TEMUX336 (PM8310)

THE Complete Driver/OAM Source Code Solution

...and it's ready for LINUX

NComm's Trunk Management Software (TMS) OAM source code provides a complete solution for the implementation of a T1/E1, T3/E3 and SONET/SDH WAN interface including Automatic Protection Switching systems using the PM8310. TMS consists of a set of software modules that perform all aspects of Layer One OAM (Operations, Administration & Maintenance) support from span alarming and maintenance to channelized signaling processing. Additionally, it is completely data driven allowing operating mode, alarm timers, and thresholds to be configurable on a static or run time basis. The suite includes 2 levels of ANSI-C Application Programming Interfaces (APIs) that encapsulate the details of the PM8310's operation and the underlying hardware elements, and provide a clean integration to the target system's operating environment.

TMS is modularized so that the software can be tailored to the exact needs of the project. For instance, while the PM8310 implements T1/E1, T3/E3 and SONET/SDH technologies, if only T1 and SONET are used then only those modules are required. Of course, all modules are coordinated within the transceiver, creating a seamless result.

The Driver

NComm drivers provide the perfect interface needed to properly design WAN framer devices into your product. Written the way drivers are supposed to be written, that is from a developer's perspective, they provide a clean partition between the device and the application. Well-defined, consistent APIs are used that are identical for similar devices. Thus, all T1 driver APIs, for example the TEMUX 336 and the COMET, will be the same.

NComm Driver Architecture

All NComm drivers follow a three-function architecture. Utilizing these functions, you can control, monitor and manage all activity within the device. These functions are Control, Poll and Call-Back. All capabilities are truly function oriented, not register oriented. Thus, a single function call may perform the setting or reading of several bits. The Control function is used to actually change some attribute of the device such as setting line format, setting alarm integration times, etc. The Poll function allows you to ask for status information without affecting the behavior of the device. The Call-Back function is used in situations where the device needs to inform you of some important event that has occurred.

Reuse and Transportability

Following a standardized, consistent API, the driver accommodates all device nuances and reduces them to hardware independent functions. Thus, should you want to move from one framer to another, you just need to replace the driver and the rest of your code stays the same. Being functionally oriented, the driver also assures configuration integrity and provides error feedback in the case of function call failures. NComm drivers implement work arounds for device issues if required and possible. They handle multiple instances of devices, are non-algorithmic, handle chattering and run in interrupt mode if possible. Special device functions are addressed through #defines. All drivers for the PM8310 are fully tested with TMS on PM8310 hardware.

Trunk Management Software (TMS)

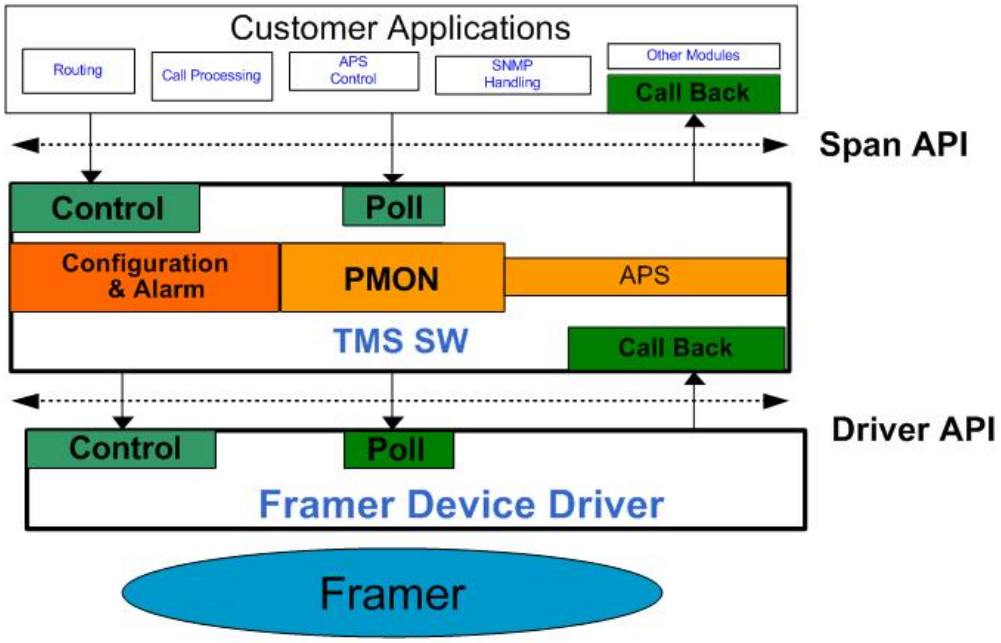

Figure 1 - TMS Layered Software

The figure above illustrates the TMS software module architecture. Above the driver API, requirements and implementation of required OAM functions are the same regardless of the underlying framer. OAM software provides non-device-specific processing and routines for the interface. While the data payload is being sent to higher layer applications, the management system performs alarm management and performance monitoring functions.

Each software component performs specific operations that represent a part of the total OAM software solution. The TMS API is comprised of a set of ANSI C functions and macros that encapsulate all functionality and data of the TMS Software. The API provides a clean interface simplifying the integration of the software to the target application. The target application is implemented above of the TMS Software API layer, using the API to access all functionality.

Alarm Management

Alarm management handles creating alarms from defects. Defects are immediate indicators of the condition of the lines. When defects like AIS-L/MS-AIS occur, they are reported up from the framer, and a timer is started. If the condition persists for 2.5 seconds, an AIS alarm is declared, AIS is propagated to the downstream entities, and an RDI-L/MS-RDI is sent back to the equipment on the other end.

PMON

Performance Monitoring (PMON) is proactive, collecting and time-stamping performance reports and other data every second. PMON data is analyzed to detect deteriorating conditions prior to service interruption, but should a hard failure occur, this information, along with loop-backs, can be used to isolate the source of the problem. Threshold Crossing Alerts (TCAs) can also be set to trigger if something exceeds an expected range.

Robbed Bit and Channel Associated Signaling

The optional Robbed Bit (RBS) and Channel Associated Signaling (CAS) modules process the robbed bit/CAS signaling information on the channels within a T1/E1 line. The signaling bits are processed according to one of several signaling modes (e.g. E+M, FXO, FXS, LS, GS, GR-303 Hybrid Signaling and E1 signaling models as well as those defined in Q.421 & Q.422.) on a channel-by-channel basis. Debouncing and Bit Freezing are also provided. Additionally, the modules allow custom signaling protocols to be defined and processed as well. For applications that do not require signaling bit processing, signaling functionality can be disabled on each timeslot.

For more information or to discuss your current project, please contact your PMC-Sierra representative or NComm:

603-893-6186 x 60
jsb@ncomm.com
www.ncomm.com

NComm's Communication Developers Handbook may also be obtained free for the asking at NComm's web page (www.ncomm.com).