvoGetDisplayMode()*

Gets the display mode

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0006000E	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
displayMode	FLR_DVO_DISPLAY_MODE_E	0:4	

1.11.3.12 *dvoSetType()*

Sets the tap at which the DVO source points to

FLIR NEUTRINO LC SOFTWARE IDD

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0006000F	N/A	N/A
tap	FLR_DVO_TYPE_E	0:4	

No output parameters.

1.11.3.13 *dvoGetType()*

Gets the tap at which the DVO source is pointing to

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00060010	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
tap	FLR_DVO_TYPE_E	0:4	

1.11.3.14 *dvoSetVideoStandard()*

Sets the analog video output to be either NTSC or PAL.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00060011	N/A	N/A
videoStandard	FLR_DVO_VIDEO_STANDARD_E	0:4	

No output parameters.

1.11.3.15 *dvoGetVideoStandard()*

Gets if the analog video output is NTSC or PAL.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00060012	N/A	N/A

Output/Receive parameters:

FLIR NEUTRINO LC SOFTWARE IDD

Name	Data Type	Bytes	Notes
videoStandard	FLR_DVO_VIDEO_STANDARD_E	0:4	

1.11.3.16 *dvoSetCheckVideoDacPresent()*

Sets the flag to either check or ignore the presence of the DAC. If we set this flag is enabled and if the video DAC is not present, we do not output the BT.656 data.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00060013	N/A	N/A
checkVideoDacPresent	FLR_ENABLE_E	0:4	

No output parameters.

1.11.3.17 *dvoGetCheckVideoDacPresent()*

Gets the state of the checkVideoDacPresent flag.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00060014	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
checkVideoDacPresent	FLR_ENABLE_E	0:4	

1.11.3.18 *dvoSetCustomLcdConfig()*

Set a custom LCD configuration.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00060015	N/A	N/A
id	FLR_DVO_LCD_CONFIG_ID_E	0:4	
config	FLR_DVO_LCD_CONFIG_T	4:52	

FLIR NEUTRINO LC SOFTWARE IDD

No output parameters.

1.11.3.19 *dvoGetCustomLcdConfig()*

Get a custom LCD configuration.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00060016	N/A	N/A
id	FLR_DVO_LCD_CONFIG_ID_E	0:4	

Output/Receive parameters:

Name	Data Type	Bytes	Notes
config	FLR_DVO_LCD_CONFIG_T	0:48	

1.11.3.20 *dvoSetLCDConfig()*

Set the current LCD configuration ID.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00060017	N/A	N/A
id	FLR_DVO_LCD_CONFIG_ID_E	0:4	

No output parameters.

1.11.3.21 *dvoGetLCDConfig()*

Get the current LCD configuration ID.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00060018	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
id	FLR_DVO_LCD_CONFIG_ID_E	0:4	

FLIR NEUTRINO LC SOFTWARE IDD

1.11.3.22 dvoGetClockInfo()

Get the current output clock configuration.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00060019	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
horizontalSyncWidth	UINT_32	0:4	
verticalSyncWidth	UINT_32	4:8	
clocksPerRowPeriod	UINT_32	8:12	
horizontalFrontPorch	UINT_32	12:16	
horizontalBackPorch	UINT_32	16:20	
frontTelemetryPixels	UINT_32	20:24	
rearTelemetryPixels	UINT_32	24:28	
videoColumns	UINT_32	28:32	
validColumns	UINT_32	32:36	
telemetryRows	UINT_32	36:40	
videoRows	UINT_32	40:44	
validRows	UINT_32	44:48	
verticalFrontPorch	UINT_32	48:52	
verticalBackPorch	UINT_32	52:56	
rowPeriodsPerFrame	UINT_32	56:60	
clocksPerFrame	UINT_32	60:64	

102-2020-42, Neutrino LC Software Interface Description Document, Rev400

Information on this page is subject to change without notice

This document does not contain any export-controlled information.



FLIR NEUTRINO LC SOFTWARE IDD

clockRateIn MHz	FLOAT	64:68		
frameRateIn nHz	FLOAT	68:72		
validOnRisingEdge	UINT_32	72:76		
dataWidthIn nBits	UINT_32	76:80		

1.11.3.23 *dvoSetAllCustomLcdConfigs()*

Set all custom LCD configurations.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x0006001A	N/A	N/A	
config0	FLR_DVO_LCD_CONFIG_T	0:48		
config1	FLR_DVO_LCD_CONFIG_T	48:96		

No output parameters.

1.11.3.24 *dvoGetAllCustomLcdConfigs()*

Get all custom LCD configurations.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x0006001B	N/A	N/A	

Output/Receive parameters:

Name	Data Type	Bytes	Notes	
config0	FLR_DVO_LCD_CONFIG_T	0:48		
config1	FLR_DVO_LCD_CONFIG_T	48:96		

1.11.3.25 *dvoSetOutputIr16Format()*

The IR16 format settings, 16B or 8B Multiplex. In order to apply these settings, dvoApplyCustomSettings function needs to be called.

Input/Send parameters:

102-2020-42, Neutrino LC Software Interface Description Document, Rev400

Information on this page is subject to change without notice

This document does not contain any export-controlled information.

FLIR NEUTRINO LC SOFTWARE IDD

Name	Data Type	Bytes	Notes
FunctionID	0x0006001C	N/A	N/A
format	FLR_DVO_OUTPUT_IR16_FORMAT_E	0:4	

No output parameters.

1.11.3.26 *dvoGetOutputIr16Format()*

The IR16 format settings, 16B or 8B Multiplex. In order to apply these settings, dvoApplyCustomSettings function needs to be called.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0006001D	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
format	FLR_DVO_OUTPUT_IR16_FORMAT_E	0:4	

1.11.3.27 *dvoSetLcdClockRate()*

LCD aux clock rate.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0006001E	N/A	N/A
clockRate	FLR_DVO_LCD_CLOCK_RATE_E	0:4	

No output parameters.

1.11.3.28 *dvoGetLcdClockRate()*

LCD aux clock rate.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0006001F	N/A	N/A

Output/Receive parameters:

102-2020-42, Neutrino LC Software Interface Description Document, Rev400

Information on this page is subject to change without notice

This document does not contain any export-controlled information.

FLIR NEUTRINO LC SOFTWARE IDD

Name	Data Type	Bytes	Notes
clockRate	FLR_DVO_LCD_CLOCK_RATE_E	0:4	

1.11.3.29 *dvoSetLcdVideoFrameRate()*

LCD Video frame rate in Hz (only supported in continuous mode).

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00060020	N/A	N/A
framerate	UINT_32	0:4	

No output parameters.

1.11.3.30 *dvoGetLcdVideoFrameRate()*

LCD Video frame rate in Hz (only supported in continuous mode).

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00060021	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
framerate	UINT_32	0:4	

1.11.3.31 *dvoSetMipiStartState()*

Initial starting state for MIPI interface following boot.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00060022	N/A	N/A
state	FLR_DVO_MIPI_STATE_E	0:4	

No output parameters.

1.11.3.32 *dvoGetMipiStartState()*

Initial starting state for MIPI interface following boot.

FLIR NEUTRINO LC SOFTWARE IDD

Input/Send parameters:

Name	DataType	Bytes	Notes
FunctionID	0x00060023	N/A	N/A

Output/Receive parameters:

Name	DataType	Bytes	Notes
state	FLR_DVO_MIPI_STATE_E	0:4	

1.11.3.33 *dvoSetMipiState()*

MIPI interface state.

Input/Send parameters:

Name	DataType	Bytes	Notes
FunctionID	0x00060024	N/A	N/A
state	FLR_DVO_MIPI_STATE_E	0:4	

No output parameters.

1.11.3.34 *dvoGetMipiState()*

MIPI interface state.

Input/Send parameters:

Name	DataType	Bytes	Notes
FunctionID	0x00060025	N/A	N/A

Output/Receive parameters:

Name	DataType	Bytes	Notes
state	FLR_DVO_MIPI_STATE_E	0:4	

1.11.3.35 *dvoSetMipiClockLaneMode()*

MIPI clock lane behavior, non-continuous or continuous.

Input/Send parameters:

Name	DataType	Bytes	Notes
FunctionID	0x00060026	N/A	N/A
mode	FLR_DVO_MIPI_CLOCK_LANE_MOD	0:4	

FLIR NEUTRINO LC SOFTWARE IDD

	E_E			
--	-----	--	--	--

No output parameters.

1.11.3.36 *dvoGetMipiClockLaneMode()*

MIPI clock lane behavior, non-continuous or continuous.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00060027	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
mode	FLR_DVO_MIPI_CLOCK_LANE_MOD E_E	0:4	

1.12 Module: EXTLENSCONTROL

This module controls attached lens hardware using an I2C interface. An accessory board with an I2C to serial converter is required hardware for this feature.

1.12.1 Enums

1.12.1.1 *FLR_EXTLENSCONTROL_FOCUS_MODE_E* — <INT_32>

FLR_EXTLENSCONTROL_FOCUS_MODE_MANUAL = 0
FLR_EXTLENSCONTROL_FOCUS_MODE_AUTO = 1
FLR_EXTLENSCONTROL_FOCUS_MODE_END = 3

1.12.1.2 *FLR_EXTLENSCONTROL_ACTION_E* — <INT_32>

FLR_EXTLENSCONTROL_ACTION_READ = 0
FLR_EXTLENSCONTROL_ACTION_READ_AND_CLEAR = 1
FLR_EXTLENSCONTROL_ACTION_END = 2

1.12.1.3 *FLR_EXTLENSCONTROL_STATE_E* — <INT_32>

FLR_EXTLENSCONTROL_STATE_OFF = 0
FLR_EXTLENSCONTROL_STATE_ON = 1
FLR_EXTLENSCONTROL_STATE_ERROR = 2
FLR_EXTLENSCONTROL_STATE_END = 3

FLIR NEUTRINO LC SOFTWARE IDD

1.12.2 Structs

1.12.2.1 FLR_EXTLENSCONTROL_PROTOCOL_STATS_T

Field Name	Data Type	Bytes
sent	UINT_32	4
received	UINT_32	4
timeouts	UINT_32	4
retries	UINT_32	4
errors	UINT_32	4

1.12.3 Functions

1.12.3.1 *extLensControlSetEnable()*

Enable/Disable the external lens control interface.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x003D0000	N/A	N/A
enable	FLR_ENABLE_E	0:4	

No output parameters.

1.12.3.2 *extLensControlGetEnable()*

Enable/Disable the external lens control interface.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x003D0001	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
enable	FLR_ENABLE_E	0:4	

1.12.3.3 *extLensControlReadStats()*

Read or Read and Clear the number of lens protocol errors in the module since the last clear.

FLIR NEUTRINO LC SOFTWARE IDD

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x003D0003	N/A	N/A
readAndClear	FLR_EXTLENSCONTROL_ACTION_E	0:4	

Output/Receive parameters:

Name	Data Type	Bytes	Notes
lensProtocolStats	FLR_EXTLENSCONTROL_PROTOCOL_STATS_T	0:20	

1.12.3.4 extLensControlCommTest()

Select Commanded Built-in Test (BIT) to verify the lens functionality including power and generally acknowledging commands. The testModule defines controller module bit number in the range 0 - 31.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x003D0004	N/A	N/A
testModule	UINT_32	0:4	

Output/Receive parameters:

Name	Data Type	Bytes	Notes
testResult	UINT_32	0:4	

1.12.3.5 extLensControlCommTestResult()

Request result of Commanded Built-in Test (BIT). The testModule defines controller module bit number in the range 0 - 31.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x003D0005	N/A	N/A
testModule	UINT_32	0:4	

Output/Receive parameters:

Name	Data Type	Bytes	Notes
------	-----------	-------	-------

FLIR NEUTRINO LC SOFTWARE IDD

testResult	UINT_32	0:4		
------------	---------	-----	--	--

1.12.3.6 *extLensControlSetZoomRelative()*

Move to a relative zoom position by incrementing or decrementing the zoom motor by user provided step size in focal length. zoomDirection defines the step direction. 1 is positive direction -1 is negative direction.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x003D0006	N/A	N/A	
zoomDirection	INT_32	0:4		

No output parameters.

1.12.3.7 *extLensControlGetZoomRelative()*

Move to a relative zoom position by incrementing or decrementing the zoom motor by user provided step size in focal length. zoomDirection defines the step direction. 1 is positive direction -1 is negative direction.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x003D0007	N/A	N/A	

Output/Receive parameters:

Name	Data Type	Bytes	Notes	
zoomDirection	INT_32	0:4		

1.12.3.8 *extLensControlSetZoomStepSize()*

Define the zoom step size to be used by the Zoom Relative command. The zoomStepSize defines the step size in zoom encoder counts in the range -1000 - 1000 with 10 as default.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x003D0008	N/A	N/A	

FLIR NEUTRINO LC SOFTWARE IDD

zoomStepSize	INT_32	0:4		
---------------------	--------	-----	--	--

No output parameters.

