

# Reference Manual

Generated by Doxygen 1.6.1

Mon May 10 10:43:25 2010



# Contents

<b>1</b>	<b>Class Index</b>	<b>1</b>
1.1	Class List . . . . .	1
<b>2</b>	<b>File Index</b>	<b>3</b>
2.1	File List . . . . .	3
<b>3</b>	<b>Class Documentation</b>	<b>5</b>
3.1	SCIP2_SCANNED_DATA Struct Reference . . . . .	5
3.1.1	Detailed Description . . . . .	5
3.2	SCIP2_SCANNED_DATA_TRI Struct Reference . . . . .	6
3.2.1	Detailed Description . . . . .	6
3.3	SCIP2_SENSOR_PARAM Struct Reference . . . . .	7
3.3.1	Detailed Description . . . . .	7
3.4	SCIP2_SENSOR_VERSION Struct Reference . . . . .	8
3.4.1	Detailed Description . . . . .	8
<b>4</b>	<b>File Documentation</b>	<b>9</b>
4.1	libscip2awd.h File Reference . . . . .	9
4.1.1	Detailed Description . . . . .	9
4.2	libscip2awd_cmd.c File Reference . . . . .	11
4.2.1	Detailed Description . . . . .	12
4.2.2	Function Documentation . . . . .	12
4.2.2.1	Scip2CMD_BM . . . . .	12
4.2.2.2	Scip2CMD_CR . . . . .	12
4.2.2.3	Scip2CMD_GS . . . . .	13
4.2.2.4	Scip2CMD_PP . . . . .	13
4.2.2.5	Scip2CMD_QT . . . . .	13
4.2.2.6	Scip2CMD_RS . . . . .	14
4.2.2.7	Scip2CMD SCIP2 . . . . .	14

4.2.2.8	Scip2CMD_SS	14
4.2.2.9	Scip2CMD_StartMS	14
4.2.2.10	Scip2CMD_StopGS	15
4.2.2.11	Scip2CMD_StopMS	15
4.2.2.12	Scip2CMD_TM	15
4.2.2.13	Scip2CMD_VV	16
4.3	libscip2awd_cmd.h File Reference	17
4.3.1	Detailed Description	18
4.3.2	Typedef Documentation	19
4.3.2.1	S2Param_t	19
4.3.2.2	S2Ver_t	19
4.3.3	Function Documentation	19
4.3.3.1	Scip2CMD_BM	19
4.3.3.2	Scip2CMD_CR	19
4.3.3.3	Scip2CMD_GS	19
4.3.3.4	Scip2CMD_PP	20
4.3.3.5	Scip2CMD_QT	20
4.3.3.6	Scip2CMD_RS	20
4.3.3.7	Scip2CMD SCIP2	20
4.3.3.8	Scip2CMD_SS	21
4.3.3.9	Scip2CMD_StartMS	21
4.3.3.10	Scip2CMD_StopGS	21
4.3.3.11	Scip2CMD_StopMS	22
4.3.3.12	Scip2CMD_TM	22
4.3.3.13	Scip2CMD_VV	22
4.4	libscip2awd_dbuffer.c File Reference	23
4.4.1	Detailed Description	24
4.4.2	Function Documentation	24
4.4.2.1	S2Sdd_Begin_Lock	24
4.4.2.2	S2Sdd_Begin_Trylock	24
4.4.2.3	S2Sdd_BeginEx	24
4.4.2.4	S2Sdd_Dest	25
4.4.2.5	S2Sdd_End	25
4.4.2.6	S2Sdd_Init	25
4.4.2.7	S2Sdd_RecvData	25
4.4.2.8	S2Sdd_RecvDataCont	25

---

4.4.2.9	S2Sdd_StopThread	25
4.5	libscip2awd_dbuffer.h File Reference	26
4.5.1	Detailed Description	27
4.5.2	Typedef Documentation	27
4.5.2.1	S2Scan_t	27
4.5.2.2	S2Sdd_t	27
4.5.3	Function Documentation	27
4.5.3.1	S2Sdd_Begin_Lock	27
4.5.3.2	S2Sdd_Begin_Trylock	28
4.5.3.3	S2Sdd_BeginEx	28
4.5.3.4	S2Sdd_Dest	28
4.5.3.5	S2Sdd_End	28
4.5.3.6	S2Sdd_Init	28
4.5.3.7	S2Sdd_RecvData	29
4.5.3.8	S2Sdd_RecvDataCont	29
4.5.3.9	S2Sdd_StopThread	29
4.6	libscip2awd_test.c File Reference	30
4.6.1	Detailed Description	30
4.6.2	Function Documentation	30
4.6.2.1	ctrlc	30
4.6.2.2	main	31
4.6.3	Variable Documentation	31
4.6.3.1	gShutoff	31



# Chapter 1

## Class Index

### 1.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

<a href="#">SCIP2_SCANNED_DATA</a> . . . . .	5
<a href="#">SCIP2_SCANNED_DATA_TRI</a> . . . . .	6
<a href="#">SCIP2_SENSOR_PARAM</a> . . . . .	7
<a href="#">SCIP2_SENSOR_VERSION</a> . . . . .	8





# Chapter 2

## File Index

### 2.1 File List

Here is a list of all documented files with brief descriptions:

<a href="#">libscip2awd.h</a> (Library for Sokuiki-Sensor "URG" ) . . . . .	9
<b>libscip2awd_base.h</b> . . . . .	??
<a href="#">libscip2awd_cmd.c</a> (Library for Sokuiki-Sensor "URG" ) . . . . .	11
<a href="#">libscip2awd_cmd.h</a> (Library for Sokuiki-Sensor "URG" ) . . . . .	17
<a href="#">libscip2awd_dbuffer.c</a> (Library for Sokuiki-Sensor "URG" ) . . . . .	23
<a href="#">libscip2awd_dbuffer.h</a> (Library for Sokuiki-Sensor "URG" ) . . . . .	26
<a href="#">libscip2awd_test.c</a> (Test program for libscip2 ) . . . . .	30



