

Soliton SCTS-E Extender

SCTS-E DLL Programming Description

Rev 1.0
May 15, 2015

Rev. No.	Description	Date	Approved
0.1	Initial	May/15/2015	Vincent
1.0	First Relase	May/15/2015	Vincent

1. Installation.....	2
2. Function Description.....	2
Int SCTSECTL_OPEN().....	2
Int SCTSECTL_OPEN(int addr)	3
int SCTSECTL_CLOSE().....	3
int SCTSECTL_BEEP(int msec , int freq)	4
int SCTSECTL_LEDGO().....	4
int SCTSECTL_LEDNG().....	4
int SCTSECTL_LEDOFF()	4
int SCTSECTL_MEAS_MV_VOLTAGE(double *volt)	5
int SCTSECTL_MEAS_MV_CURRENT(double *current).....	5
int SCTSECTL_MEAS_VBAT_VOLTAGE(double *volt)	5
int SCTSECTL_MEAS_MV_VBAT_CURRENT(double *current)	6
int SCTSECTL_CHECKSDCD()	6
int SCTSECTL_GPIOSET(int pinno)	6
int SCTSECTL_GPIORESET(int pinno)	7
int SCTSECTL_MCARD_DISCONNECT()	7
int SCTSECTL_MCARD_NORMAL()	7
int SCTSECTL_MCARD_CHECKOC()	7
int SCTSECTL_PON()	8
int SCTSECTL_POFF().....	8
3. Contact	8

1. Installation

Execute the SCTS-E_xxx_Setup.exe from the CD. All the required component will be installed in the Program files\Soliton\SCTS-E\ folder. For developer, who want to integrate test program with the SCTS-E control. Please find all the needed samples and development resources in the CD files of SCTS-E\Example Code folder.

Note : SCTS-E_xxx_Setup.exe Installation File

Windows Operating System 32-bit : SCTSEU_x86_Setup .exe

Windows Operating System 64-bit : SCTSEU_x64_Setup .exe

2. Function Description

Int SCTSECTL_OPEN()

Syntax:

int SCTSECTL_OPEN()

Description:

Call this function to scan and initialize the SCTS-E installed in PC.

Return Value:

Return 0 if success, 1 if fail, 2 if SCTS-E not exist.

Usage:

sctsectl_open();

Notes:

If multiple SCTS-E are installed, please make sure the address

Setting are different.

Please refer to Hardware setting guide.

int SCTSECTL_OPEN(int addr)

Syntax:

int SCTSECTL_OPEN(int addr)

Description:

Call this function to open and initialize the dedicate SCTS-E with the address setting.

addr: Module addr, range 0~3.

Return Value:

If successful, will return the SCTS handle of data structure setsctl_t.

Return 0 if success, 1 if fail, 2 if SCTS-E not exist.

Usage:

sctsectl_open(0);

int SCTSECTL_CLOSE()

Syntax:

int SCTSECTL_CLOSE()

Description:

Close the handle of SCTS-E board

Return Value:

Return 0 if successful; 1 if error.

Usage:

SCTSECTL_CLOSE();

Notes:

Be sure to call SCTSECTL_CLOSE before your application exits.

int SCTSECTL_BEEP(int msec , int freq)

Syntax:

int SCTSECTL_BEEP (int msec , int freq)

Description:

Play Beeper.

int msec: length Millisecond

int freq: frequency of sound, range 0~3

Return Value:

Return 0 if successful; 1 if error.

Usage:

SCTSECTL_BEEP(1000, 0);

int SCTSECTL_LEDGO()

Syntax:

int SCTSECTL_LEDGO()

Description:

Turn on the GO LED to indicate the test status.

Return Value:

Return 0 if successful; 1 if error.

Usage:

SCTSECTL_LEDGO();

int SCTSECTL_LEDNG()

Syntax:

int SCTSECTL_LEDNG()

Description:

Turn on the NO-GO LED to indicate the test status.

Return Value:

Return 0 if successful; 1 if error.

Usage:

SCTSECTL_LEDNG();

int SCTSECTL_LEDOFF()

Syntax:

int SCTSECTL_LEDOFF()

Description:

Turn off both the GO and NG LEDs

Return Value:

Return 0 if successful; 1 if error.

Usage:

SCTSECTL_LEDOFF();

int SCTSECTL_MEAS_MV_VOLTAGE(double *volt)

Syntax:

int SCTSECTL_MEAS_MV_VOLTAGE(double *volt)

Description:

Measure master voltage rail voltage.

double *volt : if success, the measured voltage will stored in the passed double pointer.

Return Value:

Return 0 if successful; 1 if error; 3 if ADC not exist.

Usage:

status = SCTSECTL_MEAS_MV_VOLTAGE(&volt);

int SCTSECTL_MEAS_MV_CURRENT(double *current)

Syntax:

int SCTSECTL_MEAS_MV_CURRENT(double *current)

Description:

Measure master voltage rail current.

double *current: if success, the measured current will stored in the passed double pointer.

Return Value:

Return 0 if successful; 1 if error; 3 if ADC not exist.

Usage:

status = SCTSECTL_MEAS_MV_CURRENT(¤t);

int SCTSECTL_MEAS_VBAT_VOLTAGE(double *volt)

Syntax:

int SCTSECTL_MEAS_VBAT_VOLTAGE(double *volt)

Description:

Measure VBAT rail voltage.

double *volt : if success, the measured voltage will stored in the passed double pointer.

Return Value:

Return 0 if successful; 1 if error; 3 if ADC not exist.

Usage:

```
status = SCTSECTL_MEAS_MV_VOLTAGE( &volt);
```

int SCTSECTL_MEAS_MV_VBAT_CURRENT(double *current)

Syntax:

```
int SCTSECTL_MEAS_VBAT_CURRENT(double *current)
```

Description:

Measure VBAT rail current.

double *current: if success, the measured current will stored in the passed double pointer.

Return Value:

Return 0 if successful; 1 if error; 3 if ADC not exist.

Usage:

```
status = SCTSECTL_MEAS_MV_CURRENT(&current);
```

int SCTSECTL_CHECKSDCD()

Syntax:

```
int SCTSECTL_CHECKSDCD()
```

Description:

Check SD card detect pin, check whether SD card is inserted.

Return Value:

Return 1 if card detected ; 0 if no card detected; -1 if error.

Usage:

```
SCTSECTL_CHECKSDCD();
```

int SCTSECTL_GPIOSET(int pinno)

Syntax:

```
int SCTSECTL_GPIOSET(int pinno)
```

Description:

Set GPIO output to high.

pinno: GPIO pin no, range 0~1

Return Value:

Return 0 if successful; 1 if error.

Usage:

```
SCTSECTL_GPIOSET(0);
```

int SCTSECTL_GPIORESET(int pinno)

Syntax:

int SCTSECTL_GPIORESET(int pinno)

Description:

Set GPIO output to low.

pinno: GPIO pin no, range 0~1

Return Value:

Return 0 if successful; 1 if error.

Usage:

SCTSECTL_GPIORESET(0);

int SCTSECTL_MCARD_DISCONNECT()

Syntax:

int SCTSECTL_MCARD_DISCONNECT ()

Description:

Manual mode disconnect socket.

Return Value:

Return 0 if successful; 1 if error.

Usage:

SCTSECTL_MCARD_DISCONNECT();

int SCTSECTL_MCARD_NORMAL()

Syntax:

int SCTSECTL_MCARD_NORMAL()

Description:

Manual mode force normal mode.

Return Value:

Return 0 if successful; 1 if error.

Usage:

SCTSECTL_MCARD_NORMAL();

int SCTSECTL_MCARD_CHECKOC()

Syntax:

int SCTSECTL_MCARD_CHECKOC()

Description:

Check over current flag.

Return Value:

Return 0 if successful; 1 if error.

Usage:

SCTSECTL_MCARD_CHECKOC();

int SCTSECTL_PON()

Syntax:

int SCTSECTL_PON()

Description:

Manual mode force power on.

Return Value:

Return 0 if successful; 1 if error.

Usage:

SCTSECTL_PON();

int SCTSECTL_POFF()

Syntax:

int SCTSECTL_POFF()

Description:

Manual mode force power off.

Return Value:

Return 0 if successful; 1 if error.

Usage:

SCTSECTL_POFF();

3. Contact

Please contact us if you have experienced any problems.

E-mail: info@soliton.com.tw

Tel: +886-3-6566996

Fax: +886-3-6566883