1.12.3.9 *extLensControlGetZoomStepSize()*

Define the zoom step size to be used by the Zoom Relative command. The zoomStepSize defines the step size in zoom encoder counts in the range -1000 - 1000 with 10 as default.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x003D0009	N/A	N/A	

Output/Receive parameters:

Name	Data Type	Bytes	Notes	
zoomStepSize	INT_32	0:4		

1.12.3.10 *extLensControlSetZoomAbsolute()*

Move to absolute zoom position in focal length. The zoomAbsolute defines the effective focal length in mm in the range of 000.000 - 999.999.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x003D000A	N/A	N/A	
zoomAbsolute	FLOAT	0:4		

No output parameters.

1.12.3.11 *extLensControlGetZoomAbsolute()*

Move to absolute zoom position in focal length. The zoomAbsolute defines the effective focal length in mm in the range of 000.000 - 999.999.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x003D000B	N/A	N/A	

FLIR NEUTRINO LC SOFTWARE IDD

Output/Receive parameters:

Name	Data Type	Bytes	Notes
zoomAbsolute	FLOAT	0:4	

1.12.3.12 *extLensControlSetTrackingState()*

Enable/disable various real-time tracking (controller initiated motion) based on lens temperature and object range. The trackingState defines the parameter: 4= temperature, 5 = range in the value range 1 - 6.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x003D000C	N/A	N/A
functionIndex	UINT_16	0:2	
axisNumber	UINT_16	2:4	

No output parameters.

1.12.3.13 *extLensControlGetTrackingState()*

Enable/disable various real-time tracking (controller initiated motion) based on lens temperature and object range. The trackingState variable defines the parameter: 4= temperature, 5 = range in the value range 1 - 6.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x003D000D	N/A	N/A
functionIndex	UINT_16	0:2	

Output/Receive parameters:

Name	Data Type	Bytes	Notes
axisNumber	UINT_16	0:2	

FLIR NEUTRINO LC SOFTWARE IDD

1.12.3.14 *extLensControlSetFocusRelative()*

Move to a relative focus position by incrementing or decrementing the focus motor by user provided step size, where focusDirection defines the step direction. 1 is positive direction. -1 is negative direction

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x003D000E	N/A	N/A
focusDirection	INT_32	0:4	

No output parameters.

1.12.3.15 *extLensControlGetFocusRelative()*

Move to a relative focus position by incrementing or decrementing the focus motor by user provided step size, where focusDirection defines the step direction. 1 is positive direction. -1 is negative direction

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x003D000F	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
focusDirection	INT_32	0:4	

1.12.3.16 *extLensControlSetFocusStepSize()*

Configure the focus motor step size used in the Focus Relative command. The focusStepSize defines the step size in focus encoder counts in the range -1000 - 1000 with a default of 10.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x003D0010	N/A	N/A
focusStepSize	INT_32	0:4	

No output parameters.

FLIR NEUTRINO LC SOFTWARE IDD

1.12.3.17 *extLensControlGetFocusStepSize()*

Configure the focus motor step size used in the Focus Relative command. The focusStepSize defines the step size in focus encoder counts in the range -1000 - 1000 with a default of 10.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x003D0011	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
focusStepSize	INT_32	0:4	

1.12.3.18 *extLensControlSetFocusRange()*

Focus the lens for specified object distance, where focusRange is object distance in meters.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x003D0012	N/A	N/A
focusRange	UINT_32	0:4	

No output parameters.

1.12.3.19 *extLensControlGetFocusRange()*

Focus the lens for specified object distance, where focusRange is object distance in meters.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x003D0013	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
focusRange	UINT_32	0:4	

1.12.3.20 *extLensControlReadLensStatus()*

Receive lens comm result string and lens comm result code. Up to 256 bytes.

FLIR NEUTRINO LC SOFTWARE IDD

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x003D0018	N/A	N/A
sizeInBytes	UINT_16	0:2	

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	BYTEARRAY	0:512	
receivedBytes	UINT_16	512:514	
resultCode	UINT_32	514:518	

1.12.3.21 *extLensControlReadStatus()*

Read the the module status. A non-zero value indicates an error has occurred.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x003D0019	N/A	N/A
readAndClear	FLR_EXTLENSCONTROL_ACTION_E	0:4	

Output/Receive parameters:

Name	Data Type	Bytes	Notes
statusCode	INT_32	0:4	

1.13 Module: FILEOPS

These APIs describe the File operations for managing files and directories.

1.13.1 Enums

No enumerations in module fileOps.

1.13.2 Structs

No struct types in module fileOps.

FLIR NEUTRINO LC SOFTWARE IDD

1.13.3 Functions

1.13.3.1 *fileOpsDir()*

Iterate through current directory contents.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00160000	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
dirent	UCHAR*128	0:128	

1.13.3.2 *fileOpsCd()*

Change the current working directory.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00160001	N/A	N/A
path	UCHAR*128	0:128	

Output/Receive parameters:

Name	Data Type	Bytes	Notes
pwd	UCHAR*128	0:128	

1.13.3.3 *fileOpsMd()*

Make a new directory.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00160002	N/A	N/A
path	UCHAR*128	0:128	

No output parameters.

FLIR NEUTRINO LC SOFTWARE IDD

1.13.3.4 *fileOpsFopen()*

Open a file pointer.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00160003	N/A	N/A
path	UCHAR*128	0:128	
mode	UCHAR*128	128:256	

Output/Receive parameters:

Name	Data Type	Bytes	Notes
id	UINT_32	0:4	

1.13.3.5 *fileOpsFclose()*

Close a file pointer.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00160004	N/A	N/A
id	UINT_32	0:4	

No output parameters.

1.13.3.6 *fileOpsFread()*

Read data from the specified file pointer.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00160005	N/A	N/A
id	UINT_32	0:4	
length	UINT_32	4:8	

Output/Receive parameters:

Name	Data Type	Bytes	Notes
buf	UCHAR*128	0:128	
ret	UINT_32	128:132	

FLIR NEUTRINO LC SOFTWARE IDD

1.13.3.7 *fileOpsFwrite()*

Write data to the specified file pointer.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00160006	N/A	N/A
id	UINT_32	0:4	
length	UINT_32	4:8	
buf	UCHAR*128	8:136	

Output/Receive parameters:

Name	Data Type	Bytes	Notes
ret	UINT_32	0:4	

1.13.3.8 *fileOpsFtell()*

Get the current data pointer location for the specified file pointer.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00160007	N/A	N/A
id	UINT_32	0:4	

Output/Receive parameters:

Name	Data Type	Bytes	Notes
offset	UINT_32	0:4	

1.13.3.9 *fileOpsFseek()*

Set the data pointer location for the specified file pointer.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00160008	N/A	N/A
id	UINT_32	0:4	
offset	UINT_32	4:8	

FLIR NEUTRINO LC SOFTWARE IDD

origin	UINT_32	8:12		
---------------	---------	------	--	--

No output parameters.

1.13.3.10 *fileOpsFtruncate()*

Truncate the specified file pointer to a maximum length.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x00160009	N/A	N/A	
id	UINT_32	0:4		
length	UINT_32	4:8		

No output parameters.

1.13.3.11 *fileOpsRmdir()*

Delete the specified folder.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x0016000A	N/A	N/A	
path	UCHAR*128	0:128		

No output parameters.

1.13.3.12 *fileOpsRm()*

Delete the specified file.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x0016000B	N/A	N/A	
path	UCHAR*128	0:128		

No output parameters.

1.13.3.13 *fileOpsRename()*

Delete the specified file or folder.

Input/Send parameters:

102-2020-42, Neutrino LC Software Interface Description Document, Rev400

Information on this page is subject to change without notice

This document does not contain any export-controlled information.



FLIR NEUTRINO LC SOFTWARE IDD

Name	Data Type	Bytes	Notes
FunctionID	0x0016000C	N/A	N/A
oldpath	UCHAR*128	0:128	
newpath	UCHAR*128	128:256	

No output parameters.

1.13.3.14 *fileOpsGetFileSize()*

Get the size of the specified file.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0016000D	N/A	N/A
path	UCHAR*128	0:128	

Output/Receive parameters:

Name	Data Type	Bytes	Notes
fileLength	UINT_32	0:4	

1.14 Module: FLASHIO

Please provide description.

1.14.1 Enums

No enumerations in module flashIO.

1.14.2 Structs

No struct types in module flashIO.

1.14.3 Functions

1.14.3.1 *flashIOSetProtectionState()*

Set the write protection state, allowing or disallowing flash write and erase operations.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00300001	N/A	N/A
protectionS	FLR_ENABLE_E	0:4	

FLIR NEUTRINO LC SOFTWARE IDD

tate				
-------------	--	--	--	--

No output parameters.

1.14.3.2 *flashIOGetProtectionState()*

Get the current write protection state.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00300002	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
protectionS tate	FLR_ENABLE_E	0:4	

1.15 Module: GAO

This module exposes functions to control the application of various gains and offsets during the Non Uniformity Corrections part of the pipeline.

1.15.1 Enums

1.15.1.1 *FLR_GAO_NUC_TYPE_E* — <INT_32>

FLR_GAO_NUC_TYPE_ONE_POINT_FFC = 0
 FLR_GAO_NUC_TYPE_TWO_POINT_FIELD = 1
 FLR_GAO_NUC_TYPE_TWO_POINT_FACTORY = 2
 FLR_GAO_NUC_TYPE_END = 3

1.15.2 Structs

1.15.2.1 *FLR_GAO_RNS_COL_CORRECT_T*

Field Name	Data Type	Bytes
value	INT_16*20	40

FLIR NEUTRINO LC SOFTWARE IDD

1.15.3 Functions

1.15.3.1 *gaoSetGainState()*

Enables / disables application of per-pixel gain coefficients. When disabled, unity gain is applied to all pixels. Most users should leave this enabled.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00000001	N/A	N/A
data	FLR_ENABLE_E	0:4	

No output parameters.

1.15.3.2 *gaoGetGainState()*

Reads the state (enabled/disabled) of per-pixel gain coefficients.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00000002	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLR_ENABLE_E	0:4	

1.15.3.3 *gaoSetFfcState()*

Enables / disables application of per-pixel Flat-Field Correction (FFC) coefficients. Most users should leave this enabled.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00000003	N/A	N/A
data	FLR_ENABLE_E	0:4	

No output parameters.

1.15.3.4 *gaoGetFfcState()*

Reads the state (enabled/disabled) of per-pixel FFC coefficients.

FLIR NEUTRINO LC SOFTWARE IDD

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00000004	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLR_ENABLE_E	0:4	

1.15.3.5 *gaoSetAveragerState()*

Enables / disables a smart-averager function which cuts frame rate in half.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0000000B	N/A	N/A
data	FLR_ENABLE_E	0:4	

No output parameters.

1.15.3.6 *gaoGetAveragerState()*

Reads the state (enabled/disabled) of the smart-averager function.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0000000C	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLR_ENABLE_E	0:4	

1.15.3.7 *gaoSetNumFFCFrames()*

Specifies the number of frames (2, 4, 8, or 16) to be integrated during flat-field correction (FFC).

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0000000D	N/A	N/A

FLIR NEUTRINO LC SOFTWARE IDD

data	UINT_16	0:2		
-------------	---------	-----	--	--

No output parameters.

1.15.3.8 *gaoGetNumFFCFrames()*

Reads the number of frames to be integrated during FFC.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x0000000E	N/A	N/A	

Output/Receive parameters:

Name	Data Type	Bytes	Notes	
data	UINT_16	0:2		

1.15.3.9 *gaoGetAveragerThreshold()*

Reads the threshold value used by the smart-averager function.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x00000010	N/A	N/A	

Output/Receive parameters:

Name	Data Type	Bytes	Notes	
data	UINT_16	0:2		

1.15.3.10 *gaoSetTestRampState()*

Enables / disables a test ramp generated by internal electronics (in lieu of data from the sensor array). Most users should leave this disabled as it is intended primarily as a diagnostic feature.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x00000013	N/A	N/A	
data	FLR_ENABLE_E	0:4		

FLIR NEUTRINO LC SOFTWARE IDD

No output parameters.

1.15.3.11 *gaoGetTestRampState()*

Reads the state (enabled/disabled) of a test ramp.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00000014	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLR_ENABLE_E	0:4	

1.15.3.12 *gaoSetSffcState()*

Enables / disables supplemental flat-field correction (SFFC)

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00000017	N/A	N/A
data	FLR_ENABLE_E	0:4	

No output parameters.

1.15.3.13 *gaoGetSffcState()*

Reads the state (enabled/disabled) of the supplemental flat-field correction.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00000018	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLR_ENABLE_E	0:4	

1.15.3.14 *gaoSetNucType()*

Sets the value of the NUC type to either one point or two point.

FLIR NEUTRINO LC SOFTWARE IDD

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00000023	N/A	N/A
nucType	FLR_GAO_NUC_TYPE_E	0:4	

No output parameters.

1.15.3.15 *gaoGetNucType()*

Reads the current value of the NUC type desired.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00000024	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
nucType	FLR_GAO_NUC_TYPE_E	0:4	

1.15.3.16 *gaoGetAveragerDesiredState()*

Return the state of the averager which will be used at the next boot of the camera if the dynamic header has been saved with this value.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0000003E	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLR_ENABLE_E	0:4	

1.16 Module: I2CBUS

Please provide description.

FLIR NEUTRINO LC SOFTWARE IDD

1.16.1 Enums

1.16.1.1 *FLR_I2CBUS_CONTROLLER_E* — <INT_32>

FLR_I2CBUS_I2C_1 = 0
FLR_I2CBUS_I2C_2 = 1
FLR_I2CBUS_I2C_3 = 2
FLR_I2CBUS_GPIO_I2C = 3

1.16.1.2 *FLR_I2CBUS_ADDR_TYPE_E* — <INT_32>

FLR_I2CBUS_7BIT_ADDR = 0
FLR_I2CBUS_10BIT_ADDR = 1

1.16.1.3 *FLR_I2CBUS_SUBADDR_TYPE_E* — <INT_32>

FLR_I2CBUS_8BIT_SUBADDR = 0
FLR_I2CBUS_16BIT_SUBADDR = 1
FLR_I2CBUS_24BIT_SUBADDR = 2
FLR_I2CBUS_32BIT_SUBADDR = 3
FLR_I2CBUS_40BIT_SUBADDR = 4
FLR_I2CBUS_NO_SUBADDR = 5

1.16.2 Structs

1.16.2.1 *FLR_I2CBUS_ADAPTER_CONFIG_T*

Field Name	Data Type	Bytes
sdaPin	UCHAR	1
sclPin	UCHAR	1
speedKhz	UINT_32	4
controller	FLR_I2CBUS_CONTROLLER_E	4

1.16.2.2 *FLR_I2CBUS_CLIENT_CONFIG_T*

Field Name	Data Type	Bytes
deviceAddress	UINT_32	4
addressType	FLR_I2CBUS_ADDR_TYPE_E	4
subaddressType	FLR_I2CBUS_SUBADDR_TYPE_E	4
adapter	FLR_I2CBUS_ADAPTER_CONFIG_T	10

FLIR NEUTRINO LC SOFTWARE IDD

1.16.3 Functions

1.16.3.1 *i2cBusOpenClient()*

Open an I2C client session.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00190000	N/A	N/A
i2cClient	FLR_I2CBUS_CLIENT_CONFIG_T	0:22	

Output/Receive parameters:

Name	Data Type	Bytes	Notes
id	UINT_32	0:4	

1.16.3.2 *i2cBusCloseClient()*

Close an I2C client session.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00190001	N/A	N/A
id	UINT_32	0:4	

No output parameters.

1.16.3.3 *i2cBusRead()*

Read data from a client session.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00190002	N/A	N/A
id	UINT_32	0:4	
subaddr	UINT_32	4:8	
length	UINT_32	8:12	

Output/Receive parameters:

Name	Data Type	Bytes	Notes
buf	UCHAR*128	0:128	

FLIR NEUTRINO LC SOFTWARE IDD

1.16.3.4 *i2cBusWrite()*

Write data to a client session.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00190003	N/A	N/A
id	UINT_32	0:4	
subaddr	UINT_32	4:8	
buf	UCHAR*128	8:136	
length	UINT_32	136:140	

No output parameters.

1.17 Module: I2CCPE

No description available

1.17.1 Enums

1.17.1.1 *FLR_I2CCPE_FLASH_INFO_E* — <UINT_32>

FLR_I2CCPE_FLASH_INFO_FLASH_SIZE = 0x23000 (143,360)

FLR_I2CCPE_FLASH_INFO_FLASH_START = 0x2B000

FLR_I2CCPE_FLASH_INFO_END = 0xFFFFFFFF

1.17.2 Structs

No struct types in module i2cCpe.

1.17.3 Functions

1.17.3.1 *i2cCpeNetworkRegisterRead()*

No description available

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x001F0001	N/A	N/A
reg_addr	UCHAR	0:1	
sizeInBytes	UCHAR	1:2	

FLIR NEUTRINO LC SOFTWARE IDD

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	BYTEARRAY	0:512	

1.17.3.2 *i2cCpeNetworkFlashRead()*

No description available

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x001F0006	N/A	N/A
reg_addr	UINT_32	0:4	
sizeInBytes	UINT_16	4:6	

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	BYTEARRAY	0:512	

1.17.3.3 *i2cCpeNetworkFlashWrite()*

No description available

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x001F0007	N/A	N/A
reg_addr	UINT_32	0:4	
sizeInBytes	UINT_16	4:6	
data	BYTEARRAY	6:262	

No output parameters.

1.17.3.4 *i2cCpeNetworkFlashImageErase()*

The image erase function erases the whole flash space available for field upgrade.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x001F0009	N/A	N/A

FLIR NEUTRINO LC SOFTWARE IDD

No output parameters.

1.17.3.5 *i2cCpeNetworkFlashImageWrite()*

The image write function is used to write the new FPGA bin image beginning at the start address of the CPE FLASH space and continuing in 256-byte chunks until the whole image is written.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x001F000A	N/A	N/A
address	UINT_32	0:4	
sizeInBytes	UINT_16	4:6	
data	BYTEARRAY	6:262	

No output parameters.

1.17.3.6 *i2cCpeNetworkFlashImageRead()*

The image read function is used to read the new FPGA bin image beginning at the start address of the CPE FLASH space and continuing in 256-byte chunks until the whole image is read.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x001F000B	N/A	N/A
address	UINT_32	0:4	
sizeInBytes	UINT_16	4:6	

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	BYTEARRAY	0:512	

1.17.3.7 *i2cCpeNetworkFlashInfo()*

The flash info command returns the start address and size of the CPE FLASH space available for field upgrade.

Input/Send parameters:

Name	Data Type	Bytes	Notes
------	-----------	-------	-------

FLIR NEUTRINO LC SOFTWARE IDD

FunctionID	0x001F000C	N/A	N/A	
------------	------------	-----	-----	--

Output/Receive parameters:

Name	Data Type	Bytes	Notes	
flash_start	UINT_32	0:4		
flash_size	UINT_32	4:8		
status	UINT_32	8:12		

1.18 Module: I2CTOSERIAL

This is the module that supports the I2C to serial feature. It assumes the presence of I2C to serial hardware. If the hardware is not present the system will respond with FLR_NOT_READY.

1.18.1 Enums

1.18.1.1 FLR_I2CTOSERIAL_ACTION_E — <INT_32>

FLR_I2CTOSERIAL_ACTION_READ_ONLY = 0
FLR_I2CTOSERIAL_ACTION_READ_AND_CLEAR = 1
FLR_I2CTOSERIAL_ACTION_END = 2

1.18.1.2 FLR_I2CTOSERIAL_COMM_LOOPBACK_E — <INT_32>

FLR_I2CTOSERIAL_OFF = 0
FLR_I2CTOSERIAL_ON = 1

1.18.1.3 FLR_I2CTOSERIAL_COMM_BITS_E — <INT_32>

FLR_I2CTOSERIAL_SEVEN = 7
FLR_I2CTOSERIAL_EIGHT = 8

1.18.1.4 FLR_I2CTOSERIAL_COMM_PARITY_E — <INT_32>

FLR_I2CTOSERIAL_NONE = 0
FLR_I2CTOSERIAL_ODD = 1
FLR_I2CTOSERIAL_EVEN = 2

1.18.1.5 FLR_I2CTOSERIAL_COMM_STOPBITS_E — <INT_32>

FLR_I2CTOSERIAL_1 = 0
FLR_I2CTOSERIAL_1_5 = 1
FLR_I2CTOSERIAL_2 = 2

FLIR NEUTRINO LC SOFTWARE IDD

1.18.1.6 FLR_I2CTOSERIAL_COMM_FLOWCONTROL_E — <INT_32>

FLR_I2CTOSERIAL_NO = 0
FLR_I2CTOSERIAL_SW = 1
FLR_I2CTOSERIAL_HW = 2

1.18.1.7 FLR_I2CTOSERIAL_COMM_TYPE_E — <INT_32>

FLR_I2CTOSERIAL_RS232_3V3 = 0
FLR_I2CTOSERIAL_RS485 = 1

1.18.1.8 FLR_I2CTOSERIAL_STATE_E — <INT_32>

FLR_I2CTOSERIAL_STATE_HW_CONFIG_OK = 0
FLR_I2CTOSERIAL_STATE_HW_NO_CONFIG = 1
FLR_I2CTOSERIAL_STATE_END = 2

1.18.2 Structs

No struct types in module i2cToSerial.

1.18.3 Functions

1.18.3.1 i2cToSerialGetEnable()

Enable or disable the i2c to serial port. NOTE: the enable will only take effect after a camera reset.

Input/Send parameters:

Name	DataType	Bytes	Notes
FunctionID	0x003A0000	N/A	N/A

Output/Receive parameters:

Name	DataType	Bytes	Notes
state	FLR_ENABLE_E	0:4	

1.18.3.2 i2cToSerialSetEnable()

Enable or disable the i2c to serial port. NOTE: the enable will only take effect after a camera reset.

Input/Send parameters:

Name	DataType	Bytes	Notes
FunctionID	0x003A0001	N/A	N/A

FLIR NEUTRINO LC SOFTWARE IDD

state	FLR_ENABLE_E	0:4		
--------------	--------------	-----	--	--

No output parameters.

1.18.3.3 *i2cToSerialSetSerialConfig()*

Set the output configuration for the I2C to Serial port.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x003A0002	N/A	N/A	
Bits	FLR_I2CTOSERIAL_COMM_BITS_E	0:4		
Parity	FLR_I2CTOSERIAL_COMM_PARITY_E	4:8		
Baudrate	UINT_32	8:12		
Stopbits	FLR_I2CTOSERIAL_COMM_STOPBITS_E	12:16		
FlowControl	FLR_I2CTOSERIAL_COMM_FLOWCONTROL_E	16:20		

No output parameters.

1.18.3.4 *i2cToSerialGetSerialConfig()*

Get the output configuration for the I2C to Serial port.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x003A0003	N/A	N/A	

Output/Receive parameters:

Name	Data Type	Bytes	Notes	
Bits	FLR_I2CTOSERIAL_COMM_BITS_E	0:4		
Parity	FLR_I2CTOSERIAL_COMM_PARITY_E	4:8		
Baudrate	UINT_32	8:12		
Stopbits	FLR_I2CTOSERIAL_COMM_STOPBITS_E	12:16		
FlowControl	FLR_I2CTOSERIAL_COMM_FLOWCONTROL_E	16:20		

FLIR NEUTRINO LC SOFTWARE IDD

1.18.3.5 *i2cToSerialRegisterWrite()*

Write one byte of data to the I2C to serial device register specified by `reg_addr`. This command requires an accessory board with an I2C to serial chip like the MAX3770.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x003A0004	N/A	N/A
reg_addr	UCHAR	0:1	
data	UCHAR	1:2	

No output parameters.

1.18.3.6 *i2cToSerialRegisterRead()*

Read one byte of data from the I2C to serial device register specified by `reg_addr`. This command requires an accessory board with an I2C to serial chip like the MAX3770.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x003A0005	N/A	N/A
reg_addr	UCHAR	0:1	

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	UCHAR	0:1	

1.18.3.7 *i2cToSerialSendData()*

Send data over the I2C to serial interface. The data will be transmitted out of the serial device. Up to 128 bytes at a time may be sent.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x003A0006	N/A	N/A
sizeInBytes	UINT_16	0:2	
data	BYTEARRAY	2:258	

FLIR NEUTRINO LC SOFTWARE IDD

Output/Receive parameters:

Name	Data Type	Bytes	Notes
sentBytes	UINT_16	0:2	

1.18.3.8 *i2cToSerialReceiveData()*

Receive data from the I2C to serial interface. The data will be the data received into the serial port. Up to 128 bytes at a time may be received.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x003A0007	N/A	N/A
sizeInBytes	UINT_16	0:2	

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	BYTEARRAY	0:512	
receivedBytes	UINT_16	512:514	

1.18.3.9 *i2cToSerialGetLoopback()*

Turn On/Off loopback mode on the i2c to serial board.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x003A0008	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
loopback	FLR_I2CTOSERIAL_COMM_LOOPBACK_E	0:4	

1.18.3.10 *i2cToSerialSetLoopback()*

Turn On/Off loopback mode on the i2c to serial board.

Input/Send parameters:

Name	Data Type	Bytes	Notes
------	-----------	-------	-------

FLIR NEUTRINO LC SOFTWARE IDD

FunctionID	0x003A0009	N/A	N/A	
loopback	FLR_I2CTOSERIAL_COMM_LOOPBACK_E	0:4		

No output parameters.

1.18.3.11 *i2cToSerialGetCommType()*

Select comm port protocol type: RS232 or RS485

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x003A000A	N/A	N/A	

Output/Receive parameters:

Name	Data Type	Bytes	Notes	
commType	FLR_I2CTOSERIAL_COMM_TYPE_E	0:4		

1.18.3.12 *i2cToSerialSetCommType()*

Select comm port protocol type: RS232 or RS485

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x003A000B	N/A	N/A	
commType	FLR_I2CTOSERIAL_COMM_TYPE_E	0:4		

No output parameters.

1.18.3.13 *i2cToSerialReceiveSwFifoInfo()*

The number of characters in the receive SW fifo.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x003A000C	N/A	N/A	

Output/Receive parameters:

Name	Data Type	Bytes	Notes	
insert	UINT_32	0:4		

FLIR NEUTRINO LC SOFTWARE IDD

remove	UINT_32	4:8		
wrap	UINT_32	8:12		

1.18.3.14 *i2cToSerialTransmitSwFifoInfo()*

The number of characters in the transmit SW fifo.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x003A000D	N/A	N/A	

Output/Receive parameters:

Name	Data Type	Bytes	Notes	
insert	UINT_32	0:4		
remove	UINT_32	4:8		
wrap	UINT_32	8:12		

1.18.3.15 *i2cToSerialReadStatus()*

Read the the module status. A non-zero value indicates an error has occurred

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x003A000E	N/A	N/A	
readAndClear	FLR_I2CTOSERIAL_ACTION_E	0:4		

Output/Receive parameters:

Name	Data Type	Bytes	Notes	
statusCode	INT_32	0:4		

1.19 Module: IMAGESTATS

The Image Stats module provides the APIs needed to gather statistics about the images being captured through the camera. The statistics are gathered from the whole image or for a region of interest selected using APIs for that purpose.

FLIR NEUTRINO LC SOFTWARE IDD

1.19.1 Enums

No enumerations in module imageStats.

1.19.2 Structs

No struct types in module imageStats.

1.19.3 Functions

1.19.3.1 *imageStatsGetTotalHistPixelsInROI()*

Return the number of pixels which are described by the ROI

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x001D0000	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
totalPixelsInROI	UINT_32	0:4	

1.19.3.2 *imageStatsGetPopBelowLowToHighThresh()*

The number of pixels that are below the low to high threshold.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x001D0001	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
popBelowLowToHighThresh	UINT_32	0:4	

1.19.3.3 *imageStatsGetPopAboveHighToLowThresh()*

The number of pixels that are above the high to low threshold.

Input/Send parameters:

FLIR NEUTRINO LC SOFTWARE IDD

Name	Data Type	Bytes	Notes
FunctionID	0x001D0002	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
popAboveHighToLowThreshold	UINT_32	0:4	

1.19.3.4 *imageStatsSetROI()*

Sets the ROI to be used when collecting ROI image stats. The ROI sent is a datatype that describes row start, row stop, column start, and column stop.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x001D0003	N/A	N/A
roi	FLR_ROI_T	0:8	

No output parameters.

1.19.3.5 *imageStatsGetROI()*

Return the coordinates of the ROI to use for the collection of image stats. The returned ROI is a datatype that describes row start, row stop, column start, and column stop.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x001D0004	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
roi	FLR_ROI_T	0:8	

1.19.3.6 *imageStatsGetFirstBin()*

Return the first bin that contains an intensity value

Input/Send parameters:

FLIR NEUTRINO LC SOFTWARE IDD

Name	Data Type	Bytes	Notes
FunctionID	0x001D0005	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
firstBin	UINT_16	0:2	

1.19.3.7 *imageStatsGetLastBin()*

Return the last bin that contains an intensity value

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x001D0006	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
lastBin	UINT_16	0:2	

1.19.3.8 *imageStatsGetMean()*

Return the mean intensity value in the image.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x001D0007	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
mean	UINT_16	0:2	

1.19.3.9 *imageStatsGetFirstBinInROI()*

Return the intensity value in the first bin in the ROI.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x001D0008	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
firstBinInROI	UINT_16	0:2	

1.19.3.10 *imageStatsGetLastBinInROI()*

Return the intensity value in the last bin in the ROI.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x001D0009	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
lastBinInROI	UINT_16	0:2	

1.19.3.11 *imageStatsGetMeanInROI()*

This function returns the mean intensity value in the selected region of interest.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x001D000A	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
meanInROI	UINT_16	0:2	

1.19.3.12 *imageStatsGetImageStats()*

This function returns the mean, peak, and base intensity values in the image.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x001D000B	N/A	N/A

FLIR NEUTRINO LC SOFTWARE IDD

Output/Receive parameters:

Name	Data Type	Bytes	Notes
meanIntensity	UINT_16	0:2	
peakIntensity	UINT_16	2:4	
baseIntensity	UINT_16	4:6	

1.20 Module: JFFS2

These APIs describe operations that may be performed on the JFFS2 file system.

1.20.1 Enums

1.20.1.1 FLR_JFFS2_STATE_E — <INT_32>

FLR_JFFS2_INITIAL = 0
FLR_JFFS2_CONFIGURED = 1
FLR_JFFS2_MOUNTING = 2
FLR_JFFS2_MOUNTED = 3
FLR_JFFS2_UNMOUNTING = 4
FLR_JFFS2_UNMOUNTED = 5
FLR_JFFS2_FAILED_MOUNT = 6
FLR_JFFS2_FAILED_UNMOUNT = 7
FLR_JFFS2_FAILED_CONFIG = 8
FLR_JFFS2_DISABLED = 9
FLR_JFFS2_STATE_END = 10

1.20.2 Structs

No struct types in module jffs2.

1.20.3 Functions

1.20.3.1 jffs2Mount()

Mount the JFFS2 filesystem. Will mount existing filesystem, format empty flash, or fail on non-empty flash.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00170001	N/A	N/A

FLIR NEUTRINO LC SOFTWARE IDD

No output parameters.

1.20.3.2 *jffs2Unmount()*

Unmount the JFFS2 filesystem.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00170002	N/A	N/A

No output parameters.

1.20.3.3 *jffs2GetState()*

Get the current JFFS2 filesystem status.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00170007	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
state	FLR_JFFS2_STATE_E	0:4	

1.21 Module: MEM

The mem module provides tools for byte level access to volatile and persistent memory objects.

1.21.1 Enums

1.21.1.1 *FLR_MEM_LOCATION_E* — <INT_32>

FLR_MEM_INVALID = 0
FLR_MEM_BOOTLOADER = 1
FLR_MEM_UPGRADE_APP = 2
FLR_MEM_LENS_NVFFC = 3
FLR_MEM_LENS_SFFC = 4
FLR_MEM_LENS_GAIN = 5
FLR_MEM_LENS_DISTORTION = 6
FLR_MEM_USER_SPACE = 7

FLIR NEUTRINO LC SOFTWARE IDD

FLR_MEM_RUN_CMDS = 8
FLR_MEM_JFFS2 = 9
FLR_MEM_MEMTEST_APP = 10
FLR_MEM_LAST = 11

1.21.2 Structs

No struct types in module mem.

1.21.3 Functions

1.21.3.1 *memReadCapture()*

Read bytes from the selected image buffer at the specified offset.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0xFFFF0003	N/A	N/A
bufferNum	UCHAR	0:1	
offset	UINT_32	1:5	
sizeInBytes	UINT_16	5:7	

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	BYTEARRAY	0:512	

1.21.3.2 *memGetCaptureSize()*

Get the size of the buffer in bytes, as well as the number of rows and columns in each capture buffer.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0xFFFF0004	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
bytes	UINT_32	0:4	
rows	UINT_16	4:6	
columns	UINT_16	6:8	

1.21.3.3 *memWriteFlash()*

Write bytes to the selected Flash enum at the specified offset. Lens enums require an additional index parameter, other enums must set this parameter to 0.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0xFFFF0005	N/A	N/A
location	FLR_MEM_LOCATION_E	0:4	
index	UCHAR	4:5	
offset	UINT_32	5:9	
sizeInBytes	UINT_16	9:11	
data	BYTEARRAY	11:267	

No output parameters.

1.21.3.4 *memReadFlash()*

Read bytes from the selected Flash enum at the specified offset. Lens enums require an additional index parameter, other enums must set this parameter to 0.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0xFFFF0006	N/A	N/A
location	FLR_MEM_LOCATION_E	0:4	
index	UCHAR	4:5	
offset	UINT_32	5:9	
sizeInBytes	UINT_16	9:11	

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	BYTEARRAY	0:512	

1.21.3.5 *memGetFlashSize()*

Get the size of a specified Flash enum in bytes.

Input/Send parameters:

102-2020-42, Neutrino LC Software Interface Description Document, Rev400

Information on this page is subject to change without notice

This document does not contain any export-controlled information.

FLIR NEUTRINO LC SOFTWARE IDD

Name	Data Type	Bytes	Notes
FunctionID	0xFFFF0007	N/A	N/A
location	FLR_MEM_LOCATION_E	0:4	

Output/Receive parameters:

Name	Data Type	Bytes	Notes
bytes	UINT_32	0:4	

1.21.3.6 *memEraseFlash()*

Prepare the specified Flash location for writing. Lens enums require an additional index parameter, other enums must set this parameter to 0.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0xFFFF0008	N/A	N/A
location	FLR_MEM_LOCATION_E	0:4	
index	UCHAR	4:5	

No output parameters.

1.21.3.7 *memEraseFlashPartial()*

Prepare subsections of the specified Flash location for writing. Flash erases must start and end on a multiple of 0x1000.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0xFFFF0009	N/A	N/A
location	FLR_MEM_LOCATION_E	0:4	
index	UCHAR	4:5	
offset	UINT_32	5:9	
length	UINT_32	9:13	

No output parameters.

1.21.3.8 *memReadCurrentGain()*

Read bytes from the current applied gain buffer.

FLIR NEUTRINO LC SOFTWARE IDD

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0xFFFF000A	N/A	N/A
offset	UINT_32	0:4	
sizeInBytes	UINT_16	4:6	

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	BYTEARRAY	0:512	

1.21.3.9 memGetGainSize()

Get the size of the buffer in bytes, as well as the number of rows and columns in applied gain buffer.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0xFFFF000B	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
bytes	UINT_32	0:4	
rows	UINT_16	4:6	
columns	UINT_16	6:8	

1.21.3.10 memGetCaptureSizeSrc()

Get the size of the capture buffer in bytes, rows, and columns according to specified capture source.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0xFFFF000C	N/A	N/A
src	FLR_CAPTURE_SRC_E	0:4	

Output/Receive parameters:

Name	Data Type	Bytes	Notes
------	-----------	-------	-------

FLIR NEUTRINO LC SOFTWARE IDD

bytes	UINT_32	0:4		
rows	UINT_16	4:6		
columns	UINT_16	6:8		

1.21.3.11 *memReadCaptureSrc()*

Read bytes from the selected image buffer at the specified offset, assuming the specified capture source.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0xFFFF000D	N/A	N/A	
src	FLR_CAPTURE_SRC_E	0:4		
bufferNum	UCHAR	4:5		
offset	UINT_32	5:9		
sizeInBytes	UINT_16	9:11		

Output/Receive parameters:

Name	Data Type	Bytes	Notes	
data	BYTEARRAY	0:512		

1.22 Module: ROIC

Interface to read FPA related variables

1.22.1 Enums

1.22.1.1 *FLR_ROIC_TEMP_MODE_E* — <INT_32>

FLR_ROIC_TEMP_NORMAL_MODE = 0
FLR_ROIC_TEMP_OFFSET_MODE = 1
FLR_ROIC_TEMP_STATIC_MODE = 2
FLR_ROIC_TEMP_MODE_END = 3

1.22.1.2 *FLR_ROIC_EXT_SYNC_MODE_E* — <INT_32>

FLR_ROIC_EXT_SYNC_DISABLE_MODE = 0
FLR_ROIC_EXT_SYNC_MASTER_MODE = 1
FLR_ROIC_EXT_SYNC_SLAVE_MODE = 2
FLR_ROIC_EXT_SYNC_END = 3

FLIR NEUTRINO LC SOFTWARE IDD

1.22.1.3 FLR_ROIC_ALL_AD1_MODE_E — <INT_32>

FLR_ROIC_ALL_AD1_DISABLE_MODE = 0
FLR_ROIC_ALL_AD1_ENABLE_MODE = 1
FLR_ROIC_ALL_AD1_TOGGLE_MODE = 2
FLR_ROIC_ALL_AD1_END = 3

1.22.2 Structs

1.22.2.1 FLR_ROIC_FPATEMP_TABLE_T

Field Name	Data Type	Bytes
value	INT_16*32	64

1.22.3 Functions

1.22.3.1 roicGetFPATemp()

Reads the raw (uncorrected) output of the focal plane array temperature sensor. Note: A different command, bosonlookupFPATempDegCx10, provides the calibrated output in degrees Celsius, and bosonlookupFPATempDegKx10 provides the output in Kelvin.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00020001	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	UINT_16	0:2	

1.22.3.2 roicGetFrameCount()

Reads the value of a frame counter which increments by one for each new frame.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00020002	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	UINT_32	0:4	

FLIR NEUTRINO LC SOFTWARE IDD

1.22.3.3 *roicSetFPARampState()*

Enables / disables a test ramp generated by the sensor array. Most users should leave this disabled as it is intended primarily as a diagnostic feature.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00020014	N/A	N/A
state	FLR_ENABLE_E	0:4	

No output parameters.

1.22.3.4 *roicGetFPARampState()*

Gets the state of the sensor array ramp.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00020015	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
state	FLR_ENABLE_E	0:4	

1.22.3.5 *roicGetSensorADC1()*

The value of an internal analog-to-digital converter. This internal ADC is not currently used.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00020019	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	UINT_16	0:2	

1.22.3.6 *roicGetSensorADC2()*

The value of an internal analog-to-digital converter. This internal ADC is not currently used.

