RTC Library for CC3200 and MSP432 - Reference Manual

Generated by Doxygen 1.8.10

Sat Aug 1 2015 18:14:59

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 DateTime Class Reference 5

3.1.1 Detailed Description 5

3.1.2 Member Function Documentation 5

3.1.2.1 getLocalTime() 5

3.1.2.2 getTime() 6

3.1.2.3 setLocalTime(uint32_t epoch) 6

3.1.2.4 setTime(uint32_t epoch) 6

3.1.2.5 setTimeZone(int32_t timeZone) 6

4 File Documentation **7**

4.1 Credentials.h File Reference 7

4.1.1 Detailed Description 7

4.2 DateTimeLibrary.h File Reference 8

4.2.1 Detailed Description 9

4.2.2 Macro Definition Documentation 9

4.2.2.1 INCLUDE_NTP 9

4.2.2.2 tz_GMT 10

4.2.2.3 tz_PDT 10

4.2.2.4 tz_PDT 10

4.2.3 Function Documentation 10

4.2.3.1 convertEpoch2Structure(time_t epoch, tm &timeStructure) 10

4.2.3.2 convertStructure2Epoch(tm timeStructure, time_t &epoch) 11

4.2.3.3 formatStringDateTime(const char *format, tm timeStructure) 11

4.2.3.4 formatStringDateTime(const char *format, time_t epoch) 11

4.2.3.5 getTimeNTP(time_t &epochNTP, IPAddress serverNTP=IPAddress(145, 238, 203, 14)) 11

4.2.3.6 stringDateTime(tm timeStructure) 12
4.2.3.7 stringDateTime(time_t epoch) 12

Index **13**

Chapter 1

Class Index

1.1 Class List

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

DateTime
Class for RTC 5

Chapter 2

File Index

2.1 File List

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

Credentials.h
    Header 7
DateTimeLibrary.h
    Library header 8

Chapter 3

Class Documentation

3.1 DateTime Class Reference

Class for RTC.

`#include <DateTimeLibrary.h>`

Public Member Functions

- DateTime ()
  Constructor.
- void begin ()
  Initialisation.
- uint32_t getTime ()
  Get GMT date and time.
- uint32_t getLocalTime ()
  Get local date and time.
- void setTime (uint32_t epoch)
  Set GMT date and time.
- void setLocalTime (uint32_t epoch)
  Set local date and time.
- void setTimeZone (int32_t timeZone)
  Set time zone.

3.1.1 Detailed Description

Class for RTC.

Note

Tested on MSP432-EMT, CC3200, LM4F120, TM4C123, TM4C129

3.1.2 Member Function Documentation

3.1.2.1 uint32_t DateTime::getLocalTime ( )

Get local date and time.

Returns

epoch = number of seconds since Jan 1st, 1970, uint32_t or time_t

Note

Set the time zone with setTimeZone()

3.1.2.2 uint32_t DateTime::getTime ( )

Get GMT date and time.

Returns

epoch = number of seconds since Jan 1st, 1970, uint32_t or time_t

3.1.2.3 void DateTime::setLocalTime (uint32_t *epoch*)

Set local date and time.

Parameters

epoch	number of seconds since Jan 1st, 1970

Note

Set the time zone with setTimeZone()

3.1.2.4 void DateTime::setTime (uint32_t *epoch*)

Set GMT date and time.

Parameters

epoch	number of seconds since Jan 1st, 1970

3.1.2.5 void DateTime::setTimeZone (int32_t *timeZone*)

Set time zone.

Parameters

timeZone	difference in seconds between local time zone and GMT

Note

Use pre-defined tz_CET, tz_CEST, tz_PST, tz_PDT, ...

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

- DateTimeLibrary.h
- DateTimeLibrary.cpp

Chapter 4

File Documentation

4.1 Credentials.h File Reference

Header.

Variables

- char ssid [ ] = "ssid"
  Network name of SSID.
- char password [ ] = "password"
  Network password.

