
Triangle MicroWorks, Inc.
IEC 60870-5-103 Master
Source Code Library

What's New?

a description of the changes included in all versions of the library

Version 3.13.00
November 17, 2011

Property of Triangle MicroWorks, Inc.



This Source Code and the associated Documentation contain proprietary information of Triangle MicroWorks, Inc. and may not be copied or distributed in any form without the written permission of Triangle MicroWorks, Inc.

Copies of the source code may be made only for backup purposes.





© 1994 - 2011 Triangle MicroWorks, Inc. All rights reserved.

This document describes features or corrections that have been added to the IEC 60870-5-103 Master Source Code Library.




The symbols to the left of each revision are used to help define the following kinds of revisions:


- 103** Additional IEC 60870-5-103 Features, giving the Library support for more IEC 60870-5-103 objects or functions.
-  Additional Implementation Features, allowing implementers to more efficiently install the Source Code Library.
-  Corrections to problems, with indications for when the problems were introduced.


Version 3.13.00 (November 17, 2011)

-  Modified 56-bit time decoding so that years 70-99 assume 1970-1999 instead of 2070-2099
-  Corrected a warning caused by an uninitialized structure in LinIOTarg RS232 implementation.
-  Implemented performance improvement by modifying locking calls in `tmwdb_storeEntry`. This change applies only when processing data asynchronously.
-  Provided big endian get and put functions.
-  *Description:* LinIOTarg could access channel structure members after marking a channel for deletion.
Introduced: v3.07
Resolution: Fixed in v3.13

Version 3.12.00 (July 7, 2011)

-  *Description:* The timer callback may continue to loop if a very short timer is added in the timer callback.
Introduced: v3.00
Resolution: Fixed in v3.12
-  *Description:* `tmwdtime_addOffset` did not follow 0 -599999 rule.
Introduced: v3.00
Resolution: Fixed in v3.12
-  *Description:* Some compilers issued warnings when a break followed a return (e.g., in `tmwphysd.c`).
Introduced: v3.00
Resolution: Fixed in v3.12

 *Description:* In function `tmwmem_lowlnit()`, a failure to allocate memory resulted in a return with no indication to the caller that a failure occurred.
Introduced: v3.00
Resolution: Fixed in v3.12

 *Description:* `LinIOTarg` computed date incorrectly when converting a “struct tm” to `TMWDTIME`.
Introduced: v3.07
Resolution: Fixed in v3.12

Version 3.11.00 (October 1, 2010)


No updates to IEC 60870-5-103 Master Source Code Library in this release.


Version 3.10.00 (August 25, 2010)


No updates to IEC 60870-5-103 Master Source Code Library in this release.


Version 3.09.00 (March 15, 2010)

 Added user data pointers to `TMWCHNL`, `TMWSCCTR`, and `TMWSESN`.


 *Description:* When multiple application contexts are used, closing one deleted the memory pool lock, which could cause problems if another application context was open.
Introduced: v3.00
Resolution: Fixed in v3.09

 *Description:* `WinIOTarg` could fail to process all available characters, causing timeouts when running at high baud rates.
Introduced: v3.00
Resolution: Fixed in v3.09




 *Description:* When configured to use system time, `WinIOTarg` System time not setting the `dstlnEffect` flag correctly.
Introduced: v3.00
Resolution: Fixed in v3.09

 *Description:* `LinIOTarg` could cause high levels of processor utilization when multiple channels were opened.
Introduced: v3.07
Resolution: Fixed in v3.09



Version 3.08.00 (September 25, 2009)

 *Description:* The `rs232Reader` thread in `WinIOTarg` was missing a stop event.
Introduced: v3.00
Resolution: Fixed in v3.08

Version 3.07.00 (July 17, 2009)

-  Added LinIOTarg (low-level target for Linux) and new command line-based examples (that can be used with Linux).
-  Integrated WinIOTarg (low-level target for Windows) into the standard release package.
-  Added FormatMessage Win32 API function to WinIoTarg to provide more descriptive errors instead of just numeric values.

Version 3.06.00 (April 14, 2009)

-  Added TMWCNFG_MAX_APPLRCVS to “break out” of loop in tmwappl_checkForInput() after the specified number of iterations.
-  Modified timer values in tmwlink_channelCallback to speed up retries.




Version 3.05.00 (December 4, 2008)

No updates to IEC 60870-5-103 Master Source Code Library in this release.

