MLX90614 Device Driver

1.0

Generated by Doxygen 1.8.6

Mon Feb 29 2016 09:37:40

Contents

1 Class Index 1

1.1 Class List 1

2 File Index 3

2.1 File List 3

3 Class Documentation 5

3.1 CRC8 Class Reference 5

3.1.1 Detailed Description 5

3.1.2 Constructor & Destructor Documentation 5

3.1.2.1 CRC8 5

3.1.3 Member Function Documentation 5

3.1.3.1 crc8 6

3.1.3.2 crc8 6

3.1.3.3 crc8Start 6

3.2 defaultEEPromData Struct Reference 6

3.2.1 Detailed Description 6

3.3 MLX90614 Class Reference 7

3.3.1 Detailed Description 8

3.3.2 Member Enumeration Documentation 8

3.3.2.1 tempSrc_t 8

3.3.2.2 tempUnit_t 8

3.3.3 Constructor & Destructor Documentation 9

3.3.3.1 MLX90614 9

3.3.4 Member Function Documentation 10

3.3.4.1 convCtoF 10

3.3.4.2 convKtoC 10

3.3.4.3 getAddr 10

3.3.4.4 getCRC8 10

3.3.4.5 getEmissivity 11

3.3.4.6 getFIRcoeff 11

3.3.4.7 getIIRcoeff 11

3.3.4.8 getPEC 11
3.3.4.9 getRwError 12
3.3.4.10 read16 12
3.3.4.11 readEEProm 12
3.3.4.12 readID 12
3.3.4.13 readTemp 12
3.3.4.14 setAddr 13
3.3.4.15 setEmissivity 13
3.3.4.16 setFIRcoeff 13
3.3.4.17 setIIRcoeff 14
3.3.4.18 write16 14
3.3.4.19 writeEEProm 14
3.3.5 Member Data Documentation 15
3.3.5.1 _addr 15
3.3.5.2 _crc8 15
3.3.5.3 _pec 15
3.3.5.4 _rwError 15
3.3.5.5 busAddr 15
3.3.5.6 crc8 15
3.3.5.7 pec 15
3.3.5.8 rwError 15

4 File Documentation 17

4.1 Crc8.cpp File Reference 17
4.1.1 Detailed Description 17
4.2 Crc8.h File Reference 18
4.2.1 Detailed Description 18
4.2.2 Macro Definition Documentation 19
4.2.2.1 CRC8_DEFAULTPOLY 19
4.3 examples/mlxtest/MelexisTest.ino File Reference 19
4.3.1 Detailed Description 20
4.3.2 Function Documentation 20
4.3.2.1 floatToStr 20
4.3.2.2 printCRC 21
4.3.2.3 printErrStr 21
4.3.2.4 printlnTemp 21
4.3.2.5 setEEPromDefaults 21
4.3.3 Variable Documentation 21
4.3.3.1 eDat 21
4.4 MLX90614.cpp File Reference 22

4.4.1 Detailed Description 22
4.5 MLX90614.h File Reference 23
4.5.1 Detailed Description 24
4.5.2 Macro Definition Documentation 24
4.5.2.1 MLX90614_ADDR 24
4.5.2.2 MLX90614_BROADCASTADDR 25
4.5.2.3 MLX90614_CONFIG 25
4.5.2.4 MLX90614_CRC8POLY 25
4.5.2.5 MLX90614_DATATOOLONG 25
4.5.2.6 MLX90614_EE_DEAD 25
4.5.2.7 MLX90614_EEBUSY 25
4.5.2.8 MLX90614_EECORRUPT 25
4.5.2.9 MLX90614_EMISS 25
4.5.2.10 MLX90614_I2CDEFAULTADDR 25
4.5.2.11 MLX90614_ID1 26
4.5.2.12 MLX90614_ID2 26
4.5.2.13 MLX90614_ID3 26
4.5.2.14 MLX90614_ID4 26
4.5.2.15 MLX90614_INIT 26
4.5.2.16 MLX90614_INVALIDATA 26
4.5.2.17 MLX90614_PWMCTRL 26
4.5.2.18 MLX90614_RAWIR1 26
4.5.2.19 MLX90614_RAWIR2 26
4.5.2.20 MLX90614_RFLAGCMD 27
4.5.2.21 MLX90614_RFLGERR 27
4.5.2.22 MLX90614_RXCRC 27
4.5.2.23 MLX90614_SUCCESS 27
4.5.2.24 MLX90614_TA 27
4.5.2.25 MLX90614_TARANGE 27
4.5.2.26 MLX90614_TOBJ1 27
4.5.2.27 MLX90614_TOBJ2 27
4.5.2.28 MLX90614_TOMAX 27
4.5.2.29 MLX90614_TOMIN 28
4.5.2.30 MLX90614_TXADDRNACK 28
4.5.2.31 MLX90614_TXDATANACK 28
4.5.2.32 MLX90614_TXOTHER 28
4.5.2.33 MLX90614_XDLY 28