FLIR NEUTRINO LC SOFTWARE IDD

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0002001A	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	UINT_16	0:2	

1.22.3.7 *roicSetFPATempOffset()*

Specifies an override of or an offset applied to the camera's internal temperature sensor, intended primarily as a diagnostic feature. Only has effect in two of the three FPA Temp modes (see *roicSetFPATempMode*)

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0002001B	N/A	N/A
data	INT_16	0:2	

No output parameters.

1.22.3.8 *roicGetFPATempOffset()*

Reads the value of an override / offset value applied to the camera's internal temperature sensor.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0002001C	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	INT_16	0:2	

1.22.3.9 *roicSetFPATempMode()*

Specifies the FPA temp mode (normal, fixed/override, or offset). Fixed/override and offset modes are intended primarily as diagnostic features, and most customers should leave this in its default state.

FLIR NEUTRINO LC SOFTWARE IDD

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0002001D	N/A	N/A
data	FLR_ROIC_TEMP_MODE_E	0:4	

No output parameters.

1.22.3.10 *roicGetFPATempMode()*

Reads the FPA temp mode (normal, fixed/override, or offset).

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0002001E	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLR_ROIC_TEMP_MODE_E	0:4	

1.22.3.11 *roicGetFPATempTable()*

The look-up table used internally for conversion of the raw output of the camera's internal temp sensor into a calibrated value (deg C or Kelvin).

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00020020	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
table	FLR_ROIC_FPATEMP_TABLE_T	0:64	

1.22.3.12 *roicSetFPATempValue()*

Sets the value of the FPA temp when the FPA temp mode is set to fixed.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00020022	N/A	N/A

FLIR NEUTRINO LC SOFTWARE IDD

data	UINT_16	0:2		
-------------	---------	-----	--	--

No output parameters.

1.22.3.13 *roicGetFPATempValue()*

Gets the value of the FPA Temp when the FPA temp mode is set to fixed. Alternately, in this mode *roicGetFPATemp* returns the same value.

Input/Send parameters:

Name	DataType	Bytes	Notes	
FunctionID	0x00020023	N/A	N/A	

Output/Receive parameters:

Name	DataType	Bytes	Notes	
data	UINT_16	0:2		

1.22.3.14 *roicGetRoicStarted()*

Get state of ROIC.

Input/Send parameters:

Name	DataType	Bytes	Notes	
FunctionID	0x0002002C	N/A	N/A	

Output/Receive parameters:

Name	DataType	Bytes	Notes	
roicStarted	FLR_ENABLE_E	0:4		

1.23 Module: SCALER

This module is used to control eZoom functionality.

1.23.1 Enums

No enumerations in module scaler.

FLIR NEUTRINO LC SOFTWARE IDD

1.23.2 Structs

1.23.2.1 FLR_SCALER_ZOOM_PARAMS_T

Field Name	Data Type	Bytes
zoom	UINT_32	4
xCenter	UINT_32	4
yCenter	UINT_32	4

1.23.3 Functions

1.23.3.1 *scalerGetMaxZoom()*

Get the maximum scaling factor allowed by the current camera configuration.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000D0001	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
zoom	UINT_32	0:4	

1.23.3.2 *scalerSetZoom()*

Set the current zoom parameters.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000D0002	N/A	N/A
zoomParams	FLR_SCALER_ZOOM_PARAMS_T	0:12	

No output parameters.

1.23.3.3 *scalerGetZoom()*

Get the current zoom parameters.

Input/Send parameters:

Name	Data Type	Bytes	Notes
------	-----------	-------	-------

FLIR NEUTRINO LC SOFTWARE IDD

FunctionID	0x000D0003	N/A	N/A	
-------------------	------------	-----	-----	--

Output/Receive parameters:

Name	Data Type	Bytes	Notes	
zoomParams	FLR_SCALER_ZOOM_PARAMS_T	0:12		

1.23.3.4 *scalerSetFractionalZoom()*

Set zoom parameters using numerator and denominator.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x000D0007	N/A	N/A	
zoomNumerator	UINT_32	0:4		
zoomDenominator	UINT_32	4:8		
zoomXCenter	UINT_32	8:12		
zoomYCenter	UINT_32	12:16		
inChangeEnable	FLR_ENABLE_E	16:20		
zoomOutXCenter	UINT_32	20:24		
zoomOutYCenter	UINT_32	24:28		
outChangeEnable	FLR_ENABLE_E	28:32		

No output parameters.

1.23.3.5 *scalerSetIndexZoom()*

Set zoom parameters pre-calculated steps.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x000D0008	N/A	N/A	

FLIR NEUTRINO LC SOFTWARE IDD

zoomIndex	UINT_32	0:4		
zoomXCenter	UINT_32	4:8		
zoomYCenter	UINT_32	8:12		
inChangeEnable	FLR_ENABLE_E	12:16		
zoomOutXCenter	UINT_32	16:20		
zoomOutYCenter	UINT_32	20:24		
outChangeEnable	FLR_ENABLE_E	24:28		

No output parameters.

1.24 Module: SCNR

Spatial Column Noise Reduction settings

1.24.1 Enums

1.24.1.1 *FLR_SCNR_CORR_SELECT_E* — <INT_32>

FLR_SCNR_STD_CORR = 0
FLR_SCNR_ABS_DIFF_CORR = 1
FLR_SCNR_CORR_END = 2

1.24.1.2 *FLR_SCNR_MODE_E* — <INT_32>

FLR_SCNR_MODE_M = 0
FLR_SCNR_MODE_I = 1
FLR_SCNR_MODE_DIFFS = 2
FLR_SCNR_MODE_OFFSETS = 3
FLR_SCNR_MODE_END = 4

1.24.2 Structs

No struct types in module scnr.

1.24.3 Functions

1.24.3.1 *scnrSetEnableState()*

Enable or disable Spatial Column Noise Reduction (scnr).

FLIR NEUTRINO LC SOFTWARE IDD

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00080001	N/A	N/A
data	FLR_ENABLE_E	0:4	

No output parameters.

1.24.3.2 *scnrGetEnableState()*

Get scnr correction's current state.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00080002	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLR_ENABLE_E	0:4	

1.24.3.3 *scnrSetThColSum()*

Set the threshold that determines if a column should increment or decrement by 1.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00080003	N/A	N/A
data	UINT_16	0:2	

No output parameters.

1.24.3.4 *scnrGetThColSum()*

Get the current value of ThColSum.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00080004	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
------	-----------	-------	-------

FLIR NEUTRINO LC SOFTWARE IDD

data	UINT_16	0:2		
-------------	---------	-----	--	--

1.24.3.5 *scnrSetThPixel()*

Set the (base) threshold that determines if a neighboring pixel is within range to affect the correction of the center.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x00080005	N/A	N/A	
data	UINT_16	0:2		

No output parameters.

1.24.3.6 *scnrGetThPixel()*

Get the current (base) value of ThPixel.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x00080006	N/A	N/A	

Output/Receive parameters:

Name	Data Type	Bytes	Notes	
data	UINT_16	0:2		

1.24.3.7 *scnrSetMaxCorr()*

Set the (base) maximum correction amount that will be applied.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x00080007	N/A	N/A	
data	UINT_16	0:2		

No output parameters.

1.24.3.8 *scnrGetMaxCorr()*

Get the (base) maximum correction.

FLIR NEUTRINO LC SOFTWARE IDD

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00080008	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	UINT_16	0:2	

1.24.3.9 *scnrGetThPixelApplied()*

Get the current (scaled with temperature) value of ThPixel.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0008000A	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	UINT_16	0:2	

1.24.3.10 *scnrGetMaxCorrApplied()*

Get the (scaled with temperature) maximum correction.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0008000B	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	UINT_16	0:2	

1.24.3.11 *scnrSetThColSumSafe()*

Set the threshold (for Safe Mode) that determines if a column should increment or decrement by 1.

Input/Send parameters:

FLIR NEUTRINO LC SOFTWARE IDD

Name	Data Type	Bytes	Notes
FunctionID	0x0008000C	N/A	N/A
data	UINT_16	0:2	

No output parameters.

1.24.3.12 *scnrGetThColSumSafe()*

Get the current value of ThColSum(Safe Mode).

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0008000D	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	UINT_16	0:2	

1.24.3.13 *scnrSetThPixelSafe()*

Set the (base) threshold (for Safe Mode) that determines if a neighboring pixel is within range to affect the correction of the center.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0008000E	N/A	N/A
data	UINT_16	0:2	

No output parameters.

1.24.3.14 *scnrGetThPixelSafe()*

Get the current (base) value of ThPixel (Safe Mode).

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0008000F	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
------	-----------	-------	-------

FLIR NEUTRINO LC SOFTWARE IDD

data	UINT_16	0:2		
-------------	---------	-----	--	--

1.24.3.15 *scnrSetMaxCorrSafe()*

Set the (base) maximum correction amount (for Safe Mode) that will be applied.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x00080010	N/A	N/A	
data	UINT_16	0:2		

No output parameters.

1.24.3.16 *scnrGetMaxCorrSafe()*

Get the (base) maximum correction (Safe Mode).

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x00080011	N/A	N/A	

Output/Receive parameters:

Name	Data Type	Bytes	Notes	
data	UINT_16	0:2		

1.24.3.17 *scnrSetCorrectionMethod()*

Correction Method (scnr).

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x00080012	N/A	N/A	
data	FLR_SCNR_CORR_SELECT_E	0:4		

No output parameters.

1.24.3.18 *scnrGetCorrectionMethod()*

Correction Method (scnr).

Input/Send parameters:

102-2020-42, Neutrino LC Software Interface Description Document, Rev400

Information on this page is subject to change without notice

This document does not contain any export-controlled information.

FLIR NEUTRINO LC SOFTWARE IDD

Name	Data Type	Bytes	Notes
FunctionID	0x00080013	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLR_SCNR_CORR_SELECT_E	0:4	

1.24.3.19 *scnrSetStdThreshold()*

Std Deviation Threshold (scnr).

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00080014	N/A	N/A
data	UINT_16	0:2	

No output parameters.

1.24.3.20 *scnrGetStdThreshold()*

Std Deviation Threshold (scnr).

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00080015	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	UINT_16	0:2	

1.24.3.21 *scnrSetNFrames()*

NumFrames for M-Mode (scnr).

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00080016	N/A	N/A
data	UINT_32	0:4	

FLIR NEUTRINO LC SOFTWARE IDD

No output parameters.

1.24.3.22 *scnrGetNFrames()*

NumFrames for M-Mode (scnr).

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00080017	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	UINT_32	0:4	

1.24.3.23 *scnrSetResetDesired()*

Reset Desired (scnr).

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00080018	N/A	N/A
data	UINT_32	0:4	

No output parameters.

1.24.3.24 *scnrGetResetDesired()*

Reset Desired (scnr).

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00080019	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	UINT_32	0:4	

1.24.3.25 *scnrSetM_modeOnly()*

M Mode Only (scnr).

FLIR NEUTRINO LC SOFTWARE IDD

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0008001A	N/A	N/A
data	UINT_32	0:4	

No output parameters.

1.24.3.26 *scnrGetM_modeOnly()*

M Mode Only (scnr).

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0008001B	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	UINT_32	0:4	

1.24.3.27 *scnrGetMode()*

SCNR Mode (scnr).

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0008001C	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLR_SCNR_MODE_E	0:4	

1.24.3.28 *scnrSetPixelBinEnableState()*

Pixel Binning Enable state.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0008001E	N/A	N/A
data	FLR_ENABLE_E	0:4	

No output parameters.

1.24.3.29 *scnrGetPixelBinEnableState()*

Pixel Binning Enable state.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0008001F	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLR_ENABLE_E	0:4	

1.25 Module: SPLASHSCREEN

Please provide description.

1.25.1 Enums

1.25.1.1 *FLR_SPLASHSCREEN_FILETYPE_E* — <INT_32>

FLR_SPLASHSCREEN_PNG = 0
 FLR_SPLASHSCREEN_BMP = 1
 FLR_SPLASHSCREEN_RAW = 2
 FLR_SPLASHSCREEN_NONE = 3
 FLR_SPLASHSCREEN_FILE_END = 4

1.25.2 Structs

No struct types in module splashScreen.

1.25.3 Functions

1.25.3.1 *splashScreenSetDuration()*

Set the duration to display the specified splash screen.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x001A0000	N/A	N/A
screen_nu	UINT_32	0:4	

FLIR NEUTRINO LC SOFTWARE IDD

m				
periodMs	UINT_32	4:8		

No output parameters.

1.25.3.2 *splashScreenSetDataType()*

Set the data type for the specified splash screen.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x001A0001	N/A	N/A	
screen_num	UINT_32	0:4		
filetype	FLR_SPLASHSCREEN_FILETYPE_E	4:8		

No output parameters.

1.25.3.3 *splashScreenSetBackground()*

Set the background color for the specified splash screen.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x001A0002	N/A	N/A	
screen_num	UINT_32	0:4		
backgroundColor	UINT_32	4:8		

No output parameters.

1.25.3.4 *splashScreenGetDuration()*

Get the duration that the specified splash screen will display.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x001A0003	N/A	N/A	
screen_num	UINT_32	0:4		

FLIR NEUTRINO LC SOFTWARE IDD

Output/Receive parameters:

Name	Data Type	Bytes	Notes
periodMs	UINT_32	0:4	

1.25.3.5 *splashScreenGetDataType()*

Get the data type of the specified splash screen.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x001A0004	N/A	N/A
screen_num	UINT_32	0:4	

Output/Receive parameters:

Name	Data Type	Bytes	Notes
filetype	FLR_SPLASHSCREEN_FILETYPE_E	0:4	

1.25.3.6 *splashScreenGetBackground()*

Get the background color of the specified splash screen.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x001A0005	N/A	N/A
screen_num	UINT_32	0:4	

Output/Receive parameters:

Name	Data Type	Bytes	Notes
backgroundColor	UINT_32	0:4	

1.26 Module: SYMBOLOGY

These APIs are used to manage the symbology of the camera for position, orientation, image type, etc.

1.26.1 Enums

1.26.1.1 *FLR_SYMBOLOGY_TEXT_ALIGNMENT_E* — <INT_16>

FLR_SYMBOLOGY_LEFT_TOP = 17
FLR_SYMBOLOGY_CENTER_TOP = 18
FLR_SYMBOLOGY_RIGHT_TOP = 19
FLR_SYMBOLOGY_LEFT_MIDDLE = 33
FLR_SYMBOLOGY_CENTER_MIDDLE = 34
FLR_SYMBOLOGY_RIGHT_MIDDLE = 35
FLR_SYMBOLOGY_LEFT_BOTTOM = 49
FLR_SYMBOLOGY_CENTER_BOTTOM = 50
FLR_SYMBOLOGY_RIGHT_BOTTOM = 51
FLR_SYMBOLOGY_ALIGNMENT_LAST = 64

1.26.1.2 *FLR_SYMBOLOGY_TRANSFORMATION_E* — <INT_16>

FLR_SYMBOLOGY_TRANSFORMATION_NONE = 0
FLR_SYMBOLOGY_TRANSFORMATION_FLIP_BOTH = 1
FLR_SYMBOLOGY_TRANSFORMATION_FLIP_HORIZONTAL = 2
FLR_SYMBOLOGY_TRANSFORMATION_FLIP_VERTICAL = 3

1.26.1.3 *FLR_SYMBOLOGY_IMAGE_TYPE_E* — <INT_16>

FLR_SYMBOLOGY_RAW_IMAGE = 0
FLR_SYMBOLOGY_PNG_IMAGE = 1
FLR_SYMBOLOGY_JPEG_IMAGE = 2
FLR_SYMBOLOGY_BMP_IMAGE = 3

1.26.1.4 *FLR_SYMBOLOGY_SCALING_MODE_E* — <INT_16>

FLR_SYMBOLOGY_SCALING_MODE_NONE = 0
FLR_SYMBOLOGY_SCALING_MODE_FIT = 1
FLR_SYMBOLOGY_SCALING_MODE_CROP = 2
FLR_SYMBOLOGY_SCALING_MODE_FILL = 3

1.26.2 Structs

No struct types in module symbology.

1.26.3 Functions

1.26.3.1 *symbologySetEnable()*

Enable / Disable symbol drawing.

Input/Send parameters:

FLIR NEUTRINO LC SOFTWARE IDD

Name	Data Type	Bytes	Notes
FunctionID	0x00140000	N/A	N/A
draw_symbols	FLR_ENABLE_E	0:4	

No output parameters.

1.26.3.2 *symbologyCreateBitmap()*

Create a bitmap symbol.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00140001	N/A	N/A
ID	UCHAR	0:1	
pos_X	INT_16	1:3	
pos_Y	INT_16	3:5	
width	INT_16	5:7	
height	INT_16	7:9	

No output parameters.

1.26.3.3 *symbologySendData()*

Send raw symbol data to specified symbol. Must declare "size" of data less than or equal to 128 bytes.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00140003	N/A	N/A
ID	UCHAR	0:1	
size	INT_16	1:3	
text	UCHAR*128	3:131	

No output parameters.

1.26.3.4 *symbologyCreateArc()*

Create an arc or ellipsoid symbol.

Input/Send parameters:

FLIR NEUTRINO LC SOFTWARE IDD

Name	Data Type	Bytes	Notes
FunctionID	0x00140004	N/A	N/A
ID	UCHAR	0:1	
pos_X	INT_16	1:3	
pos_Y	INT_16	3:5	
width	INT_16	5:7	
height	INT_16	7:9	
start_angle	FLOAT	9:13	
end_angle	FLOAT	13:17	
color	UINT_32	17:21	

No output parameters.

1.26.3.5 *symbolyCreateText()*

Create a text symbol.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00140006	N/A	N/A
ID	UCHAR	0:1	
pos_X	INT_16	1:3	
pos_Y	INT_16	3:5	
width	INT_16	5:7	
height	INT_16	7:9	
font	CHAR	9:10	
size	INT_16	10:12	
alignment	FLR_SYMBOLGY_TEXT_ALIGNME NT_E	12:14	
color	UINT_32	14:18	
text	UCHAR*128	18:146	

No output parameters.

1.26.3.6 *symbolyMoveSprite()*

Move an existing symbol to new absolute coordinates.

FLIR NEUTRINO LC SOFTWARE IDD

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x00140007	N/A	N/A	
ID	UCHAR	0:1		
pos_X	INT_16	1:3		
pos_Y	INT_16	3:5		

No output parameters.

1.26.3.7 *symbolologyAddToGroup()*

Add a symbol to a group.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x00140008	N/A	N/A	
ID	UCHAR	0:1		
group_ID	UCHAR	1:2		

No output parameters.

1.26.3.8 *symbolologyRemoveFromGroup()*

Remove a symbol from a group.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x00140009	N/A	N/A	
ID	UCHAR	0:1		
group_ID	UCHAR	1:2		

No output parameters.

1.26.3.9 *symbolologyUpdateAndShow()*

Redraw and show or hide the specified symbol.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x0014000A	N/A	N/A	

FLIR NEUTRINO LC SOFTWARE IDD

ID	UCHAR	0:1		
visible	UCHAR	1:2		

No output parameters.

1.26.3.10 *symbolologyUpdateAndShowGroup()*

Redraw and show or hide all symbols in a specified group.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x0014000B	N/A	N/A	
group_ID	UCHAR	0:1		
visible	UCHAR	1:2		

No output parameters.

1.26.3.11 *symbolologyDelete()*

Delete the specified symbol

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x0014000C	N/A	N/A	
ID	UCHAR	0:1		

No output parameters.

1.26.3.12 *symbolologyDeleteGroup()*

Delete all symbols from group.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x0014000D	N/A	N/A	
group_ID	UCHAR	0:1		

No output parameters.

1.26.3.13 *symbolologyCreateFilledRectangle()*

Create a solid color rectangle symbol.

FLIR NEUTRINO LC SOFTWARE IDD

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0014000E	N/A	N/A
ID	UCHAR	0:1	
pos_X	INT_16	1:3	
pos_Y	INT_16	3:5	
width	INT_16	5:7	
height	INT_16	7:9	
color	UINT_32	9:13	

No output parameters.

1.26.3.14 *symbologyCreateOutlinedRectangle()*

Create an outlined rectangle with transparent center.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00140010	N/A	N/A
ID	UCHAR	0:1	
pos_X	INT_16	1:3	
pos_Y	INT_16	3:5	
width	INT_16	5:7	
height	INT_16	7:9	
color	UINT_32	9:13	

No output parameters.

1.26.3.15 *symbologyCreateBitmapFromPng()*

Create a bitmap symbol from PNG encoded data.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00140012	N/A	N/A
ID	UCHAR	0:1	
pos_X	INT_16	1:3	
pos_Y	INT_16	3:5	

FLIR NEUTRINO LC SOFTWARE IDD

size	INT_16	5:7		
-------------	--------	-----	--	--

No output parameters.

1.26.3.16 *symbologyCreateCompressedBitmap()*

Create a bitmap symbol from RLE color data.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x00140014	N/A	N/A	
ID	UCHAR	0:1		
pos_X	INT_16	1:3		
pos_Y	INT_16	3:5		
width	INT_16	5:7		
height	INT_16	7:9		

No output parameters.

1.26.3.17 *symbologyCreateBitmapFromPngFile()*

Create a bitmap symbol from PNG file.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x00140016	N/A	N/A	
ID	UCHAR	0:1		
pos_X	INT_16	1:3		
pos_Y	INT_16	3:5		
path	UCHAR*128	5:133		

No output parameters.

1.26.3.18 *symbologyCreateBitmapFromFile()*

Create bitmap symbol from RAW file.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x00140017	N/A	N/A	

FLIR NEUTRINO LC SOFTWARE IDD

ID	UCHAR	0:1		
pos_X	INT_16	1:3		
pos_Y	INT_16	3:5		
path	UCHAR*128	5:133		
imageType	FLR_SYMBOLOGY_IMAGE_TYPE_E	133:135		

No output parameters.

1.26.3.19 *symbologyResetWritePosition()*

Reset write pointer for symbol ID.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x00140018	N/A	N/A	
ID	UCHAR	0:1		

No output parameters.

1.26.3.20 *symbologyMoveByOffset()*

Move an existing symbol by offset coordinates.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x00140019	N/A	N/A	
ID	UCHAR	0:1		
off_X	INT_16	1:3		
off_Y	INT_16	3:5		

No output parameters.

1.26.3.21 *symbologyMoveGroupByOffset()*

Move all symbols in a group by offset coordinates.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x0014001A	N/A	N/A	
ID	UCHAR	0:1		

FLIR NEUTRINO LC SOFTWARE IDD

off_X	INT_16	1:3		
off_Y	INT_16	3:5		

No output parameters.

1.26.3.22 *symbolologyCreateFilledEllipse()*

Create a solid color ellipsoid or circle.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x0014001B	N/A	N/A	
ID	UCHAR	0:1		
pos_X	INT_16	1:3		
pos_Y	INT_16	3:5		
width	INT_16	5:7		
height	INT_16	7:9		
color	UINT_32	9:13		

No output parameters.

1.26.3.23 *symbolologyCreateLine()*

Create a line symbol.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x0014001C	N/A	N/A	
ID	UCHAR	0:1		
pos_X	INT_16	1:3		
pos_Y	INT_16	3:5		
pos_X2	INT_16	5:7		
pos_Y2	INT_16	7:9		
color	UINT_32	9:13		

No output parameters.

1.26.3.24 *symbolologySetZorder()*

Set the Z coordinate for the specified symbol.

FLIR NEUTRINO LC SOFTWARE IDD

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0014001D	N/A	N/A
ID	UCHAR	0:1	
zorder	UCHAR	1:2	

No output parameters.

1.26.3.25 *symbologySaveConfiguration()*

Save current symbol definitions, positions, and visibility to config file.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0014001E	N/A	N/A

No output parameters.

1.26.3.26 *symbologyReloadConfiguration()*

Load symbol definitions, positions, and visibility from config file.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0014001F	N/A	N/A

No output parameters.

1.26.3.27 *symbologyGetEnable()*

Get symbol drawing enable status.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00140020	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
draw_symbols	FLR_ENABLE_E	0:4	

FLIR NEUTRINO LC SOFTWARE IDD

1.26.3.28 *symbologySetClonesNumber()*

Set the number of clones for the specified symbol.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00140021	N/A	N/A
ID	UCHAR	0:1	
numberOfClones	UCHAR	1:2	

No output parameters.

1.26.3.29 *symbologyMoveCloneByOffset()*

Move specified clone of specified symbol by offset coordinates.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00140022	N/A	N/A
ID	UCHAR	0:1	
cloneID	UCHAR	1:2	
pos_X	INT_16	2:4	
pos_Y	INT_16	4:6	

No output parameters.

1.26.3.30 *symbologyMoveCloneSprite()*

Move specified clone of specified symbol to absolute coordinates.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00140023	N/A	N/A
ID	UCHAR	0:1	
cloneID	UCHAR	1:2	
pos_X	INT_16	2:4	
pos_Y	INT_16	4:6	

No output parameters.

FLIR NEUTRINO LC SOFTWARE IDD

1.26.3.31 *symbologySetTransformation()*

Change the global symbol transformation enumeration.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00140024	N/A	N/A
transformation	FLR_SYMBOLOLOGY_TRANSFORMATION_E	0:2	

No output parameters.

1.26.3.32 *symbologyUpdateAllVisible()*

Update and redraw all currently visible symbols.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00140025	N/A	N/A

No output parameters.

1.26.3.33 *symbologySetSizeAndScalingMode()*

Set desired size and scaling mode for specified symbol.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00140026	N/A	N/A
ID	UCHAR	0:1	
width	INT_16	1:3	
height	INT_16	3:5	
scalingMode	FLR_SYMBOLOLOGY_SCALING_MODE_E	5:7	

No output parameters.

1.26.3.34 *symbologyCreateLineHVT()*

Create HVT line symbol.

Input/Send parameters:

FLIR NEUTRINO LC SOFTWARE IDD

Name	Data Type	Bytes	Notes
FunctionID	0x00140027	N/A	N/A
ID	UCHAR	0:1	
pos_X	INT_16	1:3	
pos_Y	INT_16	3:5	
pos_X2	INT_16	5:7	
pos_Y2	INT_16	7:9	
color1	UINT_32	9:13	
color2	UINT_32	13:17	
dashLen	UINT_16	17:19	
thickness	UINT_16	19:21	

No output parameters.