# Chapter 3

## Class Documentation

### 3.1 SCIP2\_SCANNED\_DATA Struct Reference

```
#include <libscip2awd_dbuffer.h>
```

#### Public Attributes

- int **start**
- int **end**
- int **group**
- int **cull**
- int **size**
- int **num**
- int **memsize**
- unsigned long **time**
- int **error**
- pthread\_mutex\_t **mutex**
- S2Port \* **port**
- unsigned long \* **data**
- S2EncType **enc**

#### 3.1.1 Detailed Description

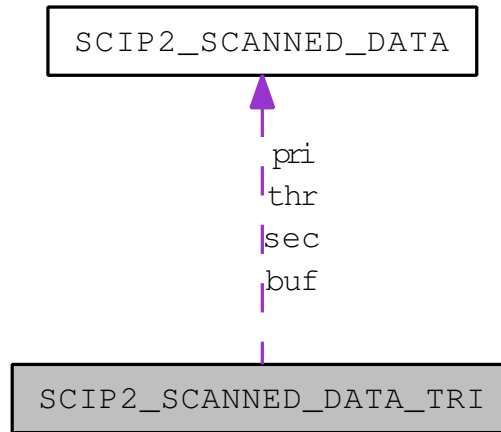
Buffer structure for scanned data

The documentation for this struct was generated from the following file:

- [libscip2awd\\_dbuffer.h](#)

## 3.2 SCIP2\_SCANNED\_DATA\_TRI Struct Reference

#include <libscip2awd\_dbuffer.h> Collaboration diagram for SCIP2\_SCANNED\_DATA\_TRI:



### Public Attributes

- [S2Scan\\_t](#) \* **pri**
- [S2Scan\\_t](#) \* **sec**
- [S2Scan\\_t](#) \* **thr**
- [S2Scan\\_t](#) **buf** [3]
- int **update**
- [pthread\\_mutex\\_t](#) **mutexr**
- [pthread\\_mutex\\_t](#) **mutexw**
- [pthread\\_t](#) **thread**
- int **nbuf**

### 3.2.1 Detailed Description

Multi buffer structure for scanned data

The documentation for this struct was generated from the following file:

- [libscip2awd\\_dbuffer.h](#)

## 3.3 SCIP2\_SENSOR\_PARAM Struct Reference

```
#include <libscip2awd_cmd.h>
```

### Public Attributes

- char **model** [SCIP2\_MAX\_LENGTH]
- int **dist\_min**
- int **dist\_max**
- int **step\_resolution**
- int **step\_min**
- int **step\_max**
- int **step\_front**
- int **revolution**

### 3.3.1 Detailed Description

Structure of PP command

The documentation for this struct was generated from the following file:

- [libscip2awd\\_cmd.h](#)

## 3.4 SCIP2\_SENSOR\_VERSION Struct Reference

```
#include <libscip2awd_cmd.h>
```

### Public Attributes

- char **vender** [SCIP2\_MAX\_LENGTH]
- char **product** [SCIP2\_MAX\_LENGTH]
- char **firmware** [SCIP2\_MAX\_LENGTH]
- char **protocol** [SCIP2\_MAX\_LENGTH]
- char **serialno** [SCIP2\_MAX\_LENGTH]

### 3.4.1 Detailed Description

Structure of VV command

The documentation for this struct was generated from the following file:

- [libscip2awd\\_cmd.h](#)

# Chapter 4

## File Documentation

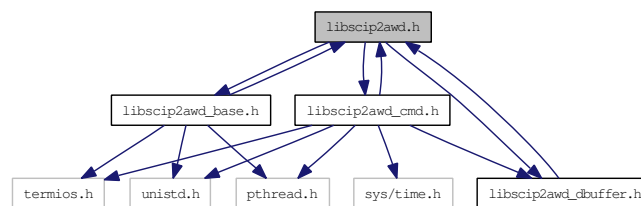
### 4.1 libscip2awd.h File Reference

Library for Sokuiki-Sensor "URG". #include "libscip2awd\_base.h"

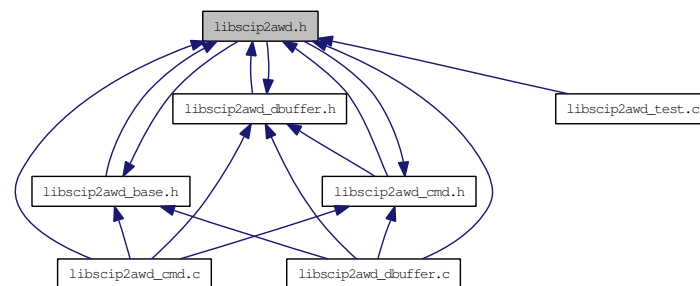
#include "libscip2awd\_cmd.h"

#include "libscip2awd\_dbuffer.h"

Include dependency graph for libscip2awd.h:



This graph shows which files directly or indirectly include this file:



#### 4.1.1 Detailed Description

Library for Sokuiki-Sensor "URG".

**Author:**

Atsushi WATANEBE <atusi\_w@roboken.esys.tsukuba.ac.jp> <atusi\_w@doramanjyu.com>



## 4.2 libscip2awd\_cmd.c File Reference

Library for Sokuiki-Sensor "URG". #include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <unistd.h>

#include <sys/time.h>

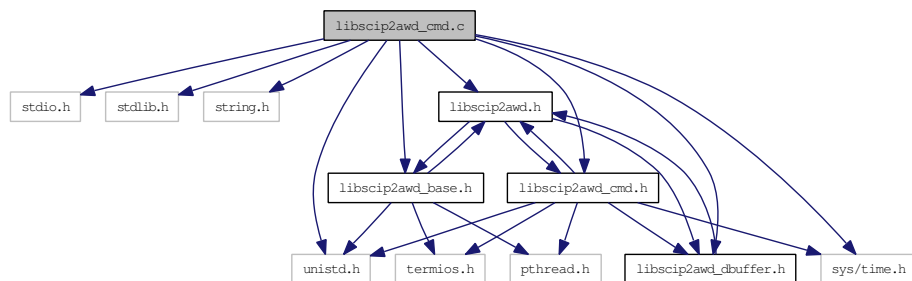
#include "libscip2awd.h"