4.1.1 Detailed Description

Header.

Credentials for WiFi LAN

Project CC3200_NTP
Developed with embedXcode+

Author

Rei Vilo
http://embeddedcomputing.weebly.com

Date

19/07/2015 11:35

Version

101

Copyright

See also

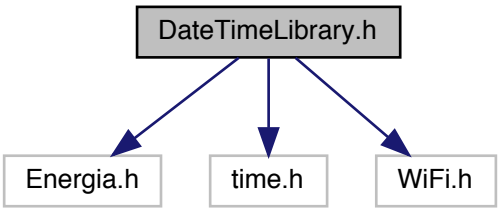
ReadMe.txt for references

4.2 DateTimeLibrary.h File Reference

Library header.

```
#include "Energia.h"
#include "time.h"
#include "WiFi.h"
```

Include dependency graph for DateTimeLibrary.h:



Classes

- class DateTime

  Class for RTC.

Macros

- #define INCLUDE_NTP 1

  Scope for NTP.

- #define tz_GMT 0

  Predefined time zones.
- #define tz_CUT 0

  Coordinated Universal Time.
- #define tz_BST 1∗60∗60

  British Summer Time.
- #define tz_CET 1∗60∗60

  Central Europe Time.
- #define tz_CEST 2∗60∗60

  Central Europe Summer Time.
- #define tz_PST -8∗60∗60

  Pacific Standard Time.
- #define tz_PDT -7∗60∗60

  Pacific Daylight Time.
- #define tz_CST -6∗60∗60

  Central Standard Time.
- #define tz_PDT -5∗60∗60

  Pacific Daylight Time.

Functions

- bool getTimeNTP (time_t &epochNTP, IPAddress serverNTP=IPAddress(145, 238, 203, 14))
  Get date and time from NTP server.
- void convertEpoch2Structure (time_t epoch, tm &timeStructure)
  Utilities.
- void convertStructure2Epoch (tm timeStructure, time_t &epoch)
  Convert structure into epoch.
- String stringDateTime (tm timeStructure)
  Standard format for date and time.
- String stringDateTime (time_t epoch)
  Standard format for date and time.
- String formatStringDateTime (const char *format, tm timeStructure)
  Custom format for date and time.
- String formatStringDateTime (const char *format, time_t epoch)
  Custom format for time.

4.2.1 Detailed Description

Library header.

RTC Date and Time Library for MSP432 and CC3200

Project MSP432_RTC
Developed with embedXcode+

Author

Rei Vilo
http://embeddedcomputing.weebly.com

Date

Jul 31, 2015

Version

102

Copyright

(c) Rei Vilo, 2015
CC = BY SA NC

See also

ReadMe.txt for references
http://www.epochconverter.com

4.2.2 Macro Definition Documentation

4.2.2.1 #define INCLUDE_NTP 1

Scope for NTP.

1 to include NTP utility, 0 otherwise

4.2.2.2 #define tz_GMT 0

Predefined time zones.

Difference in seconds to GMT/CUT
CET = 3600 = GMT + 1 hourGreenwich Mean Time

4.2.2.3 #define tz_PDT -7∗60∗60

Pacific Daylight Time.

Central Daylight Time

4.2.2.4 #define tz_PDT -5∗60∗60

Pacific Daylight Time.

Central Daylight Time

4.2.3 Function Documentation

4.2.3.1 void convertEpoch2Structure (time_t *epoch,* tm & *timeStructure*)

Utilities.

There are 2 representation for data and time.

- **Epoch**
  POSIX time = number of seconds since 00:00 Jan 1st, 1979
- **Structure**
  The MSP432 has its own not compatible structure!

MSP432 specific structure	Standard C structure
struct _RTC_C_Calendar	struct tm
{	{
uint_fast8_t seconds;	int tm_sec; // seconds after the minute [0-60]
uint_fast8_t minutes;	int tm_min; // minutes after the hour [0-59]
uint_fast8_t hours;	int tm_hour; // hours since midnight [0-23]
uint_fast8_t dayOfWeek;	(∗)
uint_fast8_t dayOfmonth;	int tm_mday; // day of the month [1-31]
uint_fast8_t month;	int tm_mon; // months since January [0-11]
uint_fast16_t year;	int tm_year; // years since 1900
(∗)	int tm_wday; // days since Sunday [0-6]
.	int tm_yday; // days since January 1 [0-365]
.	int tm_isdst; // Daylight Savings Time flag
.	long tm_gmtoff; // offset from CUT in seconds
.	char ∗tm_zone; // timezone abbreviation
};	};

Convert epoch into structure

Parameters

epoch	time as epoch

timeStructure	time as structure

4.2.3.2 void convertStructure2Epoch (tm *timeStructure*, time_t & *epoch*)

Convert structure into epoch.

Parameters

timeStructure	time as time structure
epoch	time as epoch

4.2.3.3 String formatStringDateTime (const char * *format*, tm *timeStructure*)

Custom format for date and time.

Parameters

format	see below
timeStructure	time as structure

Returns

String

4.2.3.4 String formatStringDateTime (const char * *format*, time_t *epoch*)

Custom format for time.

Parameters

format	see below
epoch	time as epoch

Returns

String

4.2.3.5 bool getTimeNTP (time_t & *epochNTP*, IPAddress *serverNTP* = `IPAddress(145, 238, 203, 14)`)

Get date and time from NTP server.

Parameters

epochNTP	time in epoch format
serverNTP	IP address of the NTP server, default =

Returns

true is successful, false otherwise

Note

epochNTP is updated only if successful.

Warning

A valid connection to Internet is required.

Note

Examples of NTP servers

- time.nist.gov IPAddress(206,246,122,250)
- wwv.nist.gov IPAddress(24,56,178,140)
- ntp-p1.obspm.fr IPAddress(145,238,203,14)

Based on UDP NTP Client, provided with Energia 16

- Created 4 Sep 2010 by Michael Margolis
- Modified 9 Apr 2012 by Tom Igoe
- Modified 1 July 2014 by Noah Luskey
- Updated July 19, 2015 by Rei Vilo with RTC for CC3200, MSP432, TM4C123 and TM4C129

See also

NTP time servers and messages
http://en.wikipedia.org/wiki/Network_Time_Protocol

4.2.3.6 String stringDateTime (tm *timeStructure*)

Standard format for date and time.

Parameters

timeStructure	time as structure

Returns

String

Note

Sun Jul 19 18:55:13 2015

4.2.3.7 String stringDateTime (time_t *epoch*)

Standard format for date and time.

Parameters

epoch	time as epoch

Returns

String

Note

Sun Jul 19 18:55:13 2015

Index

convertEpoch2Structure
    DateTimeLibrary.h, 10
convertStructure2Epoch
    DateTimeLibrary.h, 11
Credentials.h, 7

DateTime, 5
    getLocalTime, 5
    getTime, 6
    setLocalTime, 6
    setTime, 6
    setTimeZone, 6
DateTimeLibrary.h, 8
    convertEpoch2Structure, 10
    convertStructure2Epoch, 11
    formatStringDateTime, 11
    getTimeNTP, 11
    INCLUDE_NTP, 9
    stringDateTime, 12
    tz_GMT, 9
    tz_PDT, 10

formatStringDateTime
    DateTimeLibrary.h, 11

getLocalTime
    DateTime, 5
getTime
    DateTime, 6
getTimeNTP
    DateTimeLibrary.h, 11

INCLUDE_NTP
    DateTimeLibrary.h, 9

setLocalTime
    DateTime, 6
setTime
    DateTime, 6
setTimeZone
    DateTime, 6
stringDateTime
    DateTimeLibrary.h, 12

tz_GMT
    DateTimeLibrary.h, 9
tz_PDT
    DateTimeLibrary.h, 10