1.26.3.35 *symbolyCreateTextHVT()*

Create HVT text symbol.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00140028	N/A	N/A
ID	UCHAR	0:1	
pos_X	INT_16	1:3	
pos_Y	INT_16	3:5	
width	INT_16	5:7	
height	INT_16	7:9	
font	CHAR	9:10	
size	INT_16	10:12	
alignment	FLR_SYMBOLGY_TEXT_ALIGNME NT_E	12:14	
color1	UINT_32	14:18	
color2	UINT_32	18:22	
dashLen	UCHAR	22:23	
text	UCHAR*128	23:151	

No output parameters.

FLIR NEUTRINO LC SOFTWARE IDD

1.26.3.36 *symbolologyCreateTextBg()*

Create text with background color symbol.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00140029	N/A	N/A
ID	UCHAR	0:1	
pos_X	INT_16	1:3	
pos_Y	INT_16	3:5	
width	INT_16	5:7	
height	INT_16	7:9	
font	CHAR	9:10	
size	INT_16	10:12	
alignment	FLR_SYMBOLOGY_TEXT_ALIGNME NT_E	12:14	
color	UINT_32	14:18	
bgColor	UINT_32	18:22	
text	UCHAR*128	22:150	

No output parameters.

1.26.3.37 *symbolologyCreateScaledBitmapFromFile()*

Create bitmap with scaling from file pointer.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x0014002A	N/A	N/A
ID	UCHAR	0:1	
pos_X	INT_16	1:3	
pos_Y	INT_16	3:5	
width	INT_16	5:7	
height	INT_16	7:9	
scalingMode	FLR_SYMBOLOGY_SCALING_MODE_ E	9:11	
path	UCHAR*128	11:139	
imageType	FLR_SYMBOLOGY_IMAGE_TYPE_E	139:141	

No output parameters.

1.27 Module: SYSCTRL

General Pipeline controls

1.27.1 Enums

1.27.1.1 *FLR_SYSCTRL_USBIR16_MODE_E* — <INT_32>

FLR_SYSCTRL_USBIR16_MODE_16 = 0
 FLR_SYSCTRL_USBIR16_MODE_14 = 1
 FLR_SYSCTRL_USBIR16_MODE_TLINEAR = 2
 FLR_SYSCTRL_USBIR16_MODE_LAST = 3

1.27.1.2 *FLR_SYSCTRL_OPERATING_MODE_E* — <INT_32>

FLR_SYSCTRL_MODE_UNKNOWN = -1
 FLR_SYSCTRL_MODE_STARTUP = 0
 FLR_SYSCTRL_MODE_NORMAL_IMAGING = 1
 FLR_SYSCTRL_MODE_UPDATE = 2
 FLR_SYSCTRL_MODE_LOW_POWER_IMAGING = 3
 FLR_SYSCTRL_MODE_LOW_POWER = 4
 FLR_SYSCTRL_MODE_TESTRAMP = 5
 FLR_SYSCTRL_MODE_LENS_GAIN_CAL = 6
 FLR_SYSCTRL_MODE_SFFC_CAL = 7
 FLR_SYSCTRL_MODE_END = 8

1.27.2 Structs

No struct types in module sysctrl.

1.27.3 Functions

1.27.3.1 *sysctrlSetFreezeState()*

Sets the state of the pipeline freeze parameter (enable/disable)

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000E0001	N/A	N/A
data	FLR_ENABLE_E	0:4	

No output parameters.

FLIR NEUTRINO LC SOFTWARE IDD

1.27.3.2 *sysctrlGetFreezeState()*

Gets the state of the pipeline freeze parameter (enable/disable)

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000E0002	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLR_ENABLE_E	0:4	

1.27.3.3 *sysctrlGetCameraFrameRate()*

Get the framerate of the camera in frames per second (60/30 or 9).

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000E0007	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
frameRate	UINT_32	0:4	

1.27.3.4 *sysctrlGetUptimeSecs()*

Gets the current uptime of the camera in seconds.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000E0008	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
uptime	UINT_32	0:4	

1.27.3.5 *sysctrlSetUsbVideoIR16Mode()*

Data packing for USB Video sent via 'Y16' format

FLIR NEUTRINO LC SOFTWARE IDD

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000E000D	N/A	N/A
data	FLR_SYSCTRL_USBIR16_MODE_E	0:4	

No output parameters.

1.27.3.6 *sysctrlGetUsbVideoIR16Mode()*

Data packing for USB Video sent via 'Y16' format

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000E000E	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLR_SYSCTRL_USBIR16_MODE_E	0:4	

1.27.3.7 *sysctrlSetOperatingMode()*

Configures system parameters including video pipeline (when applicable) for the selected mode.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000E000F	N/A	N/A
data	FLR_SYSCTRL_OPERATING_MODE_E	0:4	

No output parameters.

1.27.3.8 *sysctrlGetOperatingMode()*

Configures system parameters including video pipeline (when applicable) for the selected mode.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000E0010	N/A	N/A

FLIR NEUTRINO LC SOFTWARE IDD

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLR_SYSCTRL_OPERATING_MODE_E	0:4	

1.27.3.9 *sysctrlGetAvgFpaTempCounts()*

The average temperature of the FPA in counts.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000E0018	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLOAT	0:4	

1.27.3.10 *sysctrlSetFpaTempFrames()*

The number of frames to average the FPA temperature over.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000E0019	N/A	N/A
data	UINT_16	0:2	

No output parameters.

1.27.3.11 *sysctrlGetFpaTempFrames()*

The number of frames to average the FPA temperature over.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000E0020	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
------	-----------	-------	-------

FLIR NEUTRINO LC SOFTWARE IDD

data	UINT_16	0:2		
------	---------	-----	--	--

1.28 Module: SYSINFO

These APIs are used to return System Information. Currently only Monitor information is returned.

1.28.1 Enums

1.28.1.1 *FLR_SYSINFO_SW_CONFIG_ID_E* — <INT_32>

FLR_SYSINFO_UNKNOWN = 0
FLR_SYSINFO_BOSON_1406 = 1
FLR_SYSINFO_BOSON_1407 = 2
FLR_SYSINFO_BOSON_1403 = 3
FLR_SYSINFO_BOSON_BLENDED = 4
FLR_SYSINFO_BOSON_OTIS = 5
FLR_SYSINFO_BOSON_MARITIME = 6
FLR_SYSINFO_NV4 = 7
FLR_SYSINFO_BOSON_SWIR = 8
FLR_SYSINFO_SENSORTEST_1406 = 9
FLR_SYSINFO_SENSORTEST_1407 = 10
FLR_SYSINFO_NEUTRINOLC = 11
FLR_SYSINFO_GLUON_1406 = 12
FLR_SYSINFO_GLUON_1407 = 13
FLR_SYSINFO_GRAVITON_1407 = 14
FLR_SYSINFO_BOSON_1801 = 15
FLR_SYSINFO_BOSON_OUTLINE_1406 = 16
FLR_SYSINFO_GRAVITON_1406 = 17

1.28.1.2 *FLR_SYSINFO_SW_PERMISSIONS_E* — <INT_32>

FLR_SYSINFO_FACTORY = 0
FLR_SYSINFO_USER = 1

1.28.2 Structs

1.28.2.1 *FLR_SYSINFO_MONITOR_BUILD_VARIANT_T*

Field Name	Data Type	Bytes
value	UCHAR*50	50

FLIR NEUTRINO LC SOFTWARE IDD

1.28.3 Functions

1.28.3.1 *sysinfoGetMonitorSoftwareRev()*

Get the software revision of the bootloader code.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x002F0001	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
major	UINT_32	0:4	
minor	UINT_32	4:8	
patch	UINT_32	8:12	

1.28.3.2 *sysinfoGetMonitorBuildVariant()*

Get the bootloader variant name.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x002F0002	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
monitorBuildVariant	FLR_SYSINFO_MONITOR_BUILD_VARIANT_T	0:50	

1.28.3.3 *sysinfoGetProductName()*

Device name.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x002F0003	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
------	-----------	-------	-------

FLIR NEUTRINO LC SOFTWARE IDD

name	UCHAR*128	0:128		
-------------	-----------	-------	--	--

1.28.3.4 *sysinfoGetCameraSN()*

Camera serial number.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x002F0005	N/A	N/A	

Output/Receive parameters:

Name	Data Type	Bytes	Notes	
number	UCHAR*128	0:128		

1.28.3.5 *sysinfoGetBootLocation()*

Boot Sw Location.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x002F0006	N/A	N/A	

Output/Receive parameters:

Name	Data Type	Bytes	Notes	
bootSwLocation	UINT_32	0:4		

1.28.3.6 *sysinfoGetSwConfigID()*

Sw config id.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x002F0007	N/A	N/A	

Output/Receive parameters:

Name	Data Type	Bytes	Notes	
swConfigID	FLR_SYSINFO_SW_CONFIG_ID_E	0:4		

FLIR NEUTRINO LC SOFTWARE IDD

1.28.3.7 *sysinfoGetSwPermissions()*

Is Software factory or User.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x002F0008	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
swPermissions	FLR_SYSINFO_SW_PERMISSIONS_E	0:4	

1.28.3.8 *sysinfoGetProductVersion()*

Returns the friendly version number of the product configuration.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x002F000A	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
major	UINT_32	0:4	
minor	UINT_32	4:8	
patch	UINT_32	8:12	

1.28.3.9 *sysinfoGetMonitorProductRev()*

Get the product revision of the bootloader code.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x002F000F	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
------	-----------	-------	-------

FLIR NEUTRINO LC SOFTWARE IDD

major	UINT_32	0:4		
minor	UINT_32	4:8		
patch	UINT_32	8:12		

1.29 Module: SYSTEMSYMBOLS

These APIs are used to control/manage the system symbols

1.29.1 Enums

1.29.1.1 FLR_SYSTEMSYMBOLS_SYMBOL_E — <INT_32>

FLR_SYSTEMSYMBOLS_FFC_IMMINEENT = 0
 FLR_SYSTEMSYMBOLS_FFC_DESIREED = 1
 FLR_SYSTEMSYMBOLS_TABLE_SWITCH_DESIREED = 2
 FLR_SYSTEMSYMBOLS_LOW_GAIN = 3
 FLR_SYSTEMSYMBOLS_OVERTEMP = 4
 FLR_SYSTEMSYMBOLS_SYMBOL_LAST = 5

1.29.1.2 FLR_SYSTEMSYMBOLS_ID_TYPE_E — <INT_32>

FLR_SYSTEMSYMBOLS_ELEMENT = 0
 FLR_SYSTEMSYMBOLS_GROUP = 1
 FLR_SYSTEMSYMBOLS_ID_LAST = 2

1.29.1.3 FLR_SYSTEMSYMBOLS_STATE_E — <INT_32>

FLR_SYSTEMSYMBOLS_ENTERED = 0
 FLR_SYSTEMSYMBOLS_EXITED = 1
 FLR_SYSTEMSYMBOLS_STATE_LAST = 2

1.29.2 Structs

No struct types in module systemSymbols.

1.29.3 Functions

1.29.3.1 systemSymbolsGetID()

Get the ID associated with the specified symbol enumeration.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x001B0002	N/A	N/A
symbol	FLR_SYSTEMSYMBOLS_SYMBOL_E	0:4	

FLIR NEUTRINO LC SOFTWARE IDD

Output/Receive parameters:

Name	Data Type	Bytes	Notes
id	UCHAR	0:1	
id_type	FLR_SYSTEMSYMBOLS_ID_TYPE_E	1:5	

1.29.3.2 *systemSymbolsSetID()*

Set the ID associated with the specified symbol enumeration

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x001B0003	N/A	N/A
symbol	FLR_SYSTEMSYMBOLS_SYMBOL_E	0:4	
id	UCHAR	4:5	
id_type	FLR_SYSTEMSYMBOLS_ID_TYPE_E	5:9	

No output parameters.

1.29.3.3 *systemSymbolsGetEnable()*

Get the enable state for the specified symbol enumeration.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x001B0004	N/A	N/A
symbol	FLR_SYSTEMSYMBOLS_SYMBOL_E	0:4	

Output/Receive parameters:

Name	Data Type	Bytes	Notes
enabled	FLR_ENABLE_E	0:4	

1.29.3.4 *systemSymbolsSetEnable()*

Set the enable state for the specified symbol enumeration.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x001B0005	N/A	N/A

FLIR NEUTRINO LC SOFTWARE IDD

symbol	FLR_SYSTEMSYMBOLS_SYMBOL_E	0:4		
enabled	FLR_ENABLE_E	4:8		

No output parameters.

1.30 Module: TELEMETRY

Boson provides the option to enable a single line of telemetry as either the first or last line in each frame. The telemetry line contains metadata describing the image stream and the camera. A complete list of the telemetry-line contents is provided in the Appendix <?>. All telemetry is aligned to 16-bit fields. If CMOS is configured for 8-bit output, only the 8 LSBs will be provided. If CMOS is configured to 24bit output, telemetry data will be provided via `cmos_data[0:15]`.