Version 3.04.01 (September 30, 2008)

-  Improved comments on compiler options such as TMWCONFIG_USE_MANAGED_SCL.


Version 3.04 (September 12, 2008)

-  *Description:* The Source Code Library did not poll properly for Class 1 data if class1PollCount ==0 and only one session was configured.
Introduced: v3.00
Resolution: Fixed in v3.04
-  *Description:* tmwdb_lockQueue and tmwdb_unlockQueue were described in the header comments for tmwdb_storeEntry in tmwdb.h, but they were not defined in tmwdb.c.
Introduced: v3.00
Resolution: Fixed in v3.04
-  *Description:* A null pointer could be referenced if a response was received before the link confirm.
Introduced: v3.0
Resolution: Fixed in v3.04



Version 3.03.00 (April 1, 2008)

-  Added ability to include session indicator in channel statistics.


Version 3.02.00 (December 7, 2007)

-  *Description:* The Source Code Library could poll other devices even though there was an outstanding response if the ACS bit was set in a “no data” response, which could cause problems in multi-drop applications.
Introduced: v3.00
Resolution: Fixed in v3.02.00



Version 3.01.01 (August 8, 2007)

-  Added ability to keep time on a per-session basis. Added pSession to tmwdtime structure; target layer can use this parameter to return a time specific to the session.
-  *Description:* Several sessions not responding could cause link layer polling to stop polling other sessions.
Introduced: v3.00
Resolution: Fixed in v3.03




Version 3.01.00 (May 23, 2007)

-  Modified online notification algorithm to facilitate notification of sessions that never go online.




Version 3.00.49 (November 1, 2006)

-  Renamed TMWCHNL_STAT_CALLBACK and TMWSESN_STAT_CALLBACK macros to TMWCHNL_STAT_CALLBACK_FUNC and TMWSESN_STAT_CALLBACK_FUNC. This change avoids having a macro and a typedef with the same name.
-  *Description:* The Source Code Library could delay for up to *class2PendingDelay*, when it could have sent the next poll immediately.
Introduced: v3.00
Resolution: Fixed in v3.00.49






Version 3.00.46 (April 27, 2006)

-  Modified release to include WinIoTarg.dll built with Visual Studio 6. This DLL is used when using Visual Studio to build the sample applications.
-  Provided a method to cancel the physical layer if a command is canceled at a higher layer.
-  Updated sample Makefiles to correctly build Source Code Library.

Version 3.00.45 (March 17, 2006)

-  Added support for Areva private ASDUs.
-  Added support for Siemens SIPROTEC private ASDU 205.
-  Added channel call to error call in i870dia1_linkAddressUnknown.




Version 3.00.44 (February 2, 2006)

-  Added a sample application to provide an example of a multi-threaded, event-driven application.
-  Set receive indication in diagnostic information to indicate the direction of the message.
-  Enhanced statistics to include additional values such as CRC errors, number of invalid start bytes, frame length errors, etc.
-  Split into tmwtarg.h/c into tmwtarg.h/c and tmwtargp.c/c. This change moves all Triangle MicroWorks, Inc. specific code into a separate file. Customers only need to modify tmwtarg.c.
-  *Description:* Some diagnostic levels were displayed incorrectly.
Introduced: v2.00
Resolution: Fixed in v3.00.44

Version 3.00.43 (December 2, 2005)

No updates to IEC 60870-5-103 Master Source Code Library in this release.






Version 3.00.42 (September 12, 2005)

-  Improved documentation by updating *IEC 60870-5-103 Master.doc* to improve readability and include all user-modifiable files.
-  Added subversion to SCL filename (e.g., m103v30042.exe). This allows the version number to be determined from the file name. The version number continues to be defined in tmwvrsn.h.
-  *Description:* The Source Code Library could potentially reference a Null pointer in tmwdlist_removeEntry
Introduced: v3.00
Resolution: Fixed in v3.00.42






Version 3.00.41 (July 20, 2005)

No updates to IEC 60870-5-103 Master Source Code Library in this release.