#include "libscip2awd\_base.h"

#include "libscip2awd\_cmd.h"

#include "libscip2awd\_dbuffer.h"

Include dependency graph for libscip2awd\_cmd.c:



## Functions

- int [Scip2CMD SCIP2](#) (S2Port \*apPort)  
*Switch to SCIP2.0 mode.*
- int [Scip2CMD SS](#) (S2Port \*apPort, const speed\_t acBitrate)  
*Change Device's Bitrate.*
- int [Scip2CMD BM](#) (S2Port \*apPort)  
*Laser ON.*
- int [Scip2CMD QT](#) (S2Port \*apPort)  
*Laser OFF.*
- int [Scip2CMD GS](#) (S2Port \*apPort, int aStart, int aEnd, int aGroup, S2Sdd\_t \*aData, const S2EncType acEnc)  
*Start getting scanned data.*
- int [Scip2CMD StopGS](#) (S2Port \*apPort, S2Sdd\_t \*aData)  
*Stop GS scanning.*
- int [Scip2CMD StartMS](#) (S2Port \*apPort, int aStart, int aEnd, int aGroup, int aCull, int aNum, S2Sdd\_t \*aData, const S2EncType acEnc)

*Start getting scanned data.*

- int `Scip2CMD_StopMS` (S2Port \*apPort, S2Sdd\_t \*aData)  
*Stop MS scanning.*
- int `Scip2CMD_TM` (S2Port \*apPort, struct timeval \*apTime)  
*Get time of device.*
- int `Scip2CMD_CR` (S2Port \*apPort, const int acDeboost)  
*Set rotating speed.*
- int `Scip2CMD_VV` (S2Port \*apPort, S2Ver\_t \*apVer)  
*Get version info.*
- int `Scip2CMD_PP` (S2Port \*apPort, S2Param\_t \*apParam)  
*Get param info.*
- int `Scip2CMD_RS` (S2Port \*apPort)  
*Reset Device.*

## 4.2.1 Detailed Description

Library for Sokuiki-Sensor "URG".

### Author:

Atsushi WATANEBE <atusi\_w@roboken.esys.tsukuba.ac.jp> <atusi\_w@doramanjyu.com>

## 4.2.2 Function Documentation

### 4.2.2.1 int Scip2CMD\_BM (S2Port \* apPort)

Laser ON.

#### Parameters:

*\*apPort* Pointer to SCIP2.0 Device Port

#### Returns:

failed: false, succeeded: true

### 4.2.2.2 int Scip2CMD\_CR (S2Port \* apPort, const int acDeboost)

Set rotating speed.

#### Parameters:

*\*apPort* Pointer to SCIP2.0 Device Port

*acDeboost* Deboost rate

**Returns:**

failed: 0, succeeded: time of device

**4.2.2.3 int Scip2CMD\_GS (S2Port \* *apPort*, int *aStart*, int *aEnd*, int *aGroup*, S2Sdd\_t \* *aData*, const S2EncType *acEnc*)**

Start getting scanned data.

**Parameters:**

\**apPort* Pointer to SCIP2.0 Device Port

*aStart* Start step

*aEnd* End step

*aGroup* Number of group

\**aData* Pointer to buffer structure

*acEnc* Encode type

**Returns:**

failed: false, succeeded: true

**Attention:**

Scip2CMD\_StopGS must be called before calling another Scip2CMD function, if the device remains sending data to PC!!

**4.2.2.4 int Scip2CMD\_PP (S2Port \* *apPort*, S2Param\_t \* *apParam*)**

Get param info.

**Parameters:**

\**apPort* Pointer to SCIP2.0 Device Port

\**apParam* Pointer to param structure

**Returns:**

failed: 0, succeeded: time of device

**4.2.2.5 int Scip2CMD\_QT (S2Port \* *apPort*)**

Laser OFF.

**Parameters:**

\**apPort* Pointer to SCIP2.0 Device Port

**Returns:**

failed: false, succeeded: true

#### 4.2.2.6 int Scip2CMD\_RS (S2Port \* *apPort*)

Reset Device.

##### Parameters:

*\*apPort* Pointer to SCIP2.0 Device Port

##### Returns:

failed: false, succeeded: true

#### 4.2.2.7 int Scip2CMD SCIP2 (S2Port \* *apPort*)

Switch to SCIP2.0 mode.

##### Parameters:

*\*apPort* Pointer to SCIP2.0 Device Port

##### Returns:

false: failed, true: succeeded

#### 4.2.2.8 int Scip2CMD\_SS (S2Port \* *apPort*, const speed\_t *acBitrate*)

Change Device's Bitrate.

##### Parameters:

*\*apPort* Pointer to SCIP2.0 Device Port

*acBitrate* Bitrate

##### Returns:

failed: false, succeeded: true

#### 4.2.2.9 int Scip2CMD\_StartMS (S2Port \* *apPort*, int *aStart*, int *aEnd*, int *aGroup*, int *aCull*, int *aNum*, S2Sdd\_t \* *aData*, const S2EncType *acEnc*)

Start getting scanned data.

##### Parameters:

*\*apPort* Pointer to SCIP2.0 Device Port

*aStart* Start step

*aEnd* End step

*aGroup* Number of group

*aCull* Culling clearance

*aNum* Number of scan

*\*aData* Pointer to buffer structure

*acEnc* Encode type

**Returns:**

failed: false, succeeded: true

**Attention:**

Scip2CMD\_StopMS must be called before calling another Scip2CMD function, if the device remains sending data to PC!!

**4.2.2.10 int Scip2CMD\_StopGS (S2Port \* *apPort*, S2Sdd\_t \* *aData*)**

Stop GS scanning.

**Parameters:**

\**apPort* Pointer to SCIP2.0 Device Port

\**aData* Pointer to buffer structure

**Returns:**

failed: false, succeeded: true

**4.2.2.11 int Scip2CMD\_StopMS (S2Port \* *apPort*, S2Sdd\_t \* *aData*)**

Stop MS scanning.

**Parameters:**

\**apPort* Pointer to SCIP2.0 Device Port

\**aData* Pointer to buffer structure

**Returns:**

failed: false, succeeded: true

**4.2.2.12 int Scip2CMD\_TM (S2Port \* *apPort*, struct timeval \* *apTime*)**

Get time of device.

**Parameters:**

\**apPort* Pointer to SCIP2.0 Device Port

\**apTime* Start time of device

**Returns:**

failed: 0, succeeded: 1

**4.2.2.13 int Scip2CMD\_VV (S2Port \* *apPort*, S2Ver\_t \* *apVer*)**

Get version info.

**Parameters:**

\**apPort* Pointer to SCIP2.0 Device Port

\**apVer* Pointer to version structure

**Returns:**

failed: 0, succeeded: time of device

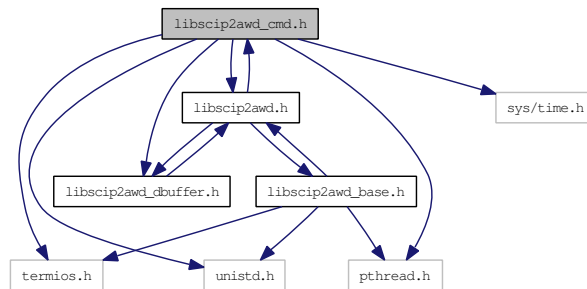
## 4.3 libscip2awd\_cmd.h File Reference

```

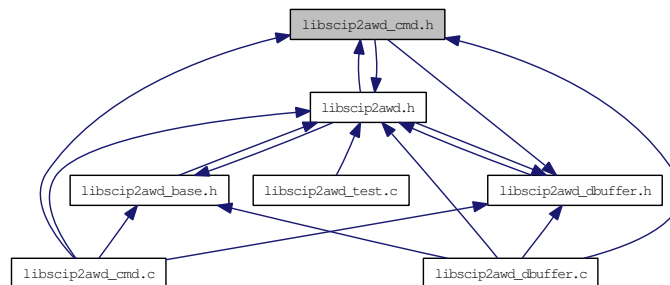
Library for Sokuiki-Sensor "URG". #include <termios.h>
#include <unistd.h>
#include <pthread.h>
#include <sys/time.h>
#include "libscip2awd.h"
#include "libscip2awd_dbuffer.h"

```

Include dependency graph for libscip2awd\_cmd.h:



This graph shows which files directly or indirectly include this file:



### Classes

- struct [SCIP2\\_SENSOR\\_VERSION](#)
- struct [SCIP2\\_SENSOR\\_PARAM](#)

### Typedefs

- typedef struct [SCIP2\\_SENSOR\\_VERSION](#) [S2Ver\\_t](#)
- typedef struct [SCIP2\\_SENSOR\\_PARAM](#) [S2Param\\_t](#)

### Functions

- int [Scip2CMD SCIP2](#) (S2Port \*apPort)

*Switch to SCIP2.0 mode.*

- int [Scip2CMD\\_SS](#) (S2Port \*apPort, const speed\_t acBitrate)  
*Change Device's Bitrate.*
- int [Scip2CMD\\_BM](#) (S2Port \*apPort)  
*Laser ON.*
- int [Scip2CMD\\_QT](#) (S2Port \*apPort)  
*Laser OFF.*
- int [Scip2CMD\\_GS](#) (S2Port \*apPort, int aStart, int aEnd, int aGroup, [S2Sdd\\_t](#) \*aData, const S2EncType acEnc)  
*Start getting scanned data.*
- int [Scip2CMD\\_StopGS](#) (S2Port \*apPort, [S2Sdd\\_t](#) \*aData)  
*Stop GS scanning.*
- int [Scip2CMD\\_TM](#) (S2Port \*apPort, struct timeval \*apTime)  
*Get time of device.*
- int [Scip2CMD\\_RS](#) (S2Port \*apPort)  
*Reset Device.*
- int [Scip2CMD\\_CR](#) (S2Port \*apPort, const int acDeboost)  
*Set rotating speed.*
- int [Scip2CMD\\_StopMS](#) (S2Port \*apPort, [S2Sdd\\_t](#) \*aData)  
*Stop MS scanning.*
- int [Scip2CMD\\_StartMS](#) (S2Port \*apPort, int aStart, int aEnd, int aGroup, int aCull, int aNum, [S2Sdd\\_t](#) \*aData, const S2EncType acEnc)  
*Start getting scanned data.*
- int [Scip2CMD\\_VV](#) (S2Port \*apPort, [S2Ver\\_t](#) \*apVer)  
*Get version info.*
- int [Scip2CMD\\_PP](#) (S2Port \*apPort, [S2Param\\_t](#) \*apParam)  
*Get param info.*

### 4.3.1 Detailed Description

Library for Sokuiki-Sensor "URG".

**Author:**

Atsushi WATANEBE <[atusi\\_w@roboken.esys.tsukuba.ac.jp](mailto:atusi_w@roboken.esys.tsukuba.ac.jp)> <[atusi\\_w@doramanjyu.com](mailto:atusi_w@doramanjyu.com)>



## 4.3.2 Typedef Documentation

### 4.3.2.1 typedef struct SCIP2\_SENSOR\_PARAM S2Param\_t

Structure of PP command

### 4.3.2.2 typedef struct SCIP2\_SENSOR\_VERSION S2Ver\_t

Structure of VV command

## 4.3.3 Function Documentation

### 4.3.3.1 int Scip2CMD\_BM (S2Port \* *apPort*)

Laser ON.

#### Parameters:

*\*apPort* Pointer to SCIP2.0 Device Port

#### Returns:

failed: false, succeeded: true

### 4.3.3.2 int Scip2CMD\_CR (S2Port \* *apPort*, const int *acDeboost*)

Set rotating speed.

