Nova Mars Series of LED Display Control Systems SDK User Manual

NovaStar Tech Co., Ltd

Ver: 1.5.2

Release: 2016-04-20

http://www.novastar-led.com/

Content

1 Overview

1.1Contents of the SDK

The SDK is for Nova Mars series of LED display control systems. And the SDK contains the files listed in the following table.

No.	File Name	Description
1	Nova.Mars.SDK.MarsInterface.dll	The dynamic link library (DLL) file based on which users can customize their applications.
2	NovaEncode64.dll	Copy the DLL files to the path of the .exe file when use the SDK for customized apps developing.
3	NovaEncode.dll	
4	The other .dll librarys	Underlying DLL files. Copy the files to the folder in which Nova.Mars.SDK.MarsInterface.dll is in when use. No references are required.
5	Server	This folder contains the files for the series port communication service with the Mars control systems. Copy the folder to the path of the .exe file when use the SDK for customized apps developing.
6	CommonData	This folder contains the Mars control systems chips lists files. Copy the folder to the path of the .exe file when use the SDK for customized apps developing.
7	Sample	A C# sample for interfaces communication test.

1.2Development Environment Requirements

The SDK is provided in the form of DLL files. Visual Studio 2005 and late are recommended as the development environment. It is better if developers have the experience in C# programming as only C# samples are available at present.

1.3Important notes for SDK usage

No.	Notes

1	The **Server** folder must be copied to the path of the exe file when use the SDK.
2	The **CommonData** folder must be copied to the path of the exe file when use the SDK.
3	The SDK can only be used for 32-bit application developing.
4	The **NovaEncode64.dll** and **NovaEncode.dll** must be copied to the path of the exe file when use the SDK.
5	The SDK does not support redundant transmit cards (controllers) RJ45 Ethernet ports.

2 DLL Interfaces

2.1MarsHardwareEnumerator

This class is used to get the IDs of the devices (LED display control systems and multifunction cards) in the overall system. The IDs are the only way to get access to the devices. Functions provided by this class are as follow.

- Initialize the object (of this class) itself.
- Release the object resources.
- Get the count of the LED display control systems connected to the computer.
- Get the count of the multifunction cards connected to the computer through serial ports.
- Get the corresponding serial port name according to the index of the control system which is connected to the computer through the serial port.
- Get the corresponding serial port name according to the index of the multifunction card which is connected to the computer through the serial port.

2.1.1 Initialize

Description

Initialize the object (of this class) itself.

Remark

Only when the initialization operation is performed successfully will the other functions be workable.

2.1.2 UnInitialize

Description

Release the resources of the object (of this class).

Remark

After the operation of uninitializatize is performed successfully, all other functions will return False when called.

2.1.3 CtrlSystemCount

Description

Get the count of the LED display control systems connected to the computer.

Remark

The return value will be 0 if the object of this class has not initialized itself or has uninitialized itself.

2.1.4 FuncCardInCommCount

Description

Get the count of the multifunction cards connected to the computer through serial ports.

Remark

The return value will be 0 if the object of this class has not initialized itself or has uninitialized itself.

2.1.5 GetComNameOfControlSystem

Description

Get the corresponding serial port name (ID) of a certain LED display control system according to the control system index.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	index	Control system index	From 0 to (CS_Count-1) where CS_Count is the count of the control systems.
2	string	comName	The retrieved serial port name (ID)	

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.1.6 Get Com Name Of Function Card

Description

Get the corresponding serial port name (ID) of a certain multifunction card according to the multifunction card index.

Parameter

No.	Type	Para Name	Description	Value Range
1	int	index	Multifunction card index	From 0 to (MC_Count-1) where MC_Count is the count of the multifunction cards.
2	string	comName	The retrieved serial port name (ID)	

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.2MarsFunctionCardInCOM

This class is used to manage the multifunction cards which are connected to the system through serial ports. Functions provided by this class are as follow.

- Set or get the power supply control mode of a multifunction card.
- Set or get the power supplies status of a multifunction card.
- Set or get the schedules in a multifunction card for automatic power supplies control
- Get the measurement results of the light sensors connected to a multifunction card.

2.2.1 SetPowerControlMode

Description

Set the power supply control mode of a multifunction card.

Remark

If the mode is set to be Auto, a schedule will be needed by the multiunction card for automatic power supply control. Use the **SetPowerSwitchAutoTime** method to set the schedule for a multifunction card of which the schedule has not been set.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	funcCardIndex	Multifunction card index	From 0 to (MC_Count-1) where MC_Count is the number of multifunction cards connected with the system through serial ports.
2	Nova.Mars.SDK.PowerControlMode	ctrlMode	Power supply control mode	3.1.6 PowerControlMode

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.2.2 SetPowerSwitchStatus

Description

Set the status (on/off) of a certain power supply output on a multifunction card.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	funcCardIndex	Multifunction card index	From 0 to (MC_Count -1) where MC_Count is the number of multifunction cards connected with the system through serial ports.
2	int	switchIndex	Power supply index	0～7
3	Nova.Mars.SDK.PowerSwitchStatus	switchStatus	Power supply status	Refer to 3.1.7 PowerSwitchStatus for more details.

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.2.3 SetPowerAllSwitchStatus

Description

Set the status (on/off) of all power supply outputs on a multifunction card.

Remark

The schedule for automatic power supply control will be disabled when all power supplies are turned off. The schedule won’t be activated untill all power supplies are turned on by calling this method again.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	funcCardIndex	Multifunction card index	From 0 to (MC_Count -1) where MC_Count is the number of multifunction cards

				connected with the system through serial ports.
2	Nova.Mars.SDK.PowerSwitchStatus	switchStatus	Power supply status	Refer to 3.1.7 PowerSwitchStatus for more details.

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.2.4 SetPowerSwitchAutoTime

Description

Set the schedule for automatic control of the power supply outputs on a multifunction card.

Remark

The schedule is workable only when the power supply control mode is set as Auto.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	funcCardIndex	Multifunction card index	From 0 to (MC_Count -1) where MC_Count is the number of multifunction cards connected with the system through serial ports.
2	DateTime[]	startTime	The time to turn on all 8 power supply outputs.	The array length is 8.
3	DateTime[]	StopTime	The time to turn off all 8 power supply outputs.	The array length is 8.

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.2.5 GetPowerControlMode

Description

Get the power supply control mode of a multifunction card.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	funcCardIndex	Multifunction card index	From 0 to (MC_Count -1) where MC_Count is the number of multifunction cards connected with the system through serial ports.
2	Nova.Mars.SDK.Power Contro lMode	ctrlMode	The retrieved power supply control mode.	Refer to 3.1.6Power Control Mode for more details.

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.2.6 GetPowerSwitchStatus

Description

Get the status of a certain power supply output on a multifunction card.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	funcCardIndex	Multifunction card index	From 0 to (MC_Count -1) where MC_Count is the number of multifunction cards connected with the system through serial ports.
2	int	switchIndex	Power supply index	0～7

3	Nova.Mars.SDK.PowerSwitchStatus	switchStatus	The retrieved power supply status.	Refer to 3.1.7PowerSwitchStatus for more details.

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.2.7 GetPowerSwitchAutoTime

Description

Get the schedule for automatic power supplies control from the multifunction card. As there are 8 power supply outputs on a multifunction card, the lengths of the startTime array and the stopTime array are both 8.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	funcCardIndex	Multifunction card index	From 0 to (MC_Count -1) where MC_Count is the number of multifunction cards connected with the system through serial ports.
2	DateTime[]	startTime	The retrieved start time of the power supplies.	The array length is 8.
3	DateTime[]	StopTime	The retrieved stop time of the power supplies.	The array length is 8.

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.2.8 GetValueOfLightSensorInFunction

Description

Get the measurement result of a certain light sensor connected to a multifunction card. Note that a multifunction card has 6 interfaces for connecting external devices, but the external devices can only be light sensors at present.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	funcCardIndex	Multifunction card index	From 0 to (MC_Count -1) where MC_Count is the number of multifunction cards connected with the system through serial ports.
2	int	devIndex	External device index	0～5
3	int	lux	The retrieved measurement result of a light sensor.	

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.3MarsControlSystem

This class is used for LED display control system cluster management. Functions provided by this class are as follow.

- Initial the devices connected with a certain serial port.
- Release the resources of an object of this class.
- Get the reverver cards (scan-boards) information of a certain LED display.
- Set the location of the display area of a LED display.
- Set or get the brightness of a LED display.
- Set or get the current gain of a LED display.

- Set or get the Gamma value of a LED display.
- Set the image display type of a LED display.
- Save the settings of transmitter cards (controllers) and receiver cards.
- Refresh the monitored statuses on the receiver cards and the monitor board.
- Get the monitored statuses from receiver cards or monitor boards.
- Check the pixels statuses of a certain receiver card on a certain LED display and report the check result.
- Set or get the power supply control mode of a multifunction card connected to a certain transmitter card.
- Set or get the power supply outputs status of a multifunction card connected to a certain transmitter card or controller through an Ethernet port.
- Set or get the automatic power supply control schedule on a multifunction card connected to a certain transmitter card or controller through an Ethernet port.
- Get the measurement result of a light sensor connected to a multifunction card which is connected to a certain transmitter card or controller through an Ethernet port.
- Get the measurement result of a light sensor directly conneced to a certain transmitter card.

Note

Redundent Ethernet ports are not supported by all operations related to LED display control system control.

2.3.1 Initialize

Description

Initialize an object of this class with a specified serial port and get the numbers of LED displays and transmitter cards (or controllers) that are connected to the computer through this serial port.

Parameters

No.	Type	Para Name	Description	Value Range

1	string	comNameOfCtrlSystem	The serial port name through which the control system is connected to the computer.	
2	int	screenCount	Number of the LED displays that are connected to the computer through the specified serial port.	
3	int	senderCount	Number of the transmitter cards or controllers that are connected to the computer through the specified serial port.	

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.3.2 UnInitialize

Description

Release the resources of an object of this class.

Remark

All member functions will return False after an object of this class is uninitialized.

2.3.3 GetScreenLocation

Description

Get the size and offsets of a specified LED display. By LED display here means the activated area (area used to show images) of a physical LED display.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	screenIndex	LED display (Screen) index	From 0 to (S_Count-1) where S_Count is the number of LED displays connected to the

				computer through the current serial ports.
2	int	X	The retrieved X (Column) offset of the LED display	
3	int	y	The retrieved Y (Row) offset of the LED display	
5	int	width	The retrieved width of the LED display	
6	int	height	The retrieved height of the LED display	

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.3.4 SetScreenLocation

Description

Set the offsets (in row and column) of the specified LED display

Parameters

No.	Type	Para Name	Description	Value Range
1	int	screenIndex	LED display (Screen) index	From 0 to (S_Count-1) where S_Count is the number of LED displays connected to the computer through the current serial ports.
2	int	X	X (Column) offset of the LED display	0~32767
3	int	y	Y (Row) offset of the LED display	0~32767

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.3.5 SetLEDScreenInfo

Description: Set up screen information.

Parameters

No.	Type	Para Name	Description	Value Range
1	UInt16	dviWidth	DVI’s Width	0～65535
2	UInt16	dviHeight	DVI’s Height	0～65535
3	List<LEDScreenInfo>	screenInfoList	Screen information list	Maximum LED display quantity: 200, display info please refer to LEDScreenInfo

Return Value:

Type	Description
bool	When the receiver card list in a certain screen is empty, then return to **ScreenHasNoSB**; When the set-up width and height of DVI are less than or equivalent to zero, then return to **DVIInfoError**; When failure in initialization or no initialization, then return to **NotInit**; When other failures occur, return to **CommunicateFailed**; When setting up succeeds, return to OK

Remark:

1. The width and height of DVI means DVI input of each controller；
2. Each LED display needs one DVI input；

2.3.6 ClearLEDScreenInfo

Description: Clear screen information。

Return Value

Type	Description
bool	As a result of clearing screen information, when failure in initialization or no initialization, then return to **NotInit**;

	When failure in clearing, return to **CommunicateFailed**； When clearing succeeds

2.3.7 ReadLEDScreenInfo

Description： Read screen information。

Parameters

No.	Type	Para Name	Description	Value Range
1	List<LEDScreenInfo>	screenInfoList	Screen information list	

Return Value：

Type	Description
bool	As a result of reading screen information, when failure in initialization or no initialization, then return to NotInit; When failure in reading screen information, return to CommunicateFailed； When the type of equipment in connection with the serial port is not control system, return to CommunicateFailed； When success in reading back screen information, return to OK, with screenInforList as the screen information read

2.3.8 GetScanBoardCount

Description

Get the receiver cards count of specified LED display.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	screenIndex	LED display (Screen) index	From 0 to (S_Count-1) where S_Count is the number of LED displays connected to the computer through the current serial ports.

2	int	scanBdCount	The retrieved receiver cards count	

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.3.9 GetScanBoardOfScreen

Description

Get the mapping information of a specified receiver card in a certain LED display.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	screenIndex	LED display (Screen) index	From 0 to (S_Count-1) where S_Count is the number of LED displays connected to the computer through the current serial ports.
2	int	index	Receiver card index	From 0 to (RC_Count-1) where RC_Count is the number of recerver cards in the LED display.
3	Nova.Mars.SDK.ScanBoardMapRegion	scanBd	The retrieved mapping information of the receiver card.	Refer to 3.1.1 ScanBoardMapRegion for more details.

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.3.10 ControlDisplay

Description

Set the image display type of a LED display.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	screenIndex	LED display (Screen) index	From 0 to (S_Count-1) where S_Count is the number of LED displays connected to the computer through the current serial ports.
2	Nova.Mars.SDK.DisplayControlType	controlType	Image display type	Refer to 3.1.5 DisplayControlType for more details.

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.3.11 SetBrightness

Description

Set the overall brightness of the LED display.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	screenIndex	LED display (Screen) index	From 0 to (S_Count-1) where S_Count is the number of LED displays connected to the computer through the current serial ports.
2	byte	bright	brightness	0～255

Return Value

Type	Description

bool	True --- Operation succeeded　　False --- Operation failed

2.3.12　SetRedBrightness

Description

Set the red channel brightness of the LED display.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	screenIndex	LED display (Screen) index	From 0 to (S_Count-1) where S_Count is the number of LED displays connected to the computer through the current serial ports.
2	byte	red	Brightness of the red channel.	0～255

Return Value

Type	Description
bool	True --- Operation succeeded　　False --- Operation failed

2.3.13　SetBlueBrightness

Description

Set the blue channel brightness of the LED display.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	screenIndex	LED display (Screen) index	From 0 to (S_Count-1) where S_Count is the number of LED displays connected to the computer through the current serial ports.
2	byte	blue	Brightness of the blue channel	0～255

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.3.14 SetGreenBrightness

Description

Set the green channel brightness of the LED display.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	screenIndex	LED display (Screen) index	From 0 to (S_Count-1) where S_Count is the number of LED displays connected to the computer through the current serial ports.
2	byte	green	Brightness of the green channel.	0～255

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.3.15 GetBrightness

Description

Get the brightness information of the LED display, including the overall brightness and the brightness of the red, blue and green channels.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	screenIndex	LED display (Screen) index	From 0 to (S_Count-1) where S_Count is the number of LED displays connected to

				the computer through the current serial ports.
2	byte	bright	The retrieved overall brightness	0～255
3	byte	red	The retrieved red channel brightness	0～255
4	byte	blue	The retrieved blue channel brightness	0～255
5	byte	green	The retrieved green channel brightness	0～255

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.3.16 SetGain

Description

Set the current gain of the LED display.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	screenIndex	LED display (Screen) index	From 0 to (S_Count-1) where S_Count is the number of LED displays connected to the computer through the current serial ports.
2	byte	redGain	Gain of the red channel	0～255
3	byte	blueGain	Gain of the blue channel	0～255
4	byte	greenGain	Gain of the green channel	0～255

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.3.17 GetGain

Description

Get the current gain of the LED display.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	screenIndex	LED display (Screen) index	From 0 to (S_Count-1) where S_Count is the number of LED displays connected to the computer through the current serial ports.
2	byte	redGain	The retrieved gain of the red channel	0～255
3	byte	blueGain	The retrieved gain of the blue channel	0～255
4	byte	greenGain	The retrieved gain of the green channel	0～255

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.3.18 SetGamma

Description

Set the Gamma value of the LED display.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	screenIndex	LED display (Screen) index	From 0 to (S_Count-1) where S_Count is the number of LED displays connected to the computer through the current serial ports.
2	float	gamma	Gamma value	1.0～4.0

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.3.19 GetGamma

Description

Get the Gamma value of the LED display.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	screenIndex	LED display (Screen) index	From 0 to (S_Count-1) where S_Count is the number of LED displays connected to the computer through the current serial ports.
2	float	gamma	The retrieved Gamma value	

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.3.20 BeginRefreshHardwareStatus

Description

Begin the operation of updating the hardware statuses.

Remark

Refreshing operation result (succeeded or failed) will be reported by the event of RefreshHardwareStatusFinishEvent.

2.3.21 IsScanBoardWorkOK

Description

Get the working status of a specified receiver card (scan-board).

Remark

Only after the operation of **BeginRefreshHardwareStatus** has been performed successfully will the retrieved receiver card status be real.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	screenIndex	LED display (Screen) index	From 0 to (S_Count-1) where S_Count is the number of LED displays connected to the computer through the current serial ports.
2	int	scanBdIndex	Receiver card index	From 0 to (RC_Count-1) where RC_Count is the Receiver cards number in the LED display.
3	Nova.Mars.SDK.StatusType	status	The retrieved receiver card working status.	Refer to 3.1.2 StatusType for more details.

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.3.22 GetScanBoardTemperature

Description

Get the temperature monitored by a specified receiver card (scan-board).

Remark

Only after the operation of **BeginRefreshHardwareStatus** has been performed successfully will the retrieved temperature be real.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	screenIndex	LED display (Screen) index	From 0 to (S_Count-1) where S_Count is the number of LED displays connected to the computer through the current serial ports.
2	int	scanBdIndex	Receiver card index	From 0 to (RC_Count-1) where RC_Count is the receiver cards number in the LED display.
3	Nova.Mars.SDK.ValueInfo	value	The retrieved temperature.	Refer to 3.1.8 ValueInfo for more details.

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.3.23 GetScanBoardVoltage

Description

Get the power supply voltage of a specified receiver card (scan-board).

Remark

Only after the operation of **BeginRefreshHardwareStatus** has been performed successfully will the retrieved voltage be real.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	screenIndex	LED display (Screen) index	From 0 to (S_Count-1) where S_Count is the number of LED displays connected to the computer through the current serial ports.

2	int	scanBdIndex	Receiver card index	From 0 to (RC_Count-1) where RC_Count is the receiver cards number in the LED display.
3	Nova.Mars.SDK.ValueInfo	value	The retrieved voltage	Refer to 3.1.8 ValueInfo for more details.

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.3.24 IsConnectWithMonitorBoard

Description

Check whether there is a monitor board connected to the specified receiver card.

Remark

Only after the operation of **BeginRefreshHardwareStatus** has been performed successfully will the retrieved result be real.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	screenIndex	LED display index	From 0 to (S_Count-1) where S_Count is the number of LED displays connected to the computer through the current serial ports.
2	int	scanBdIndex	Receiver card index	From 0 to (RC_Count-1) where RC_Count is the receiver cards number in the LED display.
3	bool	isConnectMB	The received flag indicating whether the receiver card is connected with a monitor board.	

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.3.25 GetMonitorBoardHumidity

Description

Get the humidity monitored by a specified monitor board.

Remark

Only after the operation of **BeginRefreshHardwareStatus** has been performed successfully will the retrieved humidity be real.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	screenIndex	LED display index	From 0 to (S_Count-1) where S_Count is the number of LED displays connected to the computer through the current serial ports.
2	int	scanBdIndex	Receiver card index	From 0 to (RC_Count-1) where RC_Count is the receiver cards number in the LED display.
3	Nova.Mars.SDK.ValueInfo	value	The retrieved humidity measured by the monitor board.	Refer to 3.1.8 ValueInfo for more details.

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.3.26 GetMonitorBoardSmoke

Description

Get the smoke information monitored by a specified monitor board.

Remark

Only after the operation of **BeginRefreshHardwareStatus** has been performed successfully will the retrieved smoke information be real.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	screenIndex	LED display index	From 0 to (S_Count-1) where S_Count is the number of LED displays connected to the computer through the current serial ports.
2	int	scanBdIndex	Receiver card index	From 0 to (RC_Count-1) where RC_Count is the receiver cards number in the LED display.
3	Nova.Mars.SDK.AlarmInfo	alarmInfo	The retrieved smoke alarm repoeted by the monitor board.	Refer to 3.1.9 AlarmInfo for more details.

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.3.27 GetMonitorBoardFan

Description

Get the fan speeds monitored by a monitor board.

Remark

Only after the operation of **BeginRefreshHardwareStatus** has been performed successfully will the retrieved fan speeds be real.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	screenIndex	LED display index	From 0 to (S_Count-1) where S_Count is the number of LED

				displays connected to the computer through the current serial ports.
2	int	scanBdIndex	Receiver card index	From 0 to (RC_Count-1) where RC_Count is the receiver cards number in the LED display.
3	Nova.Mars.SDK.ValueInfo[]	value	The retrieved fan speeds measured by the monitor board.	Refer to 3.1.8 ValueInfo for more details.

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.3.28 GetMonitorBoardPower

Description

Get the power supply voltages monitored by a monitor board.

Remark

Only after the operation of **BeginRefreshHardwareStatus** has been performed successfully will the retrieved voltages be real.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	screenIndex	LED display index	From 0 to (S_Count-1) where S_Count is the number of LED displays connected to the computer through the current serial ports.
2	int	scanBdIndex	Receiver card index	From 0 to (RC_Count-1) where RC_Count is the receiver cards number in the LED display.
3	Nova.Mars.SDK.ValueI	value	The retrieved voltages measured by the	Refer to 3.1.8 ValueInfo for more details.

	nfo[]		monitor board. Note that the first one voltage in this array is the power supply for the monitor board.	

Return Value

Type	**Description**
bool	True --- Operation succeeded False --- Operation failed

2.3.29 GetCabinetDoorStatus

Description

Get the status (close or open) of the door of a cabinet monitored by a certain monitor board.

Remark

Only after the operation of **BeginRefreshHardwareStatus** has been performed successfully will the retrieved door status be real.

Parameters

No.	**Type**	**Para Name**	**Description**	**Value Range**
1	int	screenIndex	LED display index	From 0 to (S_Count-1) where S_Count is the number of LED displays connected to the computer through the current serial ports.
2	int	scanBdIndex	Receiver card index	From 0 to (RC_Count-1) where RC_Count is the receiver cards number in the LED display.
3	Nova.Mars.SDK.CabinetDoorStatusType	status	The retrieved cabinet door status.	Refer to 3.1.4CabinetDoorStatusType for more details.

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.3.30 IsCabinetFPCOK

Description

Check whether the flexible printed circuit (FPC) in a cabinet is working normally.

Remark

Only after the operation of **BeginRefreshHardwareStatus** has been performed successfully will the retrieved FPC working status be real.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	screenIndex	LED display index	From 0 to (S_Count-1) where S_Count is the number of LED displays connected to the computer through the current serial ports.
2	int	scanBdIndex	Receiver card index	From 0 to (RC_Count-1) where RC_Count is the receiver cards number in the LED display.
3	Nova.Mars.SDK.StatusType	type	The retrieved status of the FPC .	Refer to 3.1.2 StatusType for more details.

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.3.31 BeginPointDetect

Description

Begin point detecting, also known as LED lights status (working normally, short circuit or open circuit) checking of the modules in the specified cabinet of a certain LED display.

Remark

This is for LED lights status checking of the modules which do not require the current gains to be fixed for LED lights status checking.

The LED lights status checking operation result will be reported by the event of **GetCabinetPixelEvent**.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	screenIndex	LED display index	From 0 to (S_Count-1) where S_Count is the number of LED displays connected to the computer through the current serial ports.
2	int	scanBdIndex	Receiver card index	From 0 to (RC_Count-1) where RC_Count is the receiver cards number in the LED display.
3	int	threshold	The threshold for LED lights status checking	The threshold may be different for different driver chips. Refer to 4.1 Threshold and Type of LED Light Status checking for more details.
4	Nova.Mars.SDK.PointDetectType	type	LED lights status checking type (open circuit or short circuit)	The type of LED light status checking may different for different driver chips. Refer to 4.1 Threshold and Type of LED Light Status checking and 3.1.3 PointDetectType for more details.

Return Value

Type	Description

bool	True --- Operation succeeded False --- Operation failed

❖ BeginPointDetect(int, int, int, Nova.Mars.SDK.PointDetectType, byte, byte, byte);

Description

Begin LED lights status (working normally, short circuit or open circuit) checking of the modules in the specified cabinet of a certain LED display.

Remark

This is for LED lights status checking of the modules which require the current gains to be fixed for LED lights status checking.

The LED lights status checking operation result will be reported by the event of **GetCabinetPixelEvent**.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	screenIndex	LED display index	From 0 to (S_Count-1) where S_Count is the number of LED displays connected to the computer through the current serial ports.
2	int	scanBdIndex	Receiver card index	From 0 to (RC_Count-1) where RC_Count is the receiver cards number in the LED display.
3	int	threshold	The threshold for LED lights status checking	The threshold may be different for different driver chips. Refer to 4.1 Threshold and Type of LED Light Status checking for more details.
4	Nova.Mars.SDK.PointDetectType	type	LED lights status checking type (open circuit or short circuit)	The type of LED light status checking may different for different driver chips. Refer to 4.1 Threshold and Type of LED

				Light Status checking and 3.1.3 PointDetectType for more details.
5	byte	redGain	Current gain of the red channel	0～255
6	byte	greenGain	Current gain of the green channel	0～255
7	byte	blueGain	Current gain of the blue channel	0～255

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.3.32 SetPowerControlMode

Description

Set the power supply control mode of the multifunction card connected to a certain transmitter card (or controller) through an Ethernet port.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	senderIndex	Transmitter card (or controller) index	From 0 to (TC_Count-1) where TC_Count is the number of the transmitter cards or controllers connected to the computer through the current serial port.
2	int	portIndex	Ethernet port index	For transmitter cards (MSD300): 0 or 1 For controllers (MCTRL500): 0, 1, 2 or 3
3	int	funcCardIndex	Multifunction card index	
4	Nova.Mars.SDK.PowerControlMod	ctrlMode	Power supply control mode	Refer to 3.1.6 PowerControlMode for more details.

	e			

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.3.33 SetPowerSwitchStatus

Description

Set the status of a certain power supply on the specified multifunction card connected to a certain transmitter card or controller.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	senderIndex	Transmitter card or controller index	From 0 to (TC_Count-1) where TC_Count is the number of the transmitter cards or controllers connected to the computer through the current serial port.
2	int	portIndex	Ethernet port index	For transmitter cards (MSD300): 0 or 1 For controllers (MCTRL500): 0, 1, 2 or 3
3	int	funcCardIndex	Multifunction card index	
4	int	switchIndex	Power supply index	0～7
5	Nova.Mars.SDK.PowerSwitchStatus	switchStatus	Power supply status	Refer to 3.1.7 PowerSwitchStatus for more details.

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.3.34 SetPowerAllSwitchStatus

Description

Set the status of all 8 power supplies on a specified multifunction card connected to a certain transmitter card or controller through an Ethernet port.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	senderIndex	Transmitter card or controller index	From 0 to (TC_Count-1) where TC_Count is the number of the transmitter cards or controllers connected to the computer through the current serial port.
2	int	portIndex	Ethernet port index	For transmitter cards (MSD300): 0 or 1 For controllers (MCTRL500): 0, 1, 2 or 3
3	int	funcCardIndex	Multifunction card index	
4	Nova.Mars.SDK.PowerSwitchStatus	switchStatus	Power supply status	Refer to 3.1.7PowerSwitchStatus for more details.

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.3.35 SetPowerSwitchAutoTime

Description

Set the schedule for automatic control of all 8 power supplies on a certain multifunction card connected to a specified transmitter card or controller.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	senderIndex	Transmitter card or controller index	From 0 to (TC_Count-1) where TC_Count is the number of the transmitter cards or controllers connected to the computer through the current serial port.
2	int	portIndex	Ethernet port index	For transmitter cards (MSD300): 0 or 1 For controllers (MCTRL500): 0, 1, 2 or 3
3	int	funcCardIndex	Multifunction card index	
4	DateTime[]	startTime	Time to turn on the powers.	Array length 8
5	DateTime[]	StopTime	Time to turn off the powers.	Array length 8

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.3.36 GetPowerSwitchAutoTime

Description

Get the schedule for automatic control of all 8 power supplies on a certain multifunction card connected to a specified transmitter card or controller.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	senderIndex	Transmitter	From 0 to (TC_Count-1) where

			card or controller index	TC_Count is the number of the transmitter cards or controllers connected to the computer through the current serial port.
2	int	portIndex	Ethernet port index	For transmitter cards (MSD300): 0 or 1 For controllers (MCTRL500): 0, 1, 2 or 3
3	int	funcCardIndex	Multifunction card index	
4	DateTime[]	startTime	The retrieved time to turn on the powers	Array length 8
5	DateTime[]	StopTime	The retrieved time to turn off the powers	Array length 8

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.3.37 GetPowerControlMode

Description

Get the power supply control mode of a certain multifunction card connected to a specified transmitter card or controller through an Ethernet port.

Parameter

No.	Type	Para Name	Description	Value Range
1	int	senderIndex	Transmitter card or controller index	From 0 to (TC_Count-1) where TC_Count is the number of the transmitter cards or controllers connected to the computer through the current serial port.
2	int	portIndex	Ethernet port index	For transmitter cards

				(MSD300): 0 or 1 For controllers (MCTRL500): 0, 1, 2 or 3
3	int	funcCardIndex	Multifunction card index	
4	Nova.Mars.SDK.PowerControlMode	ctrlMode	The retrieved power supply control mode	Refer to 3.1.6 PowerControlMode for more details.

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.3.38 GetPowerSwitchStatus

Description

Get the status of a certain power supply on a specified multifunction card connected to a certain transmitter card or controller through an Ethernet port.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	senderIndex	Transmitter card or controller index	From 0 to (TC_Count-1) where TC_Count is the number of the transmitter cards or controllers connected to the computer through the current serial port.
2	int	portIndex	Ethernet port index	For transmitter cards (MSD300): 0 or 1 For controllers (MCTRL500): 0, 1, 2 or 3
3	int	funcCardIndex	Multifunction card index	
4	int	switchIndex	Power supply index	0～7

5	Nova.Mars.SDK.PowerSwitchStatus	switchStatus	The retrieved power supply status	Refer to 3.1.7 PowerSwitchStatus for more details.

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.3.39 GetValueOfLightSensorInSender

Description

Get the measurement result of the light sensor connected to a specified transmitter card.

Remark

Light sensors can be connected to transmitter cards, but not to controllers.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	senderIndex	Transmitter card index	From 0 to (TC_Count-1) where TC_Count is the number of the transmitter cards connected to the computer through the current serial port.
2	int	lux	The retrieved measurement value of the light sensor.	

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.3.40 GetValueOfLightSensorInFunction

Description

Get the measurement value of a light sensor connected to a certain multifunction card which is connected with a transmitter card or controller through an Ethernet port.

Parameters

No.	Type	Para Name	Description	Value Range
1	int	senderIndex	Transmitter card or controller index	From 0 to (TC_Count-1) where TC_Count is the number of the transmitter cards or controllers connected to the computer through the current serial port.
2	int	portIndex	Ethernet port index	For transmitter cards (MSD300): 0 or 1 For controllers (MCTRL500): 0, 1, 2 or 3
3	int	funcCardIndex	Multifunction card index	
4	int	lux	The retrieved measurement value of the light sensor.	

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.3.41 SaveParameters

Description

Save the settings of transmitter cards and receiver cards.

Remark

The saving operation will tae about 11 seconds. Do not perform any other operation on the hardware during this time.

Return Value

Type	Description
bool	True --- Operation succeeded False --- Operation failed

2.3.42 RefreshHardwareStatusFinishEvent

Description

The event that indicates the finish of the operation of hardware status refreshing (updating the monitored statuses).

2.3.43 GetCabinetPixelEvent

Description

The event that indicates the finish of LED lights status checking.

2.3.44 SetHotBackUp

Description

Set hot backup information

Parameters

No.	Type	Para Name	Description	Value Range
1	List<SenderRedundancyInfo>	reduInfoList	Hot backup information list	
2	CompleteHotBackUpWriteConfig	callBack	Set delegation for the completion result of the hot backup information (callback function)	

2.3.45 SaveHotBackUpToHw

Description

Save hot backup information

Parameters

No.	Type	Para Name	Description	Value Range
1	List<SenderRedundancyInfo>	reduInfoList	Hot backup information list	
2	CompleteHotBackUpWriteConfig	callBack	Save delegation for the completion result of the hot backup information (callback function)	

2.3.46 GetHotBackUp

Description

Get hot backup information

Parameters

No.	Type	Para Name	Description	Value Range
1	CompleteHotBackUpReadConfig	callBack	Get delegation for the completion result of the hot backup information (callback function)	

2.3.47 DeleteHotBackUp

Description

Delete the hot backup information

Parameters

No.	Type	Para Name	Description	Value Range
1	List<SenderRedund	deletedReduInfo	Hot backup	

	ancyInfo>	List	information list to be deleted	
2	CompleteHotBackUpWriteConfig	callBack	Delete delegation for the completion result of the hot backup information (callback function)	

2.3.48 GetControlSysInfo

Description

Get type, serial number, internet access number of control system

Parameters

No.	Type	Para Name	Description	Value Range
1	CompleteControlSysInfoConfig	callBack	Get delegation for completion result of type, serial number, internet access number of control system (callback function)	

2.3.49 ReadScannerParameters

Description

Get parameters of refresh rate, grey scale and brightness effective rate of receiving card

Parameters

No.	Type	Para Name	Description	Value Range
1	int	screenIndex	Screen index	Index number starts from 0
2	int	senderIndex	Serial number of	Serial number starts from 0

			sending card	
3	int	portIndex	Serial number of internet access of sending card	Serial number starts from 0
4	int	scanBdIndex	Serial number of receiving card	Serial number starts from 0
5	CompleteControlSysInfoConfig	callBack	Get delegation for completion result of type, serial number, internet access number of control system (callback function)	

2.3.50 SetMarsDisplayResolutionRate

Description

Sets the resolution and refresh rate of a sending card.

Parameters

No.	Type	Para Name	Description	Value Range
1	SenderDisplayInfo	senderDisplayInfo	Set the resolution and refresh rate of a sending card.	

2.3.51 GetMarsDisplayResolutionRateInfo

Description

Obtain the resolution and refresh rate of the specified sending card.

Parameters

No.	Type	Para Name	Description	Value Range

1	Byte	senderCardAddr	Serial number of a sending card.	≥ 0

2.3.52 SendScanConfigFileToHW

Description

send the config file information of receiving card to hardware

Parameters

No.	Type	Para Name	Description	Value Range
1	string	scanConfigFileName	receiving card config file path	
2	int	screenIndex	Screen index	Index number starts from 0
3	CompleteSendConfigFileToHW	callback	Get delegation for the completion result (callback function), parameter type : SendConfigFileToHWState	

2.3.53 SendSysConfigFileToHW

Description

Send system config file information to hardward

Parameters

No.	Type	Para Name	Description	Value Range
1	string	sysConfigFileName	system config file path	

2	CompleteSendConfigFileToHW	callback	Get delegation for the completion result (callback function), parameter type : SendConfigFileToHWState	

2.3.54 SendScreenConfigFileToHW

Description

Send screen config file information to hardward

Parameters

No.	Type	Para Name	Description	Value Range
1	string	screenConfigFileName	Screen config file path	
2	Size	dviResolution	Resolution of the sending card(fill resolution of the present sending card, in order to ensure the size of screen contained in config file doesn’t exceed the limits of present sending card.)	

Return value

No.	Type	Para Name	Description
1	OperateResult	result	the result of sending screen config file information to hardware

2.3.55 SaveParameters

Description

Save the settings of transmitter cards and receiver cards(including hot backup information)

Remark

The saving operation will tae about 11 seconds. Do not perform any other operation on the hardware during this time.

Parameters

No.	Type	Para Name	Description	Value Range
1	SaveParamsCallBack	callback	Get delegation for the saving result (callback function), parameter type : SaveParamsResData	

2.3.56 SetScanBigTableData

Description

Send and receive card large table data (default profile box)

Remark

After the success of the transmission is not cured, need to manually operate the curing

Parameters

No.	Type	Para Name	Description	Value Range
1	byte	senderIndex	Serial number of sending card	Serial number starts from 0
2	byte	portIndex	Serial number of internet access of sending card	Serial number starts from 0
3	ushort	scanBdIndex	Serial number of receiving card	Serial number starts from 0
4	Byte[]	data	Data	
5	ComPleteSetSca	callBack	Get delegation for the sending	

	nBigTableData		result (callback function), parameter type : ReadScannerState	

2.3.57 SetEquipmentIP

Description

Setting Sender IP

Parameters

No.	Type	Para Name	Description	Value Range
1	byte	senderIndex	Serial number of sending card	Serial number starts from 0
2	Byte[]	data	IP Data	

2.3.58 GetEquipmentIP

Description

Get Sender IP

Parameters

No.	Type	Para Name	Description	Value Range
1	byte	senderIndex	Serial number of sending card	Serial number starts from 0

3 Data Structures and Delegates

3.1Data Structure

3.1.1 ScanBoardMapRegion

Description

Structure for the Mapping region info.

Parameters

No.	Type	Para Name	Description	Value Range
1	byte	SenderIndex	Transmitter card or controller index	0～255
2	byte	PortIndex	Ethernet port index	0～255
3	UInt16	ConnectIndex	Receiver card index	0～65535
4	UInt16	X	X (column) offset of the receiver card	0～32767
5	UInt16	Y	Y (row) offset of the receiver card	0～32767
6	UInt16	Width	Width of the pixel array driven by the receiver card.	0～65535
7	UInt16	Height	Height of the pixel array driven by the receiver card.	0～65535

3.1.2 StatusType

Description

Enum of the working status.

Parameters

No.	Para Name	Value	Description
1	OK	0	The object working status is good.
2	Error	1	The object is not working normally.
3	Unknown	2	The object working status is unknown.

3.1.3 PointDetectType

Description

Enum of the Point detecting (LED lights status checking) type.

Parameters

No.	Para Name	Value	Description

1	OpenCircuit	0	Checking for open circuit.
2	ShortCircuit	1	Checking for short circuit.

3.1.4 CabinetDoorStatusType

Description

Enum of cabinet door status.

Parameters

No.	Para Name	Value	Description
1	Close	0	The cabinet door is closed
2	Open	1	The cabinet door is open.
3	Unknown	2	The cabinet door status is unknown.

3.1.5 DisplayControlType

Description

Enum of image display type.

Parameters

No.	Para Name	Value	Description
1	Normal	0	Show the images normally.
2	Lock	1	Lock (keep on showing) the current image.
3	Kill	2	Show black on the LED display.

3.1.6 PowerControlMode

Description

Enum of the power supply control mode.

Parameters

No.	Para Name	Value	Description

1	Manual	0	Manual control. That is to turn on/off the power supplies by sending instructions through application software manually.
2	Auto	1	Automatic control. The power supplies are turned on/off by the system automatically according to the schedule which is set through application software.

3.1.7 PowerSwitchStatus

Description

Enum of the power supply status.

Parameters

No.	Para Name	Value	Description
1	On	0	The power supply is on.
2	Off	1	The power supply is off.

3.1.8 ValueInfo

Description

Information of a certain value.

Parameters

No.	Type	Para Name	Description	Value Range
1	bool	IsValid	Flag indicating whether the value is valid.	
2	float	Value	The value. Only valid when IsValid is True.	

3.1.9 AlarmInfo

Description

Warning information.

Parameters

No.	Type	Para Name	Description	Value Range
1	bool	IsValid	Flag indicating whether the warning is	

			valid.	
2	bool	IsAlarm	Flag indicating whether there is a warning. Only valid when IsAlarm is True.	

3.1.10 ModulePixelInfo

Description

The LED light status checking result of a module.

Parameters

No.	Type	Para Name	Description	Value Range
1	bool	IsHasErrorLight	Flag indicating whether there are error lights.	
2	List<Point>	RedErrorList	List of the positions of the error red LED lights in the pixel array of the module.	
3	List<Point>	GreenErrorList	List of the positions of the error green LED lights in the pixel array of the module.	
4	List<Point>	BlueErrorList	List of the positions of the error blue LED lights in the pixel array of the module.	
5	List<Point>	VRedErrorList	List of the positions of the error virtual red LED lights in the pixel array of the module.	

3.1.11 CabinetErrorPixelInfo

Description

The LED lights status checking result of a cabinet.

Parameters

No.	Type	Para Name	Description	Value Range

1	int	ModuleCols	Column number of the module array in the cabinet.	
2	int	ModuleRows	Row number of the module array in the cabinet.	
3	List<ModulePixelInfo>	ModuleList	The list of the LED light status checking result of each module in the cabinet.	Refer to 3.1.10ModulePixelInfo for more details about ModulePixelInfo.

3.1.12 LEDScreenInfo

Description

Screen information。

Parameters

No.	Type	Para Name	Description	Value Range
1	UInt16	ScreenX	Horizontal location of the screen	0～65535
2	UInt16	ScreenY	Vertical location of the screen	0～65535
3	ScreenVirtualMode	VirtualMode	Virtual mode of the screen	Enable, Disable
4	List<ScanBoardMapRegion>	ScanBoardInfoList	Information list of receiver card in the screen	

3.1.13 ScreenVirtualMode

Description

Virtual mode of the screen.

Parameters

No.	Para Name	Value	Description

1	Disable	0	Real pixel
2	Led4Mode1	1	Four LED Virtual Mode 1
3	Led4Mode2	2	Four LED Virtual Mode 1
4	Led3Mode	3	Three LED Virtual Mode

3.1.14 OperateResult

Description

Operation results.

Parameters

No.	Para Name	Value	Description
1	OK	0	OK
2	NotInit	1	Not initialized
3	SBAddrDuplication	2	SB address duplication
4	CommunicateFailed	3	Communicate failed
5	ScreenHasNoSB	4	Screen has no SB
6	DVIInfoError	5	DVI information error
7	ScreenLocationError	6	Screen location error
8	SBLocationError	7	SB location error
9	LoadScreenConfigFileError	8	Screen file loading errors
10	ConfigFileNameIsNone	9	config file is empty
11	Sending	10	sending
12	NoScanBdInfo	11	Information for at least one receiving card is needed
13	SenderError	12	The sending card is not connected or not working normally.
14	OutOfAange	13	Mapped areas of the sending card is out of the present screen arrange

3.1.15 HotBackUpState

Description

Operate enumeration status of hot backup

Parameters

No.	Para Name	Value
1	NoHotBackUpInfo	No hot backup information
2	Successful	Success
3	Sending	Sending hot backup information
4	Loading	Loading hot backup information
5	Saving	Saving hot backup information
6	Error	Failed
7	InitalError	Initialization failed
8	NoSelectedHotBackUpInfo	No data to be deleted
9	DeleteSuccessful	Delete successfully
10	MasterEqualToSlave	Mater equipment equals to slave equipment
11	IsExist	The same hot backup information exists
12	MasterIsExistSlave	Master sending card to be set already exists in slave sending card
13	SlaveIsExistMaster	Slave sending card to be set already exists in master sending card
14	MasterCommunicateError	Hardware of master sending card to be set is not connected
15	SlaveCommunicateError	Hardware of slave sending card to be set is not connected

16	MasterIsExist	Master sending card to be set already exists
17	SlaveIsExist	Slave sending card to be set already exists
18	MasterPortCommunicateError	Hardware of master internet access number to be set is not connected
19	SlavePortCommunicateError	Hardware of slave internet access number to be set is not connected

3.1.16 GetControlSysState

Description

Get enumeration status of type, serial number and internet access number of control system

Parameters

No.	Para Name	Value
1	NoControlSysInfo	No control system information
2	Successful	Success
3	Sending	Loading control system information
4	InitalError	Initialization failed
5	GetSNError	Get serial number failed
6	Error	Failed

3.1.17 ReadScannerState

Description

Get enumeration status of receiving card parameter

Parameters

No.	Para Name	Value
1	Successful	Success
2	Reading	Loading receiving card parameter

3	AllotypeBoxIsNotSupport	Allotype box is not supported
4	NoScannerParameter	No receiving card information
5	NoScreenInfo	No screen information
6	ScreenIndexError	Screen index error
7	InitalError	Initialization failed
8	Error	Failed

3.1.18 ControlSysInfo

Description

Type, serial number, internet access number and other information of control system

Parameters

No.	Type	Para Name	Description	Value Range
1	NSCardType	CardType	Type of control system	
2	string	CardSN	Serial number of control system	
3	int	PortNum	Internet access number of control system	

3.1.19 ScannerParameter

Description

Refresh rate, grey scale, refresh ratio, brightness and other information of receiving card

Parameters

No.	Type	Para Name	Description	Value Range
1	int	Refresh	Refresh rate	
2	byte	GrayDepth	Grey scale	
3	int	SubFields	Refresh ratio	
4	float	BrightEcyValue	Brightness	

3.1.20 SenderRedundancyInfo

Description: Hot backup information

Parameters

No.	Type	Para Name	Description	Value Range
1	byte	MasterSenderIndex	Serial number of master sending card	The serial number starts from 0
2	byte	MasterPortIndex	Serial number of master internet access	The serial number starts from 0
3	byte	SlaveSenderIndex	Serial number of slave sending card	The serial number starts from 0
4	byte	SlavePortIndex	Serial number of slave internet access	The serial number starts from 0

3.1.21 SenderDisplayInfo

Description

Indicate the information about the resolution and refresh rate.

Parameters

No.	Type	Para Name	Description	Value Range
1	byte	SenderAddr	Serial number of a sending card. 255 (0xFF) indicates the broadcast address.	≥ 0
2	int	Height	Height	
3	int	Width	Width	
4	int	Refresh	Refresh rate	
5	bool	IsCustomPix	Custom pixel	**True** or **False**

3.1.22 SendConfigFileToHWState

Description

Enumeration status of sending config file information to hardware

Parameters

NO	Para Name	value	Description
1	Successful	0	Success
2	LoadError	1	loading failed
3	SendError	2	sending failed
4	InitalError	3	Initialization failed
5	Loading	4	sending
6	ScreenIndexError	5	Screen index error
7	ConfigFileNameIsNone,	6	config file is empty
8	DVIInfoError	7	DVI information error
9	NoScanBdInfo	8	Information for at least one receiving card is needed
10	OutOfAange	9	Mapped areas of the sending card is out of the present screen arrange
11	SenderError	10	Sending card is not connected or not working normally

3.1.23 SaveParamsErrorType

Description

Enumeration status of saving parameters

Parameters

NO	Para Name	value	Description
1	OK	0	Success
2	UnInit	1	Initialization failed
3	SaveScreenInfoErr	2	Saving screen information failed
4	SaveSenderParamsErr	3	Saving sending card parameters failed
5	SaveScannerParamsErr	4	Saving receiving card parameters failed
6	SaveReduInfoErr	5	Saving hot backup information failed

3.2Delegates and Parameters

3.2.1 NotifyUnInitializeEvent

Description： Notice the uninitialized.

3.2.2 CabinetPixelInfoEventArgs

Description

The class for containing the LED lights status checking results.

Parameters

No.	Type	Para Name	Description	Value Range
1	bool	IsDetectedSuccessful	Flag indicating whether the LED lights status checking has been finished successfully.	
2	int	screenIndex	The index of the LED display on which the LED lights status checking is performed.	
3	int	scanBdIndex	The index of the cabinet on which the LED lights status checking is performed.	
4	CabinetErrorPixelInfo	PixelInfo	Result the LED lights status checking.	Refer to 3.1.11 CabinetErrorPixelInfo for more details.

3.2.3 RefreshResultEventArgs

Description

The class for containing the result of monitored statuses refreshing.

Parameters

No.	Type	Para Name	Description	Value Range
1	bool	bFinishSucceed	Flag indicating whether the	

			monitored statuses refreshing operation has been finished successfully.	
2	string	CurCommPort	The port that has finished monitored statuses refreshing refreshing	

3.2.4 CabinetPixelInfoEventHandler

Description

Delegate of the finish of the LED lights status checking.

3.2.5 RefreshResultEventHandler

Description

Delegate of the finish of the monitored statuses updating.

3.2.6 CompleteHotBackUpWriteConfig

Description

Be able to get delegation for completion result when setting, saving and deleting the hot backup information

Parameters

No.	Type	Para Name	Description	Value Range
1	HotBackUpState	Res	Set, save and delete status of hot backup information	
2	SenderRedundancyInfo	Info	Prompt wrong hot backup information when setting hot backup information	

3.2.7 CompleteHotBackUpReadConfig

Description

Be able to get delegation for completion result when getting hot backup information。

Parameters

No.	Type	Para Name	Description	Value Range
1	HotBackUpState	Res	Get the status of the hot backup information	
2	List<SenderRedundancyInfo>	Info	Obtained hot backup information list	

3.2.8 CompleteControlSysInfoConfig

Description

Be able to get delegation for the completion result when getting the type, serial number and internet access number of the control system

Parameters

No.	Type	Para Name	Description	Value Range
1	GetControlSysState	Res	Get the status of type, serial number and internet access number of the control system	
2	List<ControlSysInfo>	Info	Obtained type, serial number, internet access number and other information of the control system	

3.2.9 ComPleteReadScannerParametersConfig

Description

Be able to get delegation of completion result when getting the refresh rate, grey scale, refresh ratio and brightness of receiving card

Parameters

No.	Type	Para Name	Description	Value Range
1	ReadScannerState	Res	Get the status of the receiving card parameter	
2	ScannerParameter	Info	Obtained parameter of refresh rate, grey scale, refresh ratio and brightness of receiving card	

3.2.10 SetMarsDisplayEDIDDataEvent

Description

Set the resolution event of a sending card.

3.2.11 GetMarsDisplayPixDataEvent

Description

Obtain the resolution event of a sending card.

3.2.12 OperateResultEventHandler

Description

Set a commission for a resolution completion event of a sending card.

Parameters

No.	Type	Para Name	Description	Value Range
1	Object	sender	Sender	
2	OperateResultEventArgs	args	Returned data	

			parameters	

3.2.13 GetMarsDisplayInfoResultEventHandler

Description

Obtain a commission for a resolution completion event of a sending card.

Parameters

No.	Type	Para Name	Description	Value Range
1	Object	sender	Sender	
2	GetMarsDisplayInfoResultEventArgs	args	Returned data parameters	

3.2.14 OperateResultEventArgs

Note: set the parameter type of a resolution completion event of a sending card.

Parameters

No.	Type	Para Name	Description	Value Range
1	bool	Result	Whether the setting is successful	
2	byte	SendCardAddr	Serial number of a sending card	≥ 0

3.2.15 CompleteSendConfigFileToHW

Description

Get delegation for the completion result when sending confile file information to hardware

Parameters

No	Type	Para Name	Description	Value Range
1	SendConfigFileToHWState	res	the result of sending config file	

3.2.16 SaveParamsCallBack

Description

Get delegation for the completion result of saving parameters(including hot backup information)

Parameters

No	Type	Para Name	Description	Value Range
1	SaveParamsResData	resData	the result of saving parameters	

3.2.17 ComPleteSetScanBigTableData

Description

Get delegation for the completion result of sending parameters

Parameters

No	Type	Para Name	Description	Value Range
1	ComPleteSetScanBigTableData	res	the result of sending parameters	

3.2.18 SendEquipmentIPDataEvent

Description

Send data to complete the event triggered

Parameters

No	Type	Para Name	Description	Value Range
1	Byte[]	Data	Null	
2	bool	IsExecResult	Send Success Flag	

3.2.19 GetEquipmentIPDataEvent

Description

Access to IP notification event

Parameters

No	Type	Para Name	Description	Value Range
1	Byte[]	Data	IP Data	
2	bool	IsExecResult	Read Success Flag	

4、Version Changes

Version	Time	change content	Description
V1.5.0	2016.2.29	new chips support	the new chip details:ICN2038\SM16027\SM16159\TLS3001\GW6205\SUM2017T\SUM2033\SUM2130\SUM2131\MY9231\GW6202B

5 appendix

Threshold and Type of LED Light Status Checking

The threshold and the type of LED light status checking is related to the driver chip types used on the module. See the following table for details.

Chip Type	Threshold Range	LED Light Status Checking Type Supported	Remark
MBI5036	1～4	Do not distinguish the LED light status checking types	Supported by this SDK
MBI5034	None	Open circuit checking	Supported by this SDK
DM13H	1～2	Open/short circuit status checking	Supported by this SDK
MBI5042		Not support LED light status checking	
MBI5050		Not support LED light status checking	
P2510		Not support LED light status checking	
MBI5030	None	Open circuit status checking	Supported by this SDK

SUM2017		Not support LED light status checking	
SUM2018		Not support LED light status checking	
SUM2030		Not support LED light status checking	
MBI5040	1～3	Open/short circuit status checking	Supported by this SDK

Note

☝

1) For those driver chips that do not distinguish the LED light status checking types, any type of LED light status checking is good for the status checking functions. Refer to 3.1.3 PointDetectType for more details about LED light status checking types.
2) The none of threshold range is: the threshold must be 1 for the status checking functions.
3) For those driver chips that the Threshold Range is None, any positive integer is good for the threshold when performing LED light status checking.