drmLib Arduino Utility Library

Table of Contents

List of Tables

List of Figures

List of Equations

1 ABSTRACT

The routines in drmLib are described and enough information to use this library is provided. The source code is the ultimate documentation for these routines. It is the duty of the code author to decide if this library is appropriate for any particular use. It was originally drafted for the Logging Dive Computer project.

Use:

#include <drmLib.h>

2 LIBRARY SUMMARY

1. byte drmBcd2Dec(byte inbyte);
2. unsigned short drmSerialNo();
3. void drmStartPrint(const char *drmversion);
4. void drmPrtLead0(long in, int places);
5. void printTime(unsigned long milli_time);
6. float read_samM0_batt();
7. void print_samM0_serial();
8. ***The time routines below are not yet implemented!***
9. *int initRTC(rtc_type type);*
10. *struct parseTime readRTC(rtc_type type);*
11. *void readClock(byte *readBytes);*
12. *struct parseTime decodeTime(byte *readBytes);*

3 LIBRARY DETAILS

1. byte drmBcd2Dec(byte inbyte); – Converts a two digit BCD number into a binary representation
2. unsigned short drmSerialNo(); – Retrieves the serial number from the EEPROM of an Atmel processor
3. void drmStartPrint(const char *drmversion); – Prints out the standard start message
4. void drmPrtLead0(long in, int places); – Prints a long integer with leading zeros
5. float read_samM0_batt(); returns the the battery voltage level on an Adafruit Feather SAM M0 board
6. void print_samM0_serial(); – prints the full and shortened serial number on an Adafruit Feather SAM M0 board

7. void printTime(unsigned long milli_time); – Prints the elapsed time in the format d-h:m:s given a millisecond value
8. ***The time routines below are not yet implemented!***
9. *int initRTC(rtc_type type); – Sets up an RTC module*
10. *struct parseTime readRTC(rtc_type type); – Parses out the elements of the date/time from RTC raw data*
11. *void readClock(byte *readBytes); – Returns time from an RTC*
12. *struct parseTime decodeTime(byte *readBytes); – Parses time data returned from RTC*

4 DATA STRUCTURES

```
enum rtc_type
{
        DS3231,
        OTHER
}
```

This structure contains all the RTC data

```
typedef struct parseTime {
  byte seconds;
  byte minutes;
  byte hours;
  byte dow;
  byte dom;
  byte month;
  byte year;
  byte csr;
  byte sr;
  int int_year;
  unsigned long lsec;
  long tempf; // F temperature * 100 (poor man's float)
};
}
```

5 CONSTANTS

5.1 MISC CONSTANTS

```
#define MAX_LINE 180
#define NUMRTCREGS 19
```

5.2 STATUS VALUES

```
#define ER_BADID -20        // bad ID on requested operation
#define ER_BADOPEN -21      // error opening file
#define ER_UNEXPFIO -22     // unexpected result with file IO
#define ST_AOK 0            // Everything is good
#define ST_NOERR 1          // No error occurred
#define ST_NOOP 2           // Nothing happened
#define ER_ERR -1           // generic error
#define ER_UNK -1001        // unknown error
#define FALSE 1!=1
#define TRUE 1==1
```

6 HISTORY

Created by drm 20151213
V1.0 --> First with EEPROM access and start print
V2.0 --> adding RTC access
V2.1 --> ifdef for some M0 cases
V2.2 --> added some SAM M0 stuff to library and extended serial number support

This library is a collection of functions that find useful again and again. It will be developed into a portable library of C/C++ routines that can be used on various processors (Raspberry Pi, Arduino, Intel, ST Micro, etc)

7 REVISION HISTORY/RELEASE STATUS

REVISIONS		
REV	DESCRIPTION	DATE/APPROVED
A	Initial release	20160101 drm
B	V2.2 Enhanced for SAM M0 MCU	20170104 drm

TeamManatt
16061 Vista de Golf
Rancho Santa Fe CA 92091

Review & Approval (digital signatures)	C:\Users\manatt\Documents\GoogleDrive\Engineering\PoliciesAllProjects\SW000-100B_drmLib.docx					
DRAWN D. Manatt CHECKED ENGR D. Manatt ENGR REV	drmLib Arduino Utility Library					
PROJ MGR D. Manatt QUAL MGR	SIZE A	CAGE CODE N/A	DRAWING NUMBER SW-000-100			REV A
RELEASE D. Manatt			DRW LEVEL 3	SCALE NONE		SHEET 1 OF 5