#### Parameters:

*\*apPort* Pointer to SCIP2.0 Device Port

*acDeboost* Deboost rate

#### Returns:

failed: 0, succeeded: time of device

### 4.3.3.3 int Scip2CMD\_GS (S2Port \* *apPort*, int *aStart*, int *aEnd*, int *aGroup*, S2Sdd\_t \* *aData*, const S2EncType *acEnc*)

Start getting scanned data.

#### Parameters:

*\*apPort* Pointer to SCIP2.0 Device Port

*aStart* Start step

*aEnd* End step

*aGroup* Number of group

*\*aData* Pointer to buffer structure

*acEnc* Encode type

**Returns:**

failed: false, succeeded: true

**Attention:**

Scip2CMD\_StopGS must be called before calling another Scip2CMD function, if the device remains sending data to PC!!

**4.3.3.4 int Scip2CMD\_PP (S2Port \* *apPort*, S2Param\_t \* *apParam*)**

Get param info.

**Parameters:**

\**apPort* Pointer to SCIP2.0 Device Port

\**apParam* Pointer to param structure

**Returns:**

failed: 0, succeeded: time of device

**4.3.3.5 int Scip2CMD\_QT (S2Port \* *apPort*)**

Laser OFF.

**Parameters:**

\**apPort* Pointer to SCIP2.0 Device Port

**Returns:**

failed: false, succeeded: true

**4.3.3.6 int Scip2CMD\_RS (S2Port \* *apPort*)**

Reset Device.

**Parameters:**

\**apPort* Pointer to SCIP2.0 Device Port

**Returns:**

failed: false, succeeded: true

**4.3.3.7 int Scip2CMD SCIP2 (S2Port \* *apPort*)**

Switch to SCIP2.0 mode.

**Parameters:**

\**apPort* Pointer to SCIP2.0 Device Port

**Returns:**

false: failed, true: succeeded

#### 4.3.3.8 int Scip2CMD\_SS (S2Port \* *apPort*, const speed\_t *acBitrate*)

Change Device's Bitrate.

##### Parameters:

*\*apPort* Pointer to SCIP2.0 Device Port  
*acBitrate* Bitrate

##### Returns:

failed: false, succeeded: true

#### 4.3.3.9 int Scip2CMD\_StartMS (S2Port \* *apPort*, int *aStart*, int *aEnd*, int *aGroup*, int *aCull*, int *aNum*, S2Sdd\_t \* *aData*, const S2EncType *acEnc*)

Start getting scanned data.

##### Parameters:

*\*apPort* Pointer to SCIP2.0 Device Port  
*aStart* Start step  
*aEnd* End step  
*aGroup* Number of group  
*aCull* Culling clearance  
*aNum* Number of scan  
*\*aData* Pointer to buffer structure  
*acEnc* Encode type

##### Returns:

failed: false, succeeded: true

##### Attention:

Scip2CMD\_StopMS must be called before calling another Scip2CMD function, if the device remains sending data to PC!!

#### 4.3.3.10 int Scip2CMD\_StopGS (S2Port \* *apPort*, S2Sdd\_t \* *aData*)

Stop GS scanning.

##### Parameters:

*\*apPort* Pointer to SCIP2.0 Device Port  
*\*aData* Pointer to buffer structure

##### Returns:

failed: false, succeeded: true

**4.3.3.11 int Scip2CMD\_StopMS (S2Port \* *apPort*, S2Sdd\_t \* *aData*)**

Stop MS scanning.

**Parameters:**

\**apPort* Pointer to SCIP2.0 Device Port

\**aData* Pointer to buffer structure

**Returns:**

failed: false, succeeded: true

**4.3.3.12 int Scip2CMD\_TM (S2Port \* *apPort*, struct timeval \* *apTime*)**

Get time of device.

**Parameters:**

\**apPort* Pointer to SCIP2.0 Device Port

\**apTime* Start time of device

**Returns:**

failed: 0, succeeded: 1

**4.3.3.13 int Scip2CMD\_VV (S2Port \* *apPort*, S2Ver\_t \* *apVer*)**

Get version info.

**Parameters:**

\**apPort* Pointer to SCIP2.0 Device Port

\**apVer* Pointer to version structure

**Returns:**

failed: 0, succeeded: time of device

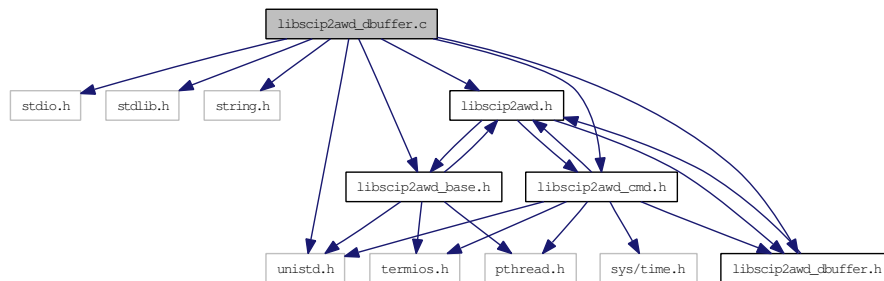
## 4.4 libscip2awd\_dbuffer.c File Reference

```

Library for Sokuiki-Sensor "URG". #include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <unistd.h>
#include "libscip2awd.h"
#include "libscip2awd_base.h"
#include "libscip2awd_cmd.h"
#include "libscip2awd_dbuffer.h"

```

Include dependency graph for libscip2awd\_dbuffer.c:



## Functions

- void [S2Sdd\\_Init](#) ([S2Sdd\\_t](#) \*aData)  
*Initialize dual buffer structure.*
- void [S2Sdd\\_Dest](#) ([S2Sdd\\_t](#) \*aData)  
*Destruct dual buffer structure.*
- int [S2Sdd\\_BeginEx](#) ([S2Sdd\\_t](#) \*aData, [S2Scan\\_t](#) \*\*aScan, int aBlock)  
*Start using Data.*
- int [S2Sdd\\_Begin\\_Trylock](#) ([S2Sdd\\_t](#) \*aData, [S2Scan\\_t](#) \*\*aScan)  
*Start using Data (non-blocking).*
- int [S2Sdd\\_Begin\\_Lock](#) ([S2Sdd\\_t](#) \*aData, [S2Scan\\_t](#) \*\*aScan)  
*Start using Data (blocking).*
- void [S2Sdd\\_End](#) ([S2Sdd\\_t](#) \*aData)  
*End using Data.*
- void [S2Sdd\\_StopThread](#) ([S2Sdd\\_t](#) \*aData)  
*Stop thread which reads data of MS/GS command.*
- void \* [S2Sdd\\_RecvData](#) (void \*aArg)

*Get Scanned data.*

- void \* [S2Sdd\\_RecvDataCont](#) (void \*aArg)  
*Recive scanned data continually.*

## 4.4.1 Detailed Description

Library for Sokuiki-Sensor "URG".

### Author:

Atsushi WATANEBE <[atusi\\_w@roboken.esys.tsukuba.ac.jp](mailto:atusi_w@roboken.esys.tsukuba.ac.jp)> <[atusi\\_w@doramanjyu.com](mailto:atusi_w@doramanjyu.com)>

## 4.4.2 Function Documentation

### 4.4.2.1 int S2Sdd\_Begin\_Lock (S2Sdd\_t \* aData, S2Scan\_t \*\* aScan)

Start using Data (blocking).

#### Parameters:

- \*aData Pointer to dual buffer structure
- \*\*aScan Pointer to buffer structure handle

#### Returns:

failed: NULL, fatal error: -1, succeeded: Pointer to front buffer

### 4.4.2.2 int S2Sdd\_Begin\_Trylock (S2Sdd\_t \* aData, S2Scan\_t \*\* aScan)

Start using Data (non-blocking).

#### Parameters:

- \*aData Pointer to dual buffer structure
- \*\*aScan Pointer to buffer structure handle

#### Returns:

failed: NULL, fatal error: -1, succeeded: Pointer to front buffer

### 4.4.2.3 int S2Sdd\_BeginEx (S2Sdd\_t \* aData, S2Scan\_t \*\* aScan, int aBlock)

Start using Data.

#### Parameters:

- \*aData Pointer to dual buffer structure
- \*\*aScan Pointer to buffer structure handle

*aBlock* Blocking: 1, Non-blocking: 0

**Returns:**

failed: NULL, fatal error: -1, succeeded: Pointer to front buffer

**4.4.2.4 void S2Sdd\_Dest (S2Sdd\_t \* aData)**

Destruct dual buffer structure.

**Parameters:**

*\*aData* Pointer to dual buffer structure

**4.4.2.5 void S2Sdd\_End (S2Sdd\_t \* aData)**

End using Data.

**Parameters:**

*\*aData* Pointer to dual buffer structure

**4.4.2.6 void S2Sdd\_Init (S2Sdd\_t \* aData)**

Initialize dual buffer structure.

**Parameters:**

*\*aData* Pointer to dual buffer structure

**4.4.2.7 void\* S2Sdd\_RecvData (void \* aArg)**

Get Scanned data.

**Parameters:**

*\*aArg* Pointer to dual buffer structure

**4.4.2.8 void\* S2Sdd\_RecvDataCont (void \* aArg)**

Recive scanned data continually.

**Parameters:**

*\*aArg* Pointer to dual buffer structure

**4.4.2.9 void S2Sdd\_StopThread (S2Sdd\_t \* aData)**

Stop thread which reads data of MS/GS command.

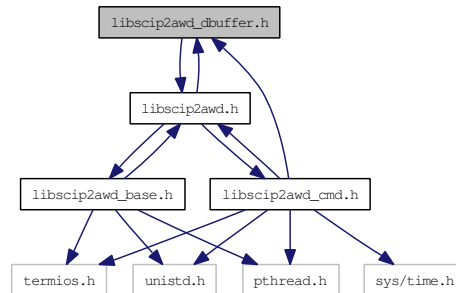
**Parameters:**

*\*aData* Pointer to buffer structure

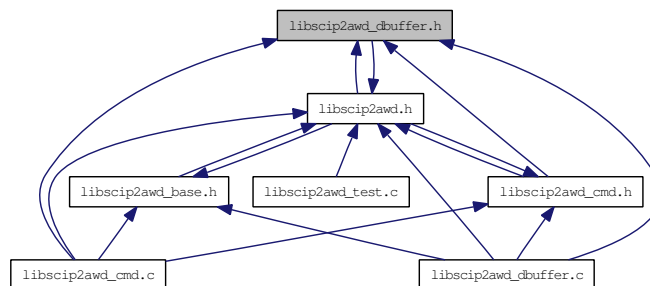
## 4.5 libscip2awd\_dbuffer.h File Reference

Library for Sokuiki-Sensor "URG". #include "libscip2awd.h"

Include dependency graph for libscip2awd\_dbuffer.h:



This graph shows which files directly or indirectly include this file:



### Classes

- struct [SCIP2\\_SCANNED\\_DATA](#)
- struct [SCIP2\\_SCANNED\\_DATA\\_TRI](#)

### Typedefs

- typedef struct [SCIP2\\_SCANNED\\_DATA](#) [S2Scan\\_t](#)
- typedef struct [SCIP2\\_SCANNED\\_DATA\\_TRI](#) [S2Sdd\\_t](#)

### Functions

- void \* [S2Sdd\\_RecvData](#) (void \*aArg)  
*Get Scanned data.*
- void [S2Sdd\\_End](#) ([S2Sdd\\_t](#) \*aData)  
*End using Data.*
- int [S2Sdd\\_BeginEx](#) ([S2Sdd\\_t](#) \*aData, [S2Scan\\_t](#) \*\*aScan, int aBlock)  
*Start using Data.*



- int `S2Sdd_Begin_Trylock` (`S2Sdd_t *aData`, `S2Scan_t **aScan`)  
*Start using Data (non-blocking).*
- int `S2Sdd_Begin_Lock` (`S2Sdd_t *aData`, `S2Scan_t **aScan`)  
*Start using Data (blocking).*
- void `S2Sdd_Dest` (`S2Sdd_t *aData`)  
*Destruct dual buffer structure.*
- void `S2Sdd_Init` (`S2Sdd_t *aData`)  
*Initialize dual buffer structure.*
- void \* `S2Sdd_RecvDataCont` (void \*`aArg`)  
*Recive scanned data continually.*
- void `S2Sdd_StopThread` (`S2Sdd_t *aData`)  
*Stop thread which reads data of MS/GS command.*

### 4.5.1 Detailed Description

Library for Sokuiki-Sensor "URG".

**Author:**

Atsushi WATANEBE <[atusi\\_w@roboken.esys.tsukuba.ac.jp](mailto:atusi_w@roboken.esys.tsukuba.ac.jp)> <[atusi\\_w@doramanjyu.com](mailto:atusi_w@doramanjyu.com)>

### 4.5.2 Typedef Documentation

#### 4.5.2.1 typedef struct SCIP2\_SCANNED\_DATA S2Scan\_t

Buffer structure for scanned data

#### 4.5.2.2 typedef struct SCIP2\_SCANNED\_DATA\_TRI S2Sdd\_t

Multi buffer structure for scanned data

### 4.5.3 Function Documentation

#### 4.5.3.1 int S2Sdd\_Begin\_Lock (S2Sdd\_t \* *aData*, S2Scan\_t \*\* *aScan*)

Start using Data (blocking).

**Parameters:**

*\*aData* Pointer to dual buffer structure

*\*\*aScan* Pointer to buffer structure handle

**Returns:**

failed: NULL, fatal error: -1, succeeded: Pointer to front buffer

**4.5.3.2 int S2Sdd\_Begin\_Trylock (S2Sdd\_t \* *aData*, S2Scan\_t \*\* *aScan*)**

Start using Data (non-blocking).

**Parameters:**

*\*aData* Pointer to dual buffer structure

*\*\*aScan* Pointer to buffer structure handle

**Returns:**

failed: NULL, fatal error: -1, succeeded: Pointer to front buffer

**4.5.3.3 int S2Sdd\_BeginEx (S2Sdd\_t \* *aData*, S2Scan\_t \*\* *aScan*, int *aBlock*)**

Start using Data.

**Parameters:**

*\*aData* Pointer to dual buffer structure

*\*\*aScan* Pointer to buffer structure handle

*aBlock* Blocking: 1, Non-blocking: 0

**Returns:**

failed: NULL, fatal error: -1, succeeded: Pointer to front buffer

**4.5.3.4 void S2Sdd\_Dest (S2Sdd\_t \* *aData*)**

Destruct dual buffer structure.

**Parameters:**

*\*aData* Pointer to dual buffer structure

**4.5.3.5 void S2Sdd\_End (S2Sdd\_t \* *aData*)**

End using Data.

**Parameters:**

*\*aData* Pointer to dual buffer structure

**4.5.3.6 void S2Sdd\_Init (S2Sdd\_t \* *aData*)**

Initialize dual buffer structure.

**Parameters:**

*\*aData* Pointer to dual buffer structure

**4.5.3.7 void\* S2Sdd\_RecvData (void \* *aArg*)**

Get Scanned data.

**Parameters:**

*\*aArg* Pointer to dual buffer structure

**4.5.3.8 void\* S2Sdd\_RecvDataCont (void \* *aArg*)**

Recive scanned data continually.

**Parameters:**

*\*aArg* Pointer to dual buffer structure

**4.5.3.9 void S2Sdd\_StopThread (S2Sdd\_t \* *aData*)**

Stop thread which reads data of MS/GS command.

**Parameters:**

*\*aData* Pointer to buffer structure

## 4.6 libscip2awd\_test.c File Reference

Test program for libscip2. #include <stdio.h>

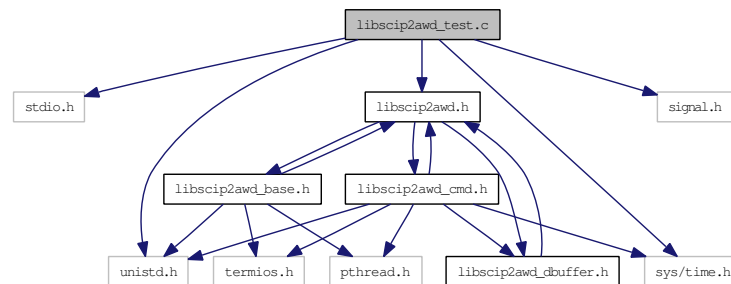
#include <unistd.h>

#include <sys/time.h>

#include <signal.h>

#include "libscip2awd.h"

Include dependency graph for libscip2awd\_test.c:



### Functions

- void `ctrlc` (int aN)  
*Ctrl+C trap.*
- int `main` (int aArgc, char \*\*appArgv)  
*Main function.*

### Variables

- int `gShutoff`

#### 4.6.1 Detailed Description

Test program for libscip2.

##### Author:

Atsushi WATANEBE <[atusi\\_w@roboken.esys.tsukuba.ac.jp](mailto:atusi_w@roboken.esys.tsukuba.ac.jp)> <[atusi\\_w@doramanjyu.com](mailto:atusi_w@doramanjyu.com)>

#### 4.6.2 Function Documentation

##### 4.6.2.1 void ctrlc (int aN)

Ctrl+C trap.

**Parameters:**

*aN* not used

**4.6.2.2 int main (int *aArgc*, char \*\* *appArgv*)**

Main function.

**Parameters:**

*aArgc* Number of Arguments

*appArgv* Arguments

**Returns:**

failed: 0, succeeded: 1

**4.6.3 Variable Documentation****4.6.3.1 int gShutoff**

Flag

# Index

- ctrlc
  - libscip2awd\_test.c, 30
- gShutoff
  - libscip2awd\_test.c, 31
- libscip2awd.h, 9
- libscip2awd\_cmd.c, 11
  - Scip2CMD\_BM, 12
  - Scip2CMD\_CR, 12
  - Scip2CMD\_GS, 13
  - Scip2CMD\_PP, 13
  - Scip2CMD\_QT, 13
  - Scip2CMD\_RS, 13
  - Scip2CMD SCIP2, 14
  - Scip2CMD\_SS, 14
  - Scip2CMD\_StartMS, 14
  - Scip2CMD\_StopGS, 15
  - Scip2CMD\_StopMS, 15
  - Scip2CMD\_TM, 15
  - Scip2CMD\_VV, 15
- libscip2awd\_cmd.h, 17
  - S2Param\_t, 19
  - S2Ver\_t, 19
  - Scip2CMD\_BM, 19
  - Scip2CMD\_CR, 19
  - Scip2CMD\_GS, 19
  - Scip2CMD\_PP, 20
  - Scip2CMD\_QT, 20
  - Scip2CMD\_RS, 20
  - Scip2CMD SCIP2, 20
  - Scip2CMD\_SS, 20
  - Scip2CMD\_StartMS, 21
  - Scip2CMD\_StopGS, 21
  - Scip2CMD\_StopMS, 21
  - Scip2CMD\_TM, 22
  - Scip2CMD\_VV, 22
- libscip2awd\_dbuffer.c, 23
  - S2Sdd\_Begin\_Lock, 24
  - S2Sdd\_Begin\_Trylock, 24
  - S2Sdd\_BeginEx, 24
  - S2Sdd\_Dest, 25
  - S2Sdd\_End, 25
  - S2Sdd\_Init, 25
  - S2Sdd\_RecvData, 25
  - S2Sdd\_RecvDataCont, 25
  - S2Sdd\_StopThread, 25
- libscip2awd\_dbuffer.h, 26
  - S2Scan\_t, 27
  - S2Sdd\_Begin\_Lock, 27
  - S2Sdd\_Begin\_Trylock, 28
  - S2Sdd\_BeginEx, 28
  - S2Sdd\_Dest, 28
  - S2Sdd\_End, 28
  - S2Sdd\_Init, 28
  - S2Sdd\_RecvData, 28
  - S2Sdd\_RecvDataCont, 29
  - S2Sdd\_StopThread, 29
  - S2Sdd\_t, 27
- libscip2awd\_test.c, 30
  - ctrlc, 30
  - gShutoff, 31
  - main, 31
- main
  - libscip2awd\_test.c, 31
- S2Param\_t
  - libscip2awd\_cmd.h, 19
- S2Scan\_t
  - libscip2awd\_dbuffer.h, 27
- S2Sdd\_Begin\_Lock
  - libscip2awd\_dbuffer.c, 24
  - libscip2awd\_dbuffer.h, 27
- S2Sdd\_Begin\_Trylock
  - libscip2awd\_dbuffer.c, 24
  - libscip2awd\_dbuffer.h, 28
- S2Sdd\_BeginEx
  - libscip2awd\_dbuffer.c, 24
  - libscip2awd\_dbuffer.h, 28
- S2Sdd\_Dest
  - libscip2awd\_dbuffer.c, 25
  - libscip2awd\_dbuffer.h, 28
- S2Sdd\_End
  - libscip2awd\_dbuffer.c, 25
  - libscip2awd\_dbuffer.h, 28
- S2Sdd\_Init
  - libscip2awd\_dbuffer.c, 25
  - libscip2awd\_dbuffer.h, 28
- S2Sdd\_RecvData

- libscip2awd\_dbuffer.c, 25
- libscip2awd\_dbuffer.h, 28
- S2Sdd\_RecvDataCont
  - libscip2awd\_dbuffer.c, 25
  - libscip2awd\_dbuffer.h, 29
- S2Sdd\_StopThread
  - libscip2awd\_dbuffer.c, 25
  - libscip2awd\_dbuffer.h, 29
- S2Sdd\_t
  - libscip2awd\_dbuffer.h, 27
- S2Ver\_t
  - libscip2awd\_cmd.h, 19
- SCIP2\_SCANNED\_DATA, 5
- SCIP2\_SCANNED\_DATA\_TRI, 6
- SCIP2\_SENSOR\_PARAM, 7
- SCIP2\_SENSOR\_VERSION, 8
- Scip2CMD\_BM
  - libscip2awd\_cmd.c, 12
  - libscip2awd\_cmd.h, 19
- Scip2CMD\_CR
  - libscip2awd\_cmd.c, 12
  - libscip2awd\_cmd.h, 19
- Scip2CMD\_GS
  - libscip2awd\_cmd.c, 13
  - libscip2awd\_cmd.h, 19
- Scip2CMD\_PP
  - libscip2awd\_cmd.c, 13
  - libscip2awd\_cmd.h, 20
- Scip2CMD\_QT
  - libscip2awd\_cmd.c, 13
  - libscip2awd\_cmd.h, 20
- Scip2CMD\_RS
  - libscip2awd\_cmd.c, 13
  - libscip2awd\_cmd.h, 20
- Scip2CMD SCIP2
  - libscip2awd\_cmd.c, 14
  - libscip2awd\_cmd.h, 20
- Scip2CMD\_SS
  - libscip2awd\_cmd.c, 14
  - libscip2awd\_cmd.h, 20
- Scip2CMD\_StartMS
  - libscip2awd\_cmd.c, 14
  - libscip2awd\_cmd.h, 21
- Scip2CMD\_StopGS
  - libscip2awd\_cmd.c, 15
  - libscip2awd\_cmd.h, 21
- Scip2CMD\_StopMS
  - libscip2awd\_cmd.c, 15
  - libscip2awd\_cmd.h, 21
- Scip2CMD\_TM
  - libscip2awd\_cmd.c, 15
  - libscip2awd\_cmd.h, 22
- Scip2CMD\_VV
  - libscip2awd\_cmd.c, 15
- libscip2awd\_cmd.h, 22