WireUtility Library

Generated by Doxygen 1.8.14

# **Contents**

1	File	File Index			
	1.1	File Lis	t	1	
2	File	File Documentation			
	2.1	WireUt	ility.cpp File Reference	1	
		2.1.1	Detailed Description	2	
		2.1.2	Function Documentation	2	
	2.2	WireUt	ility.h File Reference	6	
		2.2.1	Detailed Description	7	
		2.2.2	Function Documentation	7	
Inc	lex			11	
1	File	e Index	<b>K</b>		
1.1	File	e List			
He	re is a	list of a	ull files with brief descriptions:		
	Wire	Utility.c	pp	1	
	Wire	Utility.h		6	
2	File	e Docu	mentation		
2.1	WireUtility.cpp File Reference				
Fui	nctions	i			
	<ul> <li>int</li> <li>int</li> <li>vo</li> <li>uir</li> <li>uir</li> <li>vo</li> <li>vo</li> <li>vo</li> <li>int</li> <li>int</li> <li>int</li> </ul>	scanNe WireTes scanNe id wireR nt8_t wir nt16_t w nt32_t w id wireW id wireW readRe readMe	L_call_init_address (TwoWire &ScanWire) ext10bits (HardwareSerial &MySerial, TwoWire &ScanWire) est (TwoWire &ScanWire, int address) ext (HardwareSerial &MySerial, TwoWire &ScanWire) ext (HardwareSerial &MySerial, TwoWire &ScanWire) ext (HardwareSerial &MySerial, TwoWire &ScanWire) ext (HardwareSerial &MyWire, uint8_t_i2caddr, uint8_t addr) ext (TwoWire &MyWire, uint8_t_i2caddr, uint8_t addr) ext (TwoWire &MyWire, uint8_t_i2caddr, uint8_t addr) ext (TwoWire &MyWire, uint8_t_i2caddr, uint8_t addr, uint8_t d) ext (TwoWire &MyWire, uint8_t_i2caddr, uint8_t addr, uint16_t d) ext (TwoWire &MyWire, uint8_t_i2caddr, uint8_t addr, uint32_t d) ext (HardwareSerial &MySerial, TwoWire &ScanWire, int adddev, int addreg) ext (HardwareSerial &MySerial, TwoWire &ScanWire, int adddev, int addreg) ext (HardwareSerial &MySerial, TwoWire &ScanWire, int adddev, int addreg) ext (HardwareSerial &MySerial, TwoWire &ScanWire, int adddev, int addreg) ext (HardwareSerial &MySerial, TwoWire &ScanWire, int adddev, int addreg) ext (HardwareSerial &MySerial, TwoWire &ScanWire, int adddev, int addreg) ext (HardwareSerial &MySerial, TwoWire &ScanWire, int adddev, int addreg) ext (HardwareSerial &MySerial, TwoWire &ScanWire, int adddev, int addreg)		

#### 2.1.1 Detailed Description

### description

This library offers utility function for I2c protocol,like:

- · scan of device on the line.
- · Wiretest to check if the line is alive.
- any read/read of register inside component, evenif no standard exist, several component use the same way to do the thing.
- · dump of memory map.
- · Reset All Devices on the wire

Any thing usefull to do debug on I2c protocol with wire object.

### **Author**

inspired from: i2c scanner

Version 1 This program (or code that looks like it) can be found in many places. For example on the Arduino.cc forum. The original author is not known. Version 2, Juni 2012, Using Arduino 1.0.1 Adapted to be as simple as possible by Arduino.cc user Krodal Version 3, Feb 26 2013 V3 by louarnold Version 4, March 3, 2013, Using Arduino 1.0.3 by Arduino.cc user Krodal. Changes by louarnold removed. Scanning WireUtility\_addresses changed from 0...127 to 1...119, according to the i2c scanner by Nick Gammon http://www.gammon.com.au/forum/?id=10896 Version 5, March 28, 2013 As version 4, but WireUtility\_address scans now to 127. A sensor seems to use Wire Utility\_address 120.

version 6, based on previous job create library WireUtility (c) zoubata This sketch tests the standard 7-bit Wire Utility\_addresses Devices with higher bit WireUtility\_address might not be seen properly.

```
https://www.i2c-bus.org/WireUtility_addressing/
```

#include <Wire.h> #include <WireUtility.h>

int WireUtility\_address; int scan(HardwareSerial &MySerial,TwoWire &ScanWire) { ScanWire.begin(123); Scan ↔ Wire.begin();

WireUtility address = 0x1;// scan special address return scanNext(MySerial,ScanWire); }

/\*\* Adresse d'appel général : 0000 0000

Après l'émission d'un appel général, les circuits ayant la capacité de traiter ce genre de demande d'appel émettent un acquitement.

