Si7021 Software Manual

1.0

Generated by Doxygen 1.8.18

1 File Index 1

1.1 File List . . . 1

2 File Documentation 3

2.1 C:/Users/Dominik/Desktop/doc/i2c.c File Reference . . . 3

2.1.1 Function Documentation . . . 3

2.1.1.1 i2c_open() . . . 3

2.1.1.2 i2c_rx() . . . 4

2.1.1.3 i2c_slave_address() . . . 4

2.1.1.4 i2c_tx() . . . 4

2.1.2 Variable Documentation . . . 4

2.1.2.1 i2c_dev . . . 4

2.1.2.2 i2c_status . . . 4

2.2 C:/Users/Dominik/Desktop/doc/i2c.h File Reference . . . 5

2.2.1 Function Documentation . . . 5

2.2.1.1 i2c_open() . . . 5

2.2.1.2 i2c_rx() . . . 5

2.2.1.3 i2c_slave_address() . . . 5

2.2.1.4 i2c_tx() . . . 6

2.3 C:/Users/Dominik/Desktop/doc/si7021.c File Reference . . . 6

2.3.1 Macro Definition Documentation . . . 7

2.3.1.1 SI7021_ADDRESS . . . 8

2.3.1.2 SI7021_FIRMVERS_A . . . 8

2.3.1.3 SI7021_FIRMVERS_B . . . 8

2.3.1.4 SI7021_ID11 . . . 8

2.3.1.5 SI7021_ID12 . . . 8

2.3.1.6 SI7021_ID21 . . . 8

2.3.1.7 SI7021_ID22 . . . 8

2.3.1.8 SI7021_MEASRH_HOLD . . . 8

2.3.1.9 SI7021_MEASRH_NOHOLD . . . 8

2.3.1.10 SI7021_MEASTEMP_HOLD . . . 8

2.3.1.11 SI7021_MEASTEMP_NOHOLD . . . 9

2.3.1.12 SI7021_READHEATER_REG . . . 9

2.3.1.13 SI7021_READPREVTEMP . . . 9

2.3.1.14 SI7021_READRHT_REG . . . 9

2.3.1.15 SI7021_REG_HTRE_BIT . . . 9

2.3.1.16 SI7021_RESET . . . 9

2.3.1.17 SI7021_REV_1 . . . 9

2.3.1.18 SI7021_REV_2 . . . 9

2.3.1.19 SI7021_WRITEHEATER_REG . . . 9

2.3.1.20 SI7021_WRITERHT_REG . . . 9

2.3.2 Function Documentation . . . 9

2.3.2.1 si7021_heat_level() . . . 10
2.3.2.2 si7021_heater() . . . 10
2.3.2.3 si7021_init() . . . 10
2.3.2.4 si7021_read_firmware() . . . 10
2.3.2.5 si7021_read_humidity() . . . 11
2.3.2.6 si7021_read_serial_number() . . . 11
2.3.2.7 si7021_read_temperature() . . . 11
2.3.2.8 si7021_read_user_reg() . . . 11
2.3.2.9 si7021_reset() . . . 12
2.3.2.10 si7021_write_user_reg() . . . 12
2.3.3 Variable Documentation . . . 12
2.3.3.1 firmware_revsion . . . 12
2.3.3.2 model . . . 12
2.3.3.3 rxBuffer . . . 12
2.3.3.4 txBuffer . . . 12
2.4 C:/Users/Dominik/Desktop/doc/si7021.h File Reference . . . 13
2.4.1 Function Documentation . . . 13
2.4.1.1 si7021_heat_level() . . . 13
2.4.1.2 si7021_heater() . . . 13
2.4.1.3 si7021_init() . . . 14
2.4.1.4 si7021_read_firmware() . . . 14
2.4.1.5 si7021_read_humidity() . . . 14
2.4.1.6 si7021_read_serial_number() . . . 14
2.4.1.7 si7021_read_temperature() . . . 15
2.4.1.8 si7021_read_user_reg() . . . 15
2.4.1.9 si7021_reset() . . . 15
2.4.1.10 si7021_write_user_reg() . . . 15

Index **17**

Chapter 1

File Index

1.1 File List

Here is a list of all files with brief descriptions:

C:/Users/Dominik/Desktop/doc/i2c.c 3
C:/Users/Dominik/Desktop/doc/i2c.h 5
C:/Users/Dominik/Desktop/doc/si7021.c 6
C:/Users/Dominik/Desktop/doc/si7021.h 13

Chapter 2

File Documentation

2.1 C:/Users/Dominik/Desktop/doc/i2c.c File Reference

```
#include "i2c.h"
```

Functions

- void i2c_open (void)
  i2c_open()
- void i2c_slave_address (alt_u8 slave_address)
  i2c_slave_address()
- void i2c_tx (alt_u8 bytes, alt_u8 *data)
  i2c_tx()
- void i2c_rx (alt_u8 bytes, alt_u8 *data)
  i2c_rx()

2.1.1 Function Documentation

2.1.1.1 i2c_open()

```
void i2c_open (
            void  )
```

i2c_open()

<Open is communication
Function activates i2c device

Parameters

in	*none*	
out	*void*	

2.1.1.2 i2c_rx()

```
void i2c_rx (
            alt_u8 bytes,
            alt_u8 * data )
```

i2c_rx()
Receive data from i2c slave. Data to receive needs to be an array. If only one byte to receive data[0]

Parameters

in	*bytes*	Ammount of bytes in the receive array
in	*data*	Data to receive
out	*none*	

2.1.1.3 i2c_slave_address()

```
void i2c_slave_address (
            alt_u8 slave_address )
```

i2c_slave_address()
Transmit data to i2c slave.
Set address of I2C slave

Parameters

in	*slave_address*	Address of the i2c slave
out	*none*	

2.1.1.4 i2c_tx()

```
void i2c_tx (
            alt_u8 bytes,
            alt_u8 * data )
```

i2c_tx()
Receive data from i2c slave.
Set data to i2c slave, Data to send needs to be an array. If only one byte to send data[0]

Parameters

in	*bytes*	Ammount of bytes in the transmit array
in	*data*	Data to transmit
out	*none*	

2.1.2 Variable Documentation

2.1.2.1 i2c_dev

```
ALT_AVALON_I2C_DEV_t* i2c_dev
```

2.1.2.2 i2c_status

```
ALT_AVALON_I2C_STATUS_CODE i2c_status
```

2.2 C:/Users/Dominik/Desktop/doc/i2c.h File Reference

```
#include <alt_types.h>
#include <altera_avalon_i2c.h>
#include <io.h>
#include <system.h>
#include <stdio.h>
```

Functions

- void i2c_open (void)

  <Open is communication
- void i2c_slave_address (alt_u8 slave_address)

  Transmit data to i2c slave.
- void i2c_tx (alt_u8 bytes, alt_u8 *data)

  Receive data from i2c slave.
- void i2c_rx (alt_u8 bytes, alt_u8 *data)

  i2c_rx()

2.2.1 Function Documentation

2.2.1.1 i2c_open()

```
void i2c_open (
            void  )
```

<Open is communication
Choose slave address
<Open is communication
Function activates i2c device

Parameters

in	*none*	
out	*void*	

2.2.1.2 i2c_rx()

```
void i2c_rx (
            alt_u8 bytes,
            alt_u8 * data )
```

i2c_rx()
Receive data from i2c slave. Data to receive needs to be an array. If only one byte to receive data[0]

Parameters

in	*bytes*	Ammount of bytes in the receive array
in	*data*	Data to receive
out	*none*	

2.2.1.3 i2c_slave_address()

```
void i2c_slave_address (
            alt_u8 slave_address )
```

Transmit data to i2c slave.
Transmit data to i2c slave.
Set address of I2C slave

Parameters

in	*slave_address*	Address of the i2c slave
out	*none*	

2.2.1.4 i2c_tx()

```
void i2c_tx (
            alt_u8 bytes,
            alt_u8 * data )
```

Receive data from i2c slave.
Receive data from i2c slave.
Set data to i2c slave, Data to send needs to be an array. If only one byte to send data[0]

Parameters

in	*bytes*	Ammount of bytes in the transmit array
in	*data*	Data to transmit
out	*none*	

2.3 C:/Users/Dominik/Desktop/doc/si7021.c File Reference

```
#include "si7021.h"
```

Macros

- #define SI7021_ADDRESS 0x40

  device base address
- #define SI7021_MEASRH_HOLD 0xE5

  Measure Relative Humidity, Hold Master Mode.
- #define SI7021_MEASRH_NOHOLD 0xF5

  Measure Relative Humidity, No Hold Master Mode.
- #define SI7021_MEASTEMP_HOLD 0xE3
- #define SI7021_MEASTEMP_NOHOLD 0xF3

  Measure Temperature, No Hold Master Mode.
- #define SI7021_READPREVTEMP 0xE0

  Read Temperature Value from Previous RH Measurement.
- #define SI7021_RESET 0xFE
- #define SI7021_WRITERHT_REG 0xE6

  Write RH/T User Register 1.
- #define SI7021_READRHT_REG 0xE7
- #define SI7021_WRITEHEATER_REG 0x51

  Write Heater Control Register.
- #define SI7021_READHEATER_REG 0x11

  Read Heater Control Register.
- #define SI7021_REG_HTRE_BIT 0x02

  Control Register Heater Bit.
- #define SI7021_ID11 0xFA

Read Electronic ID 1. Byte

- #define SI7021_ID12 0x0F
- #define SI7021_ID21 0xFC

  Read Electronic ID 2. Byte.
- #define SI7021_ID22 0xC9
- #define SI7021_FIRMVERS_A 0x84

  Read Firmware Revision.
- #define SI7021_FIRMVERS_B 0xB8
- #define SI7021_REV_1 0xff

  Sensor revision 1.
- #define SI7021_REV_2 0x20

  Sensor revision 2.

Functions

- alt_u8 si7021_read_user_reg (alt_u8 reg)

  si7021_read_user_reg();
- void si7021_write_user_reg (alt_u8 reg, alt_u8 value)

  si7021_write_user_reg();
- void si7021_reset ()

  void si7021_reset();
- void si7021_init ()

  void si7021_init();
- float si7021_read_humidity ()

  si7021_read_humidity();
- float si7021_read_temperature ()

  void si7021_read_temperature()
- void si7021_read_firmware ()

  si7021_read_firmware();
- void si7021_read_serial_number ()

  si7021_read_serial_number();
- void si7021_heater (alt_u8 set)

  si7021_heater();
- void si7021_heat_level (alt_u8 level)

  si7021_heater_level();

Variables

- alt_u8 txBuffer [3]

  global buffer to hold tx
- alt_u8 rxBuffer [3]

  global buffer to hold rx
- alt_u8 firmware_revsion = 0

  firmware revision number
- alt_u8 model = 0

  sensor model

2.3.1 Macro Definition Documentation

2.3.1.1 SI7021_ADDRESS

```
#define SI7021_ADDRESS 0x40
```
device base address

2.3.1.2 SI7021_FIRMVERS_A

```
#define SI7021_FIRMVERS_A 0x84
```
Read Firmware Revision.

2.3.1.3 SI7021_FIRMVERS_B

```
#define SI7021_FIRMVERS_B 0xB8
```

2.3.1.4 SI7021_ID11

```
#define SI7021_ID11 0xFA
```
Read Electronic ID 1. Byte

2.3.1.5 SI7021_ID12

```
#define SI7021_ID12 0x0F
```

2.3.1.6 SI7021_ID21

```
#define SI7021_ID21 0xFC
```
Read Electronic ID 2. Byte.

2.3.1.7 SI7021_ID22

```
#define SI7021_ID22 0xC9
```

2.3.1.8 SI7021_MEASRH_HOLD

```
#define SI7021_MEASRH_HOLD 0xE5
```
Measure Relative Humidity, Hold Master Mode.

2.3.1.9 SI7021_MEASRH_NOHOLD

```
#define SI7021_MEASRH_NOHOLD 0xF5
```
Measure Relative Humidity, No Hold Master Mode.

2.3.1.10 SI7021_MEASTEMP_HOLD

```
#define SI7021_MEASTEMP_HOLD 0xE3
```

2.3.1.11 SI7021_MEASTEMP_NOHOLD

```
#define SI7021_MEASTEMP_NOHOLD 0xF3
```
Measure Temperature, No Hold Master Mode.

2.3.1.12 SI7021_READHEATER_REG

```
#define SI7021_READHEATER_REG 0x11
```
Read Heater Control Register.

2.3.1.13 SI7021_READPREVTEMP

```
#define SI7021_READPREVTEMP 0xE0
```
Read Temperature Value from Previous RH Measurement.

2.3.1.14 SI7021_READRHT_REG

```
#define SI7021_READRHT_REG 0xE7
```

2.3.1.15 SI7021_REG_HTRE_BIT

```
#define SI7021_REG_HTRE_BIT 0x02
```
Control Register Heater Bit.

2.3.1.16 SI7021_RESET

```
#define SI7021_RESET 0xFE
```

2.3.1.17 SI7021_REV_1

```
#define SI7021_REV_1 0xff
```
Sensor revision 1.

2.3.1.18 SI7021_REV_2

```
#define SI7021_REV_2 0x20
```
Sensor revision 2.

2.3.1.19 SI7021_WRITEHEATER_REG

```
#define SI7021_WRITEHEATER_REG 0x51
```
Write Heater Control Register.

2.3.1.20 SI7021_WRITERHT_REG

```
#define SI7021_WRITERHT_REG 0xE6
```
Write RH/T User Register 1.

2.3.2 Function Documentation

2.3.2.1 si7021_heat_level()

```
void si7021_heat_level (
            alt_u8 level )
```

si7021_heater_level();

Set the heater level.

Description: Sets the level of the internal heater At VDD 3.3 V current is level typical current draw 0000 3.09 mA 0001 9.18 mA 0010 15.24 mA .... 0100 27.39 mA ....
1000 51.69 mA 1111 94.20mA

Parameters

in	*level*	sees above
out	*void*	

2.3.2.2 si7021_heater()

```
void si7021_heater (
            alt_u8 set )
```

si7021_heater();

Activate the inbuilt heater.

Activate the sensor heater

Parameters

in	*set*	When value hold 1 heater gets activated, when 0 heater off.
out	*void*	

2.3.2.3 si7021_init()

```
void si7021_init ( )
```

void si7021_init();

Init function for sensor.

Init function for si7021 needs to be called first in order for the sensor to work.

Parameters

in	*none*	
out	*void*	

2.3.2.4 si7021_read_firmware()

```
void si7021_read_firmware ( )
```

si7021_read_firmware();

Read firmware from sensor.

Send 2 byte instruction data to the device and returns value 1 or 2 according to firmware revsion 0xFF version 1.0 0x20 version 2.0

Parameters

in	*none*	
out	*void*	

2.3.2.5 si7021_read_humidity()

```
float si7021_read_humidity ( )
```

si7021_read_humidity();

Read humidity from sensor.

Reads humidity from si7021 and in No Hold Master Mode. Retruns humidity in RH as a 16 bit integer

Parameters

in	*none*	
out	*humidity*	16 bit value

Returns

Functions returns the calculated humidity value in percent as a float

---

2.3.2.6 si7021_read_serial_number()

```
void si7021_read_serial_number ( )
```

si7021_read_serial_number();

Read serial number from sensor.

Function read the serial number. Device returns 64 bit value where of 0x00 or 0xFF are engineering samples 0x0D = Si7013 0x14 = Si7020 0x15 = Si7021 Two I2C commands are required to access the device memory and retrieve the complete serial number. see datasheet page 23

Parameters

in	*none*	
out	*void*	

2.3.2.7 si7021_read_temperature()

```
float si7021_read_temperature ( )
```

void si7021_read_temperature()

Read Temperature from sensor.

Reads temperature from si7021 and in No Hold Master Mode. Retruns it in celsius as a 16 bit integer.

Parameters

in	*none*	
out	*temperature*	16 bit value

Returns

Functions returns the calculated temperature value in celsius as a float.

2.3.2.8 si7021_read_user_reg()

```
alt_u8 si7021_read_user_reg (
            alt_u8 reg )
```

si7021_read_user_reg();

Read Si7021 user register.

Read from the si7021 user register

Parameters

in	*register*	Register to write
out	*value*	

Returns

Retruns the value of the user Register. After reset it holds 0011_1010

2.3.2.9 si7021_reset()

```
void si7021_reset ( )
```

void si7021_reset();
si7021.h
Reset function for si7021

Parameters

in	*none*	
out	*void*	

2.3.2.10 si7021_write_user_reg()

```
void si7021_write_user_reg (
            alt_u8 reg,
            alt_u8 value )
```

si7021_write_user_reg();
Write toSi7021 user register.
Write to the si7021 user register

Parameters

in	*register*	Register to write
out	*value*	Register value to write
out	*void*	

2.3.3 Variable Documentation

2.3.3.1 firmware_revsion

```
alt_u8 firmware_revsion = 0
```

firmware revision number

2.3.3.2 model

```
alt_u8 model = 0
```

sensor model

2.3.3.3 rxBuffer

```
alt_u8 rxBuffer[3]
```

global buffer to hold rx

2.3.3.4 txBuffer

```
alt_u8 txBuffer[3]
```

global buffer to hold tx

2.4 C:/Users/Dominik/Desktop/doc/si7021.h File Reference

```
#include "i2c.h"
```

Functions

- void si7021_reset ()

  si7021.h
- void si7021_init ()

  Init function for sensor.
- void si7021_read_firmware ()

  Read firmware from sensor.
- void si7021_read_serial_number ()

  Read serial number from sensor.
- float si7021_read_temperature ()

  Read Temperature from sensor.
- float si7021_read_humidity ()

  Read humidity from sensor.
- alt_u8 si7021_read_user_reg (alt_u8 reg)

  Read Si7021 user register.
- void si7021_write_user_reg (alt_u8 reg, alt_u8 value)

  Write toSi7021 user register.
- void si7021_heater (alt_u8 set)

  Activate the inbuilt heater.
- void si7021_heat_level (alt_u8 level)

  Set the heater level.

2.4.1 Function Documentation

2.4.1.1 si7021_heat_level()

```
void si7021_heat_level (
            alt_u8 level )
```

Set the heater level.

Set the heater level.

Description: Sets the level of the internal heater At VDD 3.3 V current is level typical current draw 0000 3.09 mA 0001 9.18 mA 0010 15.24 mA .... 0100 27.39 mA .... 1000 51.69 mA 1111 94.20mA

Parameters

in	*level*	sees above
out	*void*	

2.4.1.2 si7021_heater()

```
void si7021_heater (
            alt_u8 set )
```

Activate the inbuilt heater.

Activate the inbuilt heater.

Activate the sensor heater

Parameters

in	*set*	When value hold 1 heater gets activated, when 0 heater off.
out	*void*	

2.4.1.3 si7021_init()

```
void si7021_init ( )
```

Init function for sensor.

Init function for sensor.

Init function for si7021 needs to be called first in order for the sensor to work.

Parameters

in	*none*	
out	*void*	

2.4.1.4 si7021_read_firmware()

```
void si7021_read_firmware ( )
```

Read firmware from sensor.

Read firmware from sensor.

Send 2 byte instruction data to the device and returns value 1 or 2 according to firmware revsion 0xFF version 1.0 0x20 version 2.0

Parameters

in	*none*	
out	*void*	

2.4.1.5 si7021_read_humidity()

```
float si7021_read_humidity ( )
```

Read humidity from sensor.

Read humidity from sensor.

Reads humidity from si7021 and in No Hold Master Mode. Retruns humidity in RH as a 16 bit integer

Parameters

in	*none*	
out	*humidity*	16 bit value

Returns

Functions returns the calculated humidity value in percent as a float

---

2.4.1.6 si7021_read_serial_number()

```
void si7021_read_serial_number ( )
```

Read serial number from sensor.

Read serial number from sensor.

Function read the serial number. Device returns 64 bit value where of 0x00 or 0xFF are engineering samples 0x0D = Si7013 0x14 = Si7020 0x15 = Si7021 Two I2C commands are required to access the device memory and retrieve the complete serial number. see datasheet page 23

Parameters

in	*none*	
out	*void*	

2.4.1.7 si7021_read_temperature()

```
float si7021_read_temperature ( )
```

Read Temperature from sensor.

Read Temperature from sensor.

Reads temperature from si7021 and in No Hold Master Mode. Retruns it in celsius as a 16 bit integer.

Parameters

in	*none*	
out	*temperature*	16 bit value

Returns

Functions returns the calculated temperature value in celsius as a float.

2.4.1.8 si7021_read_user_reg()

```
alt_u8 si7021_read_user_reg (
            alt_u8 reg )
```

Read Si7021 user register.

Read Si7021 user register.

Read from the si7021 user register

Parameters

in	*register*	Register to write
out	*value*	

Returns

Retruns the value of the user Register. After reset it holds 0011_1010

2.4.1.9 si7021_reset()

```
void si7021_reset ( )
```

si7021.h

Created on: 07.04.2021 Author: Dominik Socher Reset the sensor

si7021.h

Reset function for si7021

Parameters

in	*none*	
out	*void*	

2.4.1.10 si7021_write_user_reg()

```
void si7021_write_user_reg (
```

```
            alt_u8 reg,
            alt_u8 value )
```

Write toSi7021 user register.

Write toSi7021 user register.

Write to the si7021 user register

Parameters

in	*register*	Register to write
out	*value*	Register value to write
out	*void*	

Index

C:/Users/Dominik/Desktop/doc/i2c.c, 3
C:/Users/Dominik/Desktop/doc/i2c.h, 5
C:/Users/Dominik/Desktop/doc/si7021.c, 6
C:/Users/Dominik/Desktop/doc/si7021.h, 13

firmware_revsion
    si7021.c, 12

i2c.c
    i2c_dev, 4
    i2c_open, 3
    i2c_rx, 4
    i2c_slave_address, 4
    i2c_status, 4
    i2c_tx, 4
i2c.h
    i2c_open, 5
    i2c_rx, 5
    i2c_slave_address, 5
    i2c_tx, 6
i2c_dev
    i2c.c, 4
i2c_open
    i2c.c, 3
    i2c.h, 5
i2c_rx
    i2c.c, 4
    i2c.h, 5
i2c_slave_address
    i2c.c, 4
    i2c.h, 5
i2c_status
    i2c.c, 4
i2c_tx
    i2c.c, 4
    i2c.h, 6

model
    si7021.c, 12

rxBuffer
    si7021.c, 12

si7021.c
    firmware_revsion, 12
    model, 12
    rxBuffer, 12
    SI7021_ADDRESS, 7
    SI7021_FIRMVERS_A, 8
    SI7021_FIRMVERS_B, 8
    si7021_heat_level, 9
    si7021_heater, 10
    SI7021_ID11, 8
    SI7021_ID12, 8
    SI7021_ID21, 8
    SI7021_ID22, 8
    si7021_init, 10
    SI7021_MEASRH_HOLD, 8
    SI7021_MEASRH_NOHOLD, 8
    SI7021_MEASTEMP_HOLD, 8
    SI7021_MEASTEMP_NOHOLD, 8
    si7021_read_firmware, 10
    si7021_read_humidity, 10
    si7021_read_serial_number, 11
    si7021_read_temperature, 11
    si7021_read_user_reg, 11
    SI7021_READHEATER_REG, 9
    SI7021_READPREVTEMP, 9
    SI7021_READRHT_REG, 9
    SI7021_REG_HTRE_BIT, 9
    SI7021_RESET, 9
    si7021_reset, 12
    SI7021_REV_1, 9
    SI7021_REV_2, 9
    si7021_write_user_reg, 12
    SI7021_WRITEHEATER_REG, 9
    SI7021_WRITERHT_REG, 9
    txBuffer, 12
si7021.h
    si7021_heat_level, 13
    si7021_heater, 13
    si7021_init, 14
    si7021_read_firmware, 14
    si7021_read_humidity, 14
    si7021_read_serial_number, 14
    si7021_read_temperature, 15
    si7021_read_user_reg, 15
    si7021_reset, 15
    si7021_write_user_reg, 15
SI7021_ADDRESS
    si7021.c, 7
SI7021_FIRMVERS_A
    si7021.c, 8
SI7021_FIRMVERS_B
    si7021.c, 8
si7021_heat_level
    si7021.c, 9
    si7021.h, 13
si7021_heater
    si7021.c, 10

si7021.h, 13
SI7021_ID11
si7021.c, 8
SI7021_ID12
si7021.c, 8
SI7021_ID21
si7021.c, 8
SI7021_ID22
si7021.c, 8
si7021_init
si7021.c, 10
si7021.h, 14
SI7021_MEASRH_HOLD
si7021.c, 8
SI7021_MEASRH_NOHOLD
si7021.c, 8
SI7021_MEASTEMP_HOLD
si7021.c, 8
SI7021_MEASTEMP_NOHOLD
si7021.c, 8
si7021_read_firmware
si7021.c, 10
si7021.h, 14
si7021_read_humidity
si7021.c, 10
si7021.h, 14
si7021_read_serial_number
si7021.c, 11
si7021.h, 14
si7021_read_temperature
si7021.c, 11
si7021.h, 15
si7021_read_user_reg
si7021.c, 11
si7021.h, 15
SI7021_READHEATER_REG
si7021.c, 9
SI7021_READPREVTEMP
si7021.c, 9
SI7021_READRHT_REG
si7021.c, 9
SI7021_REG_HTRE_BIT
si7021.c, 9
SI7021_RESET
si7021.c, 9
si7021_reset
si7021.c, 12
si7021.h, 15
SI7021_REV_1
si7021.c, 9
SI7021_REV_2
si7021.c, 9
si7021_write_user_reg
si7021.c, 12
si7021.h, 15
SI7021_WRITEHEATER_REG
si7021.c, 9
SI7021_WRITERHT_REG
si7021.c, 9

txBuffer
si7021.c, 12