Index **29**

Chapter 1

Class Index

1.1 Class List

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

CRC8 5

defaultEEPromData

EEPROM memory contents factory default values 6

MLX90614 7

Chapter 2

File Index

2.1 File List

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

Crc8.cpp
8 bit CRC helper/utility class - CPP Source file 17

Crc8.h
8 bit CRC helper/utility class - CPP Header file 18

MLX90614.cpp
Melexis MLX90614 Family Device Driver Library - CPP Source file 22

MLX90614.h
Melexis MLX90614 Family Device Driver Library - CPP Header file 23

examples/mlxtest/MelexisTest.ino
Melexis MCX90614BAA Test Program - Sensor test implementation 19

Chapter 3

Class Documentation

3.1 CRC8 Class Reference

Public Member Functions

- CRC8 (uint8_t polynomial=CRC8_DEFAULTPOLY)

  CRC8 class constructor.
- uint8_t crc8 (void)

  Return the current value of the CRC.
- uint8_t crc8 (uint8_t data)

  Update the current value of the CRC.
- void crc8Start (uint8_t poly)

  Initialize the CRC8 object.

Private Attributes

- uint8_t **_crc**
- uint8_t **_poly**

3.1.1 Detailed Description

Definition at line 42 of file Crc8.h.

3.1.2 Constructor & Destructor Documentation

3.1.2.1 CRC8::CRC8 (uint8_t *poly* = CRC8_DEFAULTPOLY)

CRC8 class constructor.

Parameters

`in`	*poly*	8 bit CRC polynomial to use.

Definition at line 41 of file Crc8.cpp.

3.1.3 Member Function Documentation

3.1.3.1 uint8_t CRC8::crc8 (void)

Return the current value of the CRC.

Returns

8 bit CRC current value.

Definition at line 47 of file Crc8.cpp.

3.1.3.2 uint8_t CRC8::crc8 (uint8_t *data*)

Update the current value of the CRC.

Parameters

in	*data*	New 8 bit data to be added to the CRC.

Returns

8 bit CRC current value.

Definition at line 54 of file Crc8.cpp.

3.1.3.3 void CRC8::crc8Start (uint8_t *poly*)

Initialize the CRC8 object.

Parameters

in	*poly*	8 bit CRC polynomial to use.

Definition at line 65 of file Crc8.cpp.

The documentation for this class was generated from the following files:

- Crc8.h
- Crc8.cpp

3.2 defaultEEPromData Struct Reference

EEPROM memory contents factory default values.

Public Attributes

- uint8_t **address**
- uint16_t **data**

3.2.1 Detailed Description

EEPROM memory contents factory default values.

Definition at line 170 of file MelexisTest.ino.

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

- examples/mlxtest/MelexisTest.ino

3.3 MLX90614 Class Reference

Public Types

- enum tempUnit_t { MLX90614_TK, MLX90614_TC, MLX90614_TF }
- enum tempSrc_t { MLX90614_SRCA, MLX90614_SRCO1, MLX90614_SRCO2 }

Public Member Functions

- MLX90614 (uint8_t addr=MLX90614_I2CDEFAULTADDR)

  MLX90614 Device class constructor.
- boolean begin ()

  Initialize the device and the i2c interface.
- uint64_t readID (void)

  Retrieve the chip ID bytes.
- uint8_t getIIRcoeff (void)

  Get the coefficients of the IIR digital filter.
- uint8_t getFIRcoeff (void)

  Get the coefficients of the FIR digital filter.
- float getEmissivity (void)

  Get the emissivity of the object.
- void setIIRcoeff (uint8_t iir)

  Set the coefficients of the IIR digital filter.
- void setFIRcoeff (uint8_t fir)

  Set the coefficients of the FIR digital filter.
- void setEmissivity (float emiss)

  Set the emissivity of the object.
- uint16_t readEEProm (uint8_t addr)

  Return a 16 bit value read from EEPROM.
- void writeEEProm (uint8_t reg, uint16_t data)

  Write a 16 bit value to EEPROM after first clearing the memory.
- double readTemp (tempSrc_t tsrc, tempUnit_t tunit)

  Return a temperature from the specified source in specified units.
- double convKtoC (double degK)

  Convert temperature in degrees K to degrees C.
- double convCtoF (double degC)

  Convert temperature in degrees C to degrees F.

Public Attributes

- Property< uint8_t, MLX90614 > busAddr
- Property< uint8_t, MLX90614 > rwError
- Property< uint8_t, MLX90614 > crc8
- Property< uint8_t, MLX90614 > pec

Private Member Functions

- float **readTemp** (uint8_t reg)
- uint16_t read16 (uint8_t cmd)

  Return a 16 bit value read from RAM or EEPROM.
- void write16 (uint8_t cmd, uint16_t data)

  Write a 16 bit value to memory.
- uint8_t getRwError (void)
- uint8_t getCRC8 (void)
- uint8_t getPEC (void)
- uint8_t getAddr (void)

  Return the device SMBus address.
- void setAddr (uint8_t v)

  Set device SMBus address.

Private Attributes

- uint8_t _addr
- uint8_t _rwError
- uint8_t _crc8
- uint8_t _pec

3.3.1 Detailed Description

Definition at line 107 of file MLX90614.h.

3.3.2 Member Enumeration Documentation

3.3.2.1 enum MLX90614::tempSrc_t

Enumerations for temperature measurement source.

Enumerator

MLX90614_SRCA Chip (ambient) sensor

MLX90614_SRCO1 IR source #1

MLX90614_SRCO2 IR source #2

Definition at line 136 of file MLX90614.h.

3.3.2.2 enum MLX90614::tempUnit_t

Enumerations for temperature units.

Enumerator

MLX90614_TK degrees Kelvin

MLX90614_TC degrees Centigrade

MLX90614_TF degrees Fahrenheit

Definition at line 131 of file MLX90614.h.

3.3.3 Constructor & Destructor Documentation

3.3.3.1 MLX90614::MLX90614 (uint8_t *i2caddr* = MLX90614_I2CDEFAULTADDR)

MLX90614 Device class constructor.

Parameters

`in`	*i2caddr*	Device address (default: published value).

Definition at line 49 of file MLX90614.cpp.

3.3.4 Member Function Documentation

3.3.4.1 double MLX90614::convCtoF (double *degC*)

Convert temperature in degrees C to degrees F.

Parameters

`in`	*degC*	Temperature in degrees Centigrade.

Returns

Temperature in degrees Fahrenheit.

Definition at line 389 of file MLX90614.cpp.

3.3.4.2 double MLX90614::convKtoC (double *degK*)

Convert temperature in degrees K to degrees C.

Parameters

`in`	*degK*	Temperature in degrees Kelvin.

Returns

Temperature in degrees Centigrade.

Definition at line 382 of file MLX90614.cpp.

3.3.4.3 uint8_t MLX90614::getAddr (void) `[private]`

Return the device SMBus address.

SMB bus address getter

Remarks

- Must be only device on the bus.
- Sets the library to use the new found address.

Returns

Device address.

Definition at line 250 of file MLX90614.cpp.

3.3.4.4 uint8_t MLX90614::getCRC8 (void) `[inline]`, `[private]`

8 bit CRC getter

Definition at line 156 of file MLX90614.h.

3.3.4.5 float MLX90614::getEmissivity (void)

Get the emissivity of the object.

Emissivity getter

Remarks

> The emissivity is stored as a 16 bit integer defined by the following:
> `emissivity = dec2hex[round(65535 x emiss)]`

Returns

> Physical emissivity value in range 0.1 ...1.0

Definition at line 122 of file MLX90614.cpp.

3.3.4.6 uint8_t MLX90614::getFIRcoeff (void)

Get the coefficients of the FIR digital filter.

IIR coefficient getter

Remarks

> The FIR digital filter coefficient N is bits 10:8 of ConfigRegister1
> The value of N is set as follows: `N = 2 ^ (csb + 3)`
> The manufacturer does not recommend `N < 128`

Parameters

in	*csb*	See page 12 of datasheet. Range 0...7

Definition at line 211 of file MLX90614.cpp.

3.3.4.7 uint8_t MLX90614::getIIRcoeff (void)

Get the coefficients of the IIR digital filter.

IIR coefficient getter

Remarks

> The IIR digital filter coefficients are set by the LS 3 bits of ConfigRegister1

Returns

> Filter coefficient table index. Range 0...7

Definition at line 168 of file MLX90614.cpp.

3.3.4.8 uint8_t MLX90614::getPEC (void) `[inline], [private]`

PEC getter

Definition at line 157 of file MLX90614.h.

3.3.4.9 uint8_t MLX90614::getRwError (void) [inline], [private]

R/W error flags getter

Definition at line 155 of file MLX90614.h.

3.3.4.10 uint16_t MLX90614::read16 (uint8_t *cmd*) [private]

Return a 16 bit value read from RAM or EEPROM.

Parameters

in	*cmd*	Command to send (register to read from).

Returns

Value read from memory.

Definition at line 272 of file MLX90614.cpp.

3.3.4.11 uint16_t MLX90614::readEEProm (uint8_t *addr*)

Return a 16 bit value read from EEPROM.

Parameters

in	*addr*	Register address to read from.

Returns

Value read from EEPROM.

Definition at line 341 of file MLX90614.cpp.

3.3.4.12 uint64_t MLX90614::readID (void)

Retrieve the chip ID bytes.

Chip ID getter

Returns

Chip ID as a 64 bit word.

Definition at line 395 of file MLX90614.cpp.

3.3.4.13 double MLX90614::readTemp (tempSrc_t *tsrc,* tempUnit_t *tunit*)

Return a temperature from the specified source in specified units.

Remarks

- Temperature is stored in ram as a 16 bit absolute value to a resolution of 0.02K
- Linearized sensor die temperature is available as Ta (ambient).
- One or two object temperatures are linearized to the range -38.2C...125C

Parameters

in	*tsrc*	Internal temperature source to read.
in	*tunit*	Temperature units to convert raw data to.

Returns

Temperature.

Definition at line 86 of file MLX90614.cpp.

3.3.4.14 void MLX90614::setAddr (uint8_t *addr*) [private]

Set device SMBus address.

SMB bus address setter

Remarks

- Must be only device on the bus.
- Must power cycle the device after changing address.

Parameters

in	*a*	New device address. Range 1...127

Definition at line 229 of file MLX90614.cpp.

3.3.4.15 void MLX90614::setEmissivity (float *emiss* = 1.0)

Set the emissivity of the object.

Emissivity setter

Remarks

The emissivity is stored as a 16 bit integer defined by the following:
`emissivity = dec2hex[round(65535 x emiss)]`

Parameters

in	*emiss*	Physical emissivity value in range 0.1 ...1.0, default 1.0

Definition at line 109 of file MLX90614.cpp.

3.3.4.16 void MLX90614::setFIRcoeff (uint8_t *csb* = 7)

Set the coefficients of the FIR digital filter.

IIR coefficient setter

Remarks

The FIR digital filter coefficient N is bits 10:8 of ConfigRegister1
The value of N is set as follows: `N = 2 ^ (csb + 3)`
The manufacturer does not recommend `N < 128`

Parameters

in	*csb*	See page 12 of datasheet. Range 0...7, default = 7 (N = 1024)

Definition at line 186 of file MLX90614.cpp.

3.3.4.17 void MLX90614::setIIRcoeff (uint8_t *csb =* 4)

Set the coefficients of the IIR digital filter.

IIR coefficient setter

Remarks

> The IIR digital filter coefficients are set by the LS 3 bits of ConfigRegister1
> The value of the coefficients is set as follows:

```
csb = 0   a1 = 0.5    a2 = 0.5
      1        0.25        0.75
      2        0.167       0.833
      3        0.125       0.875
      4        1           0 (IIR bypassed)
      5        0.8         0.2
      6        0.67        0.33
      7        0.57        0.43
```

Parameters

in	*csb*	See page 12 of datasheet. Range 0...7, default = 4 (IIR bypassed)

Definition at line 145 of file MLX90614.cpp.

3.3.4.18 void MLX90614::write16 (uint8_t *cmd,* uint16_t *data*) [private]

Write a 16 bit value to memory.

Parameters

in	*cmd*	Command to send (register to write to).
in	*data*	Value to write.

Definition at line 314 of file MLX90614.cpp.

3.3.4.19 void MLX90614::writeEEProm (uint8_t *reg,* uint16_t *data*)

Write a 16 bit value to EEPROM after first clearing the memory.

Remarks

- Erase and write time 5ms per manufacturer specification
- Manufacturer does not specify max or min erase/write times

Parameters

in	*reg*	Address to write to.
in	*data*	Value to write.

Definition at line 351 of file MLX90614.cpp.

3.3.5 Member Data Documentation

3.3.5.1 uint8_t MLX90614::_addr [private]

Slave address

Definition at line 146 of file MLX90614.h.

3.3.5.2 uint8_t MLX90614::_crc8 [private]

8 bit CRC

Definition at line 148 of file MLX90614.h.

3.3.5.3 uint8_t MLX90614::_pec [private]

PEC

Definition at line 149 of file MLX90614.h.

3.3.5.4 uint8_t MLX90614::_rwError [private]

R/W error flags

Definition at line 147 of file MLX90614.h.

3.3.5.5 Property<uint8_t, MLX90614> MLX90614::busAddr

SMBus address property

Definition at line 125 of file MLX90614.h.

3.3.5.6 Property<uint8_t, MLX90614> MLX90614::crc8

8 bit CRC property

Definition at line 127 of file MLX90614.h.

3.3.5.7 Property<uint8_t, MLX90614> MLX90614::pec

PEC property

Definition at line 128 of file MLX90614.h.

3.3.5.8 Property<uint8_t, MLX90614> MLX90614::rwError

R/W error flags property

Definition at line 126 of file MLX90614.h.

The documentation for this class was generated from the following files:

- MLX90614.h
- MLX90614.cpp

Chapter 4

File Documentation

4.1 Crc8.cpp File Reference

8 bit CRC helper/utility class - CPP Source file.

```
#include "Crc8.h"
```

4.1.1 Detailed Description

8 bit CRC helper/utility class - CPP Source file.

Author

J. F. Fitter `jfitter@eagleairaust.com.au`

Version

1.0

Date

2014-2015

Copyright

License

Definition in file Crc8.cpp.

4.2 Crc8.h File Reference

8 bit CRC helper/utility class - CPP Header file.

```
#include "WProgram.h"
```

Classes

- class CRC8

Macros

- #define CRC8_DEFAULTPOLY 7

4.2.1 Detailed Description

8 bit CRC helper/utility class - CPP Header file.

Author

J. F. Fitter `jfitter@eagleairaust.com.au`

Version

1.0

Date

2014-2015

Copyright

License

Definition in file Crc8.h.

4.2.2 Macro Definition Documentation

4.2.2.1 #define CRC8_DEFAULTPOLY 7

Default CRC polynomial = X8+X2+X1+1

Definition at line 40 of file Crc8.h.

4.3 examples/mlxtest/MelexisTest.ino File Reference

Melexis MCX90614BAA Test Program - Sensor test implementation.

```
#include <Arduino.h>
#include <Wire.h>
#include <MLX90614.h>
#include "printf.h"
```

Classes

- struct defaultEEPromData

  EEPROM memory contents factory default values.

Functions

- void setup (void)

  Program setup.
- void loop (void)

  Main processing loop.
- void printlnTemp (double temp, char src)

  Print a line of temperature, crc, pec, and error string.
- void dumpEEProm ()

  Print a complete memory dump of the EEPROM.
- char ∗ floatToStr (char ∗str, double val)

  Utility to stringify a float.
- void printCRC (uint8_t crc, uint8_t pec)

  Just print the crc and pec.
- void printErrStr (uint8_t err)

  Convert error flags to diagnostic strings and print.
- void setEEPromDefaults (void)

  Set EEPROM memory contents to factory default values.

Variables

- MLX90614 **mlx** = MLX90614(MLX90614_BROADCASTADDR)
- const struct defaultEEPromData **eDat** []

4.3.1 Detailed Description

Melexis MCX90614BAA Test Program - Sensor test implementation. Arduino test implementation of Melexis MCX90614 PIR temperature sensor driver.

Note

THIS IS ONLY A PARTIAL RELEASE. THIS DEVICE CLASS IS CURRENTLY UNDERGOING ACTIVE DEVELOPMENT AND IS STILL MISSING SOME IMPORTANT FEATURES. PLEASE KEEP THIS IN MIND IF YOU DECIDE TO USE THIS PARTICULAR CODE FOR ANYTHING.

Author

J. F. Fitter `jfitter@eagleairaust.com.au`

Version

1.0

Date

2014-2015

Copyright

Copyright (c) 2015 John Fitter. All right reserved.

License

GNU Public License. Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Definition in file MelexisTest.ino.

4.3.2 Function Documentation

4.3.2.1 char∗ floatToStr (char ∗ *str,* double *val*)

Utility to stringify a float.

Parameters

in	*str*	String to receive converted result
in	*val*	Float value

Returns

Float as string

Definition at line 133 of file MelexisTest.ino.

4.3.2.2 void printCRC (uint8_t *crc,* uint8_t *pec*)

Just print the crc and pec.

Parameters

in	*crc*	CRC
in	*pec*	PEC

Definition at line 144 of file MelexisTest.ino.

4.3.2.3 void printErrStr (uint8_t *err*)

Convert error flags to diagnostic strings and print.

Parameters

in	*err*	Error flags

Definition at line 150 of file MelexisTest.ino.

4.3.2.4 void printlnTemp (double *temp,* char *src*)

Print a line of temperature, crc, pec, and error string.

Parameters

in	*temp*	Temperature
in	*src*	Temperature source

Definition at line 97 of file MelexisTest.ino.

4.3.2.5 void setEEPromDefaults (void)

Set EEPROM memory contents to factory default values.

Remarks

A device with default adress must not be on the bus.

```
Only user allowed memory locations are written.
```

Definition at line 182 of file MelexisTest.ino.

4.3.3 Variable Documentation

4.3.3.1 const struct defaultEEPromData eDat[ ]

Initial value:

```
=  {{0x20, 0x9993}, {0x21, 0x62E3}, {0x22, 0x0201},
             {0x23, 0xF71C}, {0x24, 0xFFFF}, {0x25, 0x9FB4},
             {0x2E, 0xBE5A}, {0x2F, 0x0000}, {0x39, 0x0000}}
```

4.4 MLX90614.cpp File Reference

Melexis MLX90614 Family Device Driver Library - CPP Source file.

```
#include "MLX90614.h"
```

4.4.1 Detailed Description

Melexis MLX90614 Family Device Driver Library - CPP Source file. Based on the Melexis MLX90614 Family Data Sheet 3901090614 Rev 004 09jun2008.

- The current implementation does not manage PWM (only digital data by I2C).
- Sleep mode is not implemented yet.

Note

THIS IS ONLY A PARTIAL RELEASE. THIS DEVICE CLASS IS CURRENTLY UNDERGOING ACTIVE DEVELOPMENT AND IS STILL MISSING SOME IMPORTANT FEATURES. PLEASE KEEP THIS IN MIND IF YOU DECIDE TO USE THIS PARTICULAR CODE FOR ANYTHING.

Author

J. F. Fitter jfitter@eagleairaust.com.au

Version

1.0

Date

2014-2015

Copyright

Copyright (c) 2015 John Fitter. All right reserved.

License

GNU Public License. Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Definition in file MLX90614.cpp.

4.5 MLX90614.h File Reference

Melexis MLX90614 Family Device Driver Library - CPP Header file.

```
#include "WProgram.h"
#include <Wire.h>
#include "Property.h"
#include "Crc8.h"
```

Classes

- class MLX90614

Macros

- #define MLX90614_I2CDEFAULTADDR 0x5A
- #define MLX90614_BROADCASTADDR 0
- #define MLX90614_CRC8POLY 7
- #define MLX90614_XDLY 25
- #define MLX90614_RAWIR1 0x04
- #define MLX90614_RAWIR2 0x05
- #define MLX90614_TA 0x06
- #define MLX90614_TOBJ1 0x07
- #define MLX90614_TOBJ2 0x08
- #define MLX90614_TOMAX 0x00
- #define MLX90614_TOMIN 0x01
- #define MLX90614_PWMCTRL 0x02
- #define MLX90614_TARANGE 0x03
- #define MLX90614_EMISS 0x04
- #define MLX90614_CONFIG 0x05
- #define MLX90614_ADDR 0x0E
- #define MLX90614_ID1 0x1C
- #define MLX90614_ID2 0x1D
- #define MLX90614_ID3 0x1E
- #define MLX90614_ID4 0x1F
- #define MLX90614_RFLAGCMD 0xF0
- #define MLX90614_EEBUSY 0x80
- #define MLX90614_EE_DEAD 0x20
- #define MLX90614_INIT 0x10
- #define MLX90614_SUCCESS 0
- #define MLX90614_DATATOOLONG 1
- #define MLX90614_TXADDRNACK 2
- #define MLX90614_TXDATANACK 4
- #define MLX90614_TXOTHER 8
- #define MLX90614_RXCRC 0x10
- #define MLX90614_INVALIDATA 0x20
- #define MLX90614_EECORRUPT 0x40
- #define MLX90614_RFLGERR 0x80

4.5.1 Detailed Description

Melexis MLX90614 Family Device Driver Library - CPP Header file. Based on the Melexis MLX90614 Family Data Sheet 3901090614 Rev 004 09jun2008.

- The current implementation does not manage PWM (only digital data by I2C).
- Sleep mode is not implemented yet.

Note

THIS IS ONLY A PARTIAL RELEASE. THIS DEVICE CLASS IS CURRENTLY UNDERGOING ACTIVE DEVELOPMENT AND IS STILL MISSING SOME IMPORTANT FEATURES. PLEASE KEEP THIS IN MIND IF YOU DECIDE TO USE THIS PARTICULAR CODE FOR ANYTHING.

Author

J. F. Fitter `jfitter@eagleairaust.com.au`

Version

1.0

Date

2014-2015

Copyright

License

Definition in file MLX90614.h.

4.5.2 Macro Definition Documentation

4.5.2.1 #define MLX90614_ADDR 0x0E

EEPROM reg - SMBus address

Definition at line 79 of file MLX90614.h.

4.5.2.2 #define MLX90614_BROADCASTADDR 0

Device broadcast slave address

Definition at line 56 of file MLX90614.h.

4.5.2.3 #define MLX90614_CONFIG 0x05

EEPROM reg - Configuration register

Definition at line 78 of file MLX90614.h.

4.5.2.4 #define MLX90614_CRC8POLY 7

CRC polynomial = X8+X2+X1+1

Definition at line 57 of file MLX90614.h.

4.5.2.5 #define MLX90614_DATATOOLONG 1

R/W error bitmask - Data is too long

Definition at line 94 of file MLX90614.h.

4.5.2.6 #define MLX90614_EE_DEAD 0x20

R/W flag bitmask - EEProm double error has occurred

Definition at line 89 of file MLX90614.h.

4.5.2.7 #define MLX90614_EEBUSY 0x80

Read flags - bitmask. R/W flag bitmask - EEProm is busy (writing/erasing)

Definition at line 88 of file MLX90614.h.

4.5.2.8 #define MLX90614_EECORRUPT 0x40

R/W error bitmask - The EEProm is likely to be corrupted

Definition at line 100 of file MLX90614.h.

4.5.2.9 #define MLX90614_EMISS 0x04

EEPROM reg - Object emissivity register

Definition at line 77 of file MLX90614.h.

4.5.2.10 #define MLX90614_I2CDEFAULTADDR 0x5A

Device default slave address

Definition at line 55 of file MLX90614.h.

4.5.2.11 #define MLX90614_ID1 0x1C

EEPROM reg - ID numer (w1)

Definition at line 80 of file MLX90614.h.

4.5.2.12 #define MLX90614_ID2 0x1D

EEPROM reg - ID numer (w2)

Definition at line 81 of file MLX90614.h.

4.5.2.13 #define MLX90614_ID3 0x1E

EEPROM reg - ID numer (w3)

Definition at line 82 of file MLX90614.h.

4.5.2.14 #define MLX90614_ID4 0x1F

EEPROM reg - ID numer (w4)

Definition at line 83 of file MLX90614.h.

4.5.2.15 #define MLX90614_INIT 0x10

R/W flag bitmask - POR initialization is still ongoing

Definition at line 90 of file MLX90614.h.

4.5.2.16 #define MLX90614_INVALIDATA 0x20

R/W error bitmask - RX/TX Data fails selection criteria

Definition at line 99 of file MLX90614.h.

4.5.2.17 #define MLX90614_PWMCTRL 0x02

EEPROM reg - Pulse width modulation output control register

Definition at line 75 of file MLX90614.h.

4.5.2.18 #define MLX90614_RAWIR1 0x04

RAM addresses. RAM reg - Raw temperature, source #1

Definition at line 66 of file MLX90614.h.

4.5.2.19 #define MLX90614_RAWIR2 0x05

RAM reg - Raw temperature, source #2

Definition at line 67 of file MLX90614.h.

4.5.2.20 #define MLX90614_RFLAGCMD 0xF0

Read R/W Flags register command

Definition at line 85 of file MLX90614.h.

4.5.2.21 #define MLX90614_RFLGERR 0x80

R/W error bitmask - R/W flags register access error

Definition at line 101 of file MLX90614.h.

4.5.2.22 #define MLX90614_RXCRC 0x10

R/W error bitmask - Receiver CRC mismatch

Definition at line 98 of file MLX90614.h.

4.5.2.23 #define MLX90614_SUCCESS 0

R/W Error flags - bitmask. R/W error bitmask - No Errors

Definition at line 93 of file MLX90614.h.

4.5.2.24 #define MLX90614_TA 0x06

RAM reg - Linearized temperature, ambient

Definition at line 68 of file MLX90614.h.

4.5.2.25 #define MLX90614_TARANGE 0x03

EEPROM reg - Customer dependent ambient temperature range

Definition at line 76 of file MLX90614.h.

4.5.2.26 #define MLX90614_TOBJ1 0x07

RAM reg - Linearized temperature, source #1

Definition at line 69 of file MLX90614.h.

4.5.2.27 #define MLX90614_TOBJ2 0x08

RAM reg - Linearized temperature, source #2

Definition at line 70 of file MLX90614.h.

4.5.2.28 #define MLX90614_TOMAX 0x00

EEPROM addresses. EEPROM reg - Customer dependent object temperature range maximum

Definition at line 73 of file MLX90614.h.

4.5.2.29 #define MLX90614_TOMIN 0x01

EEPROM reg - Customer dependent object temperature range minimum

Definition at line 74 of file MLX90614.h.

4.5.2.30 #define MLX90614_TXADDRNACK 2

R/W error bitmask - TX address not acknowledged

Definition at line 95 of file MLX90614.h.

4.5.2.31 #define MLX90614_TXDATANACK 4

R/W error bitmask - TX data not acknowledged

Definition at line 96 of file MLX90614.h.

4.5.2.32 #define MLX90614_TXOTHER 8

R/W error bitmask - Unknown error

Definition at line 97 of file MLX90614.h.

4.5.2.33 #define MLX90614_XDLY 25

Experimentally determined delay to prevent read errors after calling Wire.endTransmission() *(possibly due to incompatibility between Wire library and SMBus protocol)*.

Definition at line 58 of file MLX90614.h.

Index

_addr
    MLX90614, 15
_crc8
    MLX90614, 15
_pec
    MLX90614, 15
_rwError
    MLX90614, 15

busAddr
    MLX90614, 15

CRC8, 5
    CRC8, 5
    CRC8, 5
    crc8, 5, 6
    crc8Start, 6
CRC8_DEFAULTPOLY
    Crc8.h, 19
convCtoF
    MLX90614, 10
convKtoC
    MLX90614, 10
crc8
    CRC8, 5, 6
    MLX90614, 15
Crc8.cpp, 17
Crc8.h, 18
    CRC8_DEFAULTPOLY, 19
crc8Start
    CRC8, 6

defaultEEPromData, 6

eDat
    MelexisTest.ino, 21
examples/mlxtest/MelexisTest.ino, 19

floatToStr
    MelexisTest.ino, 20

getAddr
    MLX90614, 10
getCRC8
    MLX90614, 10
getEmissivity
    MLX90614, 10
getFIRcoeff
    MLX90614, 11
getIIRcoeff
    MLX90614, 11
getPEC
    MLX90614, 11
getRwError
    MLX90614, 11

MLX90614
    MLX90614_SRCA, 8
    MLX90614_SRCO1, 8
    MLX90614_SRCO2, 8
    MLX90614_TC, 8
    MLX90614_TF, 8
    MLX90614_TK, 8
MLX90614_SRCA
    MLX90614, 8
MLX90614_SRCO1
    MLX90614, 8
MLX90614_SRCO2
    MLX90614, 8
MLX90614_TC
    MLX90614, 8
MLX90614_TF
    MLX90614, 8
MLX90614_TK
    MLX90614, 8
MLX90614, 7
    _addr, 15
    _crc8, 15
    _pec, 15
    _rwError, 15
    busAddr, 15
    convCtoF, 10
    convKtoC, 10
    crc8, 15
    getAddr, 10
    getCRC8, 10
    getEmissivity, 10
    getFIRcoeff, 11
    getIIRcoeff, 11
    getPEC, 11
    getRwError, 11
    MLX90614, 9
    MLX90614, 9
    pec, 15
    read16, 12
    readEEProm, 12
    readID, 12
    readTemp, 12
    rwError, 15
    setAddr, 13
    setEmissivity, 13

setFIRcoeff, 13
setIIRcoeff, 14
tempSrc_t, 8
tempUnit_t, 8
write16, 14
writeEEProm, 14
MLX90614.cpp, 22
MLX90614.h, 23
MLX90614_ADDR, 24
MLX90614_BROADCASTADDR, 24
MLX90614_CONFIG, 25
MLX90614_CRC8POLY, 25
MLX90614_DATATOOLONG, 25
MLX90614_EE_DEAD, 25
MLX90614_EEBUSY, 25
MLX90614_EECORRUPT, 25
MLX90614_EMISS, 25
MLX90614_I2CDEFAULTADDR, 25
MLX90614_ID1, 25
MLX90614_ID2, 26
MLX90614_ID3, 26
MLX90614_ID4, 26
MLX90614_INIT, 26
MLX90614_INVALIDATA, 26
MLX90614_PWMCTRL, 26
MLX90614_RAWIR1, 26
MLX90614_RAWIR2, 26
MLX90614_RFLAGCMD, 26
MLX90614_RFLGERR, 27
MLX90614_RXCRC, 27
MLX90614_SUCCESS, 27
MLX90614_TA, 27
MLX90614_TARANGE, 27
MLX90614_TOBJ1, 27
MLX90614_TOBJ2, 27
MLX90614_TOMAX, 27
MLX90614_TOMIN, 27
MLX90614_TXADDRNACK, 28
MLX90614_TXDATANACK, 28
MLX90614_TXOTHER, 28
MLX90614_XDLY, 28
MLX90614_ADDR
MLX90614.h, 24
MLX90614_BROADCASTADDR
MLX90614.h, 24
MLX90614_CONFIG
MLX90614.h, 25
MLX90614_CRC8POLY
MLX90614.h, 25
MLX90614_DATATOOLONG
MLX90614.h, 25
MLX90614_EE_DEAD
MLX90614.h, 25
MLX90614_EEBUSY
MLX90614.h, 25
MLX90614_EECORRUPT
MLX90614.h, 25
MLX90614_EMISS
MLX90614.h, 25
MLX90614_I2CDEFAULTADDR
MLX90614.h, 25
MLX90614_ID1
MLX90614.h, 25
MLX90614_ID2
MLX90614.h, 26
MLX90614_ID3
MLX90614.h, 26
MLX90614_ID4
MLX90614.h, 26
MLX90614_INIT
MLX90614.h, 26
MLX90614_INVALIDATA
MLX90614.h, 26
MLX90614_PWMCTRL
MLX90614.h, 26
MLX90614_RAWIR1
MLX90614.h, 26
MLX90614_RAWIR2
MLX90614.h, 26
MLX90614_RFLAGCMD
MLX90614.h, 26
MLX90614_RFLGERR
MLX90614.h, 27
MLX90614_RXCRC
MLX90614.h, 27
MLX90614_SUCCESS
MLX90614.h, 27
MLX90614_TA
MLX90614.h, 27
MLX90614_TARANGE
MLX90614.h, 27
MLX90614_TOBJ1
MLX90614.h, 27
MLX90614_TOBJ2
MLX90614.h, 27
MLX90614_TOMAX
MLX90614.h, 27
MLX90614_TOMIN
MLX90614.h, 27
MLX90614_TXADDRNACK
MLX90614.h, 28
MLX90614_TXDATANACK
MLX90614.h, 28
MLX90614_TXOTHER
MLX90614.h, 28
MLX90614_XDLY
MLX90614.h, 28
MelexisTest.ino
eDat, 21
floatToStr, 20
printCRC, 21
printErrStr, 21
printlnTemp, 21
setEEPromDefaults, 21

pec
MLX90614, 15

printCRC
    MelexisTest.ino, 21
printErrStr
    MelexisTest.ino, 21
printlnTemp
    MelexisTest.ino, 21

read16
    MLX90614, 12
readEEProm
    MLX90614, 12
readID
    MLX90614, 12
readTemp
    MLX90614, 12
rwError
    MLX90614, 15

setAddr
    MLX90614, 13
setEEPromDefaults
    MelexisTest.ino, 21
setEmissivity
    MLX90614, 13
setFIRcoeff
    MLX90614, 13
setIIRcoeff
    MLX90614, 14

tempSrc_t
    MLX90614, 8
tempUnit_t
    MLX90614, 8

write16
    MLX90614, 14
writeEEProm
    MLX90614, 14