ATCA-F125 Telco Clocking Module

Reference Guide

P/N: 6806800T64B

February 2020

SMART™
Embedded Computing

Trademarks

The stylized "S" and "SMART" is a registered trademark of SMART Modular Technologies, Inc. and "SMART Embedded Computing" and the SMART Embedded Computing logo are trademarks of SMART Modular Technologies, Inc. All other names and logos referred to are trade names, trademarks, or registered trademarks of their respective owners. These materials are provided by SMART Embedded Computing as a service to its customers and may be used for informational purposes only.

Disclaimer*

SMART Embedded Computing (SMART EC) assumes no responsibility for errors or omissions in these materials. **These materials are provided "AS IS" without warranty of any kind, either expressed or implied, including but not limited to, the implied warranties of merchantability, fitness for a particular purpose, or non-infringement.** SMART EC further does not warrant the accuracy or completeness of the information, text, graphics, links or other items contained within these materials. SMART EC shall not be liable for any special, indirect, incidental, or consequential damages, including without limitation, lost revenues or lost profits, which may result from the use of these materials. SMART EC may make changes to these materials, or to the products described therein, at any time without notice. SMART EC makes no commitment to update the information contained within these materials.

Electronic versions of this material may be read online, downloaded for personal use, or referenced in another document as a URL to a SMART EC website. The text itself may not be published commercially in print or electronic form, edited, translated, or otherwise altered without the permission of SMART EC.

It is possible that this publication may contain reference to or information about SMART EC products, programming, or services that are not available in your country. Such references or information must not be construed to mean that SMART EC intends to announce such SMART EC products, programming, or services in your country.

Limited and Restricted Rights Legend

If the documentation contained herein is supplied, directly or indirectly, to the U.S. Government, the following notice shall apply unless otherwise agreed to in writing by SMART Embedded Computing.

Use, duplication, or disclosure by the Government is subject to restrictions as set forth in subparagraph (b)(3) of the Rights in Technical Data clause at DFARS 252.227-7013 (Nov. 1995) and of the Rights in Noncommercial Computer Software and Documentation clause at DFARS 252.227-7014 (Jun. 1995).

SMART Embedded Computing, Inc.

2900 S. Diablo Way, Suite 190

Tempe, Arizona 85282

USA

*For full legal terms and conditions, visit *www.smartembedded.com/ec/legal*

Table of Contents

Table of Contents

Table of Contents

List of Tables

Telco Clocking Module

1.1 Introduction

The Artesyn ATCA-F125 Telco Clocking Module (TCM) comprises the hardware, software, and firmware specifically supporting the telecom clocking features of the ATCA-F125 blade. This section presents the software and firmware components of the TCM in detail. The TCM hardware is presented only in an overview. Refer to *the ATCA-F125 Installation and Use Guide* for details concerning the telecom clocking subsystem hardware.

1.2 Clocking Overview

A reader already possessing a basic familiarity with the ACS9510 as well as Telco clocking terminology and concepts can skip this section.

1.2.1 Clocks

A clock is a signal of some form used to synchronize a process. For example, glancing at a wall clock for the time of day might inform you when you need to leave to attend a meeting on time. Not all clocks need to provide the time of day to be useful. For instance, a clock that maintains a certain frequency, but contains no time of day information, may be sufficient to ensure that a data packet send/receive process does not exceed the capacity of overflow buffers. Other processes may not just need to be at the same frequency, but also depend on being "in phase". That is, the tasks must be performed in lock step. Imagine two cooperating robotic arms that go through a pattern of movements and have an overlapping reach. The operations of the two arms should be in phase to avoid collisions. Synchronization that is solely based on frequency without reference to phase or time of day information is known as "syntonization".

There are two major types of Telco Clocking technology:

1. Analog Time Division Multiplex (TDM). Clocks of this type provide synchronization information by oscillating between states on a regular basis.
2. Packet-based technologies such as Network Time Protocol (NTP), Global Positioning System (GPS), and IEEE1588v2 Precision Time Protocol (PTP). Clocks of this type provide synchronization information via data packets containing timing information.

TDM clocks are the traditional technology and are widely used. A down-side of TDM clocks is that they require specialized and often expensive equipment for transport and maintenance as they flow end-to-end in a system. On the other hand, packet-based technologies can reuse the packet interchange grids that already cover large parts of the globe. Specialized equipment is only needed for the original generation and final capture of the packet-based clock plus possibly to convert the clock from or to a TDM clock.

1.2.2 Reference and System Clocks

Commonly TDM Clocks are classified as either being Reference Clocks or System Clocks. These terms originated for TDM technology, but are applicable for PTP as well. A System Clock is the output of a Clock Generator Module (CGM). System Clocks are used directly to synchronize processes. Typically, a CGM can simultaneously generate multiple System Clocks with various frequencies or other characteristics.

A Reference Clock is a clock used to influence the characteristics of one or more System Clocks. Reference Clocks indirectly synchronize processes by guiding the System Clocks. A single Reference Clock can be used to simultaneously produce multiple System Clocks. When a Reference Clock is being used to guide the production of the System Clocks, the CGM is said to be locked to the Reference Clock.

To avoid single-point failures, multiple Reference Clocks should be available. A typical user task is to prioritize the selection among the available Reference Clocks in case of failure.

1.2.3 Clock Strata

Clock Stratum Levels are international clock quality standards defined by the American National Standards Institute (ANSI) and Bell Communications Research (Bellcore) that apply to TDM clocks. Other clock quality standards not presented in this document apply to packet-based clocks. The Stratum Level determines certain performance characteristics of the System Clocks generated by the CGM. The higher Strata Levels are harder or more expensive to maintain. The Strata Levels from highest to lowest quality are:

- Stratum 1
- Stratum 2
- Stratum 3E
- Stratum 3
- Stratum 4E
- Stratum 4

As clocking signals transition from system to system the typical flow is from higher to same or lower strata. Clocks may often get "cleaned up" during their trip in order to maintain their overall quality. Quality standards which are harder or more expensive to maintain such as "wander" tend to be preserved in the flow better than easier standards such as "jitter".

A clock from Stratum 1, the top Stratum Level, is known as a Primary Reference Source (PRS). A Stratum 1 clock may control (provide a Reference Clock for) all lower Strata. Strata 2 through 3 may control the same or lower Strata. Strata 4E and 4 should not control a CGM.

Stratum 1 clocks are typically generated by “atomic clocks”. The Global Positioning System (GPS) can also provide a Stratum 1 clock. A Building Integrated Timing Supply (BITS) or Synchronization Supply Unit (SSU) is typically at Stratum 2. Depending on the CGM FW and the clocking oscillator used, the TCM is capable of supporting up to Stratum 2. However, only Strata 3 is currently supported.

When the CGM is enabled for PTP operations the TCM automatically operates in Stratum 3 for TDM clocking. Attempts to set a different Stratum Level are ignored.

The Stratum Level of TDM System Clocks can be maintained only when the TDM Reference Clocks have the performance characteristics defined by the appropriate standard.

1.2.4 Clock Nomenclature

The Telco industry has many terms for clocks based on the role they play in a system or other characteristics. Two of these roles have already been defined; Reference and System clocks. The table below defines other common clock names or types.

Table 1-1 Clock types

Terminology	Description
Pure clock	A signal containing only clocking information. Examples: ATCA CLK1, CLK2, CLK3; T0, T1, T2
Recovered clock	A pure clock extracted from a signal containing other information.
Line clock	A Reference Clock recovered from a payload signal. Also known as a Network Reference Clock.
T0	Any System Clock.
T1	Any Synchronous Digital Hierarchy (SDH) line clock. SDH is a multiplexing protocol similar to Synchronous Optical Networking (SONET). SDH is used in most of the world except the U.S. and Canada. Do not confuse this T1 terminology with the T1 used to represent the SONET protocol.
T2	Any Synchronous Optical Networking (SONET) line clock. Refer to T1 above for additional information.
T3	Any Reference Clock from a Building Integrated Timing Supply (BITS) or Sync Supply Unit (SSU). This is not a pure clock.
T4	A Reference Clock from a Line Interface Unit for a BITS or SSU. This is not a pure clock.
PTP Master Clock	A PTP node providing a PTP-based Reference Clock to other PTP nodes.

Table 1-1 Clock types (continued)

Terminology	Description
PTP Slave Clock	A PTP node using a PTP-based Reference Clock to generate local clocks.
PTP Boundary Clock	A PTP node both receiving a PTP-based Reference Clock itself and providing a separate PTP-based Reference Clock to other PTP nodes.

1.2.5 Traceable Clocks

Typically, Reference Clocks should be traceable. That is, the ultimate origin of the Reference Clock should be a clock of some known and desired quality. This starter source clock is known as a Primary Reference Clock (PRC) and is typically a Stratum 1 clock. Although a traceable clock may traverse many intervening layers of clocking equipment before arriving locally, it should still exhibit the primary quality characteristics that are maintained by the PRC. Despite that, the clock may need a little easy cleaning up by the local CGM to regain some of the quality characteristics that can get lost en route. Thus, a traceable Reference Clock is the highest-quality basis from which to form high-quality System Clocks.

A Reference Clock that is not traceable has either an unknown or undesired origin. Typically the origin is undesired because the quality characteristics are not high enough. Any System Clocks derived from a lower-quality Reference will themselves be lower-quality.

1.2.6 System Clocks with No Reference

When a CGM is locked to a Reference Clock for a sufficient period, the CGM can learn the quality characteristics of the Reference Clock. If the Reference Clock were to go away and no other Reference were available, then the CGM can maintain those quality characteristics in the System Clocks for at least a while. This is known as running in holdover mode. The definition of the Stratum Level dictates how long it takes the CGM to do the learning plus how long and to what degree the CGM can maintain the quality characteristics.

If the CGM has not learned the quality characteristics of a Reference Clock and no Reference Clock is currently available, then System Clocks can still be generated, but their quality characteristics will be based on the local clocking oscillator acting as a default Reference Clock. This is known as running in freerun mode. The quality characteristics of the System Clocks will depend on the quality characteristics of the oscillator. Refer to the Stratum Level of the blade and the Stratum Level standards for more information.

1.2.7 Clock Redundancy

Redundant clocks are used to avoid single-path failures and thereby increase system reliability. Clocks redundancy is achieved by having multiple clocks available for fulfilling the same role within the system, for example, having two or more System Clocks available for synchronizing the same process. The paths traversed by the redundant clocks should diverge as far upstream as feasible to further avoid single-point failures.

Typically, only one of the redundant clocks for a role is in active use. If that clock fails, another redundant clock can be used to maintain nominal operations. Often switching between the clocks is automatic and does not cause timing anomalies.

The TCM offers the following mechanisms for supporting clock redundancy:

- Multiple Reference Clocks for the CGM. Refer to the *cgmInputTable* MIB object for more information.
- Dual ATCA backplane Clock Domains. Refer to *Clock Domain on page 19* and *ATCA Clock Domain* for more information.
- Redundant Protection Partner Master/Slave TCMs. This hot-standby Master/Slave concept should not be confused with PTP Master and Slave Clocks. Refer to *Protection Partner Pair Configuration* for more information.

1.2.8 Clock Domain

The ATCA standard defines two clock channels for each of the three ATCA Backplane Clocks known as CLK1, CLK2, and CLK3. One channel is designated as ATCA Clock Domain A and the other as B. The TCM is always assigned to one of these two domains. Based on the assignment, the TCM typically references the clocks of only one of the domains in any particular context. The two channels or domains usually supply redundant clocks for whatever role the Backplane Clock is serving. Refer to *Clock Domain on page 19* for more information.

The PTP standard has a concept of Clock Domains independent of ATCA Domains. Where the ATCA Domains are two physical communications paths, the PTP Domains are multiple logical communications channels that may share the same physical paths. This allows multiple sets of PTP clocking equipment on the same network to be logically segregated into independent working groups. The PTP clocks generated for one domain will not be confused with the PTP clocks for another domain even though the packets for one working group may be received by another.

For proper operations, all PTP nodes meant to interact with each other must be in the same PTP Domain. Further, the two PTP ports of a TCM Boundary Clock must be in different domains.

Refer to Section 8.2.1 in IEEE1588v2.

1.2.9 Clock Attributes

A user can be interested in many attributes of a clock. These attributes include:

- Validity
- Source
- Consumer
- Frequency
- Phase
- Pulse length
- Wander
- Jitter
- Traceability
- Stratum Level

The Traceability and Stratum Level attributes have already been discussed in separate sections above. The following subsections present the remaining attributes.

1.2.9.1 Validity

Clock validity refers to the extent a clock exhibits certain needed characteristics. These characteristics can include frequency, phase, pulse length, wander, jitter, and others. A valid clock is one that successfully exhibits needed characteristics and an invalid clock is one that does not. A clock may be valid in one context and invalid in another depending on the criteria used to define validity.

1.2.9.2 Sources and Consumers

An entity which outputs a clock is known as a clock source or a clock driver. An entity which inputs a clock is being driven by the clock and is the clock consumer. A single clock can simultaneously function as both an input and output.

1.2.9.3 Frequency

Clock frequency is the number of individual clock pulses (ticks) per unit time. Clock frequency is measured in pulses-per-second which is known as Hertz (Hz). Typically Telco clocks only have a small set of well-known frequency values. These values include:

- 1Hz and 1 pulse-per-second (1PPS)
- 2kHz, the Multi-Frame Sync rate
- 8kHz, the Frame Sync rate
- 1.544MHz, the T1 (SONET) rate

- 2.048MHz, the E1 (SDH) rate
- Other rates such as 19.44MHz and 156.25MHz

Unlike TDM clocks, PTP clocks do not have an inherent clock frequency.

The T1 and E1 frequencies are only available when running in the corresponding T1/E1 mode. Attempts to refer to the frequency of one mode while in the other mode are automatically changed to the frequency of the current mode. For example, setting a clock to the T1 frequency while in the E1 mode will actually set the clock to the E1 frequency. Refer to the ***cgmInterfaceMode*** **MIB and configuration variable.**

1.2.9.4 Phase

Clock phase is based on the beginning of a clock pulse. From one pulse to the next is one clock period which is also measured as 360 degrees. Two clocks with the same frequency may have certain relationships between their phases:

- In-phase/phase-aligned: The pulses of each clock begin at the same instant.
- Phase-offset: The pulses of each clock begin at a fixed offset from each other.
- Out-of-phase: The pulses of one clock begin as the pulses of the other clock end.

If a System Clock consumer has redundant input clocks available to avoid single-point failures, then these input clocks are often in-phase so switching between the clocks does not introduce timing anomalies. A CGM may also have multiple Reference Clocks available to avoid single-point failures. However, in this case it is common for the multiple Reference Clock sources to not be in-phase. Typically a CGM maintains a certain phase-offset between the current Reference Clock and the generated System Clocks of the same frequency. Simply switching between the non-phase-aligned Reference Clocks while maintaining the same phase-offset for the System Clocks can cause a sudden phase shift in the System Clocks. This may lead to timing anomalies for the System Clock consumers. To compensate, a CGM can perform Phase Build Out (PBO) when switching between Reference Clocks. With PBO, the CGM determines a new phase-offset for the new Reference Clock such that the current phases of the System Clocks are maintained.

1.2.9.5 Pulse Length

The length of a clock pulse is measure from leading edge to trailing edge of the pulse. This is also known as the duty cycle of the pulse. Some clock consumers may work best with a certain pulse length. A common pulse length is one-half of a clock period. In contrast, the difference between the standard 1Hz and 1 pulse-per second (1PPS) frequency designations is the short uniform pulse length of 1PPS.

1.2.9.6 Jitter

Jitter is a measurement of deviation between expected and actual clock phases. Jitter is a short-term, high rate-of-change characteristic that can be thought of as measuring shaky pulses. Jitter is typically easy to remove from a clock signal.

1.2.9.7 Wander

Like jitter, wander is also a measurement of deviation between expected and actual clock phases. Wander is a long-term, low rate-of-change characteristic that can be thought of as measuring drifting pulses. Wander can be introduced into a signal as equipment changes temperature or ages. Wander can be more expensive to remove than jitter.

1.2.9.8 Reference/System Clock Relationship

Reference and System Clocks typically have the following relationships:

- TDM System Clock frequencies are typically linear functions of the TDM Reference frequency.
- System and Reference Clocks may be in-phase or have a fixed offset.
- System Clocks can have improved pulse length.
- System Clocks typically have improved jitter.
- System Clocks may have improved wander.
- System Clocks inherit the traceability of Reference.
- System Clocks do not inherit Reference Stratum. A CGM typically maintains the Stratum of the Reference Clock or one step lower.

1.3 TCM Software Overview

The TCM software and firmware comprises the elements shown in the below table. Refer to *Operational Overview* for high-level details on how the major elements interact to provide the TCM services.

Table 1-2 TCM Software and Firmware Elements

TCM Software and Firmware Element	Description
snmpd	A Simple Network Management Protocol (SNMP) Master Agent executable. Refer to *Runtime Operations* for more information.

Table 1-2 TCM Software and Firmware Elements (continued)

TCM Software and Firmware Element	Description
tclk_agent	An SNMP sub-agent executable. Refer to *tclk_agent* for more information.
CGM-CONTROL-MIB.txt IEEE1588-MIB.txt	Textual forms of the SNMP Management Information Bases (MIBs), CGM-CONTROL-MIB, and IEEE1588-MIB, respectively. Refer to *MIBs* for more information.
Release_<version>.hex	The binary firmware file for the Semtech ACS9510 ToPSync Clock Generator Module (CGM). The name of the firmware file may vary by TCM release. Refer to *CGM Firmware Upload* for more information on installing the firmware.
/etc/logrotate.d/tcm	A logrotate service configuration file. Refer to *Command Line Options* for more information on logging options.
/dev/tcm	A Linux kernel device driver for servicing TCM hardware. The operations of this driver are opaque to the user and do not require user intervention or configuration.
/opt/bladeservices/tcm_track_delay	A tool used by tclk_agent.

Although not strictly considered TCM components, the front-blade and RTM FPGA firmware are key elements in the proper operation of the TCM. The version of the FPGA firmware should be maintained at the appropriate levels for the currently installed version of the TCM.

1.4 TCM Hardware Overview

The TCM hardware consists of the following components:

- A Semtech Corporation ACS9510 Clock Generator Module (CGM).
- A clocking oscillator.
- ATCA backplane, Rear Transition Module (RTM), and Advanced Mezzanine Card (AMC) interfaces.
- Building Integrated Timing Supply (BITS) or Synchronization Supply Unit (SSU) interfaces.
- Synchronous Ethernet (SyncE) interfaces.
- Extension Shelf interfaces.
- A Master/Slave sync interface.

- LEDs

There is a board variant for each supported Telco Stratum Level. There is also a board variant which does not support Telco Clocking. The BBS is identical for all board variants.

The TCM can both consume and produce Reference and System Clocks for various purposes. The following table shows how different components can participate in this process.

Table 1-3 Reference and System Clock Sources and Consumers

Component	Reference Clock		System Clock	
	Source	**Consumer**	**Source**	**Consumer**
CGM	X (for a BITS/SSU)	X	X	
AMC	X	X	X	X
RTM	X			
ATCA Backplane	X	X	X	X
Extension Shelf port	X	X	X	X
BITS/SSU	X	X		
SyncE circuitry	X			X

1.4.1 ACS9510 CGM

In general, clocks are generated using other clocks. A CGM takes a Reference Clock and makes or generates System Clocks. The Semtech ToPSync ACS9510 CGM can work with both TDM and PTP Reference and System Clocks. The ToPSync can simultaneously generate multiple TDM System Clocks of various frequencies that may also be in-phase with a Reference. The ToPSync can also be a PTP Master for up to 64 simultaneous PTP Slaves.

Although only one Reference is needed to generate multiple System Clocks, the ACS9510 supports accessing multiple TDM and PTP Reference Clocks to avoid single-point failure. If the current Reference were to fail, then the ACS9510 can quickly switch to another Reference without causing timing anomalies. In addition to supporting multiple failure-backup Reference Clocks, the ACS9510 supports using two TDM References and one PTP Reference simultaneously. Each of the TDM Reference Clocks is routed through different circuitry known as T0 and T4 paths. The T0 path is used to generate the typical System or T0 Clocks. The T4 path is used to condition a TDM Reference Clock, so it is suitable to be used as a T4 Reference Clock for a BITS/SSU. Refer to the *cgmInputTable* MIB object for additional details on identifying and prioritizing T0 and T4 Reference Clocks.

Communications with the ACS9510 are via SPI, FPGA, and Ethernet interfaces.

1.4.2 Clocking Oscillator

A clocking oscillator outputs a base clocking control signal for the ACS9510. An oscillator may be installed on the blade itself, the RTM, or on both. Depending on the oscillator and the ACS9510 firmware, the ACS9510 can support a specific Stratum Level or lower. The currently supported Stratum Level is Stratum 3.

1.4.3 ATCA Backplane Clocks

The ATCA Standard defines three backplane clocks shared by all connected blades. These TDM clocks are known as CLK1, CLK2, and CLK3. These clocks can be input or output and provide Reference or System Clocks. Each of these clocks supports two redundant channels to avoid single-point failures. These channels are known as the A and B Clock Domains.

1.4.4 AMC Clocks

When an appropriate AMC such as the Artesyn AMC-8001 is installed, the TCM can make use of the AMC clocks TCLKA, TCLKB, TCLKC, and TCLKD. These TDM clocks can be either input or output and provide Reference or System Clocks.

1.4.5 RTM Clocks

All of the currently supported RTM Clocks are part of the SyncE subsystem. Refer to *SyncE Interfaces*.

1.4.6 PTP Clocks

The TCM supports two logical PTP ports that share the single Ethernet hardware port of the ToPSync. The TCM can operate either as a PTP Slave-Only Clock using only one PTP port or as a PTP Boundary Clock using both PTP ports. For more information, refer to *PTP Operations Overview*.

1.4.7 BITS/SSU Interfaces

A Building Integrated Timing Supply (BITS) may also be known as a Synchronization Supply Unit (SSU). A BITS/SSU is typically a Stratum 2 device. These are CGM devices external to the blade used to provide a TDM Reference Clock known as a T3 clock. A T3 clock follows either the T1/SONET or E1/SDH specification.

A BITS/SSU may accept a TDM Reference Clock from a connection to other, possibly remote equipment. Typically this equipment is at Stratum 1 or 2. Otherwise the BITS should accept what is known as a T4 Reference Clock from local equipment such as the TCM. The T4 clock is often a line clock conditioned by the ACS9510. Like T3, the T4 is also a T1/E1 clock.

The TCM provides two front-panel RJ-45 connectors for BITS/SSU units. Consult the BITS/SSU and blade hardware specifications for cabling requirements. The connectors are labeled TE1/2 which is short for T1/E1 #1 and T1/E1 #2. These connectors interface the BITS/SSU equipment with Line Interface Units (LIUs) provided by Maxim Corporation DS26503 chips. The signals between the LIUs and the BITS/SSU equipment are not pure clocks and, among other things, the LIUs can embed and extract the T3 and T4 clocks in these signals.

1.4.8 SyncE Interfaces

Synchronous Ethernet (SyncE) is a merger of ordinary Ethernet and Telco Clocking. High-speed Ethernet already embeds a clock in the Ethernet signal. This clock is not necessarily of Telco quality, that is, suitable for use as a Reference Clock. Ordinary Ethernet becomes SyncE when the embedded clock is of Telco quality. SyncE is an attractive Telco solution, since only a small amount of additional circuitry for embedding and recovering the clock is needed to convert ordinary Ethernet into SyncE.

The TCM can make use of SyncE in two ways:

- SyncE can provide a basis for a CGM Reference Clock.
- A CGM System Clock can be the basis for the SyncE embedded clock.

SyncE connections can be made via the four front-panel SFP+connectors labeled ETH1 through ETH4 and the eight ATCA-F125 RTM SFP+ connectors labeled ETH5 through ETH12. For more information, refer *SyncE Configuration* on page 106.

1.4.9 Extension Shelf Interfaces

The Extension Shelf interfaces support transferring Reference and System Clocks to and from other blades in the same or nearby chassis. Connections are made using the five front-panel Extension Shelf RJ-45 connectors labeled EXT1 through EXT5. There are four clocks available per connector. Consult the blade hardware specification for cabling requirements.

1.4.10 TDM Master/Slave Sync Interface

The TDM Master/Slave sync interface supports TCM redundancy via a special TDM Master/Slave synchronization clock that can be used as a Reference Clock. There are two possible ways to transfer the TDM Master/Slave synchronization clock:

1. A front-panel RJ-45 connector labeled MS. This connector uses a standard Ethernet crossover cable. The redundant blade can be in the same or a nearby chassis.
2. ATCA Backplane Clock CLK1. No cable is needed, but the redundant blade must be in the same chassis.

Refer to *Protection Partner Pair Configuration* for more information.

1.4.11 LEDs

There are four Telco LEDs at the bottom of the front panel that show status information about the two BITS/SSU interfaces, the CGM, and the Protection Partner Master/Slave relationship. Status is indicated by the LEDs being on or off, using different colors, and by blinking or not.

The two BITS/SSU LEDs are labeled T1/E1-1 and T1/E1-2. These LEDs indicate the following status:

- Off: The BITS/SSU interface is not activated.
- Red blinking: Status is otherwise unknown at this time.
- Green: A valid BITS/SSU signal is detected.
- Yellow: No valid BITS/SSU signal is detected.

The CGM LED is labeled LOCK. This LED indicates the following status:

- Off: The CGM is not activated.
- Red blinking: Status is otherwise unknown at this time. This will be true until a Master/Slave Protection Parter, relationship has been established.
- Green: The CGM T0 PLL is locked to a TDM Reference Clock. For a Slave Protection Partner, this is the TDM Master/Slave synchronization clock.

- Yellow steady: The CGM T0 PLL is in holdover mode.
- Yellow blinking: The CGM T0 PLL is in freerun mode.

The Master/Slave Protection Partner LED is labeled MODE. This LED indicates the following status:

- Off: The CGM is not activated.
- Red blinking: Status is otherwise unknown at this time. This will be true until a Master/Slave Protection Partner relationship has been established.
- Green steady: The blade is a Master Protection Partner with a Slave.
- Green blinking: The blade is a Master Protection Partner without a Slave (a stand-alone Master).
- Yellow: The blade is a Slave Protection Partner. This always implies a controlling Master exists.

1.5 Operational Overview

The TCM user's main jobs are to configure the starting setup of the TCM and then monitor and control the runtime environment. The `tclk_agent` is the main access point for all of these jobs.

The `tclk_agent` uses configuration files for defining the startup of the TCM. Refer to *Configuration Files* and *Configuration Items* for more information. After startup, the `tclk_agent` acts as an SNMP Sub-Agent to the snmpd Master Agent and thereby offers an SNMP interface to most of the runtime functionalities provided by the TCM. The CGM-CONTROL-MIB defines the interface to the TDM and system management features of the TCM while the IEEE1588-MIB defines the interface to the PTP features.

The Semtech ACS9510 ToPSync Clock Generator Module (CGM) is the core of the TCM operations. In general, the CGM takes Reference Clocks as inputs and produces System Clocks as outputs. The CGM ensures the quality of the clocks and provides mechanisms to avoid single-point failures. The CGM inputs/outputs can be transferred among various subsystems. These subsystems include:

- Components on the F125 blade such as an AMC and Synchronous Ethernet circuitry.
- An F125 RTM.
- The ATCA chassis backplane clocks.
- Building Integrated Timing Supply (BITS) units.
- For TDM clocks, other clocking blades inside the TCM chassis or in nearby chassis.
- For PTP clocks, other PTP nodes networked to the TCM.

The user's primary job is to configure the TCM to produce the desired work. This implies the user has an overall clocking system architecture or plan in mind. This plan can act as a blueprint for the TCM configuration and define the operational mode or role for the TCM and other hardware.

Configuring the TCM requires understanding how to combine various subsystems to work as a whole. Following are the main configuration tasks:

- Define which clocks are used as inputs and which are used as outputs.
- Define which sources are driving the input clocks and which subsystems consume the output clocks.
- Define the expected frequencies of the TDM input clocks and the desired frequencies of the TDM output clocks.
- Configure and activate Ethernet interfaces and ports so PTP nodes can exchange packets.
- Activate subsystems and ports as needed to provide the inputs and consume the outputs.
- Define how redundant clock sources are used to avoid single-point failures. This should include configuring a second F125 used to form a Master/Slave Protection Partner Pair. Refer to *Protection Partner Pair Configuration* for more information.
- Cable the system as needed.

The source, consumer, and frequency attributes of a clock are typically defined by the mode of the TCM and a few TCM controls. Some clocks offer no user controls and are handled exclusively by the TCM. Other clocks may offer limited to extensive user control. Refer to *CGM-CONTROL-MIB* and *IEEE1588-MIB* for controlling the characteristics of the clocks.

Once the TCM is configured and started, the runtime user operations are typically limited to monitoring that the configured system is continuing to operate as desired. These runtime tasks may include:

- Monitoring log messages, Events, and SNMP Traps for the indication of anomalous behaviors. Refer to *Logging*, *Events*, and the *cgmLogEvent* MIB object for more information.
- Monitoring the validity and usage of the CGM inputs. Refer to the *cgmInputTable* MIB object.

The rest of this *Telco Clocking Module* chapter provides more details about the configuration, control, and operations of the TCM.

Installation

2.1 Installation

The TCM software and firmware are contained in Red Hat Package Manager (RPM) files provided as a standard part of the ATCA-F125 Basic Blade Services (BBS) package. The TCM RPMs get installed into the root file system as a part of the BBS installation.

The TCM elements are provided by the following RPMs.

Table 2-1 TCM Element RPMs

TCM Software and Firmware Element	RPM
snmpd	net-snmp-<release>.ppc_e500v2
tclk_agent CGM-CONTROL-MIB.txt IEEE1588-MIB.txt /etc/logrotate.d/tcm	bbs-tcm-<release>.ppc_e500v2
Release_<version>.hex	bbs-atcaf125-semtech-<version>
/dev/tcm	N/A Part of Linux kernel.

The TCM components are installed in the following directories.

Table 2-2 TCM Installation Directories

TCM Software and Firmware Element	Directory
snmpd	/usr/sbin
tclk_agent	/opt/bladeservices/bin
CGM-CONTROL-MIB.txt IEEE1588-MIB.txt	/opt/bladeservices/mibs
Release_<version>.hex	/opt/bladeservices/rom
tcm - logrotate configuration file	/etc/logrotate.d
`tcm` - Linux device driver	/dev

The following subsections describe the areas where additional installation steps may need to be taken for proper operations.

2.1.1 AgentX

The tclk_agent interfaces with the snmpd using the Agent Extensibility Protocol (AgentX). Either the snmpd configuration file(s) must contain a master agentx line or the snmpd must be invoked with the -x option. Additional TCM installation steps may be required depending on how AgentX is configured. For instance, the default AgentX communications mechanism is a named socket, master, that automatically gets created in the default directory/var/agentx. If these defaults are used, then an installation issue is to ensure that the directory exists and has proper permissions. If the snmpd is using a non-default AgentX mechanism, then the tclk_agent must be configured to use the same mechanism. Typically this is done by invoking the tclk_agent with the same -x command line option as used with the snmpd.

Refer to *Command Line Options* and *Configuration Files* for additional details on configuring AgentX.

2.1.2 snmpd

An ATCA-F125 BBS package comes with a generic SNMP Master Agent, snmpd. The snmpd is based on the Open Source Net-SNMP suite of Application Protocol Interfaces (APIs). Refer to http://net-snmp.sourceforge.net for information concerning Net-SNMP.

The snmpd provides the SNMP front end for multiple BBS services such as SRstackware, and the TCM. Communications with the snmpd depends on proper configuration of networking elements and the snmpd. After a normal BBS installation, the default snmpd configuration files and the configuration of networking elements such as network interfaces, Virtual Local Area Networks (VLANs), gateways, switches, etc. will possibly need adjustments to ensure the snmpd can be contacted by local and remote SNMP tools.

All SNMP v2c compliant tools should interface with the snmpd. The BBS comes with several standard Net-SNMP tools such as snmpget, snmpset, and snmptable in /usr/bin. For proper operations, these tools typically require access to the appropriate MIBs. Usually this is facilitated by setting the MIBDIRS and MIBS Linux shell environment variables. The following shell commands support finding all of the standard BBS MIBs in the current shell:

```
export MIBDIRS=/usr/share/snmp/mibs:/opt/bladeservices/mibs
export MIBS=ALL
```

Refer to *Configuration Files* for additional details on configuring the snmpd. Refer to *SRstackware* Chapter in *BBS on ATCA-F125 with SRstackware Programmer's Reference*, for additional guidance on using the snmpd with these services. Refer to other sections of this guide for information on network interfaces, VLANs, gateways, and switches.

2.1.3 CGM

Communications with the Semtech ToPSync Clock Generator Module (CGM) depends on proper configuration of networking elements and the CGM. The CGM startup configuration is controlled via the CGM firmware and the `tclk_agent` startup configuration. There is no default `tclk_agent` configuration file installed with the BBS.

Refer to *CGM Firmware Upload* for details on installing the CGM firmware.

Refer to *Configuration Files* for additional details on configuring the CGM and also refer to *CGM Ethernet Configuration* for details on configuring CGM networking elements.

2.1.4 Clocking

Proper operations of the TCM may depend on external clocking hardware being installed and configured. The external clocking hardware includes components such as:

- Additional blades in the same chassis as the TCM ATCA-F125 to perform as clock suppliers or consumers using the ATCA Backplane Clocks.
- Building Integrated Timing Supply (BITS).
- An additional ATCA-F125 to perform as a Master/Slave Protection Partner.
- Additional blades in chassis other that the one for the TCM ATCA-F125 to perform as clock suppliers or consumers.
- An ATCA Advanced Mezzanine Card (AMC) Clock Generator Module installed on the TCM ATCA-F125.
- A Crystal Oscillator signal from a Rear Transition Module (RTM).

Functional Description

3.1 Functionality

The TCM software supports the configuration, control, and monitoring of the following major functional areas:

- *Clock Generation*
- *Clock Routing*
- *Clock Distribution*

The clocks generated, routed, and distributed by the TCM are capable of serving multiple purposes. Some of these purposes may be satisfied simultaneously while others are mutually exclusive of each other. These purposes include:

- Supplying the SyncE circuitry on the blade with a System Clock or recovering a Reference Clock from the SyncE circuitry.
- Supplying BITS/SSU with a T4 clock and/or recovering a BITS/SSU T3 Reference Clock.
- Supplying an AMC with Reference or System Clocks.
- Supplying other blades within the chassis with Reference or System Clocks via the ATCA Backplane Clocks.
- Supplying other blades within the chassis or other nearby chassis with Reference or System Clocks via Extension Shelf Connectors. These clocks may be from the TCM or from other blades within the chassis or from other nearby chassis.
- Supplying local or remote PTP nodes with PTP clocks.

3.1.1 Available Clocks

The clocks available to the TCM user are categorized based on the entity commonly associated with the definition of the clock. These entities are the:

- ATCA Standard
- Semtech ToPSync CGM
- TCM itself

Each TDM clock in the TCM has an assigned TCM name. These names are used as identifiers in the TCM MIBs and configuration files. The PTP clocks do not have assigned TCM names and are instead identified by IP addresses, PTP node IDs, and PTP port numbers.

In some cases a single clock may have multiple names. This might be for historical reasons or to emphasize the different roles of the clock.

Make sure that multiple sources are not trying to drive the same clock. In particular, watch for these situations:

Both the TCM and another blade in the chassis are trying to drive an ATCA Backplane Clock within the same ATCA Clock Domain.

Both the TCM and an AMC are trying to drive an AMC clock.

Both the TCM and another blade connected through an Extension Shelf connector are trying to drive an Extension Shelf port.

3.1.1.1 ATCA Clocks

The TCM interacts with the ATCA Backplane and AMC clocks as listed below.

Table 3-1 ATCA Clocks

TCM Name	Clock
clkCLK1	CLK1 current TCM ATCA Domain
clkCLK1A	CLK1 ATCA Domain A
clkCLK1B	CLK1 ATCA Domain B
clkCLK2	CLK2 current TCM ATCA Domain
clkCLK2A	CLK2 ATCA Domain A
clkCLK2B	CLK2 ATCA Domain B
clkCLK3	CLK3 current TCM ATCA Domain
clkCLK3A	CLK3 ATCA Domain A
clkCLK3B	CLK3 ATCA Domain B
clkTCLKA	TCLKA
clkTCLKB	TCLKB
clkTCLKC	TCLKC
clkTCLKD	TCLKD

Each of the ATCA Backplane clocks which do not specify an ATCA Clock Domain in their name (`clkCLK1, clkCLK2, and clkCLK3`) represents two possible clocks, one for each of the A and B Domains. The Domain for these clocks depends on the current mode of the TCM and the context in which the clock is used. Refer to *ATCA Clock Domain*.

By tradition, the AMC clocks TCLKA and TCLKC often are used as inputs to the AMC and the clocks TCLKB and TCLKD are used as outputs. The TCM does not restrict the direction of these clocks in this manner.

3.1.1.2 CGM TDM Clocks

The Semtech ToPSync CGM defines many clocks. Some of the clocks are internal to the CGM and some are external. Most of these clocks, and in particular the internal clocks, are hidden from the TCM user and are controlled solely by the inner workings of the TCM. The TCM currently exposes to the user some of the ToPSync OPCLKs, the external CGM output clocks used as System Clocks. The OPCLKs and their frequencies are shown in the following table.

Table 3-2 ToPSync Clocks

TCM Name	Clock	Frequency and Comments
clkOPCLK0	OPCLK0	**PTP operations enabled:** Sourced from an 8 kHz output from the PTP PLL. Automatically suppressed during intervals when there is no PTP Reference Clock. **PTP operations not enabled:** Sourced from the T0 PLL. The frequency is defined by the `clkFreq` configuration feature. The clock is not automatically suppressed when the T0 PLL has no Reference Clock.
clkOPCLK1	OPCLK1	19.44MHz
clkOPCLK2	OPCLK2	Frequency defined by `cgmInterfaceMode` setting. Set to: `sonetT1(0)` for 1.544MHz or, `sdhE1 (1)` for 2.048MHz.
clkOPCLK6	OPCLK6	8kHz
clkOPCLK7	OPCLK7	2kHz

Consult Semtech documentation for more information on the Semtech clocks.

3.1.1.3 Clocks Defined By the TCM

The TCM defines several clocks for its own purposes. Some of these clocks are specific to the TCM, but others map to ATCA and CGM clocks. Clocks that have specific dedicated circuitry are known as Physical Clocks. Some TCM clocks represent multiple Physical Clocks. These are known as Abstract Clocks. The specific clock represented by an Abstract Clock depends on the modes and configuration settings in effect at the time the clock is referenced.

Available Clocks

The TCM clocks are listed in *Table 3-3*. TCM controls often can define the sources for these clocks. Refer to *CGM-CONTROL-MIB* and *Configuration Files* for more information about controls.

Table 3-3 TCM-Defined Clocks

TCM Name	Abstract/Physical	Clock
clkBase<x>	Physical	Reference Clock recovered from Clock <x> of the Base Ethernet Switch. <x> may be 0 or 1.
clkBITS1Rx	Physical	T3 Reference Clock recovered from the BITS/SSU T1/E1 #1 front panel port.
clkBITS2Rx	Physical	T3 Reference Clock recovered from the BITS/SSU T1/E1 #2 front panel port.
clkEthP<x>	Physical	Reference Clock recovered from front panel Ethernet Port <x>. <x> may be 1..4.
clkExt<x>P<y>	Physical	Clock to/from Extension Shelf connector <x> Port <y>. <x> may be 1..5 and <y> may be 1..4.
clkFrameSync	Physical	Another name used to expose the Semtech 8kHz OPCLK6 to the user.
clkMFrameSync	Physical	Another name used to expose the Semtech kHz OPCLK7 to the user.
clkMSRx	Physical	CGM Reference Clock for Master/Slave Protection Partner Pair synchronization that is input from front panel. This clock is suppressed by the TCM, if `clkMSSyncIn` is not sourced from `cgmMSRx`.
clkMSSyncIn	Abstract	Master/Slave Protection Pair Synchronization Clock input to CGM.
clkMSTx	Physical	CGM Master/Slave Protection Partner Pair synchronization clock output to front panel. This clock is driven by the TCM only if `clkMSSyncIn` is sourced from `cgmMSTx`.
clkRefA	Abstract	User-defined CGM Reference Clock. If sourced from `clkCLK3`, then from the ATCA Domain A.
clkRefB	Abstract	User-defined CGM Reference Clock. If sourced from `clkCLK3`, then from the ATCA Domain B.

Table 3-3 TCM-Defined Clocks (continued)

TCM Name	Abstract/Physical	Clock
clkRTMEthP<x>	Physical	Reference Clock recovered from the RTM Ethernet Port <x>. <x> may be 5..12.
clkRTMSyncERef	Abstract	Reference Clock recovered and frequency translated from the RTM Synchronous Ethernet circuitry. The source for this clock is clkRTMSyncERcvdRef1 when used as the source for clkSyncERefA or clkRTMSyncERcvdRef2 when used as the source for clkSyncERefB. Regardless of the frequency of the source, this clock is always 8kHz if sourced by the TCM.
clkRTMSyncERcvdRef1	Abstract	Reference Clock recovered from the RTM Synchronous Ethernet circuitry.
clkRTMSyncERcvdRef2	Abstract	Reference Clock recovered from the RTM Synchronous Ethernet circuitry.
clkSyncEGen	Physical	19.44MHz CGM System Clock used to generate the Ethernet device clocks. Only available if `cgmSyncEGenEnable` is enabled.
clkSyncERefA	Physical	CGM Reference Clock recovered and frequency-translated from the Synchronous Ethernet circuitry. The source for this clock is always same as the source for `clkSyncERcvdRefA`. If sourced from `clkCLK3`, then from the ATCA Domain A. Regardless of the frequency of the source, this clock is always 8kHz if sourced by the TCM.
clkSyncERefB	Physical	CGM Reference Clock recovered and frequency-translated from the Synchronous Ethernet circuitry. The source for this clock is always the same as the source for `clkSyncERcvdRefB`. If sourced from `clkCLK3`, then from the ATCA Domain B. Regardless of the frequency of the source, this clock is always 8kHz if sourced by the TCM.
clkSyncERcvdRefA	Abstract	Reference Clock recovered from the Synchronous Ethernet circuitry.

Table 3-3 TCM-Defined Clocks (continued)

TCM Name	Abstract/Physical	Clock
clkSyncERcvd RefB	Abstract	Reference Clock recovered from the Synchronous Ethernet circuitry.
clkUsr1	Abstract	User-define CGM System Clock 1.
clkUsr2	Abstract	User-define CGM System Clock 2.
clk8kTest	Physical	TCM 8kHz test clock from the FPGA. This clock is reserved for internal test purposes by Artesyn and is not supported for general use. The characteristics of this clock are not sufficient to be recognized as a valid Reference Clock by the CGM on a sustained basis. Better quality System Clocks are usually generated by running in holdover or free-run mode than using the test clock as a Reference Clock.

3.1.1.4 Clock Frequency

In general, the user defines the desired frequency for TDM output clocks and the expected frequency of TDM input clocks. There is no inherit frequency for a PTP clock. In some cases the TDM frequencies are fixed by the TCM and cannot be changed by the user. Once an input frequency has been established for an input clock, the TCM can determine whether the clock exhibits the needed characteristics to be correctly designated as a clock of that frequency and thus whether the clock is valid or not.

The available frequencies for a clock typically are limited to a small set of standard industry values. Refer to *Configuration Items* and *MIBs* for more information.

The TCM supports clocks with frequencies shown in *Table 3-4* and *Table 3-5*. The TCM frequency names are used as identifiers in the TCM MIBs and configuration files.

Table 3-4 Input Frequencies

TCM Name	Input Frequency
f2k	2kHz
f8k	8kHz
f1M544	1.544MHz
f2M048	2.048MHz

Table 3-4 Input Frequencies (continued)

TCM Name	Input Frequency
f19M44	19.44MHz
f25M	25MHz
f156M25	156.25MHz
f161M13	161.1328125MHz

Table 3-5 Output Frequencies

TCM Name	Output Frequency
notDriven	This is a special TCM “frequency” name used to indicate a clock is not being driven and thus has no associated frequency.
fSync2k	sync2K
fSync8k	sync8K
fT0Dig1	pathT0_Digital1
fT0Dig2	pathT0_Digital2
fT0Alog	pathT0_Analog
fT0AlogDiv2	pathT0_AnalogDiv2
fT0AlogDiv4	pathT0_AnalogDiv4
fT0AlogDiv6	pathT0_AnalogDiv6
fT0AlogDiv8	pathT0_AnalogDiv8
fT0AlogDiv12	pathT0_AnalogDiv12
fT0AlogDiv16	pathT0_AnalogDiv16
fT0AlogDiv48	pathT0_AnalogDiv48
fT4CompClk	pathT4compositeClock
fT4PriRate	pathT4PrimaryRate
fT4AlogDiv2	pathT4_AnalogDiv2
fT4AlogDiv4	pathT4_AnalogDiv4

Table 3-5 Output Frequencies (continued)

TCM Name	Output Frequency
fT4AlogDiv8	pathT4_AnalogDiv8
fT4AlogDiv16	pathT4_AnalogDiv16
fT0AlogDiv48	pathT4_AnalogDiv48
fT4AlogDiv64	pathT4_AnalogDiv64

Refer to *Semtech SETS API documentation* and the *ACS9510 Datasheet* for finding more information about output frequencies.

3.2 Clock Generation

Clock Generation is the process of taking a Reference Clock as input and producing one or more System Clocks as outputs. Clock Generation can be performed in the TCM by the Semtech ToPSync CGM and by an appropriate AMC optionally installed on the blade. The TCM also can use clocks which are generated outside of the TCM. Refer to *Master Mode* for more information.

The `cgmInputTable` object defined by the CGM-CONTROL-MIB is used to control and monitor features of the TDM Reference Clocks used for Clock Generation. The CGM-CONTROL-MIB also defines controls for the TDM System Clocks and further features of the TDM Reference Clocks.

The `ptpClockPortDSTable` and `ptpAcceptablePartnerTable` objects defined by the *IEEE1588-MIB* are used to control and monitor features of the PTP Reference Clock used for Clock Generation. The same objects also provide control for the PTP System Clocks.

To overcome single-path failure points and thereby increase system reliability, Clock Generation is often performed in a redundant manner. Redundancy can be achieved to various degrees by utilizing the following techniques separately or in combination:

- Sourcing the CGM TDM Reference Clock from multiple independent TDM Reference Clock candidates. Refer to the *cgmInputTable* MIB object for further information.
- Sourcing the CGM PTP Reference Clock from multiple PTP Masters.
- Sourcing the redundant clocks from the dual ATCA Clock Domains A and B. Refer to *TCM Mode* for more information.
- Joining the CGMs on two ATCA-F125s into a Master/Slave Protection Partner Pair. Refer to *Protection Partner Pair Configuration* for more information.

3.3 Clock Routing

Clock Routing is concerned with configuring input and output clocking pathways to enable clock generators to produce System Clocks. These pathways:

- Get Reference Clocks to the CGM.
- Route clocks internal to the CGM to generate System Clocks.
- Get System Clocks out of the CGM.

Most aspects of Clock Routing internal to the CGM are hidden from the TCM user and are controlled entirely by the TCM as needed.

The user controls Clock Routing by setting the TCM mode and by utilizing a few clock sourcing controls. Refer to *CGM-CONTROL-MIB* and *IEEE1588-MIB* for more information about controls.

3.4 Clock Distribution

Similar to Clock Routing, Clock Distribution is also concerned with configuring input and output clocking pathways. In particular, Clock Distribution takes System Clocks and guides them to consumers. As with Clock Routing, the user controls Clock Distribution by setting the TCM mode and by using a few clock sourcing controls. The user may need to enable a clock consumer before the consumer can use the clock(s) made available. Refer to *CGM-CONTROL-MIB* and *IEEE1588-MIB* for more information about controls.

Chapter 4

TCM Mode

4.1 TCM Mode

In the main, the user does not directly configure Clock Generation, Routing, and Distribution. Instead, the user specifies the TCM mode of operation. The mode largely determines the clocks used to source the Backplane, AMC, Extension Shelf, and SyncE clocks. The mode also can determine the availability of CGM Reference Clocks. Using a mode to define the system configuration instead of directly configuring clock input/output connectivity provides the following benefits:

- Ease of system configuration
- Avoidance of clock loops where a System Clock is its own Reference Clock. Such a loop defeats clock traceability and any notion that a Reference Clock is guiding the generation of the System Clocks.
- Avoidance of other misconfigurations that can lead to undesired system behaviors.

A downside is that the number of supported clocking configurations is limited to the number of modes provided.

The overall TCM mode is defined by multiple attributes, each with its own SNMP control object(s). The TCM mode attributes are:

- Master Mode
- Clock Count
- ATCA Clock Domain
- Frame Sync Mode
- Interface Mode
- Stratum Level
- PTP Operations Mode
- Reference Loopback Mode

Most of these controls influence only TDM clocks. However, since PTP operations can involve converting between PTP and TDM clocks, proper PTP behavior can depend on correctly defining TDM settings.

Refer to *CGM-CONTROL-MIB* and *IEEE1588-MIB* for more details concerning the mode control objects and their settings. Refer to *TCM Mode Summary* for more details on how TDM clocks are influenced by various modes.

TCM modes of operations cannot be changed after the CGM has been enabled.

4.1.1 Master Mode

The TCM Master Mode affects which TDM clocks are generated by the TCM and how the clocks are distributed. This TCM mode attribute is controlled using the `cgmMasterMode` object. The Master Mode can have one of the below settings:

`localClkGen(0)`- Local Clock Generation

`amcClkGen(1)`- AMC Clock Generation

`passThru(2)`- Pass Through

`extShelf(3)`- Extension Shelf

The names for the Master Mode settings are based on which entity, if any, drives the ATCA backplane clocks CLK1 and CLK2. Refer to the individual settings for more information.

Local Clock Generation is the default mode and, unless otherwise stated, the mode assumed by much of the rest of the documentation.

The only CGM controlled by the TCM is the CGM installed directly on the blade. In particular, the TCM does not control an AMC CGM, if present, or any CGM installed on a remote blade. An exception is that the TCM does have some control over the CGM installed on a Protection Partner blade, but this is only possible when the TCM on the remote blade has formed a Protection Partner Pair with the local TCM.

4.1.1.1 Local Clock Generation

When running with the Local Clock Generation setting, the TCM uses the Semtech ToPSync CGM mounted on the board itself to produce most of the TDM System Clocks and, in particular, to always drive at least the ATCA backplane clocks CLK1 and CLK2.

4.1.1.2 AMC Clock Generation

When running with the AMC Clock Generation setting, TDM clocks are largely sourced from clocks generated by an appropriate AMC installed on the board and, in particular, the AMC is used to always drive at least the ATCA backplane clocks CLK1 and CLK2. The TDM clocks generated by the CGM mounted on the board perform a minor role.

4.1.1.3 Pass Through

When running with the Pass Through setting, the TDM clocks are largely sourced from clocks generated by external blades and the TCM distributes these clocks to extension shelves and onboard clock consumers. In particular, the TCM does not source the ATCA backplane clocks CLK1 or CLK2.

4.1.1.4 Extension Shelf

When running with the Extension Shelf setting, the TDM clocks are largely sourced from clocks generated by external blades and the TCM distributes these clocks to onboard clock consumers. In particular, the Extension Shelf connection ports are not driven and certain Extension Shelf connection ports are used to source at least the ATCA backplane clocks CLK1 and CLK2.

4.1.2 Clock Count

The TCM Clock Count largely influences how the ATCA backplane and AMC clocks are sourced. This TCM mode attribute is controlled using the `cgmClockCount` object. The Clock Count can have one of the below settings:

`twoClks(0)`-Two AMC and two backplane clocks

`threeClks(1)`- Three AMC and three backplane clocks

`fourClks(2)`- Four AMC and four backplane clocks

`fourAMCthreeBPClks(3)`- Four AMC and three backplane clocks

4.1.2.1 Two Clocks

When running with the Two Clocks setting, the TCM can be configured to source two AMC Clocks (`clkTCLKA` and `clkTCLKC`), and two ATCA backplane clocks (`clkCLK1` and `clkCLK2`). The `clkTCLKB` and `clkTCLKD` are not sourced by the TCM and `clkCLK3` can be sourced from `clkTCLKB` if desired, but not from the local CGM.

4.1.2.2 Three Clocks

When running with the Three Clocks setting, the TCM can be configured to source three AMC Clocks (`clkTCLKA`, `clkTCLKC`, `clkTCLKD`) and three ATCA backplane clocks (`clkCLK1`, `clkCLK2`, `clkCLK3`). The `clkTCLKB` is not sourced by the TCM.

4.1.2.3 Four Clocks

When running with the Four Clocks setting, the TCM can be configured to source four AMC Clocks (`clkTCLKA`, `clkTCLKB`, `clkTCLKC`, `clkTCLKD`) and four ATCA backplane clocks (`clkCLK1`, `clkCLK2`, `clkCLK3 Domain A`, `clkCLK3 Domain B`). Note the unusual situation of the TCM driving both domains of ATCA CLK3.

4.1.2.4 Four AMC and Three Backplane Clocks

When running with the four AMC and three backplane clocks setting, the TCM can be configured to source four AMC clocks (`clkTCLKA, clkTCLKB, clkTCLKC, clkTCLKD`) and three ATCA backplane clocks (`clkCLK1, clkCLK2, clkCLK3`).

4.1.3 ATCA Clock Domain

The ATCA clock domain effects how the TCM handles the ATCA backplane clocks `clkCLK1, clkCLK2`, and `clkCLK3`. This TCM mode attribute is controlled using the `cgmATCAClkDomain` and `autoATCAClkDomainEnable` configuration items. The ATCA clock domain may have one of the below settings:

- `domainA(0)`- Domain A
- `domainB(1)`- Domain B

Assignment to either Domain A or B can be arbitrary as long as the dual sources are in opposite Domains. The TCM user refers to each ATCA Backplane clock pair as if it were a single clock and the TCM will automatically map the clock to the appropriate member of the pair as needed.

Depending on the context, the referenced clock may be in the same or opposite Domain of the TCM.

Unless otherwise indicated, when the TCM is in one ATCA Domain, the clocks in the other Domain are not sourced by the TCM. However, these clocks may be driven by other sources. Care should be taken to ensure that two clock sources are not activated for the same Domain of a clock.

4.1.4 Frame Sync Mode

The TCM Frame Sync Mode determines which TDM clock from the CGM is used as the source of a framing process (data chunking) synchronization clock as well as the Master/Slave Protection Partner Transmit Clock (`clkMSTx`). This TCM mode attribute is controlled using the `cgmFrameSyncMode` object. The Frame Sync Mode may have one of the below settings:

`frameSync(0)`- Frame Sync

`multiFrameSync(1)`- MultiFrame Sync

In Frame Sync mode an 8 kHz clock is the source and in MultiFrame Sync mode a 2 kHz clock is the source.

The Frame Sync Mode should have the same setting for both members of a Master/Slave Protection Partner Pair.

4.1.5 Interface Mode

The TCM Interface Mode determines the frequency and other settings for interacting with the TCM Building Integrated Timing Supply (BITS) interfaces. The BITS interfaces are also known as framers or Line Interface Units (LIUs). This TCM mode attribute is controlled using the `cgmInterfaceMode` object. The Interface Mode may have one of the below settings:

`sonetT1(0)`- SONET/T1 (North America mode)

`sdhE1(1)`- SDH/E1 (Rest of the World mode)

4.1.6 Stratum Level

The TCM Stratum Level determines certain performance characteristics of the TDM System clocks generated by the Semtech ToPSync CGM installed on the blade. This TCM mode attribute is controlled using the `cgmStratumLevel` object. The only Stratum Level supported is `stratum3(3)`- ANSI Stratum 3.

User configuration of the Stratum level is ignored when PTP clocking operations are enabled. In this case the TCM is automatically forced to Stratum 3.

4.1.7 PTP Operations Mode

If PTP operations are enabled, then the PTP Operations Mode is defined by the PTP clock type of the TCM. The supported types are Slave-Only clock and Boundary clock. Refer to *IEEE1588v2 Sections 9.2.2* and *9.2.3* for more information about the operations of a Slave-Only and Boundary clock, respectively.

There is no single variable to query for the current PTP Operations Mode. Instead, it is a synthesis of the current running states of the two PTP ports. For more details, refer to the *ptpClockPortDSTable*.

Refer to *PTP Configuration Overview* for more information on configuring PTP operations.

4.1.8 Reference Loopback Mode

When enabled, Reference Loopback Mode allows the TCM to access a certain clock as both a System clock and a Reference clock.

Enabling Reference loopback mode is only supported when PTP operations are enabled and the recommended configuration documented in PTP Routing/Distribution Configuration is followed. All other uses of this variable are reserved. Misuse of this variable may cause undesired TCM behaviors.

For more information, refer to *PTP Routing/Distribution Configuration*.

4.1.9 TCM Mode Summary

The effect of the TCM modes on the sources of various TDM clocks is shown in the tables of the following subsections. Reference loopback mode is not included due to its specialized usage. To use the tables, first determine the desired clock count setting and reference the appropriate subsection. Within a subsection, determine the desired Master Mode setting and reference the appropriate table column. Within a table column determine the desired clock and ATCA domain (if applicable) and reference the corresponding row. The resulting table entry will show the source for the indicated clock (and optional ATCA clock domain) within the context of the selected clock count and Master Mode. Note that multiple clocks may simultaneously be driven by the same source. The clocks not sourced by the TCM may be driven by other sources.

Within a table, the source may have one of the three forms:

1. A single clock that is always the source within the defined context. In this case the source is simply shown as a TCM clock name such as **clkOPCLK1**.
2. A variable clock source determined by the Frame Sync Mode setting. In this case the source is shown as `FrameSync`. The source is either `clkOPCLK6` (running at 8kHz) or `clkOPCLK7` (running at 2kHz) depending on whether `cgmFrameSyncMode` is set to `frameSync(0)` or `multiFrameSync(1)`, respectively.
3. An optional clock source if user selected. In this case the source is shown with optional criteria specified in parentheses. For example: `clkOPCLK1(een)` or `FrameSync(aen)`. The following optional criteria are used:
   - `amc:` The source is the indicated clock if `cgmAMCEnable` is set to `enable(1)`. Otherwise the clock is not sourced by the TCM.
   - `amc3:` The source is the indicated clock if `cgmAMCCLK3Enable` is set to `enable(1)`. Otherwise the clock is not sourced by the TCM.
   - `ext:` The source is the indicated clock if `cgmExtShelves` is set such that the corresponding Extension Shelf is enabled. Otherwise the clock is not sourced by the TCM.

- `msrx`: The source is the indicated clock if the source for `clkMSSyncIn` is `clkMSR`x. Otherwise the clock is not sourced by the TCM.
- `synce`: The source is the indicated clock if `cgmSyncEGenEnable` is set to `enable(0)`. Otherwise the clock is not sourced by the TCM.
- `div`: This option is currently not supported. The source is always the indicated clock.

Unlike all the other modes, the Stratum Level does not affect the sources for clocks, but only their performance characteristics.

4.1.9.1 Two Clocks

The following table rows show the source for the indicated clock when `cgmClockCount` is set to `twoClks(0)`. Select the appropriate column based on the `cgmMasterMode` setting. Refer to *TCM Mode Summary* for details on how to interpret the table entries.

Table 4-1 Two Clocks

Clock	localClkGen(0)	amcClkGen(1)	passThru(2)	extShelf(3)
clkCLK1	FrameSync	clkTCLKD	not sourced by TCM	clkExt1P2
clkCLK2	clkOPCLK1	clkTCLKB	not sourced by TCM	clkExt1P4
clkCLK3	clkTCLKB(amc3)	not sourced by TCM	clkTCLKB(amc3)	clkTCLKB(amc3)
clkTCLKA	FrameSync(amc)	clkRefB	clkCLK1(amc)	clkExt1P2(amc)
clkTCLKB	not sourced by TCM	not sourced by TCM	not sourced by TCM	not sourced by TCM
clkTCLKC	clkOPCLK1(amc)	clkRefA	clkCLK2(amc)	clkExt1P4(amc)
clkTCLKD	not sourced by TCM	not sourced by TCM	not sourced by TCM	not sourced by TCM
clkExt1P1	FrameSync(ext)	clkTCLKD(ext)	clkCLK1(ext)	clkRefA(div)
clkExt2P1 clkExt3P1 clkExt4P1 clkExt5P1	FrameSync(ext)	clkTCLKD(ext)	clkCLK1(ext)	not sourced by TCM
clkExt1P2 clkExt2P2 clkExt3P2 clkExt4P2 clkExt5P2	not sourced by TCM	not sourced by TCM	not sourced by TCM	not sourced by TCM

Table 4-1 Two Clocks (continued)

Clock	localClkGen(0)	amcClkGen(1)	passThru(2)	extShelf(3)
clkExt1P3	clkOPCLK1(ext)	clkTCLKB(ext)	clkCLK2(ext)	clkRefB(div)
clkExt2P3 clkExt3P3 clkExt4P3 clkExt5P3	clkOPCLK1(ext)	clkTCLKB(ext)	clkCLK2(ext)	not sourced by TCM
clkExt1P4 clkExt2P4 clkExt3P4 clkExt4P4 clkExt5P4	not sourced by TCM	not sourced by TCM	not sourced by TCM	not sourced by TCM
clkSyncE Gen	clkOPCLK1(synce)	clkOPCLK1(synce)	clkOPCLK1(synce)	clkOPCLK1(synce)
clkMSTx	FrameSync(msrx)	not sourced by TCM	not sourced by TCM	not sourced by TCM

4.1.9.2 Three Clocks

The following table rows show the source for the indicated clock when `cgmClockCount` is set to `threeClks(1)`. Select the appropriate column based on the `cgmMasterMode` setting. Refer to *TCM Mode Summary* for details on how to interpret the table entries.

Table 4-2 Three Clocks

Clock	localClkGen(0)	amcClkGen(1)	passThru(2)	extShelf(3)
clkCLK1	FrameSync	clkTCLKD	not sourced by TCM	clkExt1P2
clkCLK2	clkOPCLK1	clkTCLKB	not sourced by TCM	clkExt1P4
clkCLK3	clkUsr1	clkTCLKC	not sourced by TCM	clkExt1P3
clkTCLKA	FrameSync(amc)	not sourced by TCM	clkCLK1(amc)	clkExt1P2(amc)
clkTCLKB	not sourced by TCM	not sourced by TCM	not sourced by TCM	not sourced by TCM
clkTCLKC	clkOPCLK1(amc)	clkRefA	clkCLK2(amc)	clkExt1P4(amc)
clkTCLKD	clkUsr1(amc)	not sourced by TCM	clkCLK3A(amc)	clkCLK3A(amc)
clkExt1P1	FrameSync(ext)	clkTCLKD(ext)	clkCLK1(ext)	clkRefA(div)

Table 4-2 Three Clocks (continued)

Clock	localClkGen(0)	amcClkGen(1)	passThru(2)	extShelf(3)
clkExt2P1 clkExt3P1 clkExt4P1 clkExt5P1	FrameSync(ext)	clkTCLKD(ext)	clkCLK1(ext)	not sourced by TCM
clkExt1P2 clkExt2P2 clkExt3P2 clkExt4P2 clkExt5P2	not sourced by TCM	not sourced by TCM	not sourced by TCM	not sourced by TCM
clkExt1P3	clkOPCLK1(ext)	clkTCLKB(ext)	clkCLK2(ext)	clkRefB(div)
clkExt2P3 clkExt3P3 clkExt4P3 clkExt5P3	clkOPCLK1(ext)	clkTCLKB(ext)	clkCLK2(ext)	not sourced by TCM
clkExt1P4 clkExt2P4 clkExt3P4 clkExt4P4 clkExt5P4	clkUsr1(ext)	clkTCLKC(ext)	clkCLK3A(ext)	not sourced by TCM
clkSyncE Gen	clkOPCLK1(synce)	clkOPCLK1(synce)	clkOPCLK1(synce)	clkOPCLK1(synce)
clkMSTx	FrameSync(msrx)	not sourced by TCM	not sourced by TCM	not sourced by TCM

4.1.9.3 Four Clocks

The following table rows show the source for the indicated clock when `cgmClockCount` is set to `fourClks(2)`. Select the appropriate column based on the `cgmMasterMode` setting. Refer to *TCM Mode Summary* for details on how to interpret the table entries. Note the unusual situation of the TCM driving both domains of ATCA CLK3.

Table 4-3 Four Clocks

Clock	localClkGen(0)	amcClkGen(1)	passThru(2)	extShelf(3)
clkCLK1	FrameSync	clkTCLKD	not sourced by TCM	clkExt1P2

Table 4-3 Four Clocks (continued)

Clock	localClkGen(0)	amcClkGen(1)	passThru(2)	extShelf(3)
clkCLK2	clkOPCLK1	clkTCLKB	not sourced by TCM	clkExt1P4
clkCLK3A	clkUsr1	clkTCLKC	not sourced by TCM	clkExt1P3
clkCLK3B	clkUsr2	clkTCLKC	not sourced by TCM	clkExt1P3
clkTCLKA	FrameSync(amc)	not sourced by TCM	clkCLK1(amc)	clkExt1P2(amc)
clkTCLKB	clkUsr2(amc)	not sourced by TCM	clkCLK3B(amc)	clkCLK3B(amc)
clkTCLKC	clkOPCLK1(amc)	not sourced by TCM	clkCLK2(amc)	clkExt1P4(amc)
clkTCLKD	clkUsr1(amc)	not sourced by TCM	clkCLK3A(amc)	clkCLK3A(amc)
clkExt1P1	FrameSync(ext)	clkTCLKD(ext)	clkCLK1(ext)	not sourced by TCM
clkExt2P1 clkExt3P1 clkExt4P1 clkExt5P1	FrameSync(ext)	clkTCLKD(ext)	clkCLK1(ext)	not sourced by TCM
clkExt1P2 clkExt2P2 clkExt3P2 clkExt4P2 clkExt5P2	clkUsr2(ext)	clkTCLKA(ext)	clkCLK3B(ext)	not sourced by TCM
clkExt1P3	clkOPCLK1(ext)	clkTCLKB(ext)	clkCLK2(ext)	not sourced by TCM
clkExt2P3 clkExt3P3 clkExt4P3 clkExt5P3	clkOPCLK1(ext)	clkTCLKB(ext)	clkCLK2(ext)	not sourced by TCM
clkExt1P4 clkExt2P4 clkExt3P4 clkExt4P4 clkExt5P4	clkUsr1(ext)	clkTCLKC(ext)	clkCLK3A(ext)	not sourced by TCM
clkSyncE Gen	clkOPCLK1(synce)	clkOPCLK1(synce)	clkOPCLK1(synce)	clkOPCLK1(synce)

Table 4-3 Four Clocks (continued)

Clock	localClkGen(0)	amcClkGen(1)	passThru(2)	extShelf(3)
clkMSTx	FrameSync(msrx)	not sourced by TCM	not sourced by TCM	not sourced by TCM

4.1.9.4 Four AMC and Three Backplane Clocks

The following table rows show the source for the indicated clock when `cgmClockCount` is set to `fourAMCthreeBPClks(3)`. Select the appropriate column based on the `cgmMasterMode` setting. Refer to *TCM Mode Summary* for details on how to interpret the table entries.

Table 4-4 Four AMC and Three Backplane Clocks

Clock	localClkGen(0)	amcClkGen(1)	passThru(2)	extShelf(3)
clkCLK1	FrameSync	clkTCLKD	not sourced by TCM	clkExt1P2
clkCLK2	clkOPCLK1	clkTCLKB	not sourced by TCM	clkExt1P4
clkCLK3	clkUsr1	clkTCLKC	not sourced by TCM	clkExt1P3
clkTCLKA	FrameSync(amc)	not sourced by TCM	clkCLK1(amc)	clkExt1P2(amc)
clkTCLKB	clkUsr2(amc)	not sourced by TCM	clkCLK3B(amc)	clkCLK3B(amc)
clkTCLKC	clkOPCLK1(amc)	not sourced by TCM	clkCLK2(amc)	clkExt1P4(amc)
clkTCLKD	clkUsr1(amc)	not sourced by TCM	clkCLK3A(amc)	clkCLK3A(amc)
clkExt1P1	FrameSync(ext)	clkTCLKD(ext)	clkCLK1(ext)	not sourced by TCM
clkExt2P1 clkExt3P1 clkExt4P1 clkExt5P1	FrameSync(ext)	clkTCLKD(ext)	clkCLK1(ext)	not sourced by TCM
clkExt1P2 clkExt2P2 clkExt3P2 clkExt4P2 clkExt5P2	clkUsr2(ext)	clkTCLKA(ext)	clkCLK3B(ext)	not sourced by TCM
clkExt1P3	clkOPCLK1(ext)	clkTCLKB(ext)	clkCLK2(ext)	not sourced by TCM

Table 4-4 Four AMC and Three Backplane Clocks (continued)

Clock	localClkGen(0)	amcClkGen(1)	passThru(2)	extShelf(3)
clkExt2P3 clkExt3P3 clkExt4P3 clkExt5P3	clkOPCLK1(ext)	clkTCLKB(ext)	clkCLK2(ext)	not sourced by TCM
clkExt1P4 clkExt2P4 clkExt3P4 clkExt4P4 clkExt5P4	clkUsr1(ext)	clkTCLKC(ext)	clkCLK3A(ext)	not sourced by TCM
clkSyncE Gen	clkOPCLK1(synce)	clkOPCLK1(synce)	clkOPCLK1(synce)	clkOPCLK1(synce)
clkMSTx	FrameSync(msrx)	not sourced by TCM	not sourced by TCM	not sourced by TCM

Chapter 5

Runtime Operations

5.1 Overview

The runtime operations of the TCM are provide by the `tclk_agent` executable, an SNMP sub agent. Command line options and configuration files control the startup of the `tclk_agent`. The command line options are mostly concerned with SNMP features. The configuration file entries can be for both SNMP and TCM features.

Typically the server task `snmpd` is started before the client task `tclk_agent`. This prevents warnings from the `tclk_agent` that the interface to the `snmpd` is failing. The `tclk_agent` automatically registers with the `snmpd` when it becomes available. If the `snmpd` or `tclk_agent` is stopped and restarts, then the `tclk_agent` will automatically re-register.

Unwanted behaviors can arise if multiple instances of the `tclk_agent` attempt to communicate with the snmpd or the CGM. To avoid this situation, the `tclk_agent` only allows a single instance of itself to run on a given blade at the same time. Make sure to let `tclk_agent` completely shut down before attempting to restart it.

At startup, the `tclk_agent` uses a configuration file to define the initial system configuration. Before applying the configuration, the `tclk_agent` disables the CGM if it is already enabled. This is necessary since many of the TCM modes of operations cannot be set when the CGM is enabled. The CGM is also disabled when the `tclk_agent` exits properly. Thus, in normal operations the TCM is inactive at startup and no clocks are being generated or used. To enable clock generation and usage, the `cgmEnable` configuration item or MIB object must be set appropriately.

During initialization the `tclk_agent` logs all of the configurations being applied and the status of the startup of any subsystems specified to be enabled by the configuration. Typically, the only logging that occurs after startup is due to runtime changes in the status or configuration of the system.

At TCM startup, the Master/Slave Protection Partner Sync output clock and all the AMC, ATCA backplane, and Extension Shelf clocks originated by the blade are suppressed. The TCM configured for ATCA Domain A automatically assumes Mastership and unsuppresses these clocks shortly after detecting the presence of a Protection Partner. The Master Protection Partner will be in free-run if the Partner blade is detected prior to the Master becoming locked to a validated Reference clock. The Domain B TCM becomes a Slave Protection Partner after validating and locking to the Master's unsuppressed Sync clock. Two minutes later, after becoming phase-aligned with the Master, the Slave unsuppresses its clocks.

Before proper system initialization, you should avoid attempting to configure the CGM via SNMP. After full startup, the SNMP interface can be used for run-time configuration, monitoring, and control of TCM features.

The major SNMP objects are listed below:

- CGM TDM Reference Clock Input Table (refer to the *cgmInputTable* MIB object)
- PTP Clock Port Data Set Table (refer to the *ptpClockPortDSTable* MIB object)
- PTP Acceptable Partner Table (refer to the *ptpAcceptablePartnerTable* MIB object)
- PTP Visible Masters Table (refer to the *ptpVisibleMasterTable* MIB object)
- BITS/SSU Interface Table (refer to the *cgmBitsTable* MIB object)
- TCM Event Table (refer to the *cgmEventTable* MIB object)
- TCM Event Traps (refer to the *cgmLogEvent* MIB object)

If Telco Clocking or the configured Stratum Level is not supported by the blade, then startup will fail and the `tclk_agent` will exit. Refer to the *cgmStratumLevel*configuration item for more information. If the current Semtech ToPSync firmware version is not supported, startup also will exit. Refer to the *TCMFWOverrideEnable* configuration item for a way to operate for testing purposes with a non-supported firmware version.

5.1.1 TDM Operations Overview

When the CGM is enabled the TCM initially operates as a stand-alone Master and immediately seeks out its hot-standby Protection Partner. The partners negotiate their subsequent roles as either the Master or Slave Partner.

The TCM suppresses its TDM System clocks until the Protection Partnership has been negotiated and steps have been taken to make sure that the TDM clocks from the two partners are phase aligned. Once a TCM unsuppresses its TDM clocks the clocks remain unsuppressed until the TCM is successfully shut down.

A Protection Partner pair share a Master/Slave TDM Sync clock to maintain phase alignment. A role of the Slave partner is to monitor operational conditions and adjust assumptions about the transmission or track delay of the Sync clock. These adjustments help ensure the phase alignment of the two partners.

During nominal operations the two partners generate redundant phase-aligned TDM System clocks. Consumers are typically configured to access the two copies of a System clock as hot-standby backups of each other. In this manner System clocks with the same phase and frequencies are still available to the consumers even if one partner fails.

A Master's System clocks are based on the currently selected Reference clock from potentially multiple available Reference clocks which are accessed as hot-standby backups for each other. A Slave's System clocks are always based on the Master/Slave TDM Sync clock, but the Slave should also have available the same backup Reference clocks used by the Master in case the Master fails. If a Master does not fail, but it loses access to all valid Reference Clocks, the Slave can assume Mastership if it still has an external valid Reference. In this manner redundant phase-aligned TDM System Clocks that are locked to a Reference clock remain for the consumers.

For more details, refer to *Protection Partner Pair Configuration on page 93* and *Stand-Alone Masters on page 96*.

5.1.2 PTP Operations Overview

When PTP operations are enabled the TCM operates either as a Slave-Only clock or a Boundary clock as determined by the startup configuration. During nominal operations the Slave port of either clock type supplies the PTP PLL with a single PTP Reference Clock. This Reference comes from the currently selected PTP Master for the port. The Master may be restricted to one from a set of Acceptable Masters specified for the port. The Master selected at any given moment is determined by the standard PTP Best Master clock algorithm. Refer to *Section 9.3 in IEEE1588v2.*

The PTP Reference clock is used to produce an 8kHz TDM clock to drive `clkOPCLK0`. The System Clocks `clkUsr1` and `clkUsr2` can use `clkOPCLK0` as a source and thereby make the `clkOPCLK0` output available to the system. Refer to *CGM TDM Clocks on page 37* and *Clocks Defined By the TCM on page 37* for more details.

During nominal operations the Master port of a Boundary clock generates PTP Clocks for all of the PTP Slaves configured for the port. These clocks will be traceable to the current Grandmaster for the Slave port.

At startup the output of `clkOPCLK0` is suppressed. The TCM monitors the Slave port every few seconds on a continuous basis to ensure it has a PTP Master. Output from `clkOPCLK0` is enabled when a Master is detected and suppressed when no Master is detected. In this manner, the TDM clock based on the PTP clock is only valid if traceable to a Grandmaster.

Output from the PTP Master port is available soon after the port is enabled. Output is not suppressed even if the clocks produced are not traceable. Instead, the current quality of the clocks is advertised via the standard PTP mechanisms and the Slaves for the port use the Best Master clock algorithm to make their selection.

When the CGM is enabled for PTP operations the TCM automatically operates in Stratum 3 for TDM clocking. Attempts to set a different Stratum Level are ignored.

5.1.3 TDM Without PTP Operations

The TCM supports TDM operations without simultaneous PTP operations. If PTP operations are not enabled (the default), the TCM PTP ports are not enabled and PTP clocks are completely suppressed. The PTP ports will neither generate nor respond to PTP packets. However, the PTP PLL is still activated and used to support TDM operations. There may be Events generated related to the operations of the PTP PLL even when PTP operations are not enabled.

5.1.4 PTP Without TDM Operations

The TCM does not support PTP operations without simultaneous TDM operations. Nominal TCM PTP operations depend on proper TDM operations in two areas:

1. TDM clock generation is required for the Protection Partner Master/Slave Sync Clock.
2. The recommended configuration for PTP operations includes converting the PTP Reference clock into a TDM Reference clock that is distributed to both Protection Partners. Among other benefits, this setup allows each Partner to track the status of the PTP operations of the other Partner using existing TDM mechanisms.

An enabled CGM is always generating TDM clocks although the TCM suppresses the TDM System clocks until nominal operating conditions are established. There is no user control to suppress the TDM System clocks once unsuppressed by the TCM.

If no TDM System clocks are needed for the operations of the system, then the TCM can be configured to minimize the generation of superfluous Events concerning TDM operations. In particular, the BITS/SSU and SyncE subsystems should not be enabled and the priorities for all T0/T4 Reference clocks should be zero. This is the default configuration for these items.

5.2 Automatic Recovery

The TCM consists of multiple software elements that potentially could become unstable or unusable, crash, or otherwise get into a corrupt state that requires a restart for the system to resume nominal operations. These software elements include:

- `tclk_agent` and snmpd executables.
- Linux and its associated drivers and services.

As documented elsewhere in this guide, the Linux kernel supports a watchdog feature that causes an automatic reboot should Linux fail to re-arm the watchdog in a timely manner. This feature catches some, but not all, situations where Linux requires a restart to resume nominal operations. Future versions of the BBS will include a Fault Management feature. This feature will provide mechanisms to perform automatic recovery of software components for additional situations.

The default BBS configuration automatically starts the snmpd, but not the `tclk_agent` when Linux is booted. If the user desires `tclk_agent` to automatically start, then one approach is to use the standard Linux initialization services and add `tclk_agent` startup scripts to the desired run-level subdirectories of `/etc/rc.d`. This is how the snmpd gets started.

In general, `tclk_agent` should be started as late as feasible in the boot process. In particular, if any of the SyncE ports are used, then `tclk_agent` must be started after the ports have been activated and configured by the sfptool daemon. Typically, this means that the `tclk_agent` should be started by one of the last S99 rc.d scripts executed by the boot process.

In the default BBS configuration, neither the snmpd nor the `tclk_agent` executables automatically restart after a crash. If the user desires this behavior for one or both of the executables, then one approach is to use the standard Linux inittab respawn service. This service can also be used to start the executable when Linux boots as well.

The standard Net-SNMP `-f` command line option of the `snmpd` and `tclk_agent` executables must be used with the respawn service. Otherwise the executable will perform a standard forking procedure to daemonize and the initially started process will exit and trigger the respawn service to repeatedly restart the executable. Since the respawn service places the executable in the background anyway, the executable still behaves as a daemon even with the `-f` option.

For a related feature, refer to the *TCMWatchdogEnable on page 129* configuration item.

TCM Management

6.1 tclk_agent

The user configuration, control, and management of the TCM are provided by an SNMP sub-agent, `tclk_agent`. Thus, the primary user interface to the TCM is via SNMP. Like the `snmpd`, the `tclk_agent` is also based on the Net-SNMP APIs.

As a sub-agent, all SNMP traffic from the `tclk_agent` is channeled through the snmpd Master Agent. The `tclk_agent` interfaces with the snmpd using the Agent Extensibility Protocol (AgentX). This interface is configured using *Command Line Options on page 90* and *Configuration Files on page 108*.

6.1.1 Logging

The TCM logging system consolidates messages from multiple sources to a user-selected destination. Although the different sources do not all support the same message formats and logging severity codes, the TCM attempts to present all messages in a homogeneous manner. The message sources are:

1. TCM messages, Events, and SNMP traps.
2. Net-SNMP messages
3. SNMP traffic (refer to the `-d` command line option in *Table 6-10*)
4. Net-SNMP debug information (refer to the `-D` command line option in *Table 6-10*)
5. The logging destination can be any one of the following:
6. /dev/null (no logging)
7. STDOUT
8. STDERR
9. A user file
10. The syslog process

If during shutdown parts of the TCM fail to exit prior to the logging services becoming unavailable, then any subsequent logging will automatically be directed to STDERR.

The default logging destination is the file `/var/log/tcm/tclk_agent.log`. This file is overwritten if it already exists and automatically rotated by the `logrotate` service as specified by the `/etc/logrotate.d/tcm` configuration file.

All information to be logged has a priority designation. Only information with a priority matching the logging level specified by the user actually gets logged. Refer to the `-L` command line option in *Table 6-10* for information on specifying the logging priority level and the logging destination.

Logging is performed using facilities provided by the Net-SNMP APIs. Net-SNMP defines eight log message severity levels, 0-7. The TCM APIs use five of the levels. Log messages output directly by the Net-SNMP APIs may use others as well. Except for log messages output directly by the Net-SNMP APIs themselves, all TCM log messages are prefixed with the string TCM:. Except for messages with the info Severity Level, the TCM: prefix is followed by a string indicating the message severity. The TCM log Severity Levels, their meaning, and the associated prefixes are show in the below table.

Table 6-1 TCM Log Severity Levels

Severity Level	Meaning	Prefix String
2 (critical)	A critical occurrence has been detected.	"TCM: ***CRITICAL: "
3 (error)	An error condition has been detected.	"TCM: ***ERROR: "
4 (warning)	A warning is being issued.	"TCM: ***WARNING: "
5 (event)	A significant event has occurred.	"TCM: ***EVENT: "
6 (info)	An informative message is being conveyed.	"TCM: "
(various)	A log message output directly by the Net-SNMP APIs.	(none)

The Net-SNMP APIs do not support multiple threads. Thus, the various TCM threads place log messages on a queue as they are generated. This allows the Net-SNMP APIs to switch between SNMP and log processing in a serial/single-threaded manner. An exception is that log messages generated by the Net-SNMP APIs themselves bypass the queue and get processed immediately. Note that the serial nature of Net-SNMP and the higher priority given to SNMP processing compared to log processing can cause log messages to be dequeued in clusters.

All log messages are time stamped with year-month-day and hour-minute-second. The timestamps are provided by Net-SNMP when the messages are dequeued. This means that the timestamps will cluster in the same manner as the messages.

The following subsections provide details on log messages generated by the TCM for the various Severity Levels. Refer to the Net-SNMP documentation for details concerning log messages generated by the Net-SNMP APIs themselves. Since Events are also treated as log messages, refer to the Events documentation section for details on additional log messages.

6.1.1.1 Not Logged Messages

The TCM reports some messages outside of the logging subsystem. Some of the messages are due to failures occurring prior to the availability of the logging subsystem. Others are in response to user requests such as the `-v` command line option.

The following table lists all of the messages generated by TCM APIs outside of the logging subsystem. The messages are output to STDERR with no time stamp or logging prefix. Unless otherwise noted, the TCM exits after reporting these messages.

Table 6-2 Not Logged Messages

Message	Meaning
Blade does not support Telco Clocking operations...exiting	The blade does not indicate a clocking oscillator is installed.
dequeue_item: INTERNAL ERROR: unknown queue type: <code>	The queuing subsystem does not recognize the queue type code.
<TCM API>: Failed to <operation>(<entity>): <reason>	The TCM API failed to perform the operation on the entity for the reported reason.
Failed to initialize blade hardware...exiting	Communications could not be established with low-level hardware/firmware subsystems.
Invalid option: -?	getopt() returned a failure code.
Net-SNMP version: <version>	Response to the -v and -V command line options.
<error code> = pthread_mutex_lock()	An attempt to lock a Semtech API resource returned the error code. The TCM will attempt to continue executing.
Queue Logging: Creation failed	The queue used by the logging subsystem could not be created.
sendMessageToQueue(): mutex lock failed	An attempt to queue a message for a Semtech API failed. The TCM will attempt to continue executing.
tclk_agent: invalid option -- '<option>'	getopt() detected an invalid command line option.
tclk_agent: option requires an argument -- '<option>'	getopt()did not find a needed argument for a command line option.

Table 6-2 Not Logged Messages (continued)

Message	Meaning
TCM already running	The presence of a prior lock on the file /tmp/TCMLOCKFILE indicates that a copy of the TCM is already running on the blade. Only one copy of the TCM is allowed to execute at a time on a blade. Since the tclk_agent is typically run as a daemon process, the currently executing TCM may not be readily apparent.
TCM: <message to be logged>	The reported message could not be placed on the queue for the logging subsystem. The TCM will attempt to continue executing.
Telco Clocking Module version: <major>.<minor>.<patch>-<build> (<address size> bit)	Response to the -v and -V command line options. The reported address size should correspond to the address size supported by the operating system. Operating a TCM with an address size not supported by the operating system may cause undesired behavior.
ToPSync firmware version: <version>	Response to the -V command line option. The reported version is the firmware version string returned by the ACS9510.
ToPSync hardware version: 0x<version>	Response to the -V command line option. The reported version is the hex hardware version value returned by the ACS9510.
`Unknown parser option to -P: <parameter>.`	An invalid parameter was detected for the -P command line option.
<row of dots> Upload complete	Response to the -u command line option. A dot is printed as each 8k octet block of the CGM firmware is uploaded.
USAGE: tclk_agent [OPTIONS] OPTIONS: <options usage message>	Response to the -h command line option and when an invalid command line option is detected.

6.1.1.2 Critical Messages

Once the logging subsystem has been successfully started, the most severe occurrences within the TCM are reporting using log messages with the critical Severity Level. A critical occurrence may trigger shutdown or otherwise impose major operation limitations.

The following table presents additional detail concerning the various critical messages generated by the TCM. TCM operations are not shut down due to a critical condition unless otherwise indicated.

Table 6-3 Critical Messages

Message	Meaning
AcquireSharedDataLock() failed; shutting down...	An attempt to lock a Semtech API resource failed. The TCM will exit.
<TCM API>: Bad <parameter name> parameter	The named parameter did not have a value expected by the TCM API. Operations for the MIB entity referenced in the TCM API name are likely to fail.
BITS #<BITS port ID>: motDs26503LineTypeSet(<value>) failed	The BITS framer device line type could not be set to the reported value.
cgmProtectionMasterToSlaveTrackDelay: No temperature sensor reading	An attempt to read a blade temperature sensor failed. This message may be reported by either a Master or a Slave. A Slave will not update the Track Delay while this error condition persists. This may cause the skew between the A/B Domain clocks to increase.
<entity>: Cannot bind: <address>:<port>	The entity could not bind to the address and port. This message will be output repeatedly until the configuration of the network changes and the bind succeeds or the tclk_agent is stopped. Refer to the cgmProtectionLocalAddress and cgmProtectionPartnerAddress configuration variables.

Table 6-3 Critical Messages (continued)

Message	Meaning
<entity>: Cannot connect: <address>:<port>	The entity could not establish a communications channel connection using the address and port. This message will be output repeatedly until the configuration of the network changes and the connection succeeds or the tclk_agent is stopped. Refer to the cgmProtectionLocalAddress and cgmProtectionPartnerAddress configuration variables.
<entity>: Connection closed	The connection to the communications channel for the entity was closed unexpectedly.
<MIB entity>: <feature> creation failed	The feature could not be created for the MIB entity. Operations for the MIB entity are likely to fail.
Critical error; details below... <details> ...exiting	The <details> provide information on an operating system service that failed and the reason for failure. The TCM will exit.
<MIB entity>: <process step> dirty flag still set	The process step for the MIB entity could not be completed as expected. Operations for the MIB entity are likely to fail.
Error <code> converting index to OID	An SNMP MIB variable was not recognized.
<TCM API>: Error setting index	The TCM API did not perform as expected. Operations for the MIB entity referenced in the TCM API name are likely to fail.
<TCM API> failed	The TCM API returned a generic failure return code with no additional information provided. Previous failures may reveal causes for the current failure. At a minimum, the failure is likely to cause impaired operations. The failure may trigger subsequent failures which will cause the TCM to exit.
<TCM API>: Failed to <operation>(<entity>): <reason>	The TCM API failed to perform the operation on the entity for the reported reason.
<TCM API>: Failed to mmap():<reason>	The TCM API failed to perform a memory-mapping operation for the reported reason.

Table 6-3 Critical Messages (continued)

Message	Meaning
init_agent_config() failed; exiting...	TCM startup failed. Previous log messages should provide details concerning the failure.
<entity>: Invalid IP: <address>	The address is not a properly formatted IP address for the entity. This message will be output repeatedly until the tclk_agent is stopped to correct the situation. Refer to the cgmProtectionLocalAddress and cgmProtectionPartnerAddress configuration variables.
<TCM API>: ioctl(<operation>) failed: <reason>	The TCM API failed to perform the ioctl operation for the reported reason.
Logical slot: Do not understand: <value>	The value could not be interpreted as a useful blade slot number. This prevents the ATCA Clock Domain from being automatically assigned. Refer to the autoATCAClkDomainEnable configuration variable.
Logical slot: No slot returned: '<command>'	The command failed to return a useful blade slot number. This prevents the ATCA Clock Domain from being automatically assigned. Refer to the autoATCAClkDomainEnable configuration variable.
Memory allocation error: <MIB entity>: <size> bytes	Memory of the reported size could not be reserved for the MIB entity. Operations involving the MIB entity are likely to fail.
<MIB entity>: Memory allocation failed	Memory for the MIB entity could not be reserved. Operations involving the MIB entity are likely to fail.
<MIB entity>: Not enough space for value in datastore	Memory for the MIB entity could not be reserved. Operations involving the MIB entity are likely to fail.
<MIB entity>: NULL varbind data pointer	The internal representation of the MIB entity is not properly formed.
popen(<entity>) failed: <reason>	The TCM failed to open the entity for the reported reason.

Table 6-3 Critical Messages (continued)

Message	Meaning
Queue <queue type>: Creation failed	A queue of the reported type could not be created. The TCM will exit.
ReleaseSharedDataLock() failed; shutting down...	An attempt to release a Semtech API resource failed. The TCM will exit.
Startup failed	TCM startup failed. Previous log messages should provide details concerning the failure. The TCM will exit.
`synce_port_clk_recovery_enable():Failed to system(<system command>):<reason>`	An attempt to enable/disable SyncE clock recovery by invoking a system command failed.
TCM kernel driver: failed: <operation>	The TCM kernel driver failed to perform the operation.
<thread name> thread: TCM Watchdog false alarm...rearming	The named thread detected that the TCM Watchdog triggered unexpectedly. The trigger causes the ATCA Backplane and Extension Shelf clocks to be suppressed. The Watchdog is rearmed, the clocks are unsuppressed, and the TCM continues operations. The trigger may have been caused by excessive load on the CPU. Refer to the TCMWatchdogEnable configuration variable.
<TCM API>: Unhandled columns: 0x<code>	The TCM API did not perform as expected. Operations for the MIB entity referenced in the TCM API name are likely to fail.
<TCM API>: Unknown column: <column number>	The TCM API could not find the column in the SNMP table referenced in the TCM API name. Operations with the table are likely to fail.
<TCM API>: Unknown mode: <code>	The TCM API did not perform as expected. Operations for the MIB entity referenced in the TCM API name are likely to fail.
<TCM API>: Unknown return code: <code>	The TCM API did not perform as expected. Operations for the MIB entity referenced in the TCM API name are likely to fail.

Table 6-3 Critical Messages (continued)

Message	Meaning
ToPSync: PTP port <number>: ConfigurePtpPorts():Boundary Clock ports cannot be in same Domain	It is not supported to have both PTP ports of a Boundary clock configured to be in the same PTP Domain. The TCM will exit.

6.1.1.3 Error Messages

Messages with the error Severity Level identify significant non-nominal operations. Failures occurring when attempting to set/get CGM operating values are examples of significant non-nominal operations. Such failures may indicate that the set/get is with an improper value or the set/get is being performed when the system is in an improper state.

The following table presents additional details concerning the various error messages generated by the TCM. TCM operations are not shut down due to an error condition unless otherwise indicated.

Table 6-4 Error Messages

Message	Meaning
ConfigureFrequencyOutput(line <number>, frequency <number>) failed: <error>: <reason>	Setting the frequency of a ToPSync output line caused an error for the reported reason.
ConfigureMasterPtpPort(port <port number>, enable <True | False>) failed: <error>	Enabling/disabling a PTP port to be a Master failed with the reported error.
ConfigureSlavePtpPort(port <port number>, enable <True | False>) failed: <error>	Enabling/disabling a PTP port to be a Slave failed with the reported error.
Firmware upload: Failed: <failure>: <reason>	Uploading the Semtech firmware failed as indicated for the reported reason.
SETS_I[<clock ID>]: T0: Priority <value>: Set failed	The priority value of the internal CGM input clock for the T0 Path could not be set.
SETS_I[<clock ID>]: T4: Priority <value>: Set failed	The priority value of the internal CGM input clock for the T4 Path could not be set.
SETS_T0[<clock ID>]: Disable failed	The reported internal CGM clock could not be disabled.
SETS_T0[<clock ID>]: Frequency SETSDevice::PLLOutputFrequency::<frequency>: Set failed	The reported internal CGM clock could not be set to the frequency.

Table 6-4 Error Messages (continued)

Message	Meaning
SETS_T0[<clock ID>]: PECL/LVDS pin: Disable failed	The reported internal CGM clock could not be disabled.
<TCM API> failed: <reason>	The TCM API failed for the reported reason.
<TCM API> failed: <failure>: <reason>	The TCM API failed as indicated for the reported reason.
ToPSync error: <Semtech API>: <error>	The ToPSync firmware/software reported an error when the Semtech API was invoked.
ToPSyncAddAcceptablePartner(<IP>/<Priority 1>/<Priority 2>/<ucast | mcast>/<Master | Slave>) failed: <error>	Adding an Acceptable PTP Master/Slave with the indicated IP address, Priority 1&2 values, and unicast/multicast request failed with the reported error.
ToPSyncConfigurePtpPort(port <port number>) failed: <error>	Configuring the PTP port failed with the reported error.
ToPSyncDeconfigurePtpPort(port <port number>) failed: <error>	Deconfiguring the PTP port failed with the reported error.
ToPSyncDeleteAcceptablePartner(<IP>/<Master | Slave>) failed: <error>	Deleting an Acceptable PTP Master/Slave with the indicated IP address failed with the reported error.
ToPSyncGetAcceptablePartnerTableEntry(port <number>, entry <number>) failed: <error>	Fetching an entry in the ToPSync Acceptable Partner Table for a PTP port failed with the reported error.
ToPSyncGetAlarmSet(<alarm ID>) failed: <reason>	The TCM API failed to return the setting for the alarm for the reported reason.
ToPSyncSetEnablePtpPortMasterOperation(port <port number>, enable <True | False>) failed: <error>	The ToPSync API for enabling/disabling a PTP port to be a Master failed with the reported error.
ToPSyncSetEnablePtpPortSlaveOperation(port <port number>, enable <True | False>) failed: <error>	The ToPSync API for enabling/disabling a PTP port to be a Slave failed with the reported error.
ToPSyncSetFilterBandwidths() failed: <error>	Setting the filter bandwidths failed with the reported error.
ToPSyncSetOutputClockFrequencyEnable (line <number>, enable <True | False>) failed: <reason>	The Semtech API failed to enable/disable the internal CGM output line for the reported reason.

Table 6-4 Error Messages (continued)

Message	Meaning
ToPSyncSetProportionalAndIntegralGains() failed: <error>	Setting the gains for the control loop failed with the reported error.
ToPSyncSetPtpPortAddressingMode(port <port number>, addressing mode <As appropriate | Unicast only | Multicast only | Unknown >) failed: <error>	Setting a PTP port transmission addressing mode failed with the reported error.
ToPSyncSetPtpPortEnabled(port <port number>, enable <True | False>) failed: <error>	Enabling/disabling the operations of a PTP port failed with the reported error.
ToPSyncSetLocalOscillatorStratumLevel (stratum<level>) failed: <reason>	The TCM API failed to configure operations for the stratum level for the reported reason.
ToPSyncSetMUXInput(OPClkBlockMux, line <number>, setting <value>) failed: <reason>	The TCM API failed to configure the internal CGM MUX line with a value for the reported reason.
ToPSyncSetMUXInput(PTPBlockMux, line <number>, setting <value>) failed: <reason>	The TCM API failed to configure the internal CGM MUX line with a value for the reported reason.
ToPSyncSetMUXInput(TDMBlockMux, line <number>, setting <value>) failed: <reason>	The TCM API failed to configure the internal CGM MUX line with a value for the reported reason.
ToPSyncSetOutputClockFrequency (line <number>, frequency <value>) failed: <reason>	The TCM API failed to configure the frequency of the internal CGM output line for the reported reason.
ToPSyncSetPtpInputClockEnabled(line <number>, setting <value>) failed: <reason>	The TCM API failed to configure the internal CGM input line with a value for the reported reason.
ToPSyncSetPtpInputClockFrequency (line <number>, frequency <value>) failed: <reason>	The TCM API failed to configure the frequency of the internal CGM input line for the reported reason.
ToPSyncSetPtpInputClockSelected (line <number>) failed: <reason>	The TCM API failed to configure the internal CGM input line for the reported reason.
Unable to create Event with format '<string>': Memory allocation error: message buffer: <size> bytes	The Event handling subsystem could not create an entry for the cgmEventTable due to not being able to reserve the indicated amount of memory. The format string may provide details concerning the Event that is being reported.

6.1.1.4 Warning Messages

In general, log messages with the warning Severity Level occur for non-nominal situations that either may be temporary, easily corrected, or of lesser consequence. Examples of potentially temporary situations are loss of contact with a Protection Partner, sporadic communications bus congestion, Semtech Alarms (both the alarm being set and unset), and Semtech Exceptions (refer to Semtech documentation). Examples of easily corrected conditions are unrecognized configuration variables and values. Examples of situations of lesser consequence are certain anomalous conditions encountered during shutdown.

The following table presents additional details concerning the various warning messages generated by the TCM. TCM operations are not shut down due to a warning condition unless otherwise indicated.

Table 6-5 Warning Messages

Message	Meaning
Acceptable Partner Table may no longer display all current Partners	The number of Partners shown in the SNMP Acceptable Partner Table may not be consistent with the number of Partners tracked by the ToPSync. Either a logic error has occurred in the TCM or another entity such as the ToPSync Network Management application has modified the ToPSync Acceptable Partner Table.
<entity>: Change not supported while CGM is enabled	An attempt was made to change an entity that cannot be changed while the CGM is operating. Examples are Master Mode, Clock Count Mode, ATCA Clock Domain, Oscillator Selection, and the enable status of the BITS/SSU ports.
Ignoring unknown configuration: <configuration file entry>	The unsupported configuration file entry will be replaced by the default value for the configured item.
Ignoring attempt to define more than <number> PTP Acceptable <partner type>	The configuration file defines a PTP Acceptable Partner after the maximum supported number of Partners of that type, Master or Slave, has already been defined.
Protection Partner: Tx failed: <reason>	Transmission of a message to the Protection Partner via UDP failed for the reported reason. Retransmission may be requested by the Partner.
Protection Partner: Tx truncated	The message sent to the Protection Partner via UDP was truncated when transmitted. Retransmission may be requested by the Partner.

Table 6-5 Warning Messages (continued)

Message	Meaning
SPI ControlMsgReceived(): MessageReceived() failed (possible exception raised)	Transmission of the current message from the Protection Partner to the CGM via SPI has failed. Retransmission is attempted several times before transmission of the message is abandoned.
synce_fb_clk_recovery_enable(param fpga_clk_src: <src ID>): unsupported	An attempt was made to enable SyncE clock recovery from a clock source not recognized by the FPGA.
synce_port_clk_recovery_enable():Failed to system (<system command>): <reason>	An attempt to enable/disable SyncE clock recovery by invoking a system command failed.
synce_rtm_clk_recovery_enable(param fpga_clk_src: <src ID>): unsupported	An attempt was made to enable SyncE clock recovery from a clock source not recognized by the RTM FPGA.
<TCM API>: Failed to <operation>(<entity>): <reason>	The TCM API failed to perform the operation on the entity for the reported reason.
TopSync: Exception: Code <entity>: <reason>	A Semtech API has raised an exception.

6.1.1.5 Event Messages

In general, log messages with the **event** Severity Level occur for significant changes in the operations of the TCM. Event messages are related to, but not the same as, Events which are notifications sent to the SNMP Event Table. The relationship between event messages and Events is that Events with an event(2) Severity Code are treated as log messages with the event logging Severity Level.

Currently, there is only one event Severity Level log message that is not also an Event. Refer to the documentation section on Events for details on other event log messages.

The following table presents additional details concerning the event messages generated by the TCM. TCM operations are not shut down due to the notifying event.

Table 6-6 Event Messages

Message	Meaning
cgmProtectionMasterToSlaveTrackDelay: Updated from/to: <old value>/<new value>	The value of the cgmProtectionMasterToSlaveTrackDelay variable has been adjusted to compensate for changes in the temperature of the blade. This action is only performed by a Slave Protection Partner.

6.1.1.6 Informative Messages

Log messages with the **info** Severity Level are meant to keep the user informed of the nominal operational status of the TCM. Typically info messages only occur at startup and shutdown and in response to nominal TCM status changes.

The following table presents additional details concerning the various info messages generated by the TCM.

Table 6-7 Informative Messages

Message	Meaning
Alarm Client:<client ID> asserted/deasserted <alarm name>	The Semtech APIs reported that the named alarm for the Alarm Client was asserted/deasserted. Refer to Semtech documentation for more details on interpreting these messages.
BITS #<BITS port ID>: Asserting <alarm>	The alarm for the indicated BITS framer port ID is being asserted by direct request of the port.
BITS #<BITS port ID>: Asserting <alarm> (Enforced)	The alarm for the indicated BITS framer port ID is being asserted because of the status of the CGM T4 Path lock state. Refer to the bitsT4Monitor column of the cgmBITSTable.
BITS #<BITS port ID>: Clearing <alarm> (T4 is Locked)	The indicated alarm is being cleared for the indicated BITS framer port ID because the CGM T4 Path is now locked to a reference clock.

Table 6-7 Informative Messages (continued)

Message	Meaning
BITS #<BITS port ID>: Deasserting <alarm>	The alarm for the indicated BITS framer port ID is being deasserted.
BITS #<BITS port ID>: Disabling transmitter	The transmitter for the indicated BITS framer port ID is being disabled by direct request of the user.
BITS #<BITS port ID>: Disabling transmitter (Suppressed)	The transmitter for the indicated BITS framer port ID is being disabled because it is being suppressed by the status of the CGM T4 Path lock state. Refer to the bitsT4Monitor column of the cgmBITSTable.
BITS #<BITS port ID>: Disabling transmitter (T4 is Freerun)	The transmitter for the indicated BITS framer port ID is being disabled because the CGM T4 Path is not locked to a reference clock. Refer to the bitsT4Monitor column of the cgmBITSTable.
BITS #<BITS port ID>: Disabling T4 monitor	Monitoring of the CGM T4 Path lock status is being disabled for the indicated BITS framer port ID. Refer to the bitsT4Monitor column of the cgmBITSTable.
BITS #<BITS port ID>: Enabling transmitter	The transmitter for the indicated BITS framer port ID is being enabled.
BITS #<BITS port ID>: Enabling transmitter (T4 is Locked)	The transmitter for the indicated BITS framer port ID is being enabled because the CGM T4 Path is locked to a reference clock. Refer to the bitsT4Monitor column of the cgmBITSTable.
BITS #<BITS port ID>: Raising <alarm> (T4 is Freerun)	The indicated alarm is being raised for the indicated BITS framer port ID because the CGM T4 Path is now not locked to a reference clock.

Table 6-7 Informative Messages (continued)

Message	Meaning
BITS #<BITS port ID>: T4 monitor detected state change	The lock status of the CGM T4 Path or the T1/E1 operational mode for the indicated BITS framer port ID has changed. Refer to the bitsT4Monitor column of the cgmBITSTable.
cgmProtectionMasterToSlaveTrackDelay: Monitoring disabled	The TCM subsystem for automatically adjusting the Master/Slave Track Delay has stopped. Under nominal conditions this should only happen during TCM shutdown. This message will not be reported during shutdown if the output clocks have not yet been unsuppressed during startup.
cgmProtectionMasterToSlaveTrackDelay: Monitoring enabled	The TCM subsystem for automatically adjusting the Master/Slave Track Delay has been initialized. This happens at TCM startup.
cgmProtectionMasterToSlaveTrackDelay: Monitoring shut down	The TCM subsystem for automatically adjusting the Master/Slave Track Delay has been shut down. This happens during TCM shutdown.
Configuring: <configuration file entry>	The TCM is processing the indicated configuration file entry.
<thread> exit: <codes>	The named thread is exiting. Not all started TCM threads log their exit and some threads may log their exit under certain circumstances and not others. Under nominal conditions no thread is expected to exit until the shutdown of the TCM. A single thread may log multiple exit codes. The codes are for SMART EC internal use only.
`FPGA Front Blade version/build code: <build code>`	The version of the FPGA firmware on the TCM Front Blade.The TCM exits if a compatible version is not found. Refer to the *TCMFWOverrideEnable* section.

Table 6-7 Informative Messages (continued)

Message	Meaning
FPGA RTM version/build code: <build code>	The version of the FPGA firmware on the TCM RTM.The TCM exits if a compatible version is not found. Refer to the *TCMFWOverrideEnable* section.
Master: Configured T0 for current Mastership	The TCM has enabled measures specific for a Master to minimize phase shifts during Protection Partner failover. These measures are applied each time the current input for the T0 PLL changes in Stratum 3.
<entity> object: Initializing	The interface to the named entity is being initialized.
Protection Partner: IP: <IP address>	The IP address of the Protection Partner has been set.
Protection Partner: Tx/Rx socket (re)created	The UDP communications path to the Protection Partner has been (re)established.
PTP Clock Operations: <description>	The TCM is configuring for the indicated type of PTP clocking operations.
RTM not detected	At startup the TCM failed to establish communications with an RTM. This is not an issue unless attempts are made to access RTM resources.
`sfptool port <port ID>: SyncE clock recovery <enabled/disabled> [when link <established/disabled>]`	The `sfptool` was invoked to enable/disable recovery of the SyncE clock from the identified port. Recovery on a port without a link will not occur until a link is established.
`sfptool port <port_ID> clock <clock_ID>: Synce clock recovery <enabled/disabled> [when link <established/disabled>]`	The `sfptool` was invoked to enable/disable recovery of the specified SyncE clock from the identified port. Recovery on a port without a link will not occur until a link is established.

Table 6-7 Informative Messages (continued)

Message	Meaning
SPI frame stats: Tx: <Tx count> (<Tx error count> err) Rx: <Rx count> (<Rx error count> err)	The number of SPI message frames transmitted to and received from the ToPSync plus the number of frames that were not transmitted due to an internal error and the number of frames received in error. These statistics are queried from the ToPSync immediately prior to shutdown.
Slave: Configured T0 for future Mastership	The TCM has enabled measures specific for a Slave to minimize phase shifts during Protection Partner failover. These measures are applied each time the current input for the T0 PLL changes in Stratum 3. Also, these measures are applied when a T4 input priority is set for a Slave at runtime.
Starting thread: <thread> (PID: <value>)	The named thread is starting up.The Linux Process ID of the thread is shown.
Stopping thread: <thread>	The named thread is shutting down.
SyncE Clock Generation: <enabled/disabled>	The clock for the Ethernet subsystem is being supplied by the TCM (enabled) or the default Ethernet oscillator (disabled). Refer to the cgmSyncEGenEnable MIB and configuration variables.
TCM kernel driver: Activated	The TCM Linux kernel driver has been initialized.
Telco Clocking Module version: <major>.<minor>.<patch>-<build> startup.(<address size> bit)	The version of the TCM as reported at normal startup. The reported address size should correspond to the address size supported by the operating system. Operating a TCM with an address size not supported by the operating system may cause undesired behavior.
ToPSync: PTP port <number>: Acceptable Master: <IP> (p1=<Priority 1>, p2=<Priority 2>) [ucast]	The ToPSync PTP port has been configured to operate with a PTP Acceptable Master with the indicated IP address, Priority 1&2 values, and (if shown) unicast request.

Table 6-7 Informative Messages (continued)

Message	Meaning
ToPSync: PTP port <number>: Acceptable Slave: <IP>	The ToPSync PTP port has been configured to operate with a PTP Acceptable Slave with the indicated IP address.
ToPSync: PTP port <number>: Domain: <number>	The ToPSync PTP port has been configured to operate in the indicated PTP Domain.
ToPSync: PTP port <number>: Master: <Enabled | Disabled >	The ToPSync PTP port has been enabled/disabled to operate as a PTP Master.
ToPSync: PTP port <number>: Operations: <Enabled | Disabled >	The ToPSync PTP port has been enabled/disabled for PTP operations.
ToPSync: PTP port <number>: Slave: <Enabled | Disabled >	The ToPSync PTP port has been enabled/disabled to operate as a PTP Slave.
ToPSync: PTP port <number>: Transmission addressing mode: <As appropriate | Unicast only | Multicast only | Unknown >	The ToPSync PTP port has been configured to operate using the indicated IP transmission addressing mode.
ToPSync: Shutting down	The CGM is being disabled.
ToPSync SPI control: enabled	The SPI interface to the Semtech CGM has been enabled.
ToPSync: Starting up	The CGM is being enabled.

6.1.2 Signals

The `tclk_agent` responds to Linux signals as summarized in Table 6-8. Any catchable signal not listed in the table is ignored. All other signals have their normal effect.

After the `tclk_agent` is requested to start a task, for example to enable the CGM or process an SNMP command, the response to an intervening Signal will often be delayed until the completion of the task. This may take several seconds. Assuming no lengthy task is underway, the `tclk_agent` should respond to a Signal within a second.

Table 6-8 Signals

Signal	Behavior
SIGINT SIGTERM	Triggers shutdown of the tclk_agent.
SIGHUP	Reserved for logrotate.

6.1.3 Events

Events are notifications of significant occurrences in the `tclk_agent`. Events are placed on the `cgmEventTable` and also logged to the TCM Log.

Events have the following features:

- Event Number (`cgmEventNumber`):
- The sequential ordering number for the Event.
- Event Time (`cgmEventTime`):
- The timestamp for the Event. The timestamp is based on the Linux system time when the Event was generated. The displayed value depends on how the SNMP tool interprets the timestamp.
- Event Code (`cgmEventCode`):
- A unique number assigned to all Events of the same category. Refer to Table 6-9 for the various Event Codes.
- Severity Code (`cgmEventSeverity`):
- An indicator of the significance of the Event. The possible Severity Codes in decreasing order of significance are `critical(0)`, `warning(1)`, `event(2)`, and `info(3)`. When an Event gets logged, the Severity Code is used to define the priority of the log message. The Severity Codes map to Net-SNMP logging priorities as follows:
  - `critical(0)` is LOG_CRIT
  - `warning(1)` is LOG_WARN
  - `event(2)` is LOG_NOTICE
  - `info(3)` is LOG_INFO

- Event Data (`cgmEventData`):

  A 32-bit integer encapsulating information about the Event. In most cases there is no Event Data to encapsulate for the Event. The default value of Event Data is zero. Refer to Table 6-9 for the Event Data associated with an Event.

- Event Description (`cgmEventDescr`):

  A textual description of the Event. Refer to the next table for the Event Description.

The following table summarizes all Events generated by the TCM. Refer to Semtech documentation for further information concerning Events from Semtech CGM APIs.

Table 6-9 Events

Code	Severity	Data	Description and Optional Comments
1	event(2)	none	T0: Input changed From: Input <number> (<name>) To: <number> (<name>) The CGM T0 Path has switched its Reference clock from the old clock to the new clock as indicated. The clock numbers and names refer to the rows of the SNMP Input Table.
2	event(2)	none	T0: Lost lock The CGM T0 Path is no longer locked to a Reference clock.
3	event(2)	none	T0: Gained lock The CGM T0 Path has locked to a Reference clock.
4	event(2)	none	Now Master The reporting TCM has become a Protection Partner Master.
5	event(2)	none	Now Slave The reporting TCM has become a Protection Partner Slave.

Table 6-9 Events (continued)

Code	Severity	Data	Description and Optional Comments
6	event(2)	`0x<hex>` Format of value defaults to 32-bit integer, but SNMP tools may display differently.	TDM Block: Valid inputs changed: `0x<hex>` One of the CGM TDM Block potential Reference clocks has switched from valid to invalid or vice versa. Only a valid input can be used as a Reference clock. The hex number encodes a bit vector representing the current validity of the SNMP Input Table clocks. A valid clock is represented as a 1 bit. An invalid or unsupported clock is represented by a 0 bit. The clocks are listed right-to-left in the bit vector such that the right-most bit is for Input 1 (row 1) of the Input Table and the left-most bit is for the last Input/row.
7	event(2)	none	TCM: T0 selected input failed The currently selected Reference clock for the CGM T0 Path is now considered invalid and cannot be used as a Reference clock in its current state.
10	critical(0)	none	ACS85x0: Failed to initialize Log messages should reveal reasons for the failure.
11	warning(1)	Monitored Clock ID (1 to 36)	Bad frequency The Monitored clock is not within tolerance of the target frequency.
12	event(2)	cgmInputTable row index in bottom 2 octets, CGM-defined State value in top octets.	Input <Input ID> (<Input Name>): New state <New State Description> (was <Old State Description>) The indicated clock of the cgmInputTable changed state as specified.
13	event(2)	none	T4: Input changed From: Input <number> (<name>) To: <number> (<name>) The CGM T4 Path has switched its Reference clock from the old clock to the new clock as indicated. The clock numbers refer to the rows of the SNMP Input Table.
14	event(2)	none	T4: Lost lock The CGM T4 Path is no longer locked to a Reference clock.

Table 6-9 Events (continued)

Code	Severity	Data	Description and Optional Comments
15	event(2)	none	T4: Gained lock The CGM T4 Path has locked to a Reference clock.
16	event(2)	none	T4: Selected input failed The currently selected Reference clock for the CGM T4 Path is now considered invalid and cannot be used as a Reference clock in its current state.
22	event(2)	0 for Disabled 1 for Ready	ToPSync: [Ready|Disabled] Ready status means the CGM is enabled and it can respond to configuration changes. Disabled means the CGM is not operating. The CGM is Disabled when the tclk_agent shuts down and when the user configures cgmEnable to disable(1). **NOTE**: The CGM currently cannot be disabled once enabled without shutting down the tclk_agent.
30	event(2)	BITS ID (1 or 2)	BITS #<BITS ID>: Loss of frame detected
31	event(2)	BITS ID (1 or 2)	BITS #<BITS ID>: Loss of signal detected
32	event(2)	BITS ID (1 or 2)	BITS #<BITS ID>: Receive alarm detected
33	event(2)	BITS ID (1 or 2)	BITS #<BITS ID>: Received yellow alarm detected
34	event(2)	BITS ID (1 or 2)	BITS #<BITS ID>: Loss of frame cleared
35	event(2)	BITS ID (1 or 2)	BITS #<BITS ID>: Loss of signal cleared
36	event(2)	BITS ID (1 or 2)	BITS #<BITS ID>: Receive alarm cleared
37	event(2)	BITS ID (1 or 2)	BITS #<BITS ID>: Received yellow alarm cleared
38	event(2)	BITS ID (1 or 2)	BITS #<BITS ID>: SSM change (0x<new value>)
101	warning(1)	<size>	Memory allocation error: <requester>: <size> bytes
107	event(2)	none	Firmware override in effect For more information,refer to the *TCMFWOverrideEnable* section.

Table 6-9 Events (continued)

Code	Severity	Data	Description and Optional Comments
108	critical(0)	none	Required ToPSync firmware not found: <required version> (or compatible) For more information, refer to the *TCMFWOverrideEnable* section.
109	critical(0)	none	Required FPGA Front Blade firmware not found: <required version> (or compatible) For more information, refer to the *TCMFWOverrideEnable* section.
130	event(2)	none	Startup complete The configuration files have been processed, hardware has been enabled, and the tclk_agent is ready to respond to SNMP commands.
131	event(2)	none	Resetting module The TCM received a SIGHUP Signal, the NetSNMP reconfiguration request. The TCM will start logging to the new logrotate log, but does not otherwise reconfigure.
132	critical(0)	none	Startup failed The TCM will shutdown. Log messages should reveal reasons for the failure.
141	event(2)	none	Shutting down Reported when the user signals the tclk_agent to shut down, or the tclk_agent encounters an error condition and cannot continue operations, or the tclk_agent is invoked to perform a short-term task and the task has come to an end. Refer to the -h, -H, -u, -v and -V command line options in Table 6-10.
142	info(3)	0 for Domain A 1 for Domain B	ATCA Clock Domain [A|B]
143	info(3)	Monitored Clock ID (1 to 36)	Monitored Clock <ID> (<name>): Changed from <freq>Hz to <freq>Hz
144	info(3)	0 for onboard 1 for RTM	Oscillator Selection: <oscillator location>

Table 6-9 Events (continued)

Code	Severity	Data	Description and Optional Comments
145	event(2)	none	Clocks suppressed
146	event(2)	none	Clocks unsuppressed
147	warning(1)	none	Log messages discarded Reported when log messages are being generated faster than they can be logged. This often indicates that the system is in a non-nominal state.
148	warning(1)	none	Events discarded Reported when Events are being generated faster than they can be transferred to the EventTable. This often indicates that the system is in a non-nominal state.
149	critical(0)	none	Required FPGA RTM firmware not found: <required version> (or compatible) For more information, refer to the *TCMFWOverrideEnable* section.
150	event(2)	none	Requesting to Become Master: Have Ref Clock, Partner does not For more information, refer to the *TCMMastershipTimeout* section.
151	event(2)	none	Detected Partnership communications failure Reported when the UDP packets, which are expected from a Protection Partner, do not arrive in a timely manner. The TCM will become a Stand-Alone Master. For more information, refer to the *Stand-Alone Masters* section.
200	info(3)	none	Telco Clocking Ready The CGM has been fully configured as specified by the TCM configuration files.
200	info(3)	none	ACS85x0 Module: Starting Reported as part of the CGM enable.
200	info(3)	none	ToPSync: PTP Block configuration complete Reported as part of the CGM enable.

Table 6-9 Events (continued)

Code	Severity	Data	Description and Optional Comments
200	info(3)	none	Telco Clocking Module version: <major.minor.patch-build> Reported when the `tclk_agent` is started.
200	info(3)	none	Slave: Switching to Master's [T1|E1] mode Reported by the Protection Partner Slave when the Master switches to T1/E1 BITS input mode.
200	info(3)	none	FrequencyValidation: Input <ID>: rollover detected This message comes from the Semtech CGM APIs.
200	info(3)	none	PhaseHitDetector: Clock now <valid|invalid> This message comes from the Semtech CGM APIs.
200	info(3)	none	PhaseHitDetector: State changing from <old state> to <new state> This message comes from the Semtech CGM APIs.
200	info(3)	none	Phase advance is <value> This message comes from the Semtech CGM APIs.
200	info(3)	none	GetClockRawPhaseLag: Unable to read ToPSync clock phase lag This message comes from the Semtech CGM APIs.
200	info(3)	none	IncreaseRequestedClockPhaseLag: Unable to set ToPSync phase build out lag This message comes from the Semtech CGM APIs.
200	info(3)	none	ClearClockInputFilters: Unable to set ToPSync to clear clock input filters This message comes from the Semtech CGM APIs.
200	info(3)	none	ForceHoldover: Unable to force ToPSync time holdover This message comes from the Semtech CGM APIs.
200	info(3)	none	ForceHoldover: Unable to force ToPSync frequency holdover This message comes from the Semtech CGM APIs.
200	info(3)	BITS ID (1 or 2)	BITS #<BITS ID>: Initializing

Table 6-9 Events (continued)

Code	Severity	Data	Description and Optional Comments
1000	warning(1)	none	sysPtpSetsWrite(<register>, <value>): <error message> The Semtech SPI interface to the CGM TDM Registers indicated that a write attempt failed. The write is retried multiple times until either the write succeeds or a maximum number of retries is reached. Refer to the log for messages concerning the failed retries.
1000	warning(1)	none	SETSDevice::Read(<register>): <error message> The Semtech API for reading the CGM TDM Registers indicated failure.
1000	warning(1)	none	SETSDevice::Write(<register>, <value>): <error message> The Semtech API for writing the CGM TDM Registers indicated failure.
1001	critical(0)	none	ToPSync: Failed to initialize The TCM will shutdown. Log messages should reveal reasons for the failure.
1001	critical(0)	none	ToPSync: Failed to re-initialize The TCM will shutdown. Log messages should reveal reasons for the failure.
1001	critical(0)	none	BITS #<BITS ID>: Failed to initialize Log messages should reveal reasons for the failure.
1001	critical(0)	none	Logical slot: Failed to determine The autoATCAClkDomainEnable feature is enabled, but the TCM failed to determine the slot for the F125. The likely reason is that the hpmcmd executable is not properly installed. The ATCA Clock Domain will instead be set based on the cgmATCAClkDomain configuration item.

6.1.4 Command Line Options

The tclk_agent supports the command line options for a typical Net-SNMP sub-agent. There is also an additional option for uploading the ToPSync CGM firmware. The command line options are summarized in the next table.

Table 6-10 Command Line Options

Option	Description
-d	Log all SNMP traffic.
-D TOKEN[,...]	Log SNMP debugging output for specified TOKENs. ALL gives extremely verbose debugging output. Refer to *Net-SNMP documentation* for the names and definitions of possible TOKENs.
-f	Run as a foreground process instead of a background daemon process.
-h	Display help message and exit.
-H	Log list of Net-SNMP configuration file directives and exit. Use in conjunction with the -Lo command line option to see the directives easily.
-L LOGOPTS	Control Net-SNMP logging. Refer to *Net-SNMP documentation* for the LOGOPTS available. Refer to *Logging* for additional details.
-m MIB[:...]	Load list of MIBs. Can be used in conjunction with the standard Net-SNMP environment variable MIBS.
-M DIR[:...]	Look in list of directories for MIBs. Can be used in conjunction with the standard Net-SNMP environment variable MIBDIRS.
-n NAME	Use NAME for sub-agent. The name of the sub-agent determines the names of the sub-agent's configuration files and how the sub-agent is identified within a configuration file. Default name is tclk_agent.
-P MIBOPTS	Control MIB parsing. Refer to *Net-SNMP documentation* for the names and definitions of possible MIBOPTS.
-u FILE:CRC	Upload firmware in FILE to ToPSync and exit. Upload success is verified using CRC. The -f option is ignored. Refer to *CGM Firmware Upload* for additional details.
-v	Display the TCM and Net-SNMP software version numbers on STDERR and immediately exit. The -f and -L options are ignored.
-V	Display the same information as -v, as well as the CGM hardware and firmware version numbers. Automatically runs the tclk_agent in the foreground. This option can take a few seconds to process since the CGM must be booted to retrieve the version information.

Table 6-10 Command Line Options (continued)

Option	Description
-x TRANSPORT	Connect to Master Agent using TRANSPORT. Refer to Net-SNMP documentation for the TRANSPORT options. Defaults to the named socket /var/agentx/master.

Except for the `-u` command line option, refer to *Net-SNMP documentation* for more details concerning the command line options.

6.1.5 CGM Firmware Upload

The Semtech ToPSync CGM firmware should routinely be kept up-to-date. This can be done by using the `tclk_agent -u` command line option to upload the latest binary firmware file to the CGM. The release number of the current version of the ToPSync firmware can be determined by using the `-V` option.

The `-u` option automatically invokes the `-f` option to run the `tclk_agent` in the foreground. If the `-L` option is not specified, then output is sent to STDERR and the usual TCM startup messages are suppressed except errors. Whether the `-L` option is specified or not, the progress and status of the firmware upload process are sent to STDERR.

The `-u` command line option takes a single argument consisting of two elements separated by a colon. The first element is the pathname for the firmware file to be uploaded and the second element is an associated Cyclic Redundancy Check (CRC) number. Refer to *TCM Release Notes* for the firmware file name, release number, and CRC number for the latest release of the file.

The name of the CGM firmware file is controlled by Semtech and may vary by TCM release. The release number for the firmware file is based on Semtech's release numbering scheme, not the scheme used by SMART EC for the BBS or TCM.

The `tclk_agent` starts up as usual when invoked with the `-u` option. Any found configuration files are read and an attempt is made to connect to the snmpd. The settings in the configuration files have no effect on the upload process. After the configuration files are read, the CGM is enabled and the upload process begins. Although the CGM is enabled, no output clocks get generated during the upload process. Assuming there are no errors in finding and opening the file, the upload begins and can take about a minute to complete. A series of dots is given to STDERR to denote the progress of the upload. After the upload has completed the CRC verifies the upload success. Depending on the success of the upload and CRC check, appropriate messages are displayed and the `tclk_agent` exits.

A known issue with the CGM chip may cause the firmware upgrade to report failure on the first attempt. A second attempt should succeed. A failed firmware upgrade will not put the CGM into a hanged or disabled state.

6.1.6 Configuration Overview

The `tclk_agent` is not likely to operate as desired unless it has been properly configured by the user. At a minimum, the user should consider the configuration of the following items:

- TCM ATCA Clock Domain.
  Refer to the *autoATCAClkDomainEnable on page 120* or *cgmATCAClkDomain on page 156*
- TCM PTP Clocking Domains, if any.
  Refer to the *ptpClockDefaultDSDomain* and *ptpBoundaryMasterDomain on page 127ptpClockDefaultDSDomain on page 174* configuration variables.
- TCM mode. Refer to the *TCM Mode on page 45*.
- Sources, frequencies, and priorities of the CGM TDM Reference clocks.
  Refer to the `clkSrc, clkFreq, clkT0Priority, and clkT4Priority` configuration items.
- TCM PTP clock type. Refer to *PTP Operations Mode on page 49*.
- PTP port attributes such as PTP Domain and Acceptable Partners.
  Refer to *PTP Port Data Set Configuration on page 104*.
- Subsystems to be enabled at startup.
  Refer to the *cgmAMCEnable on page 155*, *cgmBITSEnable on page 156* *TCMWatchdogEnable on page 129*, *cgmSyncEGenEnable on page 168*, *ptpEnable on page 127*, and *cgmEnable on page 159* configuration items.
- Master/Slave Protection Partner Pair.
  Refer to the *Protection Partner Pair Configuration on page 93*.
- Stratum Level.
  Refer to the *cgmStratumLevel*, and *Stratum 3 Configuration on page 97* sections.

6.1.7 CGM Ethernet Configuration

The Ethernet interface to the CGM must be configured to support:

The TCM PTP features.

The Semtech ToPSync Manager GUI tools. Refer to *TSGUIEnable on page 130* configuration item for information on enabling usage of these tools with the CGM.

The CGM Ethernet interface is configured with the following items when the CGM is enabled:

1. cgmTSDfltGW
2. cgmTSIPAddress
3. cgmTSMAC
4. cgmTSNetMask

Refer to the *cgmEnable on page 159* MIB object for more information on enabling the CGM.

The CGM Ethernet port is known as interface ge20 to the SRStackware subsystem and is connected to the Base switch. To make the interface visible to an IP network, the appropriate F125 Base switch ports must be enabled and placed in an appropriate VLAN.

6.1.8 Protection Partner Pair Configuration

The expected operational environment for the TCM is to be a member of a Protection Partner Pair. A Protection Partner Pair consists of two TCMs joined to form a Master/Slave configuration. Refer to *Semtech documentation* for more information about Protection Partners and the operations of a Master and Slave.

Both members of a Protection Partner Pair operate as normal TCMs, generating and using clocks as configured. In general, the Partners should be configured redundantly such that all clock consumers outside of a single failed Partner will continue to function nominally. Typically, this means that redundant Reference clocks are supplied to the Partners and that the Partners produce redundant System clocks for the system.

A Protection Partner Pair auto-negotiate mastership such that one will be Master and the other Slave. Should either Partner fail the other will be Master. The Partnership will be restored automatically when the failed Partner gets restored and mastership will be re-negotiated. Refer to *Automatic Recovery on page 60* for more information concerning recovering a failed Partner.

Protection Partner Pair Configuration

Both members of a Protection Partner Pair generate and exchange a Master/Slave synchronization signal. This signal is used to keep the System clocks of the two TCMs in phase with each other, even if one Partner fails and is subsequently restored. Both Partners monitor the validity of the received synchronization signal. In addition, the Slave uses the signal from the Master as its exclusive CGM Reference clock. A Master never uses this signal as a Reference clock. This difference is basically what defines the Master/Slave relationship.

Each TCM of a Protection Partner Pair should be assigned to a different ATCA Clock Domain. Refer to the *autoATCAClkDomainEnable on page 120* and *cgmATCAClkDomain on page 156* configuration items. Furthermore, the interface for a Protection Partner Pair must be configured to support:

1. An Ethernet connection for continuously exchanging UDP packets at regular intervals.
2. The Master/Slave clock synchronization signal.

The Protection Partner Pair Ethernet interface is configured with the following items when the CGM is enabled:

1. cgmProtectionLocalAddress
2. cgmProtectionPartnerAddress
3. cgmProtectionPort

The TCM must be able to open a socket using the `cgmProtectionLocalAddress` and `cgmProtectionPort`. The TCM also should be able to exchange UDP packets with the `cgmProtectionPartnerAddress` and `cgmProtectionPort`.

Important Information

In a test environment, a Protection Partner Pair may not be possible or desirable. In this case, the `cgmProtectionPartnerAddress` should designate a non-enabled Ethernet interface. Otherwise the TCM will continuously log warnings concerning the inability to establish communications with the remote Protection Partner. When a Master/Slave relationship is prevented in this manner, the MODE and LOCK LEDs may not provide reliable status.

There are two possible ways to transfer the Master/Slave synchronization clock. The Master and Slave must both use the same transfer mechanism. These mechanisms are:

1. Using a short Ethernet crossover cable directly connected to the front panel Master/Slave Sync Input ports of the two ATCA-F125s. Such a cable can simultaneously transfer the clock to/from the Master and Slave. In this case the `clkMSSyncIn` clocks of the two TCMs must be sourced from `clkMSRx`. The redundant blade can be in the same or a nearby chassis.

2. Using ATCA Backplane clock CLK1. In this case, the `clkMSSyncIn` clocks of the two TCMs must be sourced from `clkCLK1` and the `cgmFrameSyncMode` must be set to `frameSync(0)`. The Master and Slave use opposite ATCA clock domains for transferring/receiving the clock. No cable is needed, but the redundant blade must be in the same chassis.

Proper phase-alignment of the Master and Slave clocks is only achievable by using CLK1 for the Master/Slave synchronization clock.

In general, a Slave will become a Master under two circumstances, they are:

1. The Slave loses either the Master/Slave sync clock or the UDP communications packets from the Partner.
2. The Master loses all valid Reference clocks, but the Slave does not.

When a Slave becomes a Master the Partner will commonly become a Slave when it gains lock to the Master/Slave synch clock and receives the UDP communications packets from the new Master.

Note that a Slave is restricted in the following operations:

- A Slave T0 CGM PLL will only lock to the `clkMSSyncIn` clock.
- A Slave cannot change its T0 CGM PLL input priorities. Instead, the T0 priorities are automatically forced to match the priorities of the Master. Refer to the `cgmT0InputPriority` column of the *cgmInputTable on page 163*. Unlike the T0, the T4 PLL priorities are not automatically updated from one Protection Partner to the other. To keep these priorities in synchronization, any T4 priority change made to one Protection Partner must also be made to the other.
- A Slave cannot change the Interface Mode of its BITS framers. Instead, the mode is automatically forced to match the mode of the Master. Refer to the *cgmInterfaceMode on page 165* MIB variable.
- A Slave cannot change its CGM Revertive Mode. Instead, the mode is automatically forced to match the Master mode. Refer to the *cgmProtectionRevertiveModeState on page 168* MIB variable.

In addition, changing any of the following items for one Partner also changes it for the other:

- Reference clock frequency
- Master/Slave Track Delay
- Path BandWidth

Given how a Protection Partner can switch between being a Master and Slave over time and how the runtime and startup configuration of one Partner can take precedence over the other, it is recommended that:

- The startup configurations for Protection Partners should be as identical as feasible. This includes:
  – TCM Modes such as Stratum Level, Interface Mode, Master Mode, Clock Count, Revertive Mode, and Frame Sync Mode.
  – T0/T4 Reference clock sources, frequencies, and priorities.
  – System clock destinations and frequencies.
- Runtime configuration changes should be avoided.

6.1.9 Stand-Alone Masters

Following are the three instances when a TCM operates as a Stand-Alone or un-partnered Master Protection Partner:

- When a TCM first starts and before it establishes communications with a Protection Partner.
- When a single member of a Protection Partnership completely fails. The remaining Partner will continue operating, but as a Stand-Alone Master.
- When neither member completely fails, but either the Master/Slave sync clock fails or the UDP packet communications fail. Both TCMs continue operating, but as dual Stand-Alone Masters. In this case the phase relationship of the system clocks from the two TCMs is no longer being maintained and may drift apart.

A Stand-Alone Master operates largely the same as a Master. A Stand-Alone Master accepts a Partnership if offered, attempts to lock to a valid external reference clock if unlocked, and will not lock to the Master/Slave Sync clock if available.

A Stand-Alone Master will remain with suppressed clocks at startup until at least a Partner is discovered and it becomes a Master or Slave. Once a Master or Slave has unsuppressed its clocks the clocks will remain unsuppressed even if the Master or Slave becomes a Stand-Alone Master. An exception is the behavior of `clkOPCLK0` as described in *PTP Operations Overview on page 59*.

If a startup Stand-Alone Master finds a Partner and becomes a Master, then the clocks are immediately unsuppressed. If a startup Stand-Alone Master becomes a Slave, then the clocks are unsuppressed after about two minutes when the clocks are sufficiently aligned with the Partner. If a Master or Slave becomes a Stand-Alone Master after the clocks have been unsuppressed, then the clocks will remain unsuppressed even if the Stand-Alone Master finds a Partner and becomes a Master or Slave.

Dual Stand-Alone Masters do not proactively attempt to restore a Protection Partnership. However, under certain circumstances it is possible for dual Stand-Alone Masters to automatically reform a full Partnership. This may happen, for instance if the UDP communications and/or Master/Slave sync clock are reestablished. In this case the Partners will bring their phase relationship back into alignment. If the phases have drifted sufficiently apart, then there may be a non-nominal phase transition (a phase hit) in the System clocks from the new Slave. A way to avoid the phase hit is to deliberately completely fail and then restart one of the dual Stand-Alone Masters prior to the full Partnership otherwise being restored.

Note that a completely failed TCM does not get restored automatically unless the user configures the operating environment to do so. Refer *Automatic Recovery on page 60* for more details.

6.1.10 Stratum 3 Configuration

In Stratum 3 the T0 and T4 PLLs can be used separately to provide output clocks. Typically, the T0 outputs are used as System clocks and the T4 output, if required, is used as a T4 Reference clock for a BITS/SSU.

Although the T0 and T4 PLLs can select from the same set of Reference clock inputs, the two PLLs should never simultaneously use the same input at the same time. This means that an individual input should not be configured to have a non-zero priority for both PLLs at the same time. Operating in such a manner may cause unexpected behaviors.

The TCM does not automatically restrict attempts to use the same input simultaneously for both PLLs. Thus, care should be taken to ensure that the inputs are not so configured. For instance, an input should not be given a non-zero priority for one PLL unless it already has a zero priority for the other PLL.

Note that runtime T0 priorities changes can only be made for a Master. A Master's new T0 input priority will be updated automatically to the Slave. It is not the same for the T4 inputs. A T4 priority can be changed on either the Master or the Slave and the new priority is not automatically propagated from one Protection Partner to the other.

When the CGM is enabled for PTP operations the TCM automatically operates in Stratum 3 for TDM clocking. Attempts to set a different Stratum Level are ignored.

6.1.11 SyncE Configuration

The Ethernet ports capable of SyncE operation are ETH1 through ETH4 on a front blade and ETH5 through ETH12 on an RTM. All of these ports support both Gb and 10Gb operations. Besides Ethernet ports, a SyncE Reference clock can also be sourced from `clkCLK3`.

If `cgmSyncEGenEnable` is configured to `enable`, then a CGM System clock is used as the source for the Synchronous Ethernet clock for all of the SyncE ports on the front blade and any installed RTM. Neither the specific CGM System clock used as the source nor its frequency are configurable. The frequency of the SyncE clock embedded into the output Ethernet signal is determined by the Ethernet circuitry and is also not configurable.

There are two SyncE clocks that can be used as Reference clocks for the CGM, `clkSyncERefA` and `clkSyncERefB`. No configuration is needed for `clkSyncERefA` and `clkSyncERefB` since their sources are fixed as `clkSyncERcvdRefA` and `clkSyncERcvdRefB`, respectively, and they are always expected to be `8kHz`. However, the sources and frequencies for `clkSyncERcvdRefA` or `clkSyncERcvdRefB` must be configured if `clkSyncERefA` or `clkSyncERefB` are used, respectively. For the available configuration options, refer to *Table 6-11*.

If either `clkSyncERcvdRefA` or `clkSyncERcvdRefB` are sourced from `clkRTMSyncERef`, then this means that the SyncE Reference clock is coming from an RTM port. No configuration is needed for `clkRTMSyncERef` since its source is fixed as `clkRTMSyncERcvdRef1` when sourcing `clkSyncERcvdRefA` and `clkRTMSyncERcvdRef2` when sourcing `clkSyncERcvdRefB`. However, the source and frequency for `clkRTMSyncERcvdRef1` or `clkRTMSyncERcvdRef2` must be configured if `clkSyncERefA` or `clkSyncERefB` are being sourced by `clkRTMSyncERef`, respectively. For the available configuration options, refer to *Table 6-11*.

When the source for a SyncE Reference clock is configured to be a SyncE port, the port is automatically enabled for SyncE clock recovery when `tclk_agent` is started. Clock recovery is automatically disabled when `tclk_agent` shuts down.These port configurations are performed by `tclk_agent` using the standard BBS sfptool.

When sfptool is requested to enable clock recovery, no clock is actually recovered unless a link is established on the port. If the link is subsequently lost, then sfptool automatically disables the clock recovery. If the link is re-established, then sfptool automatically re-enables clock recovery. In this manner the poor quality clock that a PHY may automatically substitute for a missing embedded clock is never used as a source.

Table 6-11 SyncE Configuration

Clock	**Source Configuration Options**		**Frequency Configuration Options**		**Notes**
	Default	Alternate	Default	Alternat e	
clkSyncERefA	clkSyncERcvdRefA	N/A	f8k	N/A	No configuration needed or possible.
clkSyncERefB	clkSyncERcvdRefB	N/A	f8k	N/A	No configuration needed or possible.

Table 6-11 SyncE Configuration (continued)

Clock	**Source Configuration Options**		**Frequency Configuration Options**		**Notes**
	Default	Alternate	Default	Alternate	
clkSyncERcvdRefA	notDriven	clkEthP1 clkEthP2 clkEthP3 clkEthP4 clkCLK3 clkRTMSyncERef	f156M25	f8k f161M13	`clkRTMSyncERcvdRef1` must be configured if `clkRTMSyncERef` is used as source. The frequency when sourcing from `clkEthP1-clkEthP4` depends on the link speed: `f156M25` for 1Gbit and `f161M13` for 10Gbit. The frequency when sourcing from clkRTMSyncERef is always `f8k`.
clkSyncERefB	notDriven	clkEthP1 clkEthP2 clkEthP3 clkEthP4 clkCLK3 clkRTMSyncERef	f156M25	f8k f161M13	`clkRTMSyncERcvdRef2` must be configured if `clkRTMSyncERef` is used as source.The frequency when sourcing from `clkEthP1-clkEthP4` depends on the link speed: `f156M25` for 1Gbit and `f161M13` for 10Gbit. The frequency when sourcing from `clkRTMSyncERef` is always `f8k`.
clkRTMSyncRef	`clkRTMSyncERcvdRef1` when sourcing `clkSyncERcvdRefA`, `clkRTMSyncERcvdRef2` when sourcing `clkSyncERcvdRefB`.	N/A	8KHZ	N/A	No configuration needed or possible.

Table 6-11 SyncE Configuration (continued)

Clock	Source Configuration Options		Frequency Configuration Options		Notes
	Default	Alternate	Default	Alternate	
clkRTMSyncERcvdRef1	notDriven	clkRTMEthP5 clkRTMEthP6 clkRTMEthP7 clkRTMEthP8 clkRTMEthP9 clkRTMEthP10 clkRTMEthP11 clkRTMEthP12	f156M25	f161M13	The frequency depends on the link speed: `f156M25` for 1Gbit and `f161M13` for 10Gbit.
clkRTMSyncERcvdRef2	`notDriven`	clkRTMEthP5 clkRTMEthP6 clkRTMEthP7 clkRTMEthP8 clkRTMEthP9 clkRTMEthP10 clkRTMEthP11 clkRTMEthP12	f156M25	f161M13	The frequency depends on the link speed: `f156M25` for 1Gbit and `f161M13` for 10Gbit.

6.1.12 PTP Configuration Overview

There are two major areas for consideration when configuring the TCM for PTP operations:

The PTP Operations Mode.

Routing of the TDM clock produced from the PTP Reference clock.

The PTP Operation Mode is configured using the following variables:

`ptpEnable`

`ptpClockDefaultDSSlaveOnly`

ptpBoundaryClockEnable

The below table summarizes the configuration of PTP Operations Mode.

Enabling PTP operations and configuring to be neither a Slave-Only nor a Boundary clock is reserved.

				PTP Operations Mode
Variable	ptpEnable	ptpClockDefaultDSSlave Only	ptpBoundaryClockEnable	
Value	disable	N/A	N/A	No PTP operations
	enable	true	N/A	Slave-Only clock
	enable	false	true	Boundary clock
	enable	false	false	This configuration is reserved

The following configuration file entries specify that the TCM will operate as a PTP Slave-Only clock:

```
ptpEnable true
ptpClockDefaultDSSlaveOnly true
```

The following configuration file entries specify that the TCM will operate as a PTP Boundary clock:

```
ptpEnable true
ptpBoundaryClockEnable true
```

If PTP operations are not enabled (the default), then no other PTP configuration is needed. Otherwise, refer to *PTP Slave-Only Clock Configuration on page 102* or *PTP Boundary Clock Configuration on page 102* as appropriate.

If PTP operations are enabled, then the TDM output based on the PTP Reference clock must be properly routed and distributed to meet the needs of the system. For more information, refer to *PTP Routing/Distribution Configuration on page 106*.

6.1.13 PTP Slave-Only Clock Configuration

When operating as a PTP Slave-Only clock the TCM enables port 1 as a PTP Slave clock and port 2 is not enabled. The user must ensure that the Slave port is configured as needed to interact appropriately with PTP nodes on the network.

The TCM configures the Slave port based on the settings of the IEEE1588v2 PTP Default Data Set and PTP Port Data Set. For more information, refer to *PTP Default Data Set Configuration on page 103* and *PTP Port Data Set Configuration on page 104*.

By default a Slave port is configured to use Acceptable Masters. For more information, refer to *PTP Acceptable Partner Configuration on page 105*.

6.1.14 PTP Boundary Clock Configuration

When operating as a PTP Boundary clock the TCM enables port 1 as a PTP Slave clock and port 2 as a PTP Master clock. The user must ensure that both ports are configured as needed to interact appropriately with PTP nodes on the network.

The TCM configures the Slave port based on the settings of the IEEE1588v2 PTP Default Data Set and PTP Port Data Set. For more information, refer to *PTP Default Data Set Configuration on page 103* and *PTP Port Data Set Configuration on page 104*.

The TCM configures the Master port based on the PTP Port Data Set, but not the PTP Default Data Set. Refer to *PTP Port Data Set Configuration on page 104*. In addition, the TCM configures the PTP Domain of the Master port based on the `ptpBoundaryMasterDomain` configuration variable. The domain cannot be the same as used by the Slave port.

By default a Slave port is configured to use Acceptable Masters and a Master port is configured to use Acceptable Slaves. For more information, refer to *PTP Acceptable Partner Configuration on page 105*.

6.1.15 PTP Default Data Set Configuration

The PTP Default Data Set describes startup characteristics of a TCM PTP clock. Refer to *IEEE1588v2 Section 8.2.1*. The following table maps Default Data Set member names to TCM variable names.

Default Data Set Member	TCM Name
twoStepFlag	ptpClockDefaultDSTwoStepFlag
clockIdentity	ptpClockDefaultDSClockIdentity
numberPorts	ptpClockDefaultDSNumOfPorts
clockQuality.clockClass	ptpClockDefaultDSQualityClass
clockQuality.clockAccuracy	ptpClockDefaultDSQualityAccuracy
clockQuality.offsetScaledLogVariance	ptpClockDefaultDSQualityOffset
priority1	ptpClockDefaultDSPriority1
priority2	ptpClockDefaultDSPriority2
domainNumber	ptpClockDefaultDSDomain
slaveOnly	ptpClockDefaultDSSlaveOnly

The only members of the Default Data Set that need to be configured are listed below:

```
ptpClockDefaultDSDomain
ptpClockDefaultDSSlaveOnly
```

The `ptpClockDefaultDSDomain` configuration variable specifies the PTP Domain of a TCM Slave port. The Default Data Set is not used for configuring the Domain of a Boundary Clock Master port. Refer to *PTP Port Data Set Configuration on page 104*.

Refer to *PTP Configuration Overview on page 101* for details concerning the usage of `ptpClockDefaultDSSlaveOnly`.

The remaining members of the Default Data Set do not need to be configured for the reasons described below. None of these members should be referenced in a configuration file or have a value set via SNMP.

Some Default Data Set members specify aspects of operating as a PTP Grandmaster clock. The TCM does not support operating as this clock type and these members are reserved for testing purposes and are not described further.

These members are:

`ptpClockDefaultDSQualityClass`

`ptpClockDefaultDSQualityAccuracy`

`ptpClockDefaultDSQualityOffset`

`ptpClockDefaultDSPriority1`

`ptpClockDefaultDSPriority2`

The remaining members of the Default Data Set do not need to be configured since they either are read-only or the TCM supports only a single pre-defined value for the variable. These members are:

`ptpClockDefaultDSTwoStepFlag`

`ptpClockDefaultDSClockIdentity`

`ptpClockDefaultDSNumOfPorts`

The `ptpClockDefaultDSTwoStepFlag` variable specifies whether the PTP Clock supports operations as a One-Step or Two-Step Clock. The TCM only supports One-Step operations only and this variable should not be changed.

The `ptpClockDefaultDSClockIdentity` variable specifies the PTP Clock Identity of both TCM PTP ports and is not user-configurable. The Clock Identify is derived from the ToPSync MAC address as described by *IEEE1588v2 Section 7.5.2.2.*

The `ptpClockDefaultDSNumOfPorts` variable specifies the number of PTP ports available for the TCM PTP clock. The read-only value is two.

6.1.16 PTP Port Data Set Configuration

The IEEE1588v2 PTP Port Data Set typically does not require configuration for nominal TCM PTP operations. However, the TCM extends the PTP Port Data Set with additional items concerning usage of PTP Acceptable Partners. This usage typically needs configuration for proper PTP operations. For more details, refer to *PTP Acceptable Partner Configuration on page 105*.

Refer to *ptpClockPortDSTable on page 180* for additional information concerning the purpose and usage of the PTP Port Data Set and the extensions made by the TCM.

6.1.17 PTP Acceptable Partner Configuration

Refer to *PTP Acceptable Partner Configuration on page 105* for information concerning the purpose and usage of PTP Acceptable Partners.

PTP Acceptable Partner startup configuration is specified using the following variables.

```
ptpClockPortDSUseAcceptableMasters
ptpClockPortDSUseAcceptableSlaves
ptpAcceptableMasterAddress
ptpAcceptableMasterPriority1
ptpAcceptableMasterPriority2
ptpAcceptableSlaveAddress
```

Each of these variables requires an associated port number in addition to the value for the variable. For example:

```
ptpClockPortDSUseAcceptableMasters 1 true
ptpAcceptableMasterAddress 1 10.156.1.1
```

The ptpAcceptableMasterPriority1 and ptpAcceptableMasterPriority2 variables require an associated port and address in addition to the value for the variable. For example:

```
ptpAcceptableMasterPriority1 1 10.156.1.1 128
```

A port is defined to use Acceptable Masters at startup based on the configuration of `ptpClockPortDSUseAcceptableMasters` for that port. The value true means the port will use Acceptable Masters and the value false means the port will not use Acceptable Masters. Similarly, a port is defined to use Acceptable Slaves at startup based on the configuration of `ptpClockPortDSUseAcceptableSlaves`.

By default a Slave port is configured at startup to use Acceptable Masters and a Master port to use Acceptable Slaves. This means that by default Acceptable Partners must be defined for nominal PTP operations.

Acceptable Masters and Slaves can be configured for a port at startup using the `ptpAcceptableMasterAddress` and `ptpAcceptableSlaveAddress` variables, respectively. The values for these variables are IPv4 dotted-decimal addresses of the associated Acceptable Master or Slave. These variables can be used in a configuration file multiple times to specify multiple Acceptable Partners for a port. For example:

```
ptpAcceptableMasterAddress 1 10.206.1.1
ptpAcceptableMasterAddress 1 10.206.1.2
ptpAcceptableSlaveAddress  2 10.206.1.3
ptpAcceptableSlaveAddress  2 10.206.1.4
```

The above configuration specifies that port 1 has two Acceptable Masters, one with the address `10.206.1.1` and the other with address `10.206.1.2.` Similarly, port 2 is specified to have two Acceptable Slaves with the indicated addresses.

By default, the Acceptable Masters configured using `ptpAcceptableMasterAddress` will have alternative PTP priorities of 0 (zero). In cases where non-zero alternate priorities are desired, Acceptable Masters can be configured for a port using the `ptpAcceptableMasterPriority1` and `ptpAcceptableMasterPriority2` variables. These variables behave the same as `ptpAcceptableMasterAddress` except that both an IPv4 dotted-decimal addresses and an alternate PTP *priority1* or *priority2* are specified for the Acceptable Master. For example:

```
ptpAcceptableMasterPriority1 1 10.206.1.1 128
ptpAcceptableMasterPriority1 1 10.206.1.2 129
```

The above configuration specifies that port 1 has two Acceptable Masters, one with the address `10.206.1.1` and an alternate PTP priority1 of 128 and the other with address `10.206.1.2` and an alternate PTP priority1 of 129.

By default all Acceptable Masters are requested to use unicast Announce messages. This configuration cannot be changed.

The number of Acceptable Partners for a single port is restricted to no more than 10 Acceptable Masters and 64 Acceptable Slaves. There is also a restriction of no more than 74 Acceptable Partners for all ports combined.

Duplicate Acceptable Partners do not count toward the above totals. An Acceptable Partner is a duplicate if it another Acceptable Partner of the same type, Slave or Master, and with the same address has previously been specified for the same port. If multiple *priority1* values are configured for the same Acceptable Master for the same port, then only the last *priority1* configuration value will apply. The same applies for multiple *priority2* values.

6.1.18 PTP Routing/Distribution Configuration

The TCM PTP clocks are produced only by the PTP Master port and are distributed only to the PTP Slaves for the port. Refer to *PTP Boundary Clock Configuration on page 102* for details on distributing the PTP clocks to the PTP Slaves.

When the TCM Slave port has a PTP Master, the associated PTP Reference clock is used to produce a single TDM clock. This clock is not directly available to the user and thus has no TCM name. For our purposes here the clock is known as the PTP TDM clock.

The PTP TDM Clock is fixed at 8kHz and is automatically routed to drive `clkOPCLK0`. The only configuration options for `clkOPCLK0` are to use it as the source for either or both of the System clocks `clkUsr1` and `clkUsr2`.

Various options are available for routing and distributing `clkUsr1` and `clkUsr2` to clock consumers. A recommended distribution configuration is shown next.

This distribution has the following desirable hallmarks:

- It maintains consistency with the TDM operations of the TCM by routing the PTP TDM clock through the T0 PLL. The PTP TDM clock is treated like any other TDM Reference clock and no special handling or other considerations are needed for concerns such as clock validity, reference reversion, Mastership switching, and system failover.
- It allows the TCM Protection Partners to access the PTP TDM clocks from both TCMs as redundant backups of each other. This is performed using the ATCA Domain mechanism.
- It allows each Protection Partner to know the status of the other Partner's PTP Reference clock. This is a result of the automatic suppression of the `clkOPCLK0` output when there is no PTP Master for the Slave port.
- It distributes the PTP TDM clock to clock consumers other than the TCM blades. This is performed using the ATCA backplane.

Although other distribution configurations are possible, the recommended configuration is the only one known to have all of the above desirable hallmarks and is the only PTP distribution configuration tested by SMART EC.

It may seem that the recommend configuration creates a counter-productive clock loop whereby a System clock is used as its own Reference clock. While it is the case that the PTP TDM clock is routed out of the CGM as a System clock and then routed back to the CGM as a Reference clock, no loop is formed since the producer and consumer are not the same PLL. In this case the PTP PLL is the producer and the T0 PLL is the consumer.

The recommended PTP routing and distribution configuration is as follows. Note that the configuration for other aspects of the TCM such as enabling PTP operations, selecting the PTP clock type, and enabling the CGM is not shown. For complete examples on PTP configuration refer to *Example TCM Configuration on page 187*.

```
################################################
#  PTP routing and distribution configuration  #
################################################
# Use the Local Generation Master mode as the general
# clock distribution scheme
cgmMasterMode localClkGen
# Use the Three Clocks Clock Count mode to access
# all three ATCA clocks
cgmClockCount threeClks
# Use our CLK3 System Clock also as a Reference Clock
cgmRefLoopbackEnable enable
```

```
# Source both Reference Clocks RefA and RefB from CLK3
# One reference will be from the local ATCA Domain and
# the other from the remote
clkSrc clkRefA clkCLK3
clkSrc clkRefB clkCLK3
# Use the PTP TCM Clock (source for clkOPCK0)
# as the source for the clkUsr1 System Clock
# clkUsr1 is the source for CLK3 when in
# Local Generation Mode + Three Clocks Mode
clkSrc clkUsr1 clkOPCLK0
```

6.1.19 Configuration Files

The `tclk_agent` conforms to the standard Net-SNMP configuration protocols for file locations and names, syntax, search order, environment variables, etc. Refer to *Net-SNMP documentation* and in particular the SNMP_CONFIG man page for more information about configuring Net-SNMP features and the standard configuration protocols. Although not required, a typical location for the `tclk_agent` configuration files is in the `.snmp` subdirectory of the user's home directory. Note the dot (.) prefix in the name of the subdirectory.

The snmpd supports a command line option `-c` to specify the pathname of a configuration file to load. This is not the standard for a sub-agent. Instead, the `tclk_agent` supports the `-n` command line option to specify a sub-agent name. The sub-agent name is used to identify the base name of configuration files searched for in the standard Net-SNMP directories. In this manner different configuration files can be specified at runtime.

Important Information

Specifying a sub-agent name with the -n command line option does not affect the name of the sub-agent executable. The executable is always `tclk_agent`.

The default name of the sub-agent is `tclk_agent`. Assuming the default name, the configuration files specific to the sub-agent name are:

```
tclk_agent.conf
tclk_agent.local.conf
```

If the `-n` command line option were used to change the sub-agent name to TCM, then these sub-agent configuration files would be `TCM.conf` and `TCM.local.conf`, respectively. Any configuration information in these name-specific files pertains only to the named sub-agent. It is recommended that the `.local` configuration file is used to configure only Net-SNMP features and the non-local configuration file is used to configure TCM features.

Sub-agent configuration is also controlled by configuration files not specific to the sub-agent name. These files are:

```
snmp.conf
snmp.local.conf
agentx.conf
agentx.local.conf
```

Configuration information in these files can possibly pertain to other SNMP entities. Configuration meant for a specific entity must be identified by prefixing the information with the name of the entity in square brackets (e.g., `[tclk_agent]`). If the prefix is alone on a line, then all the subsequent configuration information will be identified with the named entity as shown below:

```
[tclk_agent]
cgmEnable enable
clkSrc clkRefA clkCLK3
…
```

Configuration information for a named entity also can be identified on a line-by-line basis as shown below:

```
[tclk_agent] cgmEnable enable
[tclk_agent] clkSrc clkRefA clkCLK3
```

The above configuration files not specific to the `tclk_agent` traditionally are used to configure global Net-SNMP information pertaining to all Net-SNMP entities. It is recommended that these files are not used to configure information specific to the `tclk_agent`.

6.1.20 Configuration Items

Many TCM configuration items are also MIB objects and share the same names. Refer to *Chapter 7, MIBs* for more information.

Only configuration items which are not also MIB objects are described in this section. Refer to *Chapter 7, MIBs* for other possible configuration items. The order in which items are configured is irrelevant.

There are two types of TCM configurable items, simple and complex. A simple configurable item has only a single configurable feature while a complex item has multiple configurable features. The `cgmTSDfltGW` configuration item is an example of a simple item. Its sole feature is to specify the Default Gateway for the ToPSync CGM. A clock (e.g., `clkRefA`) is an example of a complex configurable item. It has multiple configurable features such as a source, a frequency, a CGM T0 Path Priority, etc.

A simple item is configured by listing the name of the item followed by the configuration value for the item. This is how the `cgmTSDfltGW` is configured to the IP 10.206.154.56:

```
cgmTSDfltGW 10.206.154.56
```

A complex item is configured by listing the name of the feature being configured, the configuration item, and the configuration value. A clock has the following configurable feature names:

```
clkSrc - clock source
clkFreq - clock frequency
clkT0Priority - CGM T0 Path Priority
clkT4Priority - CGM T4 Path Priority
```

The source for the RefA clock is configured to the ATCA CLK3 clock as shown below:

clkSrc clkRefA clkCLK3

Table 6-12 summarizes various features of the TCM configuration items.

Note the following concerning *Table 6-12*:

- The **Configuration Item** column lists a configurable TCM item. This can be an item name for configuring a simple item or a feature name plus item name for configuring a complex item.
- A configuration item with the prefix `cgm` may become a MIB object in a future release.
- The **Value Range** column lists the possible values for the item. An asterisk following a value indicates it is the default value.
- The **Related SNMP Object** column lists any TCM MIB object that supports the same feature as the configuration item. This means that configuring the item is similar to doing an SNMP Set on the object. For instance, this object might be the column of a table used to Get/Set the same feature of a clock.

Table 6-12 Configuration Items

Configuration Item	Value Range	Related SNMP Object
autoATCAClkDomainEnable	enable disable*	cgmATCAClkDomain
CLIEnable	enable disable*	
cgmProtectionLocalAddress	<IP address> 192.168.21.110*	
cgmProtectionPartnerPort	<port number> 8123*	

Table 6-12 Configuration Items (continued)

Configuration Item	Value Range	Related SNMP Object
cgmRefLoopbackEnable	enable `disable*` Enabling this item is only supported when PTP operations are enabled and the recommended configuration documented in *Table 6.1.18* is followed.	
cgmTSDfltGW	<IP address> `192.168.21.1*`	
cgmTSIPAddress	<IP address> `192.168.21.110*`	
cgmTSMAC	<MAC address> <Semtech-assigned MAC address>*	
cgmTSNetMask	<IP netmask> `255.255.255.0*`	
clkFreq clkOPCLK0	`notDriven*` (default when PTP operations not enabled) fSync2k fSync8k* (default and only supported value when PTP operations enabled) fT0Dig1 fT0Dig2 fT0AlogDiv4 fT0AlogDiv6 fT0AlogDiv8 fT0AlogDiv12 fT0AlogDiv16 fT0AlogDiv48 fT4AlogDiv4 fT4AlogDiv8 fT4AlogDiv16 fT0AlogDiv48 fT4AlogDiv64	

Table 6-12 Configuration Items (continued)

Configuration Item	Value Range	Related SNMP Object
clkFreq clkRefA	f8k* f1M544 f2M048 f19M44	cgmInputFrequency of cgmInputTable
clkFreq clkRefB	same as for clkRefA	cgmInputFrequency of cgmInputTable
clkFreq clkRTMSyncERcvdRef1	f156M25* f161M13 Reserved: f8k	
clkFreq clkRTMSyncERcvdRef2	f156M25* f161M13 Reserved: f8k	
clkFreq clkSyncERcvdRefA	f8k f25M f156M25* f161M13	
clkFreq clkSyncERcvdRefB	f8k f25M f156M25* f161M13	
clkFreq clkTCLKB	f8k* f1M544 f2M048 f19M44	cgmInputFrequency of cgmInputTable

Table 6-12 Configuration Items (continued)

Configuration Item	Value Range	Related SNMP Object
clkSrc clkMSSyncIn	`notDriven*` clkMSRx clkCLK1 Reserved: clk8kTest	
clkSrc clkRefA	`notDriven*` clkCLK3 clkExt1P2 clkExt2P2 clkExt3P2 clkExt4P2 clkExt5P2 (clkCLK3 is only available in 2 Clock Count mode; other above clocks are available only in 2 or 3 Clock Count mode) Reserved: clk8kTest clk8kBITS1Test clk19M44BITS1Test clkCLK1Opp clkCLK2Opp clkCLK3Opp	

Table 6-12 Configuration Items (continued)

Configuration Item	Value Range	Related SNMP Object
clkSrc clkRefB	notDriven* clkCLK3 clkExt1P4 clkExt2P4 clkExt3P4 clkExt4P4 clkExt5P4 (above sources are only available in 2 Clock Count mode) Reserved: clk8kTest clk8kBITS1Test clk19M44BITS1Test clkCLK1Opp clkCLK2Opp clkCLK3Opp	
clkSrc clkRTMSyncERcvdRef1	notDriven* clkRTMEthP5 clkRTMEthP6 clkRTMEthP7 clkRTMEthP8 clkRTMEthP9 clkRTMEthP10 clkRTMEthP11 clkRTMEthP12 Reserved: clk8kTest	

Table 6-12 Configuration Items (continued)

Configuration Item	Value Range	Related SNMP Object
clkSrc clkRTMSyncERcvdRef2	notDriven* clkRTMEthP5 clkRTMEthP6 clkRTMEthP7 clkRTMEthP8 clkRTMEthP9 clkRTMEthP10 clkRTMEthP11 clkRTMEthP12 Reserved: clk8kTest	
clkSrc clkSETSPTP	clkTCLKD clkPTPOut3* Reserved: clk8kTest	
clkSrc clkSyncERcvdRefA	notDriven* clkEthP1 clkEthP2 clkEthP3 clkEthP4 clkCLK3 clkRTMSyncERef Reserved: clk8kTest clkBaseC0 clkBaseC1	
clkSrc clkSyncERcvdRefB	same as for clkSyncERcvdRefA	

Table 6-12 Configuration Items (continued)

Configuration Item	Value Range	Related SNMP Object
clkSrc clkUsr1	clkOPCLK0* clkOPCLK2 clkFrameSync clkMFrameSync clkBIT1Rx Future releases: clkOnePPS clkPhaseOPCLK clkDivPhaseOPCLK	
clkSrc clkUsr2	clkOPCLK0 clkOPCLK2* clkFrameSync clkMFrameSync clkBIT1Rx Future releases: clkOnePPS clkPhaseOPCLK clkDivPhaseOPCLK	
clkT0Priority clkBITS1Rx	0*,2..15	cgmT0InputPriority of cgmInputTable
clkT0Priority clkBITS2Rx	0*,2..15	cgmT0InputPriority of cgmInputTable
clkT0Priority clkRefA	0*,2..15	cgmT0InputPriority of cgmInputTable
clkT0Priority clkRefB	0*,2..15	cgmT0InputPriority of cgmInputTable
clkT0Priority clkSETSPTP	0*,2..15	cgmT0InputPriority of cgmInputTable
clkT0Priority clkSyncERefA	0*,2..15	cgmT0InputPriority of cgmInputTable

Table 6-12 Configuration Items (continued)

Configuration Item	Value Range	Related SNMP Object
clkT0Priority clkSyncERefB	0*,2..15	cgmT0InputPriority of cgmInputTable
clkT0Priority clkTCLKB	0*,2..15	cgmT0InputPriority of cgmInputTable
clkT4Priority clkBITS1Rx	0*,2..15	cgmT4InputPriority of cgmInputTable
clkT4Priority clkBITS2Rx	0*,2..15	cgmT4InputPriority of cgmInputTable
clkT4Priority clkRefA	0*,2..15	cgmT4InputPriority of cgmInputTable
clkT4Priority clkRefB	0*,2..15	cgmT4InputPriority of cgmInputTable
clkT4Priority clkSETSPTP	0*,2..15	cgmT4InputPriority of cgmInputTable
clkT4Priority clkSyncERefA	0*,2..15	cgmT4InputPriority of cgmInputTable
clkT4Priority clkSyncERefB	0*,2..15	cgmT4InputPriority of cgmInputTable
clkT4Priority clkTCLKB	0*,2..15	cgmT4InputPriority of cgmInputTable
ptpAcceptableMasterAddress <PTP Port number 1 or 2>	<IPv4 dotted decimal address of Acceptable Master> There is no default value.	ptpClockPortDSAcceptablePartnerAddressAdd for specified PTP port
ptpAcceptableMasterAddressType <PTP Port number 1 or 2>	Configuring this variable is reserved.	ptpClockPortDSAcceptablePartnerAddressType for specified PTP port
ptpAcceptableMasterClockIdentity <PTP Port number 1 or 2>	Configuring this variable is reserved.	

Table 6-12 Configuration Items (continued)

Configuration Item	Value Range	Related SNMP Object
ptpAcceptableMasterPriority 1 <PTP Port number 1 or 2> <IPv4 dotted decimal address of Master>	0*..255	ptpClockPortDSAcceptablePartnerPriority1 for specified PTP port
ptpAcceptableMasterPriority 2 <PTP Port number 1 or 2> <IPv4 dotted decimal address of Master>	0*..255	ptpClockPortDSAcceptablePartnerPriority2 for specified PTP port
ptpAcceptableMasterRqstUnicastAnnounce <PTP Port number 0 or 1> <IPv4 dotted decimal address of Master>	Configuring this variable is reserved.	ptpClockPortDSAcceptablePartnerRqstUnicast for specified PTP port
ptpAcceptableMasterTransmissionProtocol <PTP Port number 1 or 2> < IPv4 dotted decimal address of Master>	Configuring this variable is reserved.	
ptpAcceptableSlaveAddress <PTP Port number 1 or 2>	<IPv4 dotted decimal address of Acceptable Slave> There is no default value.	ptpClockPortDSAcceptablePartnerAddressAdd for specified PTP port
ptpBoundaryClockEnable	enable disable*	
ptpBoundaryMasterDomain	default* alt1 alt2 alt3	ptpClockDefaultDSDomain
ptpEnable	enable disable*	
TCMCoarsePhaseLossLimit	1..1023*..4097	
TCMCritThreadPriority	-1000..45*..1000	
TCMCritThreadSched	SCHED_FIFO SCHED_OTHER SCHED_RR*	

Table 6-12 Configuration Items (continued)

Configuration Item	Value Range	Related SNMP Object
TCMFWOverrideEnable	enable disable*	
TCMMasterShipTimeout	0..150*..2147483647 seconds	
TCM_S3_PathBandwidth	f500uHz f1mHz f2mHz f4mHz f8mHz f15mHz f30mHz f60mHz f100mHz* f300mHz f600mHz f1_2Hz f2_5Hz f4Hz f8Hz f18Hz f35Hz f70Hz	
TCM_S3E_PathBandwidth	same as for TCM_S3_PathBandwidth	
TCMUDPCtrlAddress	This variable is reserved for SMART EC testing.	
TCMWatchdogEnable	enable* disable	
TSGUIEnable	enable disable*	

6.1.20.1 autoATCAClkDomainEnable

At startup, `autoATCAClkDomainEnable` enables or disables the automatic assignment of the ATCA Clock Domain based on the logical slot number of the board. If enabled, a board in logical slot 1 is assigned to Domain A and a board in any other slot is assigned to Domain B. This assignment will override any assignment specified using the `cgmATCAClkDomain` configuration item.

If the logical slot of the board cannot be determined, then the Domain will be set based on the `cgmATCAClkDomain` configuration item. The most likely reason for failing to determine the logical slot is that the `hpmcmd` executable is not properly installed.

6.1.20.2 cgmRefLoopbackEnable

This configuration variable is used to enable/disable the Reference Loopback Mode.

Enabling Reference Loopback Mode is only supported when PTP operations are enabled and the recommended configuration documented in *PTP Routing/Distribution Configuration on page 106* is followed. All other uses of this variable are reserved. Misuse of this variable may cause undesired TCM behaviors.

6.1.20.3 CLIEnable

At startup, it enables or disables a simple Command Line Interface (CLI) to the `tclk_agent` for peeking and poking the CGM TDM registers, FPGA registers, and the two sets of DS26503 BITS Framer registers. This feature is intended only for debugging purposes and is not a supported feature of the TCM. Refer to *Semtech documentation* concerning the ToPSync ACS9510 TDM registers. Refer to *Maxim documentation* concerning the DS26503 registers. The FPGA registers are not documented for user access.

To ensure successful communications with the CGM TDM registers, a SPI tool external to the `tclk_agent` should not be used to interact with the CGM while the `tclk_agent` is running.

The `tclk_agent` must be run as a foreground process to make the CLI accessible. Refer to the `-f` command line option in Table 6-10. If the `tclk_agent` log messages are going to STDOUT, then the messages will be interspersed with CLI interactions. Refer to the `-L` command line option in Table 6-10.

Depending on the mode, the CLI presents the user with one of the following prompts on STDOUT:

```
TDM CLI>
FPGA CLI>
BITS-1 CLI>
BITS-2 CLI>
```

The user can enter through STDIN any one of the following commands followed by a carriage return:

```
t
f
1
2
r <whitespace> <addr>
r <whitespace> <addr1> <whitespace> <addr2>
w  <whitespace> <addr> <whitespace> <value>
```

Any other input will result in an error message.

The command components are as follows:

`t` is for switching to the TDM register peek/poke mode.

`f` is for switching to the FPGA register peek/poke mode.

`1` is for switching to the BITS Framer #1 register peek/poke mode.

`2` is for switching to the BITS Framer #2 register peek/poke mode.

`r` is for reading.

`w` is for writing.

<whitespace> can be any combination of blank characters and tabs.

<addr>, <addr1>, <addr2>, and <value> are hexadecimal numerals and depending on context may range from 0x0 to 0xff. The 0x is optional and is the same as 0X. The hexadecimal digits a..f may also be A..F.

<addr>, <addr1>, and <addr2> are register addresses for the device selected by the current mode.

<addr1> and <addr2> are a range of registers to be read, starting at <addr1> and ending at <addr2>.

<value> is the value to be written to the specified <addr>.

A response to a read is to show the address of the requested register followed by the corresponding value that was read:

```
0x48: 0x04
```

If a range of registers was read, then each read result is shown on a separate line. A response to a write is to show the result of reading the same address that was written.

6.1.20.4 cgmProtectionLocalAddress

The configuration item `cgmProtectionLocalAddress` is the IP address of the local Protection Partner. The local Protection Partner uses this address to exchange UDP packets with a remote Protection Partner. Refer to *Protection Partner Pair Configuration on page 93*.

6.1.20.5 cgmProtectionPartnerPort

The configuration item `cgmProtectionPartnerPort` is the port number for both the local and remote members of a Protection Partner Pair. The Protection Partners use this port to exchange UDP packets with each other. Refer to *Protection Partner Pair Configuration on page 93*.

6.1.20.6 cgmTSDfltGW

The configuration item `cgmTSDfltGW` is the IP for the CGM Default Gateway. Refer to *CGM Ethernet Configuration on page 93*.

6.1.20.7 cgmTSIPAddress

The configuration item `cgmTSIPAddress` is the IP for the CGM. Refer to *CGM Ethernet Configuration on page 93*.

6.1.20.8 cgmTSMAC

The configuration item `cgmTSMAC` is the MAC address for the CGM. Refer to *CGM Ethernet Configuration on page 93*.

6.1.20.9 cgmTSNetMask

The configuration item `cgmTSNetMask` is the IP netmask for the CGM. Refer to *CGM Ethernet Configuration on page 93*.

6.1.20.10 clkFreq clkOPCLK0

The configuration item `clkFreq clkOPCLK0` is the desired frequency for `clkOPCLK0`. When PTP operations are enabled this configuration item is ignored and the `clkOPCLK0` frequency is always `fSync8k`. Refer to *Clock Frequency on page 40* for more information.

6.1.20.11 clkFreq clkRefA

The configuration item `clkFreq clkRefA` is the frequency `clkRefA` is expected to be. Refer to *Clock Frequency on page 40* for more information.

6.1.20.12 clkFreq clkRefB

The configuration item `clkFreq clkRefB` is the frequency `clkRefB` is expected to be. Refer to *Clock Frequency on page 40* for more information.

6.1.20.13 clkFreq clkRTMSyncERcvdRef1

The configuration item `clkFreq clkRTMSyncERcvdRef1` is the frequency `clkRTMSyncERcvdRef1` is expected to be. Refer to *Clock Frequency* for more information.

6.1.20.14 clkFreq clkRTMSyncERcvdRef2

The configuration item `clkFreq clkRTMSyncERcvdRef2` is the frequency `clkRTMSyncERcvdRef2` is expected to be. Refer to *Clock Frequency* for more information.

6.1.20.15 clkFreq clkSyncERcvdRefA

The configuration item `clkFreq clkSyncERcvdRefA` is the frequency `clkSyncERcvdRefA` is expected to be. Refer to *Clock Frequency* for more information.

6.1.20.16 clkFreq clkSyncERcvdRefB

The configuration item `clkFreq clkSyncERcvdRefB` is the frequency `clkSyncERcvdRefB` is expected to be. Refer to *Clock Frequency* for more information.

6.1.20.17 clkFreq clkTCLKB

The configuration item `clkFreq clkTCLKB` is the frequency `clkTCLKB` is expected to be. Refer to *Clock Frequency* for more information.

6.1.20.18 clkSrc clkMSSyncIn

The configuration item `clkSrc clkMSSyncIn` is the source for `clkMSSyncIn`.

6.1.20.19 clkSrc clkRefA

The configuration item `clkSrc clkRefA` is the source for `clkRefA`.

6.1.20.20 clkSrc clkRefB

The configuration item `clkSrc clkRefB` is the source for `clkRefB`.

6.1.20.21 clkSrc clkRTMSyncERcvdRef1

The configuration item `clkSrc clkRTMSyncERcvdRef1` is the source for `clkRTMSyncERcvdRef1`.

6.1.20.22 clkSrc clkRTMSyncERcvdRef2

The configuration item `clkSrc clkRTMSyncERcvdRef2` is the source for `clkRTMSyncERcvdRef2`.

6.1.20.23 clkSrc clkSETSPTP

The configuration item `clkSrc clkSETSPTP` is the source for `clkSETSPTP`. Refer to Stratum Level for more information about using this clock source.

6.1.20.24 clkSrc clkSyncERcvdRefA

The configuration item `clkSrc clkSyncERcvdRefA` is the source for `clkSyncERcvdRefA`.

6.1.20.25 clkSrc clkSyncERcvdRefB

The configuration item `clkSrc clkSyncERcvdRefB` is the source for `clkSyncERcvdRefB`.

6.1.20.26 clkSrc clkUsr1

The configuration item `clkSrc clkUsr1` is the source for `clkUsr1`.

6.1.20.27 clkSrc clkUsr2

The configuration item `clkSrc clkUsr2` is the source for `clkUsr2`.

6.1.20.28 clkT0Priority clkBITS1Rx

The configuration item `clkT0Priority clkBITS1Rx` is the CGM T0 Path priority for `clkBITS1Rx`.

6.1.20.29 clkT0Priority clkBITS2Rx

The configuration item `clkT0Priority clkBITS2Rx` is the CGM T0 Path priority for `clkBITS2Rx`.

6.1.20.30 clkT0Priority clkRefA

The configuration item `clkT0Priority clkRefA` is the CGM T0 Path priority for `clkRefA`.

6.1.20.31 clkT0Priority clkRefB

The configuration item `clkT0Priority clkRefB` is the CGM T0 Path priority for `clkRefB`.

6.1.20.32 clkT0Priority clkSETSPTP

The configuration item clkT0Priority clkSETSPTP is the CGM T0 Path priority for the clkSETSPTP.

6.1.20.33 clkT0Priority clkSyncERefA

The configuration item clkT0Priority clkSyncERefA is the CGM T0 Path priority for clkSyncERefA.

6.1.20.34 clkT0Priority clkSyncERefB

The configuration item clkT0Priority clkSyncERefB is the CGM T0 Path priority for clkSyncERefB.

6.1.20.35 clkT0Priority clkTCLKB

The configuration item clkT0Priority clkTCLKB is the CGM T0 Path priority for the clkTCLKB.

6.1.20.36 clkT4Priority clkBITS1Rx

The configuration item clkT4Priority clkBITS1Rx is the CGM T4 Path priority for clkBITS1Rx.

6.1.20.37 clkT4Priority clkBITS2Rx

The configuration item clkT4Priority clkBITS2Rx is the CGM T4 Path priority for clkBITS2Rx.

6.1.20.38 clkT4Priority clkRefA

The configuration item clkT4Priority clkRefA is the CGM T4 Path priority for clkRefA.

6.1.20.39 clkT4Priority clkRefB

The configuration item clkT4Priority clkRefB is the CGM T4 Path priority for clkRefB.

6.1.20.40 clkT4Priority clkSETSPTP

The configuration item clkT4Priority clkSETSPTP is the CGM T4 Path priority for the clkSETSPTP.

6.1.20.41 clkT4Priority clkSyncERefA

The configuration item `clkT4Priority clkSyncERefA` is the CGM T4 Path priority for `clkSyncERefA`.

6.1.20.42 clkT4Priority clkSyncERefB

The configuration item `clkT4Priority clkSyncERefB` is the CGM T4 Path priority for `clkSyncERefB`.

6.1.20.43 clkT4Priority clkTCLKB

The configuration item `clkT4Priority clkTCLKB` is the CGM T4 Path priority for the `clkTCLKB`.

6.1.20.44 ptpAcceptableMasterAddress <port>

This configuration variable is used to add the address of an Acceptable Master to the Acceptable Partner Table for the designated port. For more details, refer to *PTP Acceptable Partner Configuration on page 105*.

6.1.20.45 ptpAcceptableMasterAddressType <port>

This configuration variable is reserved.

6.1.20.46 ptpAcceptableMasterClockIdentity <port>

This configuration variable is reserved.

6.1.20.47 ptpAcceptableMasterPriority1 <port> <address>

This configuration variable is used to add the address of an Acceptable Master to the Acceptable Partner Table for the designated port. The Acceptable Master will have the specified alternate PTP priority1 value. For more details, refer to *PTP Acceptable Partner Configuration*.

6.1.20.48 ptpAcceptableMasterPriority2 <port> <address>

This configuration variable is used to add the address of an Acceptable Master to the Acceptable Partner Table for the designated port. The Acceptable Master will have the specified alternate PTP priority2 value. For more details, refer to *PTP Acceptable Partner Configuration on page 105*.

6.1.20.49 ptpAcceptableMasterRqstUnicastAnnounce <port> <address>

This configuration variable is reserved.

6.1.20.50 ptpAcceptableMasterTransmissionProtocol <port> <address>

This configuration variable is reserved.

6.1.20.51 ptpAcceptableSlaveAddress <port>

This configuration variable is used to add the address of an Acceptable Slave to the Acceptable Partner Table for the designated port. For more details, refer to *PTP Acceptable Partner Configuration on page 105*.

6.1.20.52 ptpBoundaryClockEnable

This configuration variable is used to enable the TCM as a PTP Boundary clock. For more details, refer to *PTP Configuration Overview on page 101*.

6.1.20.53 ptpBoundaryMasterDomain

This configuration variable is used to configure the PTP Domain for TCM PTP port 2. The variable represents the domain Number member of the PTP Default Data Set. Refer to *IEEE1588v2 Section 8.2.1.*

6.1.20.54 ptpEnable

This configuration variable is used to enable TCM PTP operations. For more details, refer to *PTP Configuration Overview on page 101*.

6.1.20.55 TCMCoarsePhaseLossLimit

The configuration item `TCMCoarsePhaseLossLimit` defines the number of Unit Intervals (UIs) of phase error required to generate a phase loss condition. The limit value is rounded up to the nearest integer power of 2 minus 1 UI. Refer to *Semtech documentation* for additional information.

6.1.20.56 TCMCritThreadPriority

The configuration item `TCMCritThreadPriority` is the Linux thread priority assigned to certain time-critical TCM procedures. Typically a user would use the default value. The valid range for this item depends on the setting of the `TCMCritThreadSched` configuration item, use the standard Linux shell command `chrt -m` to determine the supported range.

6.1.20.57 TCMCritThreadSched

The configuration item `TCMCritThreadSched` is the Linux thread scheduling policy assigned to certain time-critical TCM procedures. Typically a user would use the default value.

6.1.20.58 TCMFWOverrideEnable

Enable this configuration item to force `tclk_agent` to startup with unsupported or unrecognized versions of firmware. This includes any combination of the Semtech ToPSync firmware, the FPGA Front Blade firmware, and the FPGA RTM firmware. The operations of the `tclk_agent` are undefined and unsupported when running with an unsupported version of the firmware.

A firmware version may be unrecognized, for instance, if Semtech releases a newer version of the firmware that does not follow the expected version numbering nomenclature. Contact SMART EC to determine if an unrecognized version of firmware is supported by the TCM.

6.1.20.59 TCMMastershipTimeout

The configuration item TCMMastershipTimeout defines the number of seconds during which both the Slave and Master must consistently agree on a Mastership switch before the Slave requests the switch.

The state of a Protection Partner Pair is queried multiple times a second on an ongoing basis. Among other things, a consideration is made whether the Slave Partner is in a better state to guide the output clocks than the Master. If so, then the Slave will request takeover of Mastership. Under certain circumstances, such as when Reference clocks rapidly fluctuate in quality for an extended period, a Slave may repeatedly appear to be a better Master for brief periods. Performing a Mastership switch under these conditions may be non-productive since it will not lead to overall better system quality. Under other circumstances, such as when the Master is in freerun or holdover and the Slave has a valid external Reference clock, performing a Mastership switch can be productive. Thus, a balance must be struck between longer TCMMastershipTimeout values to avoid frequent unproductive switches and shorter values to allow rapid productive switches

The state that can trigger a Mastership switch depends on the Stratum Level. For each Stratum Level, both Partners must report being in the appropriate state each time queried for an entire TCMMastershipTimeout interval before the Slave will request Mastership.

The Mastership switching states are shown below:

Stratum 3:

- The Slave has at least one valid external Reference clock.
- The Master has no valid external Reference clock.

Note that in Stratum 3 the state of a PLL is based on having valid Reference clocks and not whether they are currently selectable (have a non-zero PLL priority). However, only selectable valid clocks can be used as a reference. To prevent non-productive Mastership switchovers, at least one valid clock for a Slave should either already be selectable or be made selectable soon after the switchover. Conversely, to prevent productive switchovers from not triggering, at least one valid clock for a Master should be selectable.

6.1.20.60 TCM_S3_PathBandwidth

The configuration item `TCM_S3_PathBandwidth` is the CGM T0 PLL bandwidth for a Master Protection Partner when configured for Stratum 3. The Slave uses a higher T0 bandwidth automatically chosen to suit the Master's bandwidth. Refer to *Semtech documentation* for more information.

6.1.20.61 TCMWatchdogEnable

At startup, it enables or disables the TCM Watchdog. If enabled, then the TCM must continuously rearm a watchdog timer multiple times a second. Failure to do so will cause the clocks `clkCLK1A, clkCLK1B, clkCLK2A, clkCLK2B,` and `clkMSTx` to be squelched. This could trigger consumers of the clocks to fail and, depending on configuration, possibly switch to redundant clock sources.

The intent of the TCM Watchdog is to help ensure that the major output clocks are only being generated by a TCM that is operating in a nominal manner. However, when running in a non-real-time operating environment, it is not possible to guarantee the timing of the rearming procedure. This means that the TCM could be operating in an otherwise nominal manner except than that at some instant it is unable to rearm the Watchdog on a timely basis. In this case, the triggering of the Watchdog is a false alarm and not an indication that the TCM has otherwise failed. In such cases, the Watchdog is reset and the TCM resumes normal operations.

To prevent false alarms, the Watchdog rearming procedure and other timing-critical procedures within the TCM are assigned a high priority and a real time scheduling policy within Linux. Refer to the *TCMCritThreadPriority on page 127* and *TCMCritThreadSched on page 128* configuration items for more information.

6.1.20.62 TSGUIEnable

At startup, it enables or disables the ability of the CGM to interact with the Semtech ToPSync Manager Graphical User Interface (GUI) tools. Refer to *Semtech documentation* for more information on using the GUI tools. Refer to *CGM Ethernet Configuration on page 93* for additional information on configuration.

Important Information

Operations of the TCM is not supported while the Semtech Manager tool is communicating with the ToPSync. After the Manager tool has been shut down the TCM does not return to a supported state until the tclk_agent has gone through a complete cycle of starting up (e.g., at least the Startup complete Event has been generated) and then successfully shutting down.

MIBs

7.1 MIBs

The `tclk_agent` SNMP interface is defined by the `CGM-CONTROL-MIB.txt` and `IEEE1588-MIB.txt` MIB text files. These files and other BBS MIBs are located in the standard BBS MIB directory, `/opt/bladeservices/mibs`. The BBS MIBs may reference other standard MIBs supplied in `/usr/share/snmp/mibs`.

To maintain consistency across product lines, the TCM MIBs use many of the same file and MIB object names for referencing the same or similar features found on other Telco products (for example, the ATCA AMC-8001). However, same-named items may vary in purpose or usage across products and the product-specific documentation for each item should be consulted for proper understanding. The SNMP object Identifiers (OIDs) assigned to MIB objects, even those with the same names on other products are unique to each product line in order to maintain interoperability.

The CGM-CONTROL-MIB defines the Time Division Multiplex (TDM) and system management features of the TCM. The IEEE1588-MIB defines the Precision Time Protocol (PTP) features. The contents of these two MIBs are summarized in *Table 7-1* and *Table 7-3*.

IEEE1588-MIB Objects

The following are concerning *Table 7-1* and *Table 7-2*:

The SNMP Object Name column lists the MIB object names defined by the CGM-CONTROL-MIB and IEEE1588-MIB.

The Value Range column lists the possible values for the object if it is a scalar. An asterisk following a value indicates it is the default value. A value prefixed with (R) is a read-only value. Specifically, the value cannot be used with the SNMP Set command, but is a possible return value for a Get command.

MIBs allow enumerated values for objects to be expressed as names and as numerals. For example, the enumerated values for the `cgmEnabled` object can be expressed as the names `enable` and `disable` or as the numerals `0` and `1`. Enumerated values for MIB objects are presented in this documentation by showing the name form of the value followed by the associated numeral in parentheses (for example, `enable(0)`). Some tools, such as the Net-SNMP `snmpget` tool, support using both the name and numeric forms for values interchangeably, but others may only support one form or the other.

The **R/W** column indicates whether the object is readable R and/or writable W. A readable object can respond to an SNMP Get command and a writable object can respond to an SNMP Set command.

The **Configuration Item** column indicates whether or not the object name can also be used as a configuration file item name. If so, then the object can be initialized in a configuration file using one of the same values listed in the Value Range column that is suitable for the Set command. Currently, the TCM configuration files only support the name form of enumerated values.

Table 7-1 CGM CONTROL-MIB Objects

SNMP Object Name	Value Range	R/W	Configuration Item
bitsCurSsm	32-bit integer. Refer to the *cgmBitsTable on page 156* MIB object.	R	No
bitsE1RxSyncC	fasError3(0)* fasOrB23(1)	R/W	No
bitsEnable	disable(0) enable(1) Refer to the *cgmBitsTable on page 156* MIB object.	R/W	No
bitsInterfaceNumber	1..2 Refer to the *cgmBitsTable on page 156* MIB object.	R	No
bitsLiuE1Lbo	o120(1)* o120hrl(3) notDefined(255)	R/W	No
bitsLiuEgl	db36or43(0)* db15or12(1)	R/W	No
bitsLiuJabDs	b128(0)* b32(1)	R/W	No
bitsLiuJatEn	off(0) on(1)*	R/W	No

Table 7-1 CGM CONTROL-MIB Objects (continued)

SNMP Object Name	Value Range	R/W	Configuration Item
bitsLiuRxLevel	db2p5(0) db5p0(1) db7p5(2) db10p0(3) db12p5(4) db15p0(5) db17p5(6) db20p0(7) db22p5(8) db25p0(9) db27p5(10) db30p0(11) db32p5(12) db35p0(13) db37p5(14) db40p0(15)*	R	No
bitsLiuT1Lbo	`ft0to133(0)*` (T1 mode) ft133to266(1) ft266to399(2) ft399to533(3) ft533to655(4) `notDefined(255)*` (E1 mode)	R/W	No
bitsLiuTais	transmitAllOnes(0) transmitData(1)*	R/W	No
bitsSsmE1SaSelect	4*..8 Refer to the *cgmBitsTable on page 156* MIB object.	R/W	No
bitsT1RxSyncC	opt1(0)* opt2(1)	R/W	No
bitsT1TaisCi	off(0)* on(1)	R/W	No

Table 7-1 CGM CONTROL-MIB Objects (continued)

SNMP Object Name	Value Range	R/W	Configuration Item
bitsT1TraiCi	off(0)* on(1)	R/W	No
bitsT1TxB7zs	noStuffing(0)* on(1)	R/W	No
bitsT1TxFbCT1	trigger(0)*	R/W	No
bitsT1TxYel	off(0)* on(1)	R/W	No
bitsT3ClkForce	off(0)* on(1)	R/W	No
bitsT3ClkSquelch	enabled(0) disabled(1)	R/W	No
bitsT4Monitor	ignore(0) disableTx(1)* raiseAlarm(2)	R/W	No
bitsTestPattern	32-bit integer	R/W	No
bitsTestPatternLength	bit8(0)* bit16(1)	R/W	No
bitsTestPatternType	fixed(0) prbs11(1)* prbs15(2) qrss20(3)	R/W	No
bitsTxClockDomain	t4(0)* t0(1)	R/W	No
bitsTxEnable	off(0) on(1) Refer to the *cgmBitsTable on page 156* MIB object.	R/W	No
cgm8kTestEnable	enable(0) disable(1)*	R/W	Yes

Table 7-1 CGM CONTROL-MIB Objects (continued)

SNMP Object Name	Value Range	R/W	Configuration Item
cgmAMCCLK3Enable	enable(0) disable(1)*	R/W	Yes
cgmAMCEnable	enable(0) disable(1)*	R/W	Yes
cgmATCAClkDomain	`domainA(0)*` domainB(1)	R/W	Yes
cgmBITSEnable	enable(0) disable(1)*	R/W	Yes
cgmBitsTable	N/A (not a scalar object)	R/W	No Refer to the *cgmBITSEnable on page 156* and *cgmInterfaceMode on page 165* configuration items for related features.
cgmCLK1Src	notDriven(0) unknown(1) invalid(2) clkTCLKD(103) clkExt1P2(312) clkOPCLK6(506) clkOPCLK7(507)	R	No Refer to the `clkSrc` configuration item for a related feature.
cgmCLK2Src	notDriven(0) unknown(1) invalid(2) clkTCLKB(101) clkExt1P4(314) clkOPCLK1(501)	R	No Refer to the `clkSrc` configuration item for a related feature.

Table 7-1 CGM CONTROL-MIB Objects (continued)

SNMP Object Name	Value Range	R/W	Configuration Item
cgmCLK3Src	notDriven(0) unknown(1) invalid(2) clkTCLKB(101) clkTCLKC(102) clkUsr1(204) clkExt1P3(313)	R	No Refer to the `clkSrc` configuration item for a related feature.
cgmClockCount	`twoClks(0)*` threeClks(1) fourClks(2) fourAMCthreeBPClks(3)	R/W	Yes
cgmEnable	enable(0) `disable(1)*`	R/W	Yes
cgmEventCode	Refer to *Events on page 82* for all of the Event Codes.	R	No
cgmEventData	(32-bit integer) Refer to *Events on page 82* for description of the data supplied for an Event.	R	No
cgmEventDescr	(0..255 character string) Refer to *Events on page 82* for each Event description.	R	No
cgmEventNumber	1..4294967295	R	No
cgmEventSeverity	critical(0) warning(1) event(2) info(3)	R	No
cgmEventTable	N/A (not a scalar object)	R	No
cgmEventTime	SNMP TimeStamp	R	No

Table 7-1 CGM CONTROL-MIB Objects (continued)

SNMP Object Name	Value Range	R/W	Configuration Item
cgmExtShelves	<string of 5 binary digits> `00000*`	R/W	Yes
cgmFrameSyncMode	`frameSync(0)*` multiFrameSync(1)	R/W	Yes
cgmFrameSyncPulse	notInvertedNotPulsed(0)* notInvertedPulsed(1) invertedNotPulsed(2) invertedPulsed(3)	R/W	Yes
cgmInputActivityMonitorEna	enabled(0) disabled(1)* (Refer to the *cgmInputTable on page 163* MIB object)	R/W	No
cgmInputActivityMonitorState	working(0) failing(1) failed(2) recovering(3) unknown(4) (Refer to the *cgmInputTable on page 163* MIB object)	R	No
cgmInputFrequency	Refer to the *cgmInputTable on page 163* MIB object.	R/W	No Refer to the `clkFreq` configuration item for a related feature.
cgmInputName	Refer to the *cgmInputTable on page 163* MIB object.	R	No
cgmInputNumber	1..12 (Refer to the *cgmInputTable on page 163* MIB object)	R	No
cgmInputSrc	Refer to the *cgmInputTable on page 163* MIB object.	R	No Refer to the `clkFreq` configuration item for a related feature.

Table 7-1 CGM CONTROL-MIB Objects (continued)

SNMP Object Name	Value Range	R/W	Configuration Item
cgmInputState	valid(0) invalid(1) (Refer to the *cgmInputTable on page 163* MIB object)	R	No
cgmInputTable	N/A (not a scalar object)	R/W	No Refer to the `clkSrc`, `clkFreq`, `clkT0Priority`, and `clkT4Priority` configuration item for related features.
cgmInterfaceMode	`sonetT1(0)*` sdhE1(1)	R/W	Yes
cgmLogEvent	N/A (not a scalar object)	R	No Refer to the *cgmSysEventTrapLevel on page 169* MIB object for a related feature.
cgmMasterMode	`localClkGen(0)*` amcClkGen(1) passThru(2) extShelf(3)	R/W	Yes
cgmOXSelect	onboard(0)* fromRTM(1)	R/W	Yes

Table 7-1 CGM CONTROL-MIB Objects (continued)

SNMP Object Name	Value Range	R/W	Configuration Item
cgmPathInputT0	input_1(1) input_2(2) input_3(3) input_4(4) input_5(5) input_6(6) input_7(7) input_8(8) input_9(9) input_10(10) input_11(11) input_12(12) noInput(255)	R	No
cgmPathInputT4	input_1(1) input_2(2) input_3(3) input_4(4) input_5(5) input_6(6) input_7(7) input_8(8) input_9(9) input_10(10) input_11(11) input_12(12) noInput(255)	R	No
cgmPathStateT0	`freerun(0)` locked(1) holdover(2) preLocked(3) preLocked2(4) phaseLost(5) forcedFreerun(6) forcedHoldover(7)	R	No

Table 7-1 CGM CONTROL-MIB Objects (continued)

SNMP Object Name	Value Range	R/W	Configuration Item
cgmPathStateT0Time	32-bit integer	R	No
cgmPathT0LockedTime	32-bit integer	R	No
cgmProtectionCmd	`(R) noCmd(0)*` becomeMaster(1) becomeSlave(2) forceMaster(3) unforceMaster(4)	R/W	No
cgmProtectionMasterToSlaveTrackDelay	`0..2147483647 picoseconds` (default varies)	R/W	Yes
cgmProtectionPartnerAddress	<IP address> 192.168.21.111*	R/W	Yes
cgmProtectionRevertiveModeState	enabled disabled*	R/W	Yes
cgmProtectionState	standaloneMaster(0) slave(1) master(2)	R	No
cgmStratumLevel	stratum3(3)*	R	Yes
cgmSyncEGenEnable	enable(0) disable(1)*	R/W	Yes
cgmSysEventLogClear	retrievedEvents(0)* allEvents(1)	R/W	No
cgmSysEventLogCount	0..100	R	No
cgmSysEventLogLevel	nothing(0)* critical(1) warning(2) event(3) all(4)	R/W	Yes

Table 7-1 CGM CONTROL-MIB Objects (continued)

SNMP Object Name	Value Range	R/W	Configuration Item
cgmSysEventLogSize	100	R	No
cgmSysEventTrapLevel	nothing(0)* critical(1) warning(2) event(3) all(4)	R/W	Yes
cgmT0InputPriority	0*,2..15 Refer to the *cgmInputTable on page 163* MIB object.	R/W	No Refer to the *cgmT0InputPriority on page 170* configuration item for a related feature.
cgmT4ClkSquelch	enable(0)* disable(1)	R/W	No
cgmT4InputPriority	0*,2..15 Refer to the *cgmInputTable on page 163* MIB object.	R/W	No Refer to the *cgmT4InputPriority on page 170* configuration item for a related feature.
cgmTCMVersion	N/A	R	No
cgmTSFWVersion	N/A	R	No

dsx1LineCode	E1 Line Codes:
	`dsx1HDB3(3)*`
	dsx1AMI(5)
	
	T1 Line Codes:
	`dsx1B8ZS(2)*`
	dsx1AMI(5)
	Refer to the *cgmBitsTable on page 156* MIB object.
		R/W	No

Table 7-1 CGM CONTROL-MIB Objects (continued)

SNMP Object Name	Value Range	R/W	Configuration Item
dsx1LineStatus	Bit mask encoding the following status: Bit 0: Yellow alarm condition Bit 3: Alarm condition Bit 5: Loss of Frame condition Bit 6: Loss of Signal condition Refer to the *cgmBitsTable on page 156* MIB object.	R	No
dsx1LineType	E1 Line Types: dsx1E1(4) dsx1E1CRCMF(7)* dsx1E1UnframedAll1(9) dsx1E1UnframedAlt(11) dsx1E1G703(12) dsx1E1Unframed(13) T1 Line Types: dsx1ESF(2)* dsx1D4(3) dsx1UnframedAll1(8) dsx1UnframedAlt(10) Refer to the *cgmBitsTable on page 156* MIB object.	R/W	No
dsx1LoopbackConfig	dsx1NoLoop(1)* dsx1PayloadLoop(2) dsx1LineLoop(3) Refer to the *cgmBitsTable on page 156* MIB object.	R/W	No
tcmCmd	`1..1000` character string	R/W	No

Table 7-2 IEEE1588-MIB Objects

SNMP Object Name	Value Range	R/W	Configuration Item
ptpAcceptablePartnerAddress	Refer to the *ptpAcceptablePartnerTable on page 172* MIB object.	R	No Refer to configuration variables ptpAcceptableMasterAddress and ptpAcceptableSlaveAddress.
ptpAcceptablePartnerAddressType	Refer to the *ptpAcceptablePartnerTable on page 172* MIB object.	R	No Refer to configuration variable ptpAcceptableMasterAddressType.
ptpAcceptablePartnerIsSlave	Refer to the *ptpAcceptablePartnerTable on page 172* MIB object.	R	No
ptpAcceptablePartnerPortNumber	Refer to the *ptpAcceptablePartnerTable on page 172* MIB object.	R	No
ptpAcceptablePartnerPriority1	Refer to the *ptpAcceptablePartnerTable on page 172* MIB object.	R	No Refer to configuration variable ptpAcceptableMasterPriority1
ptpAcceptablePartnerPriority2	Refer to the *ptpAcceptablePartnerTable on page 172* MIB object.	R	No Refer to configuration variable ptpAcceptableMasterPriority2

Table 7-2 IEEE1588-MIB Objects (continued)

SNMP Object Name	Value Range	R/W	Configuration Item
ptpAcceptablePartnerRqstUnicast	Refer to the *ptpAcceptablePartnerTable on page 172* MIB object.	R	No Refer to configuration variable ptpAcceptableMasterRqstUnicastAnnounce
ptpAcceptablePartnerTable	N/A (not a scalar object)	R/W	No Refer to the configuration items prefixed with either ptpAcceptableMaster or ptpAcceptableSlave.
ptpClockCurrentDSMeanPathDelay	0..18446744073709551615 nanoseconds	R	No
ptpClockCurrentDSOffsetFromMaster	-2147483647.. 2147483647 nanoseconds	R	No
ptpClockCurrentDSStepsRemoved	0..2147483647	R	No
ptpClockDefaultDSClockIdentity	<8-octet string> <IEEE1588v2 Section 7.5.2.2 version of ToPSync MAC address>*	R	No
ptpClockDefaultDSDomain	default(0)* alt1(1) alt2(2) alt3(3)	R	Yes
ptpClockDefaultDSNumOfPorts	2*	R	No
ptpClockDefaultDSPriority1	0..128*..255	R	Reserved
ptpClockDefaultDSPriority2	0..128*..255	R	Reserved

Table 7-2 IEEE1588-MIB Objects (continued)

SNMP Object Name	Value Range	R/W	Configuration Item
ptpClockDefaultDSQualityAccuