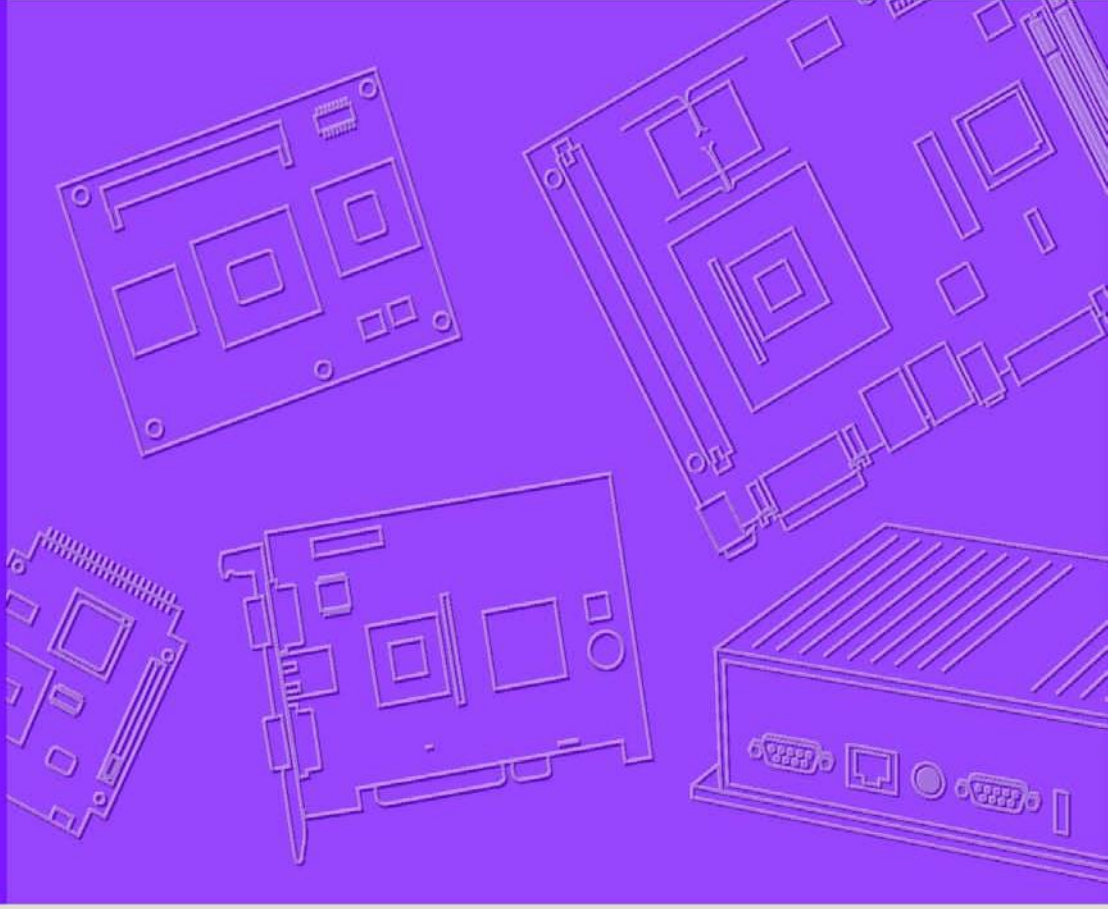


User Manual

SUSI 4.0

Secured & Unified Smart Interface Software APIs

ADVANTECH EmbCore

Edition 4.0 Part. No. N/A
April 14 2015 Printed in Taiwan

Contents

List of Tables

1 Introduction

SUSI – A Bridge to Simplify & Enhance H/W & Application Implementation Efficiency

When developers want to write an application that involves hardware access, they have to study the specifications to write the drivers. This is a time-consuming job and requires lots of expertise.

Advantech has done all the hard work for our customers with the release of a suite of Software APIs (Application Programming Interfaces), called **Secured & Unified Smart Interface** (SUSI).

SUSI provides not only the underlying drivers required but also a rich set of user-friendly, intelligent and integrated interfaces, which speeds development, enhances security and offers add-on value for Advantech platforms. SUSI plays the role of catalyst between developer and solution, and makes Advantech embedded platforms easier and simpler to adopt and operate with customer applications.

1.1 Functions

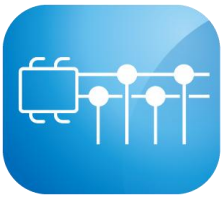
1.1.1 GPIO

General Purpose Input/Output is a flexible parallel interface that allows a variety of custom connections. It supports various Digital I/O devices – input devices like buttons, switches; output devices such as cash drawers, LED lights…etc. And, allows users to monitor the level of signal input or set the output status to switch on/off the device. Our APIs also provided Programmable GPIO and allows developers to dynamically set the GPIO input or output status.

1.1.2 SMBus

SMBus is the System Management Bus defined by Intel® Corporation in 1995. It is used in personal computers and servers for low-speed system management communications. Today, SMBus is used in all types of embedded systems. The SMBus APIs allows a developer to interface a platform to a downstream embedded system environment and transfer serial messages using the SMBus protocols, allowing multiple simultaneous device control.

1.1.3 I²C



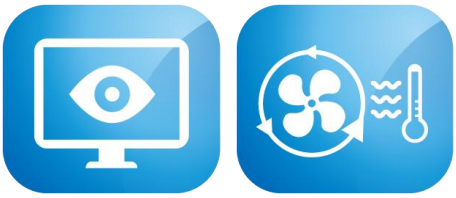
I²C is a bi-directional two wire bus that was developed by Philips for use in their televisions in the 1980s. Today, I²C is used in all types of embedded systems. The I²C API allows a developer to interface a platform to a downstream embedded system environment and transfer serial messages using the I²C protocols, allowing multiple simultaneous device control.

1.1.4 Watchdog

A watchdog timer (WDT) is a device or electronic card that performs a specific operation after a certain period of time if something goes wrong with an electronic system and the system does not recover on its own.

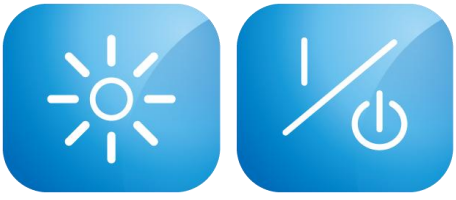
A watchdog timer can be programmed to perform a warm boot (restarting the system) after a certain number of seconds during which a program or computer fails to respond following the most recent mouse click or keyboard action.

1.1.5 Hardware Monitor



The Hardware Monitor (HWM) APIs is a system health supervision API that inspects certain condition indexes, such as smart fan, fan speed, temperature, current, case open and voltage.

1.1.6 Backlight Control

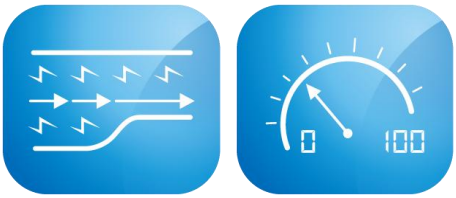


The Backlight Control APIs allows a developer to interface platform to easily control brightness through PWM and backlight on/off.

1.1.7 Storage

Storage is a non-volatile storage, the storage APIs allows a developer to access storage information, read/write data to storage and lock/unlock data area (same like write protection) by a key.

1.1.8 Thermal Protection



Thermal Protection can select a thermal source to monitor. When source temperature reach the limit, SUSI can act protect function to protect system.

1.2 Benefits

- ✓ **Faster Time to Market**

  SUSI's unified API helps developers write applications to control the hardware without knowing the hardware specs of the chipsets and driver architecture.

- ✓ **Reduced Project Effort**

  When customers have their own devices connected to the onboard bus, they can either: study the data sheet and write the driver & API from scratch, or they can use SUSI to start the integration with a 50% head start. Developers can reference the sample program on the CD to see and learn more about the software development environment.

- ✓ **Enhances Hardware Platform Reliability**

  SUSI provides a trusted custom ready solution which combines chipset and library function support, controlling application development through SUSI enhances reliability and brings peace of mind.

- ✓ **Flexible Upgrade Possibilities**

  SUSI supports an easy upgrade solution for customers. Customers just need to install the new version SUSI that supports the new functions.

- ✓ **Backward compatibility**

  Support SUSI 3.0, iManager 2.0 and EAPI 1.0 interface. Customers don't need to change any APIs in their applications.

1.3 Environment Requirements

1.3.1 Operating Systems

Windows XP Embedded

Windows XP 32-bit

Windows 7 (x86 / x64)

WES7 (x86 / x64)

Windows 8 Desktop (x86 / x64)

Windows CE 5 / 6 / 7

Linux (Project based, request from your local FAE)

Android (Project based, request from your local FAE)

QNX (Project based, request from your local FAE)

VxWorks (Project based, request from your local FAE)

2 SUSI Definition

Susi4.h file includes the API declaration, constants and flags that are required for programming.

2.1 Status Codes

All SUSI API functions immediately return a status code from a common list of possible errors. Any function may return any of the defined status codes. See the Appendix for more detailed information.

#define SUSI_STATUS_NOT_INITIALIZED 0xFFFFFFFF

Description

The SUSI API library is not yet or unsuccessfully initialized. SusiLibInitialize needs to be called prior to the first access of any other SUSI API functions.

Actions

Call SusiLibInitialize.

#define SUSI_STATUS_INITIALIZED 0xFFFFFFFE

Description

Library is initialized.

Actions

None.

#define SUSI_STATUS_ALLOC_ERROR 0xFFFFFFFD

Description

Memory Allocation Error.

Actions

Free memory and try again.

#define SUSI_STATUS_DRIVER_TIMEOUT 0xFFFFFFFC

Description

Time out in driver. This is Normally caused by hardware/software semaphore timeout.

Actions

Retry.

#define SUSI_STATUS_INVALID_PARAMETER 0xFFFFFEFF

Description

One or more of the SUSI API functions call parameters are out of defined range.

Actions

Verify Function Parameters.

#define SUSI_STATUS_INVALID_BLOCK_ALIGNMENT 0xFFFFFEFE

Description

The Block Alignment is incorrect.

Actions

Use Inputs and Outputs to correctly select input and outputs.

#define SUSI_STATUS_INVALID_BLOCK_LENGTH 0xFFFFFEFD

Description

This means that the Block length is too long.

Actions

Use Alignment Capabilities information to correctly align write access.

#define SUSI_STATUS_INVALID_DIRECTION 0xFFFFFEFC

Description

The current Direction Argument attempts to set GPIOs to a unsupported direction. I.E. Setting GPI to Output.

Actions

Use Inputs and Outputs to correctly select input and outputs.

#define SUSI_STATUS_INVALID_BITMASK 0xFFFFFEFB

Description

The Bitmask Selects bits/GPIOs which are not supported for the current ID.

Actions

Use Inputs and Outputs to probe supported bits.

#define SUSI_STATUS_RUNNING 0xFFFFFEFA

Description

Watchdog timer already started.

Actions

Call SusiWDogStop before retrying.

#define SUSI_STATUS_UNSUPPORTED 0xFFFFFCFF

Description

This function or ID is not supported at the actual hardware environment.

Actions

None.

#define SUSI_STATUS_NOT_FOUND 0xFFFFFBFF

Description

Selected device was not found

Actions

None.

#define SUSI_STATUS_TIMEOUT 0xFFFFFBFE

Description

Device has no response.

Actions

None.

#define SUSI_STATUS_BUSY_COLLISION 0xFFFFFBFD

Description

The selected device or ID is busy or a data collision is detected.

Actions

Retry.

#define SUSI_STATUS_READ_ERROR 0xFFFFFAFF

Description

An error is detected during a read operation.

Actions

Retry.

#define SUSI_STATUS_WRITE_ERROR 0xFFFFFAFE

Description

An error is detected during a write operation.

Actions

Retry.

#define SUSI_STATUS_MORE_DATA 0xFFFFF9FF

Description

The amount of available data exceeds the buffer size. Storage buffer overflow was prevented. Read count was larger than the defined buffer length.

Actions

Either increase the buffer size or reduce the block length.

#define SUSI_STATUS_ERROR 0xFFFFF0FF

Description

Generic error message. No further error details are available.

Actions

None.

```
#define SUSI_STATUS_SUCCESS                                    0
```

Description

The operation is successful.

Actions

None.

2.2 ID

```
#define SUSI_ID_UNKNOWN 0xFFFFFFFF
```

Description

Undefined/Unknown ID

```
#define SUSI_ID_BOARD_MANUFACTURER_STR 0
#define SUSI_ID_BOARD_NAME_STR 1
#define SUSI_ID_BOARD_REVISION_STR 2
#define SUSI_ID_BOARD_SERIAL_STR 3
#define SUSI_ID_BOARD_BIOS_REVISION_STR 4
#define SUSI_ID_BOARD_HW_REVISION_STR 5
#define SUSI_ID_BOARD_PLATFORM_TYPE_STR 6
```

Description

Board information string ID, use in SusiBoardGetStringA.

```
#define SUSI_ID_GET_SPEC_VERSION 0x00000000
#define SUSI_ID_BOARD_BOOT_COUNTER_VAL 0x00000001
#define SUSI_ID_BOARD_RUNNING_TIME_METER_VAL 0x00000002
#define SUSI_ID_BOARD_PNPID_VAL 0x00000003
#define SUSI_ID_BOARD_PLATFORM_REV_VAL 0x00000004
#define SUSI_ID_BOARD_DRIVER_VERSION_VAL 0x00010000
#define SUSI_ID_BOARD_LIB_VERSION_VAL 0x00010001
#define SUSI_ID_BOARD_FIRMWARE_VERSION_VAL 0x00010002
```

Description

Board information value ID, use in SusiBoardGetValue.

```
#define SUSI_ID_HWM_TEMP_CPU 0x00020000
#define SUSI_ID_HWM_TEMP_CHIPSET 0x00020001
#define SUSI_ID_HWM_TEMP_SYSTEM 0x00020002
#define SUSI_ID_HWM_TEMP_CPU2 0x00020003
#define SUSI_ID_HWM_TEMP_OEM0 0x00020004
#define SUSI_ID_HWM_TEMP_OEM1 0x00020005
#define SUSI_ID_HWM_TEMP_OEM2 0x00020006
#define SUSI_ID_HWM_TEMP_OEM3 0x00020007
#define SUSI_ID_HWM_TEMP_OEM4 0x00020008
#define SUSI_ID_HWM_TEMP_OEM5 0x00020009
```

Description

Board temperature value ID, use in SusiBoardGetValue.

```
#define SUSI_ID_HWM_VOLTAGE_VCORE 0x00021000
#define SUSI_ID_HWM_VOLTAGE_VCORE2 0x00021001
#define SUSI_ID_HWM_VOLTAGE_2V5 0x00021002
#define SUSI_ID_HWM_VOLTAGE_3V3 0x00021003
#define SUSI_ID_HWM_VOLTAGE_5V 0x00021004
#define SUSI_ID_HWM_VOLTAGE_12V 0x00021005
#define SUSI_ID_HWM_VOLTAGE_5VSB 0x00021006
#define SUSI_ID_HWM_VOLTAGE_3VSB 0x00021007
#define SUSI_ID_HWM_VOLTAGE_VBAT 0x00021008
#define SUSI_ID_HWM_VOLTAGE_5NV 0x00021009
#define SUSI_ID_HWM_VOLTAGE_12NV 0x0002100A
#define SUSI_ID_HWM_VOLTAGE_VTT 0x0002100B
#define SUSI_ID_HWM_VOLTAGE_24V 0x0002100C
#define SUSI_ID_HWM_VOLTAGE_DC 0x0002100D
#define SUSI_ID_HWM_VOLTAGE_DCSTBY 0x0002100E
#define SUSI_ID_HWM_VOLTAGE_OEM3 0x0002100F
#define SUSI_ID_HWM_VOLTAGE_OEM0 0x00021010
#define SUSI_ID_HWM_VOLTAGE_OEM1 0x00021011
#define SUSI_ID_HWM_VOLTAGE_OEM2 0x00021012
#define SUSI_ID_HWM_VOLTAGE_OEM3 0x00021013
#define SUSI_ID_HWM_VOLTAGE_1V05 0x00021014
#define SUSI_ID_HWM_VOLTAGE_1V5 0x00021015
#define SUSI_ID_HWM_VOLTAGE_1V8 0x00021016
```

Description

Board voltage value ID, use in SusiBoardGetValue.

```
#define SUSI_ID_HWM_FAN_CPU 0x00022000
#define SUSI_ID_HWM_FAN_SYSTEM 0x00022001
#define SUSI_ID_HWM_FAN_CPU2 0x00022002
#define SUSI_ID_HWM_FAN_OEM0 0x00022003
#define SUSI_ID_HWM_FAN_OEM1 0x00022004
#define SUSI_ID_HWM_FAN_OEM2 0x00022005
#define SUSI_ID_HWM_FAN_OEM3 0x00022006
#define SUSI_ID_HWM_FAN_OEM4 0x00022007
#define SUSI_ID_HWM_FAN_OEM5 0x00022008
#define SUSI_ID_HWM_FAN_OEM6 0x00022009
```

Description

Board fan speed value ID, use in SusiBoardGetValue and Smart Fan Functions.

```
#define SUSI_ID_HWM_CURRENT_OEM0 0x00023000
#define SUSI_ID_HWM_CURRENT_OEM1 0x00023001
#define SUSI_ID_HWM_CURRENT_OEM2 0x00023002
```

Description

Board current value ID, use in SusiBoardGetValue.

```
#define SUSI_ID_SMBUS_SUPPORTED 0x00030000
#define SUSI_ID_I2C_SUPPORTED 0x00030100
```

Description

Board supported information value ID, use in SusiBoardGetValue.

```
#define SUSI_ID_SMBUS_EXTERNAL 0
#define SUSI_ID_SMBUS_OEM0 1
#define SUSI_ID_SMBUS_OEM1 2
#define SUSI_ID_SMBUS_OEM2 3
#define SUSI_ID_SMBUS_OEM3 4
```

Description

SMBus device ID, use in SMBus Functions.

```
#define SUSI_ID_I2C_EXTERNAL 0
#define SUSI_ID_I2C_OEM0 1
#define SUSI_ID_I2C_OEM1 2
#define SUSI_ID_I2C_OEM2 3
```

Description

I^2C device ID, use in I2C Functions.

```
#define SUSI_ID_GPIO(GPIO_NUM) (GPIO_NUM)
#define SUSI_ID_GPIO_BANK(BANK_NUM) (0x00010000 + BANK_NUM)
```

Description

GPIO device ID, use in GPIO Functions.

```
#define SUSI_ID_BACKLIGHT_1 0
#define SUSI_ID_BACKLIGHT_2 1
#define SUSI_ID_BACKLIGHT_3 2
```

Description

Backlight device ID, use in Backlight Functions.

```
#define SUSI_ID_STORAGE_STD 0x00000000
#define SUSI_ID_STORAGE_OEM0 0x00000001
#define SUSI_ID_STORAGE_OEM1 0x00000002
```

Description

Storage device ID, use in Storage Functions.

```
#define SUSI_ID_THERMAL_PROTECT_1 0
#define SUSI_ID_THERMAL_PROTECT_2 1
#define SUSI_ID_THERMAL_PROTECT_3 2
#define SUSI_ID_THERMAL_PROTECT_4 3
```

Description

Thermal protection device ID, use in Thermal Protection Functions

```
#define SUSI_ID_WATCHDOG_1 0
#define SUSI_ID_WATCHDOG_2 1
#define SUSI_ID_WATCHDOG_3 2
```

Description

Watchdog device ID, use in Watchdog Functions.

2.3 Item ID

#define SUSI_ID_I2C_MAXIMUM_BLOCK_LENGTH	0x00000000

Description

Watchdog capabilities item ID, use in SusiI2CGetCaps.

#define SUSI_ID_GPIO_INPUT_SUPPORT	0x00000000
#define SUSI_ID_GPIO_OUTPUT_SUPPORT	0x00000001

Description

GPIO capabilities item ID, use in SusiGPIOGetCaps.

#define SUSI_ID_VGA_BRIGHTNESS_MAXIMUM	0x00010000
#define SUSI_ID_VGA_BRIGHTNESS_MINIMUM	0x00010001

Description

VGA capabilities item ID, use in SusiVgaGetCaps.

#define SUSI_ID_STORAGE_TOTAL_SIZE	0x00000000
#define SUSI_ID_STORAGE_BLOCK_SIZE	0x00000001
#define SUSI_ID_STORAGE_LOCK_STATUS	0x00010000
#define SUSI_ID_STORAGE_PSW_MAX_LEN	0x00010000

Description

Storage capabilities item ID, use in SusiStorageGetCaps.

#define SUSI_ID_WDT_SUPPORT_FLAGS	0x00000000
#define SUSI_ID_WDT_DELAY_MAXIMUM	0x00000001
#define SUSI_ID_WDT_DELAY_MINIMUM	0x00000002
#define SUSI_ID_WDT_EVENT_MAXIMUM	0x00000003
#define SUSI_ID_WDT_EVENT_MINIMUM	0x00000004
#define SUSI_ID_WDT_RESET_MAXIMUM	0x00000005
#define SUSI_ID_WDT_RESET_MINIMUM	0x00000006
#define SUSI_ID_WDT_UNIT_MINIMUM	0x0000000F
#define SUSI_ID_WDT_DELAY_TIME	0x00010001
#define SUSI_ID_WDT_EVENT_TIME	0x00010002
#define SUSI_ID_WDT_RESET_TIME	0x00010003
#define SUSI_ID_WDT_EVENT_TYPE	0x00010004

Description

Watchdog capabilities item ID, use in SusiWDogGetCaps.

#define SUSI_ID_FC_CONTROL_SUPPORT_FLAGS	0x00000000

```
#define SUSI_ID_FC_AUTO_SUPPORT_FLAGS                    0x00000001
```

Description

Fan control capabilities item ID, use in

SusiFanControlGetCaps.

#define SUSI_ID_TP_EVENT_SUPPORT_FLAGS	0x00000000
#define SUSI_ID_TP_EVENT_TRIGGER_MAXIMUM	0x00000001
#define SUSI_ID_TP_EVENT_TRIGGER_MINIMUM	0x00000002
#define SUSI_ID_TP_EVENT_CLEAR_MAXIMUM	0x00000003
#define SUSI_ID_TP_EVENT_CLEAR_MINIMUM	0x00000004

Description

Thermal protection capabilities item ID, use in SusiThermalProtectionGetCaps.

3 SUSI API

The SUSI APIs provide functions to control ADVANTECH platforms. SUSI API functions are based on a dynamic library. SUSI API can be implemented in various other programming languages.

3.1 Initialization Functions

3.1.1 SusiLibInitialize

uint32_t SUSI_API SusiLibInitialize(void)

Description:

General initialization of the SUSI API. Prior to calling any SUSI API functions, the library needs to be initialized by calling this function. The status code for all SUSI API function will be SUSI_STATUS_NOT_INITIALIZED unless this function is called.

Parameters:

None

Return Status Code:

Condition	Return Value
Library initialized	SUSI_STATUS_INITIALIZED
Fail	SUSI_STATUS_NOT_INITIALIZED
Success	SUSI_STATUS_SUCCESS

3.1.2 SusiLibUninitialize

uint32_t SUSI_API SusiLibUninitialize(void)

Description:

General function to uninitialize the SUSI API library that should be called before program exit. In a dynamic library environment this function is not expected to replace the native uninitialized routines. It is expected that in this environments this function has no functionality.

Parameters:

None

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
Success	SUSI_STATUS_SUCCESS

3.2 Information Functions

3.2.1 SusiBoardGetValue

uint32_t SUSI_API SusiBoardGetValue(uint32_t Id, uint32_t *pValue)

Description:

Getting information about the hardware platform in value format.

Parameters:

Id

Selects target value. See **Table 1** to **Table 5**.

pValue

Pointer to a buffer that receives the value's data.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
pValue==NULL	SUSI _STATUS_INVALID_PARAMETER
Unknown Id	SUSI _STATUS_UNSUPPORTED
Device unsupported	SUSI _STATUS_UNSUPPORTED
Success	SUSI_STATUS_SUCCESS

Table 1 Board information value ID

Id	Description	Unit
SUSI_ID_GET_SPEC_VERSION	API Specification Version	
SUSI_ID_BOARD_BOOT_COUNTER_VAL	Boot Counter	boot
SUSI_ID_BOARD_RUNNING_TIME_METER_VAL	Running Time Meter	hour
SUSI_ID_BOARD_PNPID_VAL	Board Vendor PNPID	
SUSI_ID_BOARD_PLATFORM_REV_VAL	Platform revision	
SUSI_ID_BOARD_DRIVER_VERSION_VAL	Driver version	
SUSI_ID_BOARD_LIB_VERSION_VAL	Library version	
SUSI_ID_BOARD_FIRMWARE_VERSION_VAL	Firmware version	

Table 2 Board voltage value ID

Id	Description	Unit
SUSI_ID_HWM_VOLTAGE_VCORE	CPU Core voltage	millivolt

SUSI_ID_HWM_VOLTAGE_VCORE2	Second CPU Core voltage	millivolt
SUSI_ID_HWM_VOLTAGE_2V5	2.5V	millivolt
SUSI_ID_HWM_VOLTAGE_3V3	3.3V	millivolt
SUSI_ID_HWM_VOLTAGE_5V	5V	millivolt
SUSI_ID_HWM_VOLTAGE_12V	12V	millivolt
SUSI_ID_HWM_VOLTAGE_5VSB	5V Standby	millivolt
SUSI_ID_HWM_VOLTAGE_3VSB	3V Standby	millivolt
SUSI_ID_HWM_VOLTAGE_VBAT	CMOS Battery voltage	millivolt
SUSI_ID_HWM_VOLTAGE_5NV	-5V	millivolt
SUSI_ID_HWM_VOLTAGE_12NV	-12V	millivolt
SUSI_ID_HWM_VOLTAGE_VTT	DIMM voltage	millivolt
SUSI_ID_HWM_VOLTAGE_24V	24V	millivolt
SUSI_ID_HWM_VOLTAGE_OEM0~3	Other voltages	millivolt

Table 3 Board temperature value ID

Id	Description	Unit
SUSI_ID_HWM_TEMP_CPU	CPU temperature	0.1 Kelvin
SUSI_ID_HWM_TEMP_CHIPSET	Chipset temperature	0.1 Kelvin
SUSI_ID_HWM_TEMP_SYSTEM	System temperature	0.1 Kelvin
SUSI_ID_HWM_TEMP_CPU2	CPU2 temperature	0.1 Kelvin
SUSI_ID_HWM_TEMP_OEM0~5	Other temperatures	0.1 Kelvin

Table 4 Board fan speed value ID

Id	Description	Unit
SUSI_ID_HWM_FAN_CPU	CPU fan speed	RPM
SUSI_ID_HWM_FAN_SYSTEM	System fan speed	RPM
SUSI_ID_HWM_FAN_CPU2	Second CPU fan speed	RPM
SUSI_ID_HWM_FAN_OEM0~6	Other fans	RPM

Table 5 Board support information value ID

Id	Description
SUSI_ID_SMBUS_SUPPORTED	Mask flags: SUSI_SMBUS_EXTERNAL_SUPPORTED SUSI_SMBUS_OEM0_SUPPORTED SUSI_SMBUS_OEM1_SUPPORTED SUSI_SMBUS_OEM2_SUPPORTED SUSI_SMBUS_OEM3_SUPPORTED

SUSI_ID_I2C_SUPPORTED	Mask flags: SUSI_ I2C_EXTERNAL_SUPPORTED SUSI_ I2C_OEM0_SUPPORTED SUSI_ I2C_OEM1_SUPPORTED SUSI_ I2C_OEM2_SUPPORTED

3.2.2 SusiBoardGetStringA

uint32_t SUSI_API SusiBoardGetStringA(uint32_t Id, char *pBuffer, uint32_t *pBufLen)

Description:

Text information about the hardware platform.

Parameters:

Id

Selects target string. See **Table 6**.

pBuffer

Pointer to a buffer that receives the value's data.

pBufLen

Pointer to a variable that specifies the size, in bytes, of the buffer pointed to by the pBuffer parameter. When the function returns, this variable contains the size of the data copied to pBuffer including the terminating null character.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
pBufLen==NULL	SUSI _STATUS_INVALID_PARAMETER
pBufLen!=NULL&&*pBufLen&&pBuffer==NULL	SUSI _STATUS_INVALID_PARAMETER
Unknown Id	SUSI _STATUS_UNSUPPORTED
Device unsupported	SUSI _STATUS_UNSUPPORTED
strlength + 1 > *pBufLen	SUSI_STATUS_MORE_DATA
Success	SUSI_STATUS_SUCCESS

Table 6 Board information string ID

Id	Description
SUSI_ID_BOARD_MANUFACTURER_STR	Board Manufacturer Name
SUSI_ID_BOARD_NAME_STR	Board Name
SUSI_ID_BOARD_REVISION_STR	Board Revision

SUSI_ID_BOARD_SERIAL_STR	Board Serial Number
SUSI_ID_BOARD_BIOS_REVISION_STR	Board BIOS Revision
SUSI_ID_BOARD_HW_REVISION_STR	Hardware Revision
SUSI_ID_BOARD_PLATFORM_TYPE_STR	Platform type

3.3 Backlight Functions

This function sub set facilitates backlight control for Integrated flat panel displays, typically LVDS.

Table 7 Backlight ID

Id	Description
SUSI_ID_BACKLIGHT_1	Backlight Local Flat Panel 1
SUSI_ID_BACKLIGHT_2	Backlight Local Flat Panel 2
SUSI_ID_BACKLIGHT_3	Backlight Local Flat Panel 3

Table 8 Backlight Enable Values

Name	Description
SUSI_BACKLIGHT_SET_ON	Signifies that the Backlight be Enabled
SUSI_BACKLIGHT_SET_OFF	Signifies that the Backlight be Disabled

3.3.1 SusiVgaGetCaps

uint32_t SUSI_API SusiVgaGetCaps(uint32_t Id, uint32_t ItemId, uint32_t *pValue);

Description:

Gets VGA capabilities.

Parameters:

Id

Selects target device. See **Table 7**.

ItemId

Selects target capability. See **Table 9**.

pValue

Pointer to a buffer that receives the target capability.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
pValue==NULL	SUSI _STATUS_INVALID_PARAMETER
Unknown Id or ItemId	SUSI _STATUS_UNSUPPORTED
Success	SUSI_STATUS_SUCCESS

Table 9 VGA capabilities item Id

Item Id	Description
SUSI_ID_VGA_BRIGHTNESS_MAXIMUM	Maximum backlight value
SUSI_ID_VGA_BRIGHTNESS_MINIMUM	Minimum backlight value

3.3.2 SusiVgaGetBacklightEnable

uint32_t SUSI_API SusiVgaGetBacklightEnable(uint32_t Id, uint32_t *pEnable)

Description:

Gets current Backlight Enable state for specified Flat Panel.

Parameters:

Id

Selects target device. See **Table 7**.

pEnable

Pointer to a buffer that receives the current backlight enable state. See **Table 8**.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
pEnable==NULL	SUSI _STATUS_INVALID_PARAMETER
Unknown Id	SUSI _STATUS_UNSUPPORTED
Device unsupported	SUSI _STATUS_UNSUPPORTED
Success	SUSI_STATUS_SUCCESS

3.3.3 SusiVgaSetBacklightEnable

uint32_t SUSI_API SusiVgaSetBacklightEnable(uint32_t Id, uint32_t Enable)

Description:

Enables or disable the backlight of the selected flat panel display

Parameters:

Id

Selects target device. See **Table 7**.

Enable

Backlight Enable options. See **Table 8**.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
Unknown Id	SUSI _STATUS_UNSUPPORTED
Device unsupported	SUSI _STATUS_UNSUPPORTED
Success	SUSI_STATUS_SUCCESS

3.3.4 SusiVgaGetBacklightBrightness

uint32_t SUSI_API SusiVgaGetBacklightBrightness(uint32_t Id, uint32_t *pBright)

Description:

Reads the current brightness of the selected flat panel display.

Parameters:

Id

Selects target device. See **Table 7**.

pBright

Pointer to a buffer that receives the current backlight brightness value.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
pBright==NULL	SUSI _STATUS_INVALID_PARAMETER
Unknown Id	SUSI _STATUS_UNSUPPORTED
Device unsupported	SUSI _STATUS_UNSUPPORTED
Success	SUSI_STATUS_SUCCESS

3.3.5 SusiVgaSetBacklightBrightness

uint32_t SUSI_API SusiVgaSetBacklightBrightness(uint32_t Id, uint32_t Bright)

Description:

Reads the current brightness of the selected flat panel display.

Parameters:

Id

Selects target device. See Table 7.

Bright

Backlight Brightness value.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
Bright > MAX value || Bright < MIN value	SUSI _STATUS_INVALID_PARAMETER
Unknown Id	SUSI _STATUS_UNSUPPORTED
Device unsupported	SUSI _STATUS_UNSUPPORTED
Success	SUSI_STATUS_SUCCESS

3.3.6 SusiVgaGetBacklightLevel

uint32_t SUSI_API SusiVgaGetBacklightLevel(uint32_t Id, uint32_t *pLevel)

Description:

Reads the current brightness level of the selected flat panel display.

Parameters:

Id

Selects target device. See **Table 7**.

pLevel

Pointer to a buffer that receives the current backlight brightness level. See **Table 10**.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
pLevel==NULL	SUSI _STATUS_INVALID_PARAMETER
Unknown Id	SUSI _STATUS_UNSUPPORTED
Device unsupported	SUSI _STATUS_UNSUPPORTED
Success	SUSI_STATUS_SUCCESS

Table 10 Brightness level range definition

Name	Description
SUSI_BACKLIGHT_LEVEL_MAXIMUM	Maximum backlight level is 9
SUSI_BACKLIGHT_LEVEL_MINIMUM	Minimum backlight level is 0

3.3.7 SusiVgaSetBacklightLevel

uint32_t SUSI_API SusiVgaSetBacklightLevel(uint32_t Id, uint32_t Level)

Description:

Sets the brightness level of the selected flat panel display.

Parameters:

Id

Selects target device. See Table 7.

Level

Backlight Brightness level. See **Table 10**.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
Level>SUSI_BACKLIGHT_LEVEL_MAXIMUM	SUSI _STATUS_INVALID_PARAMETER
Unknown Id	SUSI _STATUS_UNSUPPORTED
Device unsupported	SUSI _STATUS_UNSUPPORTED
Success	SUSI_STATUS_SUCCESS

3.3.8 SusiVgaGetPolarity

uint32_t SUSI_API SusiVgaGetPolarity(uint32_t Id, uint32_t *pPolarity)

Description:

Reads the current backlight polarity of the selected flat panel display.

Parameters:

Id

Selects target device. See **Table 7**.

pPolarity

Pointer to a buffer that receives the current backlight polarity. See **Table 11**.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
pPolarity==NULL	SUSI _STATUS_INVALID_PARAMETER
Unknown Id	SUSI _STATUS_UNSUPPORTED
Success	SUSI_STATUS_SUCCESS

Table 11 Brightness polarity definition

Name	Description
SUSI_BACKLIGHT_POLARITY_ON	Backlight signal polarity ON
SUSI_BACKLIGHT_ POLARITY_OFF	Backlight signal polarity OFF

3.3.9 SusiVgaSetPolarity

uint32_t SUSI_API SusiVgaSetPolarity(uint32_t Id, uint32_t Polarity)

Description:

Sets the polarity of the selected flat panel display.

Parameters:

Id

Selects target device. See **Table 7**.

Polarity

Polarity state. See **Table 11**.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
Unknown Id	SUSI _STATUS_UNSUPPORTED
Success	SUSI_STATUS_SUCCESS

3.3.10 SusiVgaGetFrequency

uint32_t SUSI_API SusiVgaGetFrequency(uint32_t Id, uint32_t *pFrequency)

Description:

Reads the current backlight frequency of the selected flat panel display.

Parameters:

Id

Selects target device. See **Table 7**.

pFrequency

Pointer to a buffer that receives the current backlight frequency. (Unit: Hz)

Return Status Code:

Condition	Return Value

Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
pFrequency==NULL	SUSI _STATUS_INVALID_PARAMETER
Unknown Id	SUSI _STATUS_UNSUPPORTED
Device unsupported	SUSI _STATUS_UNSUPPORTED
Success	SUSI_STATUS_SUCCESS

3.3.11 SusiVgaSetFrequency

uint32_t SUSI_API SusiVgaSetFrequency(uint32_t Id, uint32_t Frequency)

Description:

Sets the frequency of the selected flat panel display.

Parameters:

Id

Selects target device. See **Table 7**.

Polarity

Frequency value. (Unit: Hz)

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
Unknown Id	SUSI _STATUS_UNSUPPORTED
Device unsupported	SUSI _STATUS_UNSUPPORTED
Success	SUSI_STATUS_SUCCESS

3.4 I2C Functions

I^{2}C APIs support standard 7 and 10 bits slave address mode. I^{2}C APIs also support word command, it needs encode before set to parameter, see Table 13

Table 12 I^{2}C ID

Id	Description
SUSI_ID_I2C_EXTERNAL	Main I^{2}C host device
SUSI_ID_I2C_OEM0~2	Other I^{2}C host devices

Table 13 I^{2}C command encode

Type	Description
Standard command	Byte command
Extend command	Word command | 0x800000000 Ex. 0x8000FABC
No command	0x4000xxxx, ignore command parameter

3.4.1 SusiI2CGetCaps

uint32_t SUSI_API SusiI2CGetCaps(uint32_t Id, uint32_t ItemId, uint32_t *pValue)

Description:

Gets I^{2}C capabilities.

Parameters:

Id

Selects target device. See **Table 12**.

ItemId

Selects target capability. See **Table 14**.

pValue

Pointer to a buffer that receives the target capability.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
pValue==NULL	SUSI _STATUS_INVALID_PARAMETER
Unknown Id or ItemId	SUSI _STATUS_UNSUPPORTED
Success	SUSI_STATUS_SUCCESS

Table 14 I^2C capabilities item Id

Item Id	Description
SUSI_ID_I2C_MAXIMUM_BLOCK_LENGTH	I^2C maximum block length

3.4.2 SusiI2CWriteReadCombine

uint32_t SUSI_API SusiI2CWriteReadCombine(uint32_t Id, uint8_t Addr, uint8_t *pWBuffer, uint32_t WriteLen, uint8_t *pRBuffer, uint32_t ReadLen)

Description:

Universal function for read and write operations to the I^2C bus.

Parameters:

Id

Selects target device. See **Table 14**.

Addr

First byte of I^2C device address. 7-bit address only.

pWBuffer

Pointer to a buffer containing the data to be transferred. This parameter can be NULL if the data is not required.

WriteLen

Size in bytes of the information pointed to by the pWBuffer parameter. If pWBuffer is NULL this will be ignored.

pRBuffer

Pointer to a buffer that receives the read data. This parameter can be NULL if the data is not required.

ReadLen

Size in bytes of the buffer pointed to by the pRBuffer parameter. If pRBuffer is NULL this will be ignored.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
(WriteLen>1)&&(pWBuffer==NULL)	SUSI _STATUS_INVALID_PARAMETER
(RBufLen>1)&&(pRBuffer==NULL)	SUSI _STATUS_INVALID_PARAMETER
(WriteLen==0)&&(RBufLen==0)	SUSI _STATUS_INVALID_PARAMETER

Unknown Id	SUSI _STATUS_UNSUPPORTED
Bus Busy SDA/SDC low	SUSI_STATUS_BUSY_COLLISION
Arbitration Error/Collision Error On Write 1 write cycle SDA Remains low	SUSI_STATUS_BUSY_COLLISION
Time-out due to clock stretching	SUSI_STATUS_TIMEOUT
Address Non-ACK	SUSI_STATUS_NOT_FOUND
Write Non-ACK	SUSI_STATUS_WRITE_ERROR
Success	SUSI_STATUS_SUCCESS

3.4.3 SusiI2CReadTransfer

uint32_t SUSI_API SusiI2CReadTransfer(uint32_t Id, uint32_t Addr, uint32_t Cmd, uint8_t *pBuffer, uint32_t ReadLen)

Description:

Reads from a specific register in the selected I^2C device. Reads from I^2C device at the I^2C address Addr the amount of ReadLen bytes to the buffer pBuffer while using the device specific command Cmd. Depending on the addressed I^2C device Cmd can be a specific command or a byte offset.

Parameters:

Id

Selects target device. See **Table 14**.

Addr

Encoded 7/10 Bit I^2C Device Address.

Cmd

Encoded I^2C Device Command / Index. See **Table 13**.

pBuffer

Pointer to a buffer that receives the read data.

ReadLen

Size in bytes of the buffer pointed to by the pBuffer parameter.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
pBuffer==NULL || ReadLen==0	SUSI _STATUS_INVALID_PARAMETER
Unknown Id	SUSI _STATUS_UNSUPPORTED

Bus Busy SDA/SDC low	SUSI_STATUS_BUSY_COLLISION
Arbitration Error/Collision Error On Write 1 write cycle SDA Remains low	SUSI_STATUS_BUSY_COLLISION
Time-out due to clock stretching	SUSI_STATUS_TIMEOUT
Address Non-ACK	SUSI_STATUS_NOT_FOUND
Write Non-ACK	SUSI_STATUS_WRITE_ERROR
Success	SUSI_STATUS_SUCCESS

3.4.4 SusiI2CWriteTransfer

uint32_t SUSI_API SusiI2CWriteTransfer(uint32_t Id, uint32_t Addr, uint32_t Cmd, uint8_t *pBuffer, uint32_t ByteCnt)

Description:

Write to a specific register in the selected I^2C device. Writes to an I^2C device at the I^2C address Addr the amount of ByteCnt bytes from the buffer *pBuffer while using the device specific command Cmd. Depending on the addressed I^2C device Cmd can be a specific command or a byte offset

Parameters:

Id

Selects target device. See **Table 14**.

Addr

Encoded 7/10 Bit I^2C Device Address.

Cmd

Encoded I^2C Device Command / Index. See **Table 13**.

pBuffer

Pointer to a buffer that receives the write data.

ByteCnt

Size in bytes of the buffer pointed to by the pBuffer parameter.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
pBuffer==NULL || ByteCnt==0	SUSI _STATUS_INVALID_PARAMETER
ByteCnt > MaxLength	SUSI_STATUS_INVALID_BLOCK_LENGTH
Unknown Id	SUSI _STATUS_UNSUPPORTED

Bus Busy SDA/SDC low	SUSI_STATUS_BUSY_COLLISION
Arbitration Error/Collision Error On Write 1 write cycle SDA Remains low	SUSI_STATUS_BUSY_COLLISION
Time-out due to clock stretching	SUSI_STATUS_TIMEOUT
Address Non-ACK	SUSI_STATUS_NOT_FOUND
Write Non-ACK	SUSI_STATUS_WRITE_ERROR
Success	SUSI_STATUS_SUCCESS

3.4.5 SusiI2CProbeDevice

uint32_t SUSI_API SusiI2CProbeDevice(uint32_t Id, uint32_t Addr)

Description:

Probes I^2C address to test I^2C device present.

Parameters:

Id

Selects target device. See **Table 14**.

Addr

Encoded 7/10 Bit I^2C Device Address.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
Unknown Id	SUSI _STATUS_UNSUPPORTED
Bus Busy SDA/SDC low	SUSI_STATUS_BUSY_COLLISION
Arbitration Error/Collision Error On Write 1 write cycle SDA Remains low	SUSI_STATUS_BUSY_COLLISION
Time-out due to clock stretching	SUSI_STATUS_TIMEOUT
Address Non-ACK	SUSI_STATUS_NOT_FOUND
Write Non-ACK	SUSI_STATUS_WRITE_ERROR
Success	SUSI_STATUS_SUCCESS

3.4.6 SusiI2CGetFrequency

uint32_t SUSI_API SusiI2CGetFrequency(uint32_t Id, uint32_t *pFreq)

Description:

Get I^{2}C clock frequency.

Parameters:

Id

Selects target device. See **Table 14**.

pFreq

Pointer to a buffer that receives the I^{2}C clock frequency value. (Unit: Hz)

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
Unknown Id	SUSI _STATUS_UNSUPPORTED
Device unsupported	SUSI _STATUS_UNSUPPORTED
pFreq == NULL	SUSI _STATUS_INVALID_PARAMETER
Success	SUSI_STATUS_SUCCESS

3.4.7 SusiI2CSetFrequency

uint32_t SUSI_API SusiI2CSetFrequency(uint32_t Id, uint32_t Freq)

Description:

Set I^{2}C clock frequency.

Parameters:

Id

Selects target device. See **Table 14**.

Freq

I^{2}C clock frequency value. (Unit: Hz)

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
Unknown Id	SUSI _STATUS_UNSUPPORTED
Device unsupported	SUSI _STATUS_UNSUPPORTED
Success	SUSI_STATUS_SUCCESS

3.5 SMBus Functions

SMBus is the System Management Bus defined by Intel® Corporation in 1995. It is used in personal computers and servers for low-speed system management communications.

Table 15 SMBus ID

Id	Description
SUSI_ID_ SMBUS _EXTERNAL	Main SMBus host device
SUSI_ID_SMBUS_OEM0~3	Other SMBus host devices

3.5.1 SusiSMBReadByte

uint32_t SUSI_API SusiSMBReadByte(uint32_t Id, uint8_t Addr, uint8_t Cmd, uint8_t *pData)

Description:

Read a byte of data from the target slave device in the SMBus.

Parameters:

Id

Selects target device. See **Table 15**.

Addr

Specifies the 8-bit device address, ranging from 0x00 to 0xFF. Whether to give a 1 (read) or 0 (write) to the LSB of slave address could be ignored.

Cmd

Specifies the offset or command of the device register to read data from.

pData

Pointer to a variable in which the function reads the byte data.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
pData==NULL	SUSI _STATUS_INVALID_PARAMETER
Unknown Id	SUSI _STATUS_UNSUPPORTED
Bus Busy SDA/SDC low	SUSI_STATUS_BUSY_COLLISION
Arbitration Error/Collision Error On Write 1 write cycle SDA Remains low	SUSI_STATUS_BUSY_COLLISION

Time-out due to clock stretching	SUSI_STATUS_TIMEOUT
Address Non-ACK	SUSI_STATUS_NOT_FOUND
Write Non-ACK	SUSI_STATUS_WRITE_ERROR
Success	SUSI_STATUS_SUCCESS

3.5.2 SusiSMBWriteByte

uint32_t SUSI_API SusiSMBWriteByte(uint32_t Id, uint8_t Addr, uint8_t Cmd, uint8_t Data)

Description:

Write a byte of data to the target slave device in the SMBus.

Parameters:

Id

Selects target device. See **Table 15**.

Addr

Specifies the 8-bit device address, ranging from 0x00 to 0xFF. Whether to give a 1 (read) or 0 (write) to the LSB of slave address could be ignored.

Cmd

Specifies the offset or command of the device register to write data to.

Data

Specifies the byte data to be written.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
Unknown Id	SUSI _STATUS_UNSUPPORTED
Bus Busy SDA/SDC low	SUSI_STATUS_BUSY_COLLISION
Arbitration Error/Collision Error On Write 1 write cycle SDA Remains low	SUSI_STATUS_BUSY_COLLISION
Time-out due to clock stretching	SUSI_STATUS_TIMEOUT
Address Non-ACK	SUSI_STATUS_NOT_FOUND
Write Non-ACK	SUSI_STATUS_WRITE_ERROR
Success	SUSI_STATUS_SUCCESS

3.5.3 SusiSMBReadWord

uint32_t SUSI_API SusiSMBReadWord(uint32_t Id, uint8_t Addr, uint8_t Cmd, uint16_t

*pData)

Description:

Read a word of data from the target slave device in the SMBus.

Parameters:

Id

Selects target device. See **Table 15**.

Addr

Specifies the 8-bit device address, ranging from 0x00 to 0xFF. Whether to give a 1 (read) or 0 (write) to the LSB of slave address could be ignored.

Cmd

Specifies the offset or command of the device register to read data from.

pData

Pointer to a variable in which the function reads the word data.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
pData==NULL	SUSI _STATUS_INVALID_PARAMETER
Unknown Id	SUSI _STATUS_UNSUPPORTED
Bus Busy SDA/SDC low	SUSI_STATUS_BUSY_COLLISION
Arbitration Error/Collision Error On Write 1 write cycle SDA Remains low	SUSI_STATUS_BUSY_COLLISION
Time-out due to clock stretching	SUSI_STATUS_TIMEOUT
Address Non-ACK	SUSI_STATUS_NOT_FOUND
Write Non-ACK	SUSI_STATUS_WRITE_ERROR
Success	SUSI_STATUS_SUCCESS

3.5.4 SusiSMBWriteWord

uint32_t SUSI_API SusiSMBWriteWord(uint32_t Id, uint8_t Addr, uint8_t Cmd, uint16_t Data)

Description:

Write a word of data to the target slave device in the SMBus.

Parameters:

Id

Selects target device. See **Table 15**.

Addr

Specifies the 8-bit device address, ranging from 0x00 to 0xFF. Whether to give a 1 (read) or 0 (write) to the LSB of slave address could be ignored.

Cmd

Specifies the offset or command of the device register to write data to.

Data

Specifies the word data to be written.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
Unknown Id	SUSI _STATUS_UNSUPPORTED
Bus Busy SDA/SDC low	SUSI_STATUS_BUSY_COLLISION
Arbitration Error/Collision Error On Write 1 write cycle SDA Remains low	SUSI_STATUS_BUSY_COLLISION
Time-out due to clock stretching	SUSI_STATUS_TIMEOUT
Address Non-ACK	SUSI_STATUS_NOT_FOUND
Write Non-ACK	SUSI_STATUS_WRITE_ERROR
Success	SUSI_STATUS_SUCCESS

3.5.5 SusiSMBReceiveByte

uint32_t SUSI_API SusiSMBReceiveByte(uint32_t Id, uint8_t Addr, uint8_t *pData)

Description:

Receive a byte of data from the target slave device in the SMBus.

Parameters:

Id

Selects target device. See **Table 15**.

Addr

Specifies the 8-bit device address, ranging from 0x00 to 0xFF. Whether to give a 1 (read) or 0 (write) to the LSB of slave address could be ignored.

pData

Pointer to a variable in which the function receive the byte data.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
pBuffer==NULL	SUSI _STATUS_INVALID_PARAMETER
Unknown Id	SUSI _STATUS_UNSUPPORTED
Bus Busy SDA/SDC low	SUSI_STATUS_BUSY_COLLISION
Arbitration Error/Collision Error On Write 1 write cycle SDA Remains low	SUSI_STATUS_BUSY_COLLISION
Time-out due to clock stretching	SUSI_STATUS_TIMEOUT
Address Non-ACK	SUSI_STATUS_NOT_FOUND
Write Non-ACK	SUSI_STATUS_WRITE_ERROR
Success	SUSI_STATUS_SUCCESS

3.5.6 SusiSMBSendByte

uint32_t SUSI_API SusiSMBSendByte(uint32_t Id, uint8_t Addr, uint8_t Data)

Description:

Send a byte of data to the target slave device in the SMBus.

Parameters:

Id

Selects target device. See **Table 15.**

Addr

Specifies the 8-bit device address, ranging from 0x00 to 0xFF. Whether to give a 1 (read) or 0 (write) to the LSB of slave address could be ignored.

Data

Specifies the word data to be sent.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
Unknown Id	SUSI _STATUS_UNSUPPORTED
Bus Busy SDA/SDC low	SUSI_STATUS_BUSY_COLLISION
Arbitration Error/Collision Error On Write 1 write cycle SDA Remains low	SUSI_STATUS_BUSY_COLLISION

Time-out due to clock stretching	SUSI_STATUS_TIMEOUT
Address Non-ACK	SUSI_STATUS_NOT_FOUND
Write Non-ACK	SUSI_STATUS_WRITE_ERROR
Success	SUSI_STATUS_SUCCESS

3.5.7 SusiSMBReadQuick

uint32_t SUSI_API SusiSMBReadQuick(uint32_t Id, uint8_t Addr)

Description:

Turn SMBus device function off (on) or disable (enable) a specific device mode.

Parameters:

Id

Selects target device. See **Table 15**.

Addr

Specifies the 8-bit device address, ranging from 0x00 to 0xFF. Whether to give a 1 (read) or 0 (write) to the LSB of slave address could be ignored.

Return Status Code:

Condition	**Return Value**
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
Unknown Id	SUSI _STATUS_UNSUPPORTED
Bus Busy SDA/SDC low	SUSI_STATUS_BUSY_COLLISION
Arbitration Error/Collision Error On Write 1 write cycle SDA Remains low	SUSI_STATUS_BUSY_COLLISION
Time-out due to clock stretching	SUSI_STATUS_TIMEOUT
Address Non-ACK	SUSI_STATUS_NOT_FOUND
Success	SUSI_STATUS_SUCCESS

3.5.8 SusiSMBWriteQuick

uint32_t SUSI_API SusiSMBWriteQuick(uint32_t Id, uint8_t Addr)

Description:

Turn SMBus device function off (on) or disable (enable) a specific device mode.

Parameters:

Id

Selects target device. See **Table 15**.

Addr

Specifies the 8-bit device address, ranging from 0x00 to 0xFF. Whether to give a 1 (read) or 0 (write) to the LSB of slave address could be ignored.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
Unknown Id	SUSI _STATUS_UNSUPPORTED
Bus Busy SDA/SDC low	SUSI_STATUS_BUSY_COLLISION
Arbitration Error/Collision Error On Write 1 write cycle SDA Remains low	SUSI_STATUS_BUSY_COLLISION
Time-out due to clock stretching	SUSI_STATUS_TIMEOUT
Address Non-ACK	SUSI_STATUS_NOT_FOUND
Success	SUSI_STATUS_SUCCESS

3.5.9 SusiSMBReadBlock

uint32_t SUSI_API SusiSMBReadBlock(uint32_t Id, uint8_t Addr, uint8_t Cmd, uint8_t *pBuffer, uint32_t *pLength)

Description:

Read multi-data from the target slave device in the SMBus.

Parameters:

Id

Selects target device. See **Table 15**.

Addr

Specifies the 8-bit device address, ranging from 0x00 to 0xFF. Whether to give a 1 (read) or 0 (write) to the LSB of slave address could be ignored.

Cmd

Specifies the offset or command of the device register to read data from.

pBuffer

Pointer to a byte array in which the function reads the block data.

pLength

Pointer to a byte in which specifies the number of bytes to be read and also return succeed bytes.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
pBuffer==NULL	SUSI _STATUS_INVALID_PARAMETER
Unknown Id	SUSI _STATUS_UNSUPPORTED
Bus Busy SDA/SDC low	SUSI_STATUS_BUSY_COLLISION
Arbitration Error/Collision Error On Write 1 write cycle SDA Remains low	SUSI_STATUS_BUSY_COLLISION
Time-out due to clock stretching	SUSI_STATUS_TIMEOUT
Address Non-ACK	SUSI_STATUS_NOT_FOUND
Write Non-ACK	SUSI_STATUS_WRITE_ERROR
Success	SUSI_STATUS_SUCCESS

3.5.10 SusiSMBWriteBlock

uint32_t SUSI_API SusiSMBWriteBlock(uint32_t Id, uint8_t Addr, uint8_t Cmd, uint8_t *pBuffer, uint32_t Length)

Description:

Write multi-data from the target slave device in the SMBus.

Parameters:

Id

Selects target device. See **Table 15**.

Addr

Specifies the 8-bit device address, ranging from 0x00 to 0xFF. Whether to give a 1 (read) or 0 (write) to the LSB of slave address could be ignored.

Cmd

Specifies the offset or command of the device register to write data to.

pBuffer

Pointer to a byte array in which the function writes the block data.

Length

Specifies the number of bytes to be write.

Return Status Code:

Condition	Return Value

Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
pBuffer==NULL	SUSI _STATUS_INVALID_PARAMETER
Unknown Id	SUSI _STATUS_UNSUPPORTED
Bus Busy SDA/SDC low	SUSI_STATUS_BUSY_COLLISION
Arbitration Error/Collision Error On Write 1 write cycle SDA Remains low	SUSI_STATUS_BUSY_COLLISION
Time-out due to clock stretching	SUSI_STATUS_TIMEOUT
Address Non-ACK	SUSI_STATUS_NOT_FOUND
Write Non-ACK	SUSI_STATUS_WRITE_ERROR
Success	SUSI_STATUS_SUCCESS

3.5.11 SusiSMBI2CReadBlock

uint32_t SUSI_API SusiSMBI2CReadBlock(uint32_t Id, uint8_t Addr, uint8_t Cmd, uint8_t *pBuffer, uint32_t *pLength)

Description:

Read multi-data using I^2C block protocol from the target slave device in the SMBus.

Parameters:

Id

Selects target device. See **Table 15**.

Addr

Specifies the 8-bit device address, ranging from 0x00 to 0xFF. Whether to give a 1 (read) or 0 (write) to the LSB of slave address could be ignored.

Cmd

Specifies the offset or command of the device register to read data from.

pBuffer

Pointer to a byte array in which the function reads the block data.

pLength

Pointer to a byte in which specifies the number of bytes to be read and also return succeed bytes.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
pBuffer==NULL	SUSI _STATUS_INVALID_PARAMETER

Unknown Id	SUSI _STATUS_UNSUPPORTED
Device unsupported	SUSI _STATUS_UNSUPPORTED
Bus Busy SDA/SDC low	SUSI_STATUS_BUSY_COLLISION
Arbitration Error/Collision Error On Write 1 write cycle SDA Remains low	SUSI_STATUS_BUSY_COLLISION
Time-out due to clock stretching	SUSI_STATUS_TIMEOUT
Address Non-ACK	SUSI_STATUS_NOT_FOUND
Write Non-ACK	SUSI_STATUS_WRITE_ERROR
Success	SUSI_STATUS_SUCCESS

3.5.12 SusiSMBI2CWriteBlock

uint32_t SUSI_API SusiSMBI2CWriteBlock(uint32_t Id, uint8_t Addr, uint8_t Cmd, uint8_t *pBuffer, uint32_t Length)

Description:

Write multi-data using I^2C block protocol from the target slave device in the SMBus.

Parameters:

Id

Selects target device. See **Table 15**.

Addr

Specifies the 8-bit device address, ranging from 0x00 to 0xFF. Whether to give a 1 (read) or 0 (write) to the LSB of slave address could be ignored.

Cmd

Specifies the offset or command of the device register to write data to.

pBuffer

Pointer to a byte array in which the function writes the block data.

Length

Specifies the number of bytes to be write.

Return Status Code:

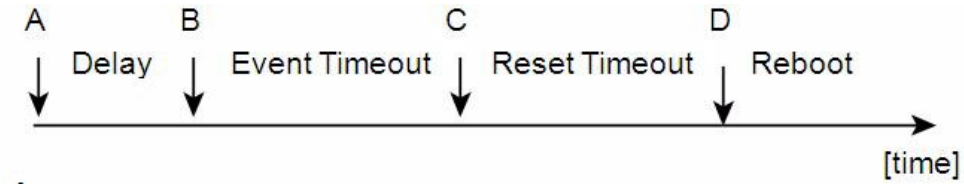
Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
pBuffer==NULL	SUSI _STATUS_INVALID_PARAMETER
Unknown Id	SUSI _STATUS_UNSUPPORTED
Device unsupported	SUSI _STATUS_UNSUPPORTED

Bus Busy SDA/SDC low	SUSI_STATUS_BUSY_COLLISION
Arbitration Error/Collision Error On Write 1 write cycle SDA Remains low	SUSI_STATUS_BUSY_COLLISION
Time-out due to clock stretching	SUSI_STATUS_TIMEOUT
Address Non-ACK	SUSI_STATUS_NOT_FOUND
Write Non-ACK	SUSI_STATUS_WRITE_ERROR
Success	SUSI_STATUS_SUCCESS

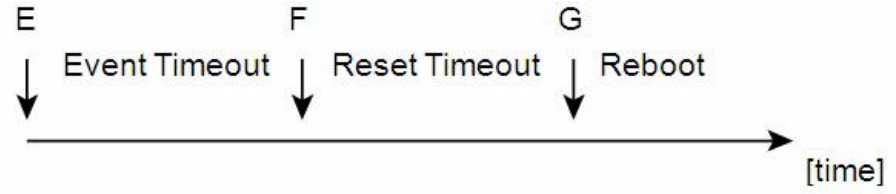
3.6 Watchdog Functions

After the watchdog timer has been start function it must be triggered within (Delay + Event Timeout) milliseconds as set with the start function, following the initial trigger every subsequent trigger must occur within (Event Timeout) milliseconds. Should trigger not be called within the relevant time limit a system reset will occur. The SUSI watchdog timer may support two stages. If the watchdog is not triggered within the event timeout, an NMI, IRQ, or hardware output will be generated. Then the reset timeout becomes active. If the watchdog timer is not triggered within the reset timeout a reset will be generated

Initial timing:



Timing after trigger:



Where:

Stage A

Watchdog is started.

Stage B

Initial Delay Period is exhausted.

Stage C/F

Event is triggered, NMI, IRQ, or PIN is Triggered. To Allow for possible Software Recovery.

Stage D/G

System is reset.

Stage E

- Watchdog is Triggered.
- Trigger / Stop must be called before Stage C/F to prevent event from being generated.
- Trigger / Stop must be called before Stage D/G to prevent The system from being reset.

Table 16 Watchdog ID

Id	Description
SUSI_ID_WATCHDOG_1	First watchdog timer
SUSI_ID_WATCHDOG_2	Second watchdog timer
SUSI_ID_WATCHDOG_3	Third watchdog timer

3.6.1 SusiWDogGetCaps

uint32_t SUSI_API SusiWDogGetCaps(uint32_t Id, uint32_t ItemId, uint32_t *pValue)

Description:

Gets watchdog capabilities.

Parameters:

Id

Selects target device. See **Table 16**.

ItemId

Selects target capability. See **Table 17**.

pValue

Pointer to a buffer that receives the target capability.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
pValue==NULL	SUSI _STATUS_INVALID_PARAMETER
Unknown Id or ItemId	SUSI _STATUS_UNSUPPORTED
Success	SUSI_STATUS_SUCCESS

Table 17 Watchdog capabilities item Id

Item Id	Description
SUSI_ID_WDT_SUPPORT_FLAGS	Event support flags (**Table 18**)
SUSI_ID_WDT_DELAY_MAXIMUM	The maximum delay time value
SUSI_ID_WDT_DELAY_MINIMUM	The minimum delay time value
SUSI_ID_WDT_EVENT_MAXIMUM	The maximum event time value
SUSI_ID_WDT_EVENT_MINIMUM	The minimum event time value
SUSI_ID_WDT_RESET_MAXIMUM	The maximum reset time value
SUSI_ID_WDT_RESET_MINIMUM	The minimum reset time value
SUSI_ID_WDT_UNIT_MINIMUM	The minimum unit value

SUSI_ID_WDT_DELAY_TIME	Current delay time setting
SUSI_ID_WDT_EVENT_TIME	Current event time setting
SUSI_ID_WDT_RESET_TIME	Current reset time setting
SUSI_ID_WDT_EVENT_TYPE	Current event type (**Table 19**)

3.6.2 SusiWDogStart

uint32_t SUSI_API SusiWDogStart(uint32_t Id, uint32_t DelayTime, uint32_t EventTime, uint32_t ResetTime, uint32_t EventType)

Description:

Start the watchdog timer and set the parameters. To adjust the parameters, the watchdog must be stopped and then start again with the new values. If the hardware implementation of the watchdog timer does not allow a setting at the exact time selected, the SUSI API selects the next possible longer timing.

Parameters:

Id

Selects target device. See **Table 16**.

DelayTime

Initial delay for the watchdog timer in milliseconds.

EventTime

Watchdog timeout interval in milliseconds to trigger an event.

ResetTime

Watchdog timeout interval in milliseconds to trigger a reset.

EventType

To select one kind of event type. See **Table 19**.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
Unknown Id	SUSI _STATUS_UNSUPPORTED
Wrong time range	SUSI _STATUS_INVALID_PARAMETER
Success	SUSI_STATUS_SUCCESS

Table 18 Watchdog Support Flags

Flag Name	Description	Value

SUSI_WDT_FLAG_SUPPORT_IRQ	Support IRQ event	0x01
SUSI_WDT_FLAG_SUPPORT_SCI	Support SCI event	0x02
SUSI_WDT_FLAG_SUPPORT_PWRBTN	Support power button event	0x04

Table 19 Watchdog timer event type

Event Type	Description
SUSI_WDT_EVENT_TYPE_NONE	No event
SUSI_WDT_EVENT_TYPE_SCI	SCI event
SUSI_WDT_EVENT_TYPE_IRQ	IRQ event
SUSI_WDT_EVENT_TYPE_PWRBTN	Power button event

3.6.3 SusiWDogStop

uint32_t SUSI_API SusiWDogStop(uint32_t Id)

Description:

Stops the operation of the watchdog timer.

Parameters:

Id

Selects target device. See **Table 16**.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
Unknown Id	SUSI _STATUS_UNSUPPORTED
Success	SUSI_STATUS_SUCCESS

3.6.4 SusiWDogTrigger

uint32_t SUSI_API SusiWDogTrigger(uint32_t Id)

Description:

Trigger the watchdog timer.

Parameters:

Id

Selects target device. See **Table 16**.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
Unknown Id	SUSI _STATUS_UNSUPPORTED
Success	SUSI_STATUS_SUCCESS

3.6.5 SusiWDogSetCallBack

uint32_t SUSI_API SusiWDogSetCallBack(uint32_t Id,
SUSI_WDT_INT_CALLBACK pfnCallback, void *Context)

Description:

The call back function pointer can be transmit from Application when IRQ triggered.

Parameters:

Id

Selects target device. See **Table 16**.

pfnCallback

Call back function pointer, SUSI_WDT_INT_CALLBACK is function pointer type, it can set NULL to clear. The type definition just like show below,

*typedef void (*SUSI_WDT_INT_CALLBACK)(void*);*

Context

Pointer to a user context structure for callback function.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
Unknown Id	SUSI _STATUS_UNSUPPORTED
Success	SUSI_STATUS_SUCCESS

3.7 GPIO Functions

Programmable GPIO allows developers to dynamically set the GPIO input or output status

Table 20 GPIO ID

Id	Description
SUSI_ID_GPIO(X)	X is GPIO pin number, definition as below: #define SUSI_ID_GPIO(x) (0x0000 | x) This ID control single pin only.
SUSI_ID_GPIO_BANK(Y)	Y is GPIO bank number, definition as below: #define SUSI_ID_GPIO_BANK(Y) (0x10000 | Y) This ID control maximum 32 pins per bank.

3.7.1 SusiGPIOGetCaps

uint32_t SUSI_API SusiGPIOGetCaps(uint32_t Id, uint32_t ItemId, uint32_t *pValue)

Description:

Reads the capabilities of the current GPIO implementation from the selected GPIO interface.

Parameters:

Id

Selects target device. See **Table 20**.

ItemId

Selects target capability. See **Table 21**.

pValue

Pointer to a buffer that receives the target capability. Each bit of the buffer value represents support situation of a GPIO, according to the order. 1 is supportive, and 0 is unsupportive.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
pValue==NULL	SUSI _STATUS_INVALID_PARAMETER
Unknown Id	SUSI _STATUS_UNSUPPORTED
Success	SUSI_STATUS_SUCCESS

Table 21 GPIO capabilities item Id

Item Id	Description

SUSI_ID_GPIO_INPUT_SUPPORT	Get GPIO input support state
SUSI_ID_GPIO_OUTPUT_SUPPORT	Get GPIO output support state

3.7.2 SusiGPIOGetDirection

uint32_t SUSI_API SusiGPIOGetDirection(uint32_t Id, uint32_t Bitmask, uint32_t *pDirection)

Description:

Gets the configuration for the selected GPIO ports.

Parameters:

Id

Selects target device. See **Table 20**.

Bitmask

Value for a bit mask. Only selected bits are changed, unselected bits remain unchanged.

This parameter will be ignored when single pin mode.

pDirection

Pointer to a buffer that receives the direction of the selected GPIO ports.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
pDirection==NULL	SUSI _STATUS_INVALID_PARAMETER
Bitmask==0 when bank mode	SUSI _STATUS_INVALID_PARAMETER
Unknown Id	SUSI _STATUS_UNSUPPORTED
Success	SUSI_STATUS_SUCCESS

3.7.3 SusiGPIOSetDirection

uint32_t SUSI_API SusiGPIOSetDirection(uint32_t Id, uint32_t Bitmask, uint32_t Direction)

Description:

Sets the configuration for the selected GPIO ports.

Parameters:

Id

Selects target device. See **Table 20**.

Bitmask

Value for a bit mask. Only selected bits are changed, unselected bits remain unchanged.

This parameter will be ignored when single pin mode.

Direction

Sets the direction of the selected GPIO ports.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
Bitmask==0 when bank mode	SUSI _STATUS_INVALID_PARAMETER
Unknown Id	SUSI _STATUS_UNSUPPORTED
Success	SUSI_STATUS_SUCCESS

3.7.4 SusiGPIOGetLevel

uint32_t SUSI_API SusiGPIOGetLevel(uint32_t Id, uint32_t Bitmask, uint32_t *pLevel)

Description:

Read level the from GPIO ports.

Parameters:

Id

Selects target device. See **Table 20**.

Bitmask

Value for a bit mask. Only selected bits are changed, unselected bits remain unchanged.

This parameter will be ignored when single pin mode.

pLevel

Pointer to a buffer that receives the GPIO level.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
pLevel==NULL	SUSI _STATUS_INVALID_PARAMETER
Bitmask==0 when bank mode	SUSI _STATUS_INVALID_PARAMETER
Unknown Id	SUSI _STATUS_UNSUPPORTED
Success	SUSI_STATUS_SUCCESS

3.7.5 SusiGPIOSetLevel

uint32_t SUSI_API SusiGPIOSetLevel(uint32_t Id, uint32_t Bitmask, uint32_t Level)

Description:

Write level to GPIO ports. Depending on the hardware implementation writing multiple GPIO ports with the bit mask option does not guarantee a time synchronous change of the output levels.

Parameters:

Id

Selects target device. See **Table 20**.

Bitmask

Value for a bit mask. Only selected bits are changed, unselected bits remain unchanged. This parameter will be ignored when single pin mode.

Level

Input level of the selected GPIO port.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
Bitmask==0 when bank mode	SUSI _STATUS_INVALID_PARAMETER
Unknown Id	SUSI _STATUS_UNSUPPORTED
Success	SUSI_STATUS_SUCCESS

3.8 Smart Fan Functions

The Smart Fan function call is used to set fan speed configuration. You can use this function to easily control the fan speed. It takes a pointer to an instance of structure *SusiFanControl*, which is defined as follows:

```
#define SUSI_FAN_AUTO_CTRL_OPMODE_PWM   0
#define SUSI_FAN_AUTO_CTRL_OPMODE_RPM   1
typedef struct _AutoFan {
        uint32_t TmlSource;     // Thermal source
        uint32_t OpMode;
        uint32_t LowStopLimit;  // Temperature (0.1 Kelvins)
        uint32_t LowLimit;      // Temperature (0.1 Kelvins)
        uint32_t HighLimit;     // Temperature (0.1 Kelvins)
        uint32_t MinPWM;        // Enable when OpMode == FAN_AUTO_CTRL_OPMODE_PWM
        uint32_t MaxPWM;        // Enable when OpMode == FAN_AUTO_CTRL_OPMODE_PWM
        uint32_t MinRPM;        // Enable when OpMode == FAN_AUTO_CTRL_OPMODE_RPM
        uint32_t MaxRPM;        // Enable when OpMode == FAN_AUTO_CTRL_OPMODE_RPM
} AutoFan , *PAutoFan ;

// Mode
#define SUSI_FAN_CTRL_MODE_OFF          0
#define SUSI_FAN_CTRL_MODE_FULL         1
#define SUSI_FAN_CTRL_MODE_MANUAL       2
#define SUSI_FAN_CTRL_MODE_AUTO         3
typedef struct _SusiFanControl {
        uint32_t Mode;
        uint32_t PWM;           // Manual mode only (0 - 100%)
        AutoFan AutoControl;    // Auto mode only
} SusiFanControl, *PSusiFanControl;
```

If Mode member of SusiFanControl is not Auto, AutoControl member will be ignored. In auto mode, parameter “TmlSource” is use *SUSI_ID_HWM_TEMP_XXX* (**Table 3**) to select which thermal type to reference. If TmlSource is not match any temperature ID means unknown or unsupported.

3.8.1 SusiFanControlGetCaps

uint32_t SUSI_API SusiFanControlGetCaps(uint32_t Id, uint32_t ItemId, uint32_t *pValue)

Description:

Gets fan control capabilities.

Parameters:

Id

Smart fan ID is same as Fan Speed Value ID. See **Table 4**.

ItemId

Selects target capability. See **Table 22**. This parameter can also input temperature ID (**Table 3**) to get is it supports in *SusiFanControl* function.

pValue

Pointer to a buffer that receives the target capability.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
pValue==NULL	SUSI _STATUS_INVALID_PARAMETER
Unknown Id or ItemId	SUSI _STATUS_UNSUPPORTED
Success	SUSI_STATUS_SUCCESS

Table 22 Fan control capabilities item Id

Item Id	Description
SUSI_ID_FC_CONTROL_SUPPORT_FLAGS	Control support flags (See **Table 23**)
SUSI_ID_FC_AUTO_SUPPORT_FLAGS	Auto support flags (See **Table 24**)

Table 23 Control Support Flags

Flag Name	Description	Value
SUSI_FC_FLAG_SUPPORT_OFF_MODE	Support off mode	0x01
SUSI_FC_FLAG_SUPPORT_FULL_MODE	Support full mode	0x02
SUSI_FC_FLAG_SUPPORT_MANUAL_MODE	Support manual mode	0x04
SUSI_FC_FLAG_SUPPORT_AUTO_MODE	Support auto mode More detail to get *Auto Support Flags*	0x08

Table 24 Auto Support Flags

Flag Name	Description	Value

SUSI_FC_FLAG_SUPPORT_AUTO_LOW_STOP	Auto mode support Low Stop	0x01
SUSI_FC_FLAG_SUPPORT_AUTO_LOW_LIMIT	Auto mode support Low Limit	0x02
SUSI_FC_FLAG_SUPPORT_AUTO_HIGH_LIMIT	Auto mode support High Limit	0x04
SUSI_FC_FLAG_SUPPORT_AUTO_PWM	Auto mode support PWM operation	0x0100
SUSI_FC_FLAG_SUPPORT_AUTO_RPM	Auto mode support RPM operation	0x0200

3.8.2 SusiFanControlGetConfig

uint32_t SUSI_API SusiFanControlGetConfig(uint32_t Id, SusiFanControl *pConfig)

Description:

Get information about smart fan function mode and configuration.

Parameters:

Id

Smart fan ID is same as Fan Speed Value ID. See **Table 4**.

pConfig

Pointer to the smart fan function configuration.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
pConfig==NULL	SUSI _STATUS_INVALID_PARAMETER
Unknown Id	SUSI _STATUS_UNSUPPORTED
Device not support smart mode	SUSI _STATUS_UNSUPPORTED
Success	SUSI_STATUS_SUCCESS

3.8.3 SusiFanControlSetConfig

uint32_t SUSI_API SusiFanControlSetConfig(uint32_t Id, SusiFanControl *pConfig)

Description:

Set smart fan function mode and configuration.

Parameters:

Id

Smart fan ID is same as Fan Speed Value ID. See **Table 4**.

pConfig

Pointer to the smart fan function configuration.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
pConfig==NULL	SUSI _STATUS_INVALID_PARAMETER
Wrong configuration	SUSI _STATUS_INVALID_PARAMETER
Unknown Id	SUSI _STATUS_UNSUPPORTED
Success	SUSI_STATUS_SUCCESS

3.9 Storage Functions

Access storage information and read / write data to the selected user data area. Developers can use this area to store their own data.

Table 25 Storage ID

Id	Description
SUSI_ID_STORAGE_STD	Standard storage device
SUSI_ID_STORAGE_OEM0~1	Other storage devices

3.9.1 SusiStorageGetCaps

uint32_t SUSI_API SusiStorageGetCaps(uint32_t Id, uint32_t ItemId, uint32_t *pValue)

Description:

Reads the capabilities of the current storage implementation from the selected storage interface.

Parameters:

Id

Selects target device. See **Table 25**.

ItemId

Selects target capability. See **Table 26**.

pValue

Pointer to a buffer that receives the target capability.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
pValue==NULL	SUSI _STATUS_INVALID_PARAMETER
Unknown Id	SUSI _STATUS_UNSUPPORTED
Device not support	SUSI _STATUS_UNSUPPORTED
Success	SUSI_STATUS_SUCCESS

Table 26 Storage capabilities item Id

Item Id	Description
SUSI_ID_STORAGE_TOTAL_SIZE	Get storage total size in bytes
SUSI_ID_STORAGE_BLOCK_SIZE	Get storage block size in bytes
SUSI_ID_STORAGE_LOCK_STATUS	Get storage lock status. See **Table 27**.

SUSI_ID_STORAGE_PSW_MAX_LEN	Get maximum length in byte of storage lock key

Table 27 Storage Lock Status

Lock Status	Description
SUSI_STORAGE_STATUS_LOCK	Storage is lock
SUSI_STORAGE_STATUS_UNLOCK	Storage is unlock

3.9.2 SusiStorageAreaRead

uint32_t SUSI_API SusiStorageAreaRead(uint32_t Id, uint32_t Offset, uint8_t *pBuffer, uint32_t BufLen)

Description:

Reads data from the selected user data area.

Parameters:

Id

Selects target device. See **Table 25**.

Offset

Storage area start address offset in bytes.

pBuffer

Size in bytes of the buffer pointed to by the pBuffer parameter.

BufLen

Size in bytes of the information read to the buffer pointed to by the pBuffer parameter.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
pBuffer==NULL || BufLen==0	SUSI _STATUS_INVALID_PARAMETER
Offset+BufLen>TotalSize	SUSI_STATUS_INVALID_BLOCK_LENGTH
Unknown Id	SUSI _STATUS_UNSUPPORTED
Read error	SUSI_STATUS_READ_ERROR
Success	SUSI_STATUS_SUCCESS

3.9.3 SusiStorageAreaWrite

uint32_t SUSI_API SusiStorageAreaWrite(uint32_t Id, uint32_t Offset, uint8_t *pBuffer, uint32_t BufLen)

Description:

Writes data to the selected user data area.

Parameters:

Id

Selects target device. See **Table 25**.

Offset

Storage area start address offset in bytes.

pBuffer

Size in bytes of the buffer pointed to by the pBuffer parameter.

BufLen

Size in bytes of the information read to the buffer pointed to by the pBuffer parameter.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
pBuffer==NULL || BufLen==0	SUSI _STATUS_INVALID_PARAMETER
Offset+BufLen>TotalSize	SUSI_STATUS_INVALID_BLOCK_LENGTH
Unknown Id	SUSI _STATUS_UNSUPPORTED
Write error	SUSI_STATUS_WRITE_ERROR
Success	SUSI_STATUS_SUCCESS

3.9.4 SusiStorageAreaSetLock

uint32_t SUSI_API SusiStorageAreaSetLock(uint32_t Id, uint8_t *pBuffer, uint32_t BufLen)

Description:

Lock a storage area for write protect.

Parameters:

Id

Selects target device. See **Table 25**.

pBuffer

Lock of key buffer.

BufLen

Number of key buffer

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
pBuffer==NULL || BufLen==0	SUSI _STATUS_INVALID_PARAMETER
Lock error	SUSI_STATUS_WRITE_ERROR
Unknown Id	SUSI _STATUS_UNSUPPORTED
Device not support	SUSI _STATUS_UNSUPPORTED
Success	SUSI_STATUS_SUCCESS

3.9.5 SusiStorageAreaSetUnlock

uint32_t SUSI_API SusiStorageAreaSetUnlock(uint32_t Id, uint8_t *pBuffer, uint32_t BufLen)

Description:

Unlock a storage area for write protect.

Parameters:

Id

Selects target device. See **Table 25**.

pBuffer

Unlock of key buffer.

BufLen

Number of key buffer

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
pBuffer==NULL || BufLen==0	SUSI _STATUS_INVALID_PARAMETER
Unlock error	SUSI_STATUS_WRITE_ERROR
Unknown Id	SUSI _STATUS_UNSUPPORTED
Device not support	SUSI _STATUS_UNSUPPORTED
Success	SUSI_STATUS_SUCCESS

3.10 Thermal Protection Functions

The Thermal Protection function call is used to set hardware base thermal monitoring and notify. It takes a pointer to an instance of structure *SusiThermalProtect*, which is defined as follows:

```
typedef struct _SusiThermalProtect{
        uint32_t SourceId;
        uint32_t EventType;
        uint32_t SendEventTemperature;
        uint32_t ClearEventTemperature;
} SusiThermalProtect, *PSusiThermalProtect;
```

Where:

SourceId

Setting thermal source ID here. See **Table 3**.

EventType

This byte can set up a thermal protect event, see **Table 28**. NOT every platform supports all event type.

SendEevntTemperature

Unit is 0.1 Kelvins. When thermal source goes over this value, SUSI will send event according Event Type.

ClearEventTemperature

Unit is 0.1 Kelvins. When thermal source goes below this value and Event is sent, SUSI will clear event according Event Type

Table 28 Thermal Protection Event Type

Event Type Name	Description	Value
SUSI_THERMAL_EVENT_SHUTDOWN	Shutdown event	0x00
SUSI_THERMAL_EVENT_THROTTLE	Throttle event	0x01
SUSI_THERMAL_EVENT_POWEROFF	Power off event	0x02
SUSI_THERMAL_EVENT_NONE	No event	0xFF

Table 29 Thermal Protection ID

Id	Description
SUSI_ID_THERMAL_PROTECT_1	Thermal protection zone 1
SUSI_ID_THERMAL_PROTECT_2	Thermal protection zone 2
SUSI_ID_THERMAL_PROTECT_3	Thermal protection zone 3
SUSI_ID_THERMAL_PROTECT_4	Thermal protection zone 4

3.10.1 SusiThermalProtectionGetCaps

uint32_t SUSI_API SusiThermalProtectionGetCaps(uint32_t Id, uint32_t ItemId, uint32_t *pValue)

Description:

Gets Thermal Protection capabilities.

Parameters:

Id

Selects target device. See **Table 29**.

ItemId

Selects target capability. See **Table 30**. This parameter can also input temperature ID (**Table 3**) to get is it supports in *SusiThermalProtection* function.

pValue

Pointer to a buffer that receives the target capability.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
pValue==NULL	SUSI _STATUS_INVALID_PARAMETER
Unknown Id or ItemId	SUSI _STATUS_UNSUPPORTED
Success	SUSI_STATUS_SUCCESS

Table 30 Thermal Protection capabilities item Id

Item Id	Description
SUSI_ID_TP_EVENT_SUPPORT_FLAGS	Event support flags (See **Table 31**)
SUSI_ID_TP_EVENT_TRIGGER_MAXIMUM	The maximum temperature to trigger event
SUSI_ID_TP_EVENT_TRIGGER_MINIMUM	The minimum temperature to trigger event
SUSI_ID_TP_EVENT_CLEAR_MAXIMUM	The maximum temperature to clear event
SUSI_ID_TP_EVENT_CLEAR_MINIMUM	The minimum temperature to clear event

Table 31 Thermal Protection Support Flags

Flag Name	Description	Value
SUSI_THERMAL_FLAG_SUPPORT_SHUTDOWN	Support shutdown event	0x01
SUSI_THERMAL_FLAG_SUPPORT_THROTTLE	Support throttle event	0x02
SUSI_THERMAL_FLAG_SUPPORT_POWEROFF	Support power off event	0x04

3.10.2 SusiThermalProtectionSetConfig

uint32_t SUSI_API SusiThermalProtectionSetConfig(uint32_t Id, SusiThermalProtect *pConfig)

Description:

Set Thermal Protection configuration.

Parameters:

Id

Selects target device. See **Table 29**.

pConfig

A data package for thermal protection.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
pConfig==NULL	SUSI _STATUS_INVALID_PARAMETER
Config invalid	SUSI _STATUS_INVALID_PARAMETER
Source Id or event type not support	SUSI _STATUS_UNSUPPORTED
Unknown Id	SUSI _STATUS_UNSUPPORTED
Success	SUSI_STATUS_SUCCESS

3.10.3 SusiThermalProtectionGetConfig

uint32_t SUSI_API SusiThermalProtectionGetConfig(uint32_t Id, SusiThermalProtect *pConfig)

Description:

Get Thermal Protection configuration.

Parameters:

Id

Selects target device. See **Table 29**.

pConfig

A data package for thermal protection.

Return Status Code:

Condition	Return Value
Library uninitialized	SUSI_STATUS_NOT_INITIALIZED
pConfig==NULL	SUSI _STATUS_INVALID_PARAMETER
Unknown Id	SUSI _STATUS_UNSUPPORTED
Success	SUSI_STATUS_SUCCESS