1.30.1 Enums

1.30.1.1 *FLR_TELEMETRY_LOC_E* — <INT_32>

FLR_TELEMETRY_LOC_TOP = 0
FLR_TELEMETRY_LOC_BOTTOM = 1
FLR_TELEMETRY_LOC_END = 2

1.30.1.2 *FLR_TELEMETRY_PACKING_E* — <INT_32>

FLR_TELEMETRY_PACKING_DEFAULT = 0
FLR_TELEMETRY_PACKING_Y = 1
FLR_TELEMETRY_PACKING_8BITS = 2
FLR_TELEMETRY_PACKING_END = 3

1.30.1.3 *FLR_TELEMETRY_ORDER_E* — <INT_32>

FLR_TELEMETRY_ORDER_DEFAULT = 0
FLR_TELEMETRY_ORDER_SWAP16B = 1
FLR_TELEMETRY_ORDER_END = 2

1.30.2 Structs

No struct types in module telemetry.

1.30.3 Functions

1.30.3.1 *telemetrySetState()*

Set the telemetry state to Enabled or Disabled.

Input/Send parameters:

102-2020-42, Neutrino LC Software Interface Description Document, Rev400

Information on this page is subject to change without notice

This document does not contain any export-controlled information.

FLIR NEUTRINO LC SOFTWARE IDD

Name	Data Type	Bytes	Notes
FunctionID	0x00040001	N/A	N/A
data	FLR_ENABLE_E	0:4	

No output parameters.

1.30.3.2 *telemetryGetState()*

Return the current telemetry state.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00040002	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLR_ENABLE_E	0:4	

1.30.3.3 *telemetrySetLocation()*

Set the telemetry to before(top) or after(bottom) the image.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00040003	N/A	N/A
data	FLR_TELEMETRY_LOC_E	0:4	

No output parameters.

1.30.3.4 *telemetryGetLocation()*

Return the current telemetry location.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00040004	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLR_TELEMETRY_LOC_E	0:4	

FLIR NEUTRINO LC SOFTWARE IDD

1.30.3.5 *telemetrySetPacking()*

Sets the type of packing that the telemetry data is presented - 16 Bit, Color or 8 -Bit

Input/Send parameters:

Name	DataType	Bytes	Notes
FunctionID	0x00040005	N/A	N/A
data	FLR_TELEMETRY_PACKING_E	0:4	

No output parameters.

1.30.3.6 *telemetryGetPacking()*

Gets the telemetry packing format.

Input/Send parameters:

Name	DataType	Bytes	Notes
FunctionID	0x00040006	N/A	N/A

Output/Receive parameters:

Name	DataType	Bytes	Notes
data	FLR_TELEMETRY_PACKING_E	0:4	

1.30.3.7 *telemetrySetOrder()*

The half-word (16 bits) order for telemetry data.

Input/Send parameters:

Name	DataType	Bytes	Notes
FunctionID	0x00040007	N/A	N/A
data	FLR_TELEMETRY_ORDER_E	0:4	

No output parameters.

1.30.3.8 *telemetryGetOrder()*

The half-word (16 bits) order for telemetry data.

Input/Send parameters:

Name	DataType	Bytes	Notes
------	----------	-------	-------

FLIR NEUTRINO LC SOFTWARE IDD

FunctionID	0x00040008	N/A	N/A	
-------------------	------------	-----	-----	--

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLR_TELEMETRY_ORDER_E	0:4	

1.31 Module: TESTRAMP

The test ramp can be used to replace the live video feed for calibration and error checking of the camera. The actual enable switch is located in the GAO module.

1.31.1 Enums

1.31.1.1 FLR_TESTRAMP_TYPE_E -- <INT_32>

```

FLR_TESTRAMP_ZERO = 0
FLR_TESTRAMP_INCREMENTING = 1
FLR_TESTRAMP_VERT_SHADE = 2
FLR_TESTRAMP_HORIZ_SHADE = 3
FLR_TESTRAMP_BIG_VERT_SHADE = 4
FLR_TESTRAMP_SIMPLE_VERTICAL = 5
FLR_TESTRAMP_VTST_CHECKERBOARD = 6
FLR_TESTRAMP_VTST_DIAGONAL_STRIPE = 7
FLR_TESTRAMP_VTST_MOVING_LINE_BLACK = 8
FLR_TESTRAMP_VTST_DIAGONAL_LR = 9
FLR_TESTRAMP_VTST_DIAGONAL_RL = 10
FLR_TESTRAMP_PN9_FILL = 11
FLR_TESTRAMP_HORIZ_BARS = 12
FLR_TESTRAMP_VERT_BARS = 13
FLR_TESTRAMP_BPR_MAP = 14
FLR_TESTRAMP_CORN_2_CORN = 15
FLR_TESTRAMP_PN9_CONTINUOUS = 16
FLR_TESTRAMP_TYPE_LAST = 17
    
```

1.31.2 Structs

1.31.2.1 FLR_TESTRAMP_SETTINGS_T

Field Name	Data Type	Bytes
start	UINT_16	2
end	UINT_16	2
increment	UINT_16	2

102-2020-42, Neutrino LC Software Interface Description Document, Rev400

Information on this page is subject to change without notice

This document does not contain any export-controlled information.



1.31.3 Functions

1.31.3.1 *testRampSetType()*

Set the selected test ramp buffer to one of the pre-configured patterns. The simulated video frame is redrawn on set.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00100000	N/A	N/A
index	UCHAR	0:1	
data	FLR_TESTRAMP_TYPE_E	1:5	

No output parameters.

1.31.3.2 *testRampGetType()*

Get the selected test ramp buffer's current pattern type.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00100001	N/A	N/A
index	UCHAR	0:1	

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLR_TESTRAMP_TYPE_E	0:4	

1.31.3.3 *testRampSetSettings()*

Change the selected buffer's ramp settings. The buffer is redrawn on set. At present, the "Incrementing" pattern is the only configurable ramp.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00100002	N/A	N/A
index	UCHAR	0:1	
data	FLR_TESTRAMP_SETTINGS_T	1:7	

FLIR NEUTRINO LC SOFTWARE IDD

No output parameters.

1.31.3.4 *testRampGetSettings()*

Get the selected test ramp buffer's current ramp settings.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00100003	N/A	N/A
index	UCHAR	0:1	

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLR_TESTRAMP_SETTINGS_T	0:6	

1.31.3.5 *testRampSetMotionState()*

Enable or disable looping through the test ramp buffers. If the Boson is configured with more than one test ramp: the video will display each ramp buffer once, then repeat.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00100004	N/A	N/A
data	FLR_ENABLE_E	0:4	

No output parameters.

1.31.3.6 *testRampGetMotionState()*

Determine whether the test ramp is in motion or still mode.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00100005	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLR_ENABLE_E	0:4	

FLIR NEUTRINO LC SOFTWARE IDD

1.31.3.7 *testRampSetIndex()*

Display the selected ramp buffer on the next frame.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00100006	N/A	N/A
data	UCHAR	0:1	

No output parameters.

1.31.3.8 *testRampGetIndex()*

Get the ramp buffer index that will be displayed on the next frame.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00100007	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	UCHAR	0:1	

1.31.3.9 *testRampGetMaxIndex()*

Determine the last valid index for a ramp buffer. MaxIndex=1 or two buffers is the default configuration.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00100008	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	UCHAR	0:1	

1.32 Module: TF

The Temporal Filter module provides API's to control or interrogate the functionality of the temporal noise filter.

FLIR NEUTRINO LC SOFTWARE IDD

1.32.1 Enums

1.32.1.1 *FLR_TF_MOTION_MODE_E* — <INT_32>

FLR_TF_MOTION_MODE_FRAME_BASED = 0
FLR_TF_MOTION_MODE_MOTION_BASED = 1
FLR_TF_MOTION_MODE_END = 2

1.32.2 Structs

1.32.2.1 *FLR_TF_WLUT_T*

Field Name	DataType	Bytes
value	UCHAR*32	32

1.32.2.2 *FLR_TF_NF_LUT_T*

Field Name	DataType	Bytes
value	UINT_16*17	34

1.32.2.3 *FLR_TF_TEMP_SIGNAL_COMP_FACTOR_LUT_T*

Field Name	DataType	Bytes
value	UINT_16*17	34

1.32.3 Functions

1.32.3.1 *tfSetEnableState()*

Enable or disable Temporal Noise Reduction (tnr)

Input/Send parameters:

Name	DataType	Bytes	Notes
FunctionID	0x000A0001	N/A	N/A
data	FLR_ENABLE_E	0:4	

No output parameters.

1.32.3.2 *tfGetEnableState()*

Get Temporal Noise Reduction (tnr) correction's current enable state.

Input/Send parameters:

102-2020-42, Neutrino LC Software Interface Description Document, Rev400

Information on this page is subject to change without notice

This document does not contain any export-controlled information.



FLIR NEUTRINO LC SOFTWARE IDD

Name	Data Type	Bytes	Notes
FunctionID	0x000A0002	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLR_ENABLE_E	0:4	

1.32.3.3 *tfSetDelta_nf()*

Sets the Delta NF value. The delta_nf modifies the filter behavior by scaling the index into the table of weights (wLUT)

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000A0003	N/A	N/A
data	UINT_16	0:2	

No output parameters.

1.32.3.4 *tfGetDelta_nf()*

Gets the Delta NF value. The delta_nf modifies the filter behavior by scaling the index into the table of weights (wLUT)

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000A0004	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	UINT_16	0:2	

1.32.3.5 *tfSetTHDeltaMotion()*

Sets the Delta Motion threshold. The Delta Motion specifies a threshold to determine if there was motion in the scene.

Input/Send parameters:

Name	Data Type	Bytes	Notes
------	-----------	-------	-------

FLIR NEUTRINO LC SOFTWARE IDD

FunctionID	0x000A0005	N/A	N/A	
data	UINT_16	0:2		

No output parameters.

1.32.3.6 *tfGetTHDeltaMotion()*

Gets the Delta Motion threshold. The Delta Motion specifies a threshold to determine if there was motion in the scene.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x000A0006	N/A	N/A	

Output/Receive parameters:

Name	Data Type	Bytes	Notes	
data	UINT_16	0:2		

1.32.3.7 *tfSetWLut()*

Sets the values in the Table of Weights - (wLUT) . The weight table specifies the ration of the averaging of the current with the previous frame.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x000A0007	N/A	N/A	
data	FLR_TF_WLUT_T	0:32		

No output parameters.

1.32.3.8 *tfGetWLut()*

Gets the values in the Table of Weights - (wLUT) . The weight table specifies the ration of the averaging of the current with the previous frame.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x000A0008	N/A	N/A	

Output/Receive parameters:

FLIR NEUTRINO LC SOFTWARE IDD

Name	Data Type	Bytes	Notes
data	FLR_TF_WLUT_T	0:32	

1.32.3.9 *tfGetMotionCount()*

Gets the current motion count from the camera. The motion count is the number of pixels in the image that is classified as have moved from the previous frame

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000A0009	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	UINT_32	0:4	

1.32.3.10 *tfSetMotionThreshold()*

Sets the motion detection threshold. If the number of pixels in a frame detected as having moved exceeds this threshold.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000A000E	N/A	N/A
data	UINT_32	0:4	

No output parameters.

1.32.3.11 *tfGetMotionThreshold()*

Gets the motion detection threshold. If the number of pixels in a frame detected as having moved exceeds this threshold, the frame is considered to have motion.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x000A000F	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
------	-----------	-------	-------

FLIR NEUTRINO LC SOFTWARE IDD

data	UINT_32	0:4		
-------------	---------	-----	--	--

1.33 Module: UART

UART Device on Myriad SOC

1.33.1 Enums

1.33.1.1 *FLR_UART_STARTUP_BAUDRATE_E* — <INT_32>

FLR_UART_921600_BAUD = 0
 FLR_UART_460800_BAUD = 1
 FLR_UART_230400_BAUD = 2
 FLR_UART_115200_BAUD = 3
 FLR_UART_57600_BAUD = 4
 FLR_UART_38400_BAUD = 5
 FLR_UART_19200_BAUD = 6
 FLR_UART_14400_BAUD = 7
 FLR_UART_9600_BAUD = 8
 FLR_UART_4800_BAUD = 9
 FLR_UART_2400_BAUD = 10
 FLR_UART_1200_BAUD = 11
 FLR_UART_600_BAUD = 12
 FLR_UART_300_BAUD = 13
 FLR_UART_110_BAUD = 14
 FLR_UART_BAUDRATE_END = 15

1.33.2 Structs

No struct types in module uart.

1.33.3 Functions

1.33.3.1 *uartSetStartupBaudRate()*

Start up baudrate for UART.

Input/Send parameters:

Name	Data Type	Bytes	Notes	
FunctionID	0x00400000	N/A	N/A	
data	FLR_UART_STARTUP_BAUDRATE_ E	0:4		

FLIR NEUTRINO LC SOFTWARE IDD

No output parameters.

1.33.3.2 *uartGetStartupBaudRate()*

Start up baudrate for UART.

Input/Send parameters:

Name	Data Type	Bytes	Notes
FunctionID	0x00400001	N/A	N/A

Output/Receive parameters:

Name	Data Type	Bytes	Notes
data	FLR_UART_STARTUP_BAUDRATE_ E	0:4	