Version 3.00.40 (June 3, 2005)

-  Added 103Master sample application executables and required DLLs to allow running this application without having to build it.
-  Improved documentation of memory type configuration.
-  Fixed compiler warning for TMWTYPES_BOOL type.
-  *Description:* Fixed frame class polls did not retry.
Introduced: v3.00
Resolution: Fixed in v3.00.40
-  *Description:* TMWMEM_FREE_FUNC was doubly defined.
Introduced: v3.00.39
Resolution: Fixed in v3.00.40

Version 3.00.39 (March 29, 2005)

-  Improved memory management within Source Code Library. Added new #define (TMWCNFG_ALLOC_ONLY_AT_STARTUP) to tmwcnfg.h. Also added new functions that the target to set the maximum number of each buffer in each pool at runtime. Both “only at startup” and static allocations use a linked list of free and allocated buffers instead of just an array that was accessed sequentially.
-  Added description and example of custom ASDU support to User Manual.
-  *Description:* Disabling one session of a multi-drop channel stopped polling for other sessions.
Introduced: v3.00
Resolution: Fixed in v3.00.39
-  *Description:* The Source Code Library did not properly handle an invalid response to a “Reset Frame Count Bit” message.
Introduced: v3.00
Resolution: Fixed in v3.00.39
-  *Description:* s3sim_msrndAddPoint() and s3sim_msrn2AddPoint() used incorrect data types.
Introduced: v3.00
Resolution: Fixed in v3.00.39

Version 3.00.38 (January 18, 2005)

-  Modified the API for the channel, session, sector modify functions to initialize a structure with the current settings. Any of these settings may then be modified by the target code.






Version 3.00.37 (December 15, 2004)

No updates to IEC 60870-5-103 Master Source Code Library in this release.



Version 3.00.36 (November 2, 2004)

No updates to IEC 60870-5-103 Master Source Code Library in this release.

Version 3.00.35 (September 22, 2004)

-  Added #define TMWCNFG_MEMORY_ALIGN_NEEDED to tmwcnfg.h to support processors requiring long word (4 byte) alignment and compilers that create unpacked structures.
-  Added m3data_distribEndOfTransStore() function to indicate when signal when the ASDU 31 end of transmission of disturbance data is received.
-  Added Tag Position (Tap) database interface (m3data_distribTagStore).
-  Added additional diagnostics for disturbance data.
-  Added tmwapp_startTimer and tmwapp_cancelTimer to tmwtarg.c.
NOTE: These functions must be defined in the target in order for the Source Code Library to link.

Version 3.00.34 (July 30, 2004)

-  *Description:* The Source Code Library could call tmwtarg_startTimer with a negative value. For some operating systems, this causes timers to stop running.
Introduced: v3.00.32
Resolution: Fixed in v3.00.34
-  *Description:* The algorithm for sorting timers did not work properly if the two times being compared were more than 17 days apart.
Introduced: v3.00
Resolution: Fixed in v3.00.34



Description: The simulated database did not check for an error return from tmwsim_newPoint, potentially causing the sample application to crash.

Introduced: v3.00

Resolution: Fixed in v3.00.34



Description: i870util.h was sometimes included as l870util.h, causing compilation problems with some compilers.

Introduced: v3.00

Resolution: Fixed in v3.00.34

Version 3.00.33 (June 17, 2004)



Added ability to queue requests with duplicate ASDU types. Instead of failing the request, the Source Code Library now queues the new request on a pendingMsgQueue. If the request is a duplicate, the request already on the pendingMsgQueue is canceled.

Version 3.00.32 (June 4, 2004)



Enhanced support for multiple threads. Added support for one timer queue per channel. Also improved locking/unlocking to prevent threads from being interrupted or deadlocked. This enhancement required two changes to the API:

- Added new #define(TMWCNFG_MULTIPLE_TIMER_QS) to allow support of multiple timer queues (default is TMWDEFS_FALSE)
- Added new parameter to tmwdb_storeEntry() to indicate whether the queue should be locked (default is not to lock the queue).

Any implementation using multiple threads should upgrade to this release



Improved detection of sessions going offline. The session is no longer marked offline if an application request times out.



Added a session-level statistic to indicate that an application layer response timed out.








Modified m103msrnd and m103msrnd2 to retrieve the value portion separately from the invalid flag and overflow flag.





Description: The _timerCallback function in tmwtimer.c did not always lock the proper channel.