Le deuxième octet permet de définir le contenu de l'appel :

0000 0100 Les circuits définissant leur adresse de façon matérielle réinitialisent leur adresse esclave. Cela ne réinitialise pas les circuits.

### 2.1.2 Function Documentation

## 2.1.2.1 dump()

dump a 8 bit address data memory map from addreg up to (addreg+size)

Definition at line 340 of file WireUtility.cpp.

### 2.1.2.2 general\_call\_init\_address()

Definition at line 70 of file WireUtility.cpp.

### 2.1.2.3 readMem2()

Definition at line 282 of file WireUtility.cpp.

### 2.1.2.4 readMem3()

Definition at line 311 of file WireUtility.cpp.

## 2.1.2.5 readRegister()

read a 8 bit address data(addreg)

Definition at line 255 of file WireUtility.cpp.

### 2.1.2.6 scanNext()

scan next 7 bit device address. if a 10bit device is found it can scanNext10bits

Definition at line 127 of file WireUtility.cpp.

### 2.1.2.7 scanNext10bits()

scan next 10 bit device address.

Definition at line 78 of file WireUtility.cpp.

## 2.1.2.8 wireRead16()

read 16 bits register at address addr on device from i2C called MyWire at address \_i2caddr

Definition at line 205 of file WireUtility.cpp.

### 2.1.2.9 wireRead32()

read 32 bits register at address addr on device from i2C called MyWire at address \_i2caddr

Definition at line 216 of file WireUtility.cpp.

## 2.1.2.10 wireRead8()

read 8 bits register at address addr on device from i2C called MyWire at address \_i2caddr

Definition at line 194 of file WireUtility.cpp.

## 2.1.2.11 wireResetAllDevices()

```
void wireResetAllDevices (
          TwoWire & MyWire )
```

perform a reset of all device on the line that manage Software Reset Call

The Software Reset address (SWRST Call) must be used with R/W = logic 0. so do a frame with address : 0x0 and write 0b00000110

Definition at line 186 of file WireUtility.cpp.

## 2.1.2.12 WireTest()

test if a device is present and acknoledging.

Definition at line 115 of file WireUtility.cpp.

## 2.1.2.13 wireWrite16()

write 16 bits register at address addr on device from i2C called MyWire at address \_i2caddr

Definition at line 235 of file WireUtility.cpp.

## 2.1.2.14 wireWrite32()

```
void wireWrite32 (
          TwoWire & MyWire,
           uint8_t _i2caddr,
          uint8_t addr,
          uint32_t d)
```

write 32 bits register at address addr on device from i2C called MyWire at address \_i2caddr

Definition at line 244 of file WireUtility.cpp.

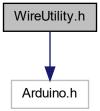
## 2.1.2.15 wireWrite8()

write 8 bits register at address addr on device from i2C called MyWire at address \_i2caddr

Definition at line 227 of file WireUtility.cpp.

# 2.2 WireUtility.h File Reference

```
#include "Arduino.h"
Include dependency graph for WireUtility.h:
```



#### **Functions**

- int WireTest (TwoWire &ScanWire, int address)
- void dump (HardwareSerial &MySerial, TwoWire &ScanWire, int adddev, int addreg, int size)
- int readRegister (HardwareSerial &MySerial, TwoWire &ScanWire, int adddev, int addreg)
- int readMem2 (TwoWire &ScanWire, int adddev, int addreg)
- int readMem3 (TwoWire &ScanWire, int adddev, int addreg)
- int scan (HardwareSerial &MySerial, TwoWire &ScanWire)
- int scanNext (HardwareSerial &MySerial, TwoWire &ScanWire)
- int scanNext10bits (HardwareSerial &MySerial, TwoWire &ScanWire)
- void wireResetAllDevices (TwoWire &MyWire)
- uint8\_t wireRead8 (TwoWire &MyWire, uint8\_t \_i2caddr, uint8\_t addr)
- uint16 t wireRead16 (TwoWire &MyWire, uint8 t i2caddr, uint8 t addr)
- uint32\_t wireRead32 (TwoWire &MyWire, uint8\_t \_i2caddr, uint8\_t addr)
- void wireWrite8 (TwoWire &MyWire, uint8\_t \_i2caddr, uint8\_t addr, uint8\_t d)
- void wireWrite16 (TwoWire &MyWire, uint8\_t \_i2caddr, uint8\_t addr, uint16\_t d)
- void wireWrite32 (TwoWire &MyWire, uint8\_t \_i2caddr, uint8\_t addr, uint32\_t d)

### 2.2.1 Detailed Description

This file content function to help to debug wire interface on component.

#### 2.2.2 Function Documentation

#### 2.2.2.1 dump()

dump a 8 bit address data memory map from addreg up to (addreg+size)

Definition at line 340 of file WireUtility.cpp.

#### 2.2.2.2 readMem2()

read a 16 bit address data(addreg)

## 2.2.2.3 readMem3()

```
int readMem3 (
                 TwoWire & ScanWire,
                 int adddev,
                 int addreg )
```

read a 24 bit address data(addreg)

### 2.2.2.4 readRegister()

read a 8 bit address data(addreg)

Definition at line 255 of file WireUtility.cpp.

## 2.2.2.5 scan()

scan a 7 bit device address. and stop after 1st find, use scanNext to get next.

## 2.2.2.6 scanNext()

scan next 7 bit device address. if a 10bit device is found it can scanNext10bits

Definition at line 127 of file WireUtility.cpp.

## 2.2.2.7 scanNext10bits()

scan next 10 bit device address.

Definition at line 78 of file WireUtility.cpp.

## 2.2.2.8 wireRead16()

read 16 bits register at address addr on device from i2C called MyWire at address \_i2caddr

Definition at line 205 of file WireUtility.cpp.

#### 2.2.2.9 wireRead32()

read 32 bits register at address addr on device from i2C called MyWire at address \_i2caddr

Definition at line 216 of file WireUtility.cpp.

## 2.2.2.10 wireRead8()

read 8 bits register at address addr on device from i2C called MyWire at address \_i2caddr

Definition at line 194 of file WireUtility.cpp.

## 2.2.2.11 wireResetAllDevices()

```
void wireResetAllDevices (
          TwoWire & MyWire )
```

perform a reset of all device on the line that manage Software Reset Call

The Software Reset address (SWRST Call) must be used with R/W = logic 0. so do a frame with address : 0x0 and write 0b00000110

Definition at line 186 of file WireUtility.cpp.

## 2.2.2.12 WireTest()

test if a device is present and acknoledging.

Definition at line 115 of file WireUtility.cpp.

## 2.2.2.13 wireWrite16()

```
void wireWrite16 (
          TwoWire & MyWire,
           uint8_t _i2caddr,
          uint8_t addr,
          uint16_t d )
```

write 16 bits register at address addr on device from i2C called MyWire at address \_i2caddr

Definition at line 235 of file WireUtility.cpp.

## 2.2.2.14 wireWrite32()

write 32 bits register at address addr on device from i2C called MyWire at address \_i2caddr

Definition at line 244 of file WireUtility.cpp.

## 2.2.2.15 wireWrite8()

write 8 bits register at address addr on device from i2C called MyWire at address \_i2caddr

Definition at line 227 of file WireUtility.cpp.

# Index

dum	wire\MriteQ C
dump	wireWrite8, 6
WireUtility.cpp, 2	WireUtility.h, 6
WireUtility.h, 7	dump, 7
supposed and init address.	readMem2, 7
general_call_init_address	readMem3, 7
WireUtility.cpp, 3	readRegister, 8
INA O	scan, 8
readMem2	scanNext, 8
WireUtility.cpp, 3	scanNext10bits, 8
WireUtility.h, 7	wireRead16, 8
readMem3	wireRead32, 9
WireUtility.cpp, 3	wireRead8, 9
WireUtility.h, 7	wireResetAllDevices, 9
readRegister	WireTest, 9
WireUtility.cpp, 3	wireWrite16, 10
WireUtility.h, 8	wireWrite32, 10
	wireWrite8, 10
scan	wireWrite16
WireUtility.h, 8	WireUtility.cpp, 5
scanNext	WireUtility.h, 10
WireUtility.cpp, 3	wireWrite32
WireUtility.h, 8	WireUtility.cpp, 5
scanNext10bits	WireUtility.h, 10
WireUtility.cpp, 4	wireWrite8
WireUtility.h, 8	WireUtility.cpp, 6
• •	WireUtility.h, 10
wireRead16	wireduity.n, 10
WireUtility.cpp, 4	
WireUtility.h, 8	
wireRead32	
WireUtility.cpp, 4	
WireUtility.h, 9	
wireRead8	
WireUtility.cpp, 4	
WireUtility.h, 9	
wireResetAllDevices	
WireUtility.cpp, 5	
WireUtility.h, 9	
WireTest	
WireUtility.cpp, 5	
WireUtility.h, 9	
WireUtility.cpp, 1	
dump, 2	
general_call_init_address, 3	
readMem2, 3	
readMem3, 3	
readRegister, 3	
scanNext, 3	
scanNext10bits, 4	
wireRead16, 4	
wireRead32, 4	
wireRead8, 4	
wireResetAllDevices, 5	
WireTest, 5	
wireWrite16, 5	
wireWrite32, 5	