Introduced: v3.00


Resolution: Fixed in v3.00.32


-  *Description:* Modifying a session to set “active” to TMWDEFS_TRUE did not make the session active.
Introduced: v3.00
Resolution: Fixed in v3.00.32
-  *Description:* The Source Code Library did not calculate the first byte time correctly for variable and fixed length frames.
Introduced: v3.00
Resolution: Fixed in v3.00.32
-  *Description:* The link layer could continue polling on the same session while a request was pending.
Introduced: v3.00.27
Resolution: Fixed in v3.00.32
-  *Description:* tmwappl_initApplication() could not be called more than once.
Introduced: v3.00.29
Resolution: Fixed in v3.00.32
-  *Description:* A 31-day timer would expire immediately.
Introduced: v3.00
Resolution: Fixed in v3.00.32

Version 3.00.29 (March 31, 2004)

-  *Description:* The Source Code Library did not calculate times correctly
Introduced: v3.00.27
Resolution: Fixed in v3.00.29
-  *Description:* When configured for Static memory support, the Source Code Library could fail to allocate buffers if they contained non-zero data on startup.
Introduced: v3.00.23
Resolution: Fixed in v3.00.29

Version 3.00.28 (March 2, 2004)




-  *Description:* The Source Code Library could attempt to access a Null pointer if the Link Address Size was configured to 0.
Introduced: v3.00
Resolution: Fixed in v3.00.28

-  *Description:* If the Source Code Library was configured to use an asynchronous database, it could store Identity Elements instead of Data Sets.
Introduced: v3.00
Resolution: Fixed in v3.00.28

Version 3.00.27 (February 3, 2004)

No updates to IEC 60870-5-103 Master Source Code Library in this release.

Version 3.00.26 (January 15, 2004)

-  *Description:* The Source Code Library could stop polling devices if a Slave answered with NO DATA on the wrong session.
Introduced: v3.00
Resolution: Fixed in v3.00.26
-  *Description:* The Source Code Library gave priority to polling sessions that were ready over sending out the RESET FCB. This could prevent some devices from coming on line when a large number of sessions was configured.
Introduced: v3.00
Resolution: Fixed in v3.00.26
-  *Description:* Closing and reopening a session could cause the system to stop polling.
Introduced: v3.00
Resolution: Fixed in v3.00.26




Version 3.00.25 (December 12, 2003)

No updates to IEC 60870-5-103 Master Source Code Library in this release.





Version 3.00.24 (November 10, 2003)

No updates to IEC 60870-5-103 Master Source Code Library in this release.



Version 3.00.23 (October 31, 2003)

-  Added locks to tmwmem.c to add support for multiple threads to memory management routines.
-  Provided optimizations when diagnostics are not compiled in.
-  *Description:* The Source Code Library did not deallocate the txData structure when using a serial link and the confirm was not received.
Introduced: v3.00
Resolution: v3.00.23





Version 3.00.22 (October 8, 2003)

-  The Source Code Library no longer generates an error when it receives a message it sent. This change facilitates use on a fiber optic ring.
-  Improved detection of sessions going offline. Any application layer error will now take a session offline.
-  Added pCheckAddrCallbackFunc function. This function facilitates the use of modem pools with the Source Code Library.
-  Added support for ASDU 123 to support Procomoe specification.


Version 3.00.21 (September 22, 2003)

-  Added support for ASDU 123 for Procome Specification.
-  Added interface to disable automatic Class polling and to send Class1 or Class2 poll. Also modified FT1.2 link layer to be tolerant of not receiving a response to a Class1 or Class2 poll.

Version 3.00.20 (September 2, 2003)

-  Added callback function to let target application know when a channel is idle and can be disconnected.
-  An evaluation version of a working sample of a Windows application is now shipped with the Source Code Library. The evaluation expires after running for two hours.
-  *Description:* The Source Code Library could make repeated recursive calls when configured for balanced mode while initializing a session. This could cause an application to appear to “freeze.”
Introduced: v3.00
Resolution: Fixed in v3.00.19
-  *Description:* If TMWCNFG_SUPPORT_DOUBLE wasn't supported, compilation issues would occur.
Introduced: v3.00
Resolution: Fixed in v3.00.20








Version 3.00.19 (August 8, 2003)

-  *Description:* The Source Code Library could call tmwtarg_startTimer even if the timer was already running. This issue only affected event driven implementations (i.e., implementations that did not use the polled timer implementation).
Introduced: v3.00
Resolution: Fixed in v3.00.19




Version 3.00.18 (July 22, 2003)

No updates to IEC 60870-5-103 Master Source Code Library in this release.

Version 3.00.17 (July 10, 2003)

-  Added link layer configuration parameters (LinkClass1PendingDly, LinkClass1PollCnt, LinkClass1PollDly, LinkClass2PendingDly, LinkClass2PollDly, and LinkClassPendingCnt) for consistency with v2.x Source Code Library. These parameters allow tuning the unbalanced link polling behavior in restricted throughput data links such as the satellite (low throughput and long delays) and multi-point radio (channel sharing and high Tx/Rx changeover times).
-  Implemented m103brm functions to provide reset user link layer function code in FT1.2.
-  Improved implementation of internal data passed to callback functions to return a well defined structure and status codes.
-  Consolidated M14SESN_RESPONSE_INFO, M103SESN_RESPONSE_INFO etc. into a single structure (I870CHNL_RESPONSE_INFO).
-  Added TMWTARG_CONFIG structure to allow specification of parameters that are common to multiple protocols.
-  *Description:* When multiple sessions were offline, the Source Code Library could fail to communicate with online sessions.
Introduced: v3.00
Resolution: Fixed in v3.00.17
-  *Description:* If a command required multiple responses, the reference returned by the m103brm_xxx callback functions was not the original reference.
Introduced: v3.00
Resolution: Fixed in v3.00.17

Version 3.00.16 (June 19, 2003)

-  Added limit checks for sprintf statements. This prevents buffer corruption caused by writing past the end of the current buffer.
-  Modified Source Code Library to avoid string copies for diagnostics when the diagnostics aren't enabled.
-  Restructured Source Code Library to allow Master and Slave sessions on a channel simultaneously to support Peer or Dual mode. Removed p10x files.



Added current version number to *tmwvrsn.c/h* in the *utils* directory to simplify determining which version of the Source Code Library is in use.



Improved Protocol Analyzer display to use terminology from standards documents; now display "SQ" instead of "SEQ" for Single/Sequence bit.



Cleaned up compiler warnings and issues identified by *lint*.



Description: The Source Code Library could crash when processing Type 2 data with the Invalid flag set.

Introduced: v3.00

Resolution: Fixed in v3.00.16



Description: When compiled with static memory, M103MEM_SIM_DISTRB_CHNL_TYPE and M103MEM_SIM_GNRC_DATA_TYPE inadvertently used the same memory array

Introduced: v3.00

Resolution: Fixed in v3.00.16

Version 3.00.15 (May 27, 2003)

No updates to IEC 60870-5-103 Master Source Code Library in this release.

Version 3.00.14 (May 9, 2003)



Improved detection of sessions going offline. Any application layer error will now take a session offline. Receiving any valid fragment from that session will take it back online.



Improved memory management. Implemented a new memory management scheme that supports static memory as well as providing limits on the number of each type of structure that can be allocated. It also provides details on how many structures of each type have been allocated and keeps track of the maximum number that was allocated. This provides feedback that can be used in sizing applications and monitoring allocated memory. Cleaned several areas in which memory was not being freed properly.

Version 3.00.12 (April 28, 2003):





Added Incremental Application Layer Response Timeout configuration parameter.




The Incremental Timeout timer is now disabled if the timeout value is set to zero

Version 3.00.11 (April 24, 2003):


-  If a connection is broken or the remote device restarts, the Source Code Library will attempt to reopen the connection.


-  Added support for custom response ASDU types.

Version 3.00 (April 28, 2003):

-  The Triangle MicroWorks, Inc. Source Code architecture was redesigned and reimplemented to follow modern software engineering practices and leverage new techniques in software design. We have also incorporated suggestions from our existing customer base. The main advantages to this redesign include:
 - 1) Common source code architecture across all TMW libraries - The use of a common architecture across all libraries significantly reduces time required to port additional protocols to a target device. In addition, using common software across all products results in better code, since common functions are utilized much more frequently than protocol specific functions.
 - 2) Source Code Library calling routines are now compatible with a wider variety of event driven techniques, allowing the target application to be achieve higher performance.
 - 3) Most configuration parameters are now passed as arguments in Source Code Library function calls instead of macros. This allows for data hiding and protects configuration parameters from accidentally being changed while the Source Code Library is running. In addition, macro calls to target hardware and the database interface routines were replaced with function calls.
 - 4) Significant reduction in the time to implement new features - In addition to being more flexible, the use of modern software design practices significantly reduces the time required for Triangle MicroWorks to support new features. As the existing protocols are constantly being improved by the associated Technical Committees and Working Groups, it is essential that the Source Code Libraries are able to remain up to date with the latest standards.

Version 2.14 (July 10, 2002):

-  Removed `M103CNFG_LINK_SUPPORT_BALANCED()` and `M103CNFG_LINK_SUPPORT_UNBALANCED()` from `m103cnfg.h`. Both are supported at all times now.

-  Fixed a bug with broadcast messages which contained a time stamp. The time stamp was added at the time the message was created, not when the message was transmitted through callbacks. Now the timestamp is added at the time of transmission and is more accurate.



Fixed bug with the reset command which was attempting to send the reset command as a variable frame message rather than a fixed frame message.

Version 2.12 (January 31, 2001):

103 Added functionality to immediately try to reestablish a connection to devices that are offline when a new application level request is issued to that device. This only effects unbalanced communications channels.



Changed argument for **M03DATA_NOTIFY_TX/RX_FRAME** macro from comIndex to sessionIndex.



Added functionality to cancel link level requests if the application level times out. The link layer should not continue to try and send a request if the application layer has timed out.



Fixed unbalanced/balanced mode only compilation issues.

Version 2.11 (December 11, 2001):



Fixed a bug that was keeping the DIR bit from being set correctly when operating in balanced mode. IEC 60870-5-103 does not officially support balanced mode but the Triangle MicroWorks Inc. M103 Source Code Library will support it if configured to do so.



The **M103CNFG_LINK_CLASS1/2_POLL_DLY** parameters for a given channel where being applied in sequence as opposed to independently. Specifically, if two sessions were assigned to a single channel each session would be polled at a rate equal to the sum of the polling periods for each session as opposed to each session's specific polling period. This is now fixed so that each session is polled at its configured polling period independent of other sessions on that channel.



Fixed a bug that caused an unbalanced master to delay polling a slave for the response to a request until the end of the current polling period. The master now immediately starts polling a session with a request pending at the rate determined by the **M103CNFG_LINK_CLASS1/2_PENDING_DLY** when a request is sent.

Version 2.08 (September 4, 2001):














103 Added **M103TARG_MALLOC()** and **M103TARG_FREE()** macros for dynamic memory allocation. Also added **M103TARG_MEMCPY()** to allow copying one location of memory to another. Changed all **memcpy** references to use **M103TARG_MEMCPY()** macro so end users can specify their own implementation of **memcpy**. **M103CNFG_OFFLINE_POLL_DLY()** to specify how often to poll a slave device if it is detected to be offline.



Added **I870FT12_NUM_IN_LENGTH** and **I870FT12_ASDU_MAX_SIZE**.



Renamed **M103CNFG_DIAG_FEATURE_MASK()** to **M103TARG_DIAG_FEATURE_MASK()** so that initial phases of installation will not require editing of **m103cnfg.h**.

-  Removed the following modules: `m103cmrw`, `m103cmli`, `m103dvrs`, `m103poll` and `m103link`. Added the following modules: `tmwclraw`, `tmwclprm`, `i870anl1`, `tmwanl1zr` and `tmwdblnk`. Added the following files: `i870scl.h` and `tmwstat.h`.
-  Renamed `m103cmts_parseValue()` function to `tmwclprm_parseValue()` and moved it from `m103cmts.c` to `tmwclprm.c`. Renamed `m103dvrs_collectComData()` to `m103task_checkAllChannels()` and moved it from `m103dvrs.c` to `m103task.c`. Renamed `tmwphys_initChannel()` to `tmwphys_openChannel()`.
-  Added check to make sure information object address is within range limited by field size.
-  Database storage functions now return a status code that is eventually returned by `m103db3_processDbas()` and then to the target application by `m103task_main()`.
-  Changed from using `M103SCL_STAT_xxxx` to `TMWSTAT_xxxx` codes and moved them to `tmwstat.h` so they can be shared across all TMW source code libraries. Renamed several `TMWSTAT_xxxx` codes, and assigned them specific values that can be documented more easily, and that will allow future codes to be added without affecting existing values. Also added new codes `TMWSTAT_LINK_UNEXPECTED_ACK` and `TMWSTAT_LINK_REQUEST_CANCELLED`.
-  Function `m103task_processFrames()` is no longer called by `m103task_main()` and can be called directly by the target application. `m103task_main()` now returns `TMWSTAT` error codes.
-  Removed public access to `m870task_libraryTimers()`.
-  Renamed `M103LINK_FLAGS_xxxx` from `m103link.h` to `m103task.h` and renamed them `M103TASK_TX_FLAGS_xxxx`.
-  Made `i870lnk1_primaryEngine()`, `i870lnk1_parseChar()` and `i870lnk1_getNeededChars()` static (local), and added `i870lnk1_checkForFrames()` which is called from the main application task and in turn calls `tmwphys_checkForReceive()`. This hides the physical layer and link layer parsing routines from the main application task.
-  Modules `i870dtim.c`, `tmwintel.c`, `tmwprntf.c`, `tmwscript.c`, `tmwsftmr.c`, `tmwclprm.c` and `tmwclraw.c` now use `tmwomni.c` module to allow them to be universal (identical for all source code libraries). When multiple source code libraries are installed, now searches through installed protocols to uniquely qualify communication channel indices and session indices.
-  Added `SESSION_NOTIFY_ONLINE` method to support target notification of online/offline events.
-  Added prototype for new `m103appl_init()` function in `m103appl.h` module.
-  Added `#if` statements in `simdata.c` to prevent errors when diagnostic features are disabled. Introduced in initial release.



Added `pTransmitStartAsdu` to `M103APPL_TRANS_DATA` for use in callback functions that store time immediately before transmission. This fixed problem in version 2.05 that did not store time correctly in such messages.



Fixed compile time errors when `M103CNFG_LINK_SUPPORT_UNBALANCD()` is set false or when `M103TARG_DIAG_FEATURE_MASK()` is set to zero. Introduced in version 2.04.



Corrected bugs with running multiple sessions and sectors on the same communication channel. Introduced in version 2.04.

Version 2.05 (January 24, 2001):



The `m103db3_processQue()` function is no longer required to be called by Target Application software. This function is now called by the Source Code Library through `m103task_main()` based on a timer configured by `M103CNFG_PROCESS_DBAS_PRD()`. However, if desired, the function, which has been renamed to `m103db3_processDbas()`, can be called by the Target Application (perhaps within a separate task) if `M103CNFG_PROCESS_DBAS_PRD()` is set to zero.



Changed from using `M103CNFG_TASK_RATIO` to using a timer configured by `M103CNFG_PROCESS_FRAMES_PRD()`.



The `m103task_main()` function now returns a `M103SCL_STAT` status code.



Renamed several `M103SCL_STAT` codes, and assigned them specific values that can be documented more easily, and that will allow future codes to be added without affecting existing values.



Moved and renamed `M103CNFG_DIAG_FEATURE_MASK()` to `M103TARG_DIAG_FEATURE_MASK()` to further consolidate target-hardware-specific interface parameters.



For the optional command line interface, the following changes were made:

- Renamed M103tscm module to M103cmts.
- Added new M103cmrw and M103cmli modules.
- Added new command to initiate a general command request message.
- Wildcards may now be used on the command line to specify, when appropriate, any valid channel, any valid link address, or any valid ASDU address.
- Default values for channel, link address, and ASDU address pre-determined if only one valid value exists.
- Script commands may now be executed without preceding the script name with "script".



Enhanced protocol analyzer display to:

- Show the cause of transmission value along with a text meaning.
- Display errors when invalid parameters are specified to build request functions.



Added more `#if` statements to prevent errors when diagnostic features are disabled.

Version 2.04 (December 4, 2000):

- 103 Now, depending upon the value of `M103CNFG_APPL_AUTO_MODE_MASK()`, clock synchronization and general interrogation commands may be automatically initiated when the end-of-init ASDU is received from remote devices.
- 103 The link-layer polling algorithm is now configurable to support a maximum number of consecutive polls to a single device as well as a configurable delay after each poll. This can be used to avoid excessive load on the target processor or communication media.
- 103 Added `M103CNFG_SESSION_POLL_ENBLD()` so end user can take remote device offline (no polling for class 1 or 2 data).
- 103 Changed application layer confirmation timeout to be measured from the absolute time the message was transmitted instead of restarted when a non-cyclic data message is received.
- 103 Added macro for `M103DATA_SESSION_NOTIFY_ONLINE()` when communications are lost or restored so that points in Target Application database can be marked accordingly. Replaced performance statistics counters with calls of other `M103DATA_XXXX_NOTIFY_XXXX()` macros.
- 103 The ability of `M103DATA` disturbance macros to request processing delays by returning `M103SCL_DISTURB_TRY_AGAIN_LATER` is now fully supported.
- 103 Changed duplicate request checking to only check the application layer portion of messages (not the link layer headers).
- 103 Automatic link resets (initiated upon confirmation failures) for 103 sessions now use `I870FT12_FC_LINK_RESET_FCB` (link function code 7) instead of `I870FT12_FC_LINK_RESET_CU` (link function code 0).

New files:



Added several optional modules that are used to provide a simple command line user-interface. If not needed, this feature can be disabled, and these optional modules can be removed.




Added `M103cfts` module, which contains code to check the consistency of configuration parameters.

TA-to-SCL interface changes:




With a single function call, request messages can now be configured to be periodic. For example a single function call can initiate a periodic general interrogation request.

 Added `m103brm_send103ResetComm()` function to initiate a request to reset the communications in a remote Slave device.

 Added `m103brm_stopRequest()` function that is primarily used to stop a request that is transmitting periodically.

Target Hardware Interface changes:


 Now, communication channels are only initialized when sessions that use them are initialized (they are no longer all initialized at startup).

 Added more parameters to `M103TARG_COMM_RECEIVE()` to allow more efficient implementations of the receive operation in the future.


Database Interface changes:


 Removed `pDelayPrd` from some `M103DATA` macros because it was not used.

General changes

 Made the following changes to the protocol analyzer:

- Enhanced display of disturbance data in protocol analyzer.
- Display communication channel name rather than channel index.

 Renamed several `m103brm`, `M103CNFG`, and `M103DATA` symbols for readability and consistency.

 To prevent warnings emitted by specific strict compilers, rearranged code, changed some data types, added example `#pragma`'s, initialized local variables, and added type-casts.

Version 2.01 (March 20, 2000):

103 Added support for disturbance data. More specifically, added the following database macros and build-request-message entry points to process data received, and to request disturbance data:

<code>M103DATA_NOTIFY_DISTURB_LIST()</code>	Processes ASDU type 23
<code>M103DATA_NOTIFY_DISTURB_DATA()</code>	Processes ASDU type 26
<code>M103DATA_NOTIFY_DISTURB_TAGS()</code>	Processes ASDU type 28
<code>M103DATA_NOTIFY_DISTURB_CHANNEL()</code>	Processes ASDU type 27
<code>M103DATA_NOTIFY_DISTURB_END()</code>	Processes ASDU type 31
<code>M103DATA_STOR_DISTURB_TAG()</code>	Processes ASDU type 29
<code>M103DATA_STOR_DISTURB_VALUE()</code>	Processes ASDU type 30
<code>m103brm_send103DisturbReqList()</code>	Public API function to request list of recorded disturbances
<code>m103brm_send103DisturbOrder()</code>	Internal function that sends ASDU type 24
<code>m103brm_send103DisturbAck()</code>	Internal function that sends ASDU type 25

103 The 60870-5-103 protocol does not require link status before link reset, so **performLinkStartup()** was modified to (only) send link reset.



For disturbance data, added several new definitions to m103def3.h and m103scl.h, including:

M103DEF3_ACC	Actual channel
M103DEF3_SOF	Status of fault
M103DEF3_TOO	Type of order
M103DEF3_TOV	Type of values
M103SCL_DISTURB_LIST_ACTION	Enumerated return value when database processing list of recorded disturbances
M103SCL_DISTURB_ACTION	Enumerated return value when database processing other disturbance data notifications and storage
M103SCL_DISTURB_END	Enumerated return value when processing end of transmission of disturbance data
M103SCL_DISTURB_FAULT_DESC	Structure that describes a single “member” of the list of disturbances
M103SCL_DISTURB_DESC	Structure used to describe disturbance data used in both transmission of requests and in responses
M103SCL_DISTURB_DATA_DESC	Structure used to describe a disturbance data set for a specific fault
M103SCL_DISTURB_CHANNEL_DESC	Structure used to describe disturbance data for a specific channel within a disturbance data set.
M103SCL_DISTURB_TAG	Structure used to describe a single disturbance tag value
M103SCL_DISTURB_VALUE	Structure used to describe a single disturbance channel value.



Changed **m103dvrs_lockwait()** and **m103dvrs_unlock()** to target macros **M103TARG_RESOURCE_LOCK()** and **M103TARG_RESOURCE_UNLOCK()** and provided more in-source documentation, with examples, for their implementation. Added **M103SCL_RESOURCE_XXXX** enumerated type to replace **M103DVRS_RESOURCE_XXXX** definitions.



Moved **SIMDATA_DBAS_SECTOR_LAST_POINT()** to inside simdata.c since it is used only as part of the specific (example) database design. Final database design implementations may not use such a parameter.



Fixed a memory leak introduced in version 2.00. The problem was that some transmit buffers were not always being free'd.

Version 2.00 (February 22, 2000):



Initial release of the IEC 60870-5-103 Master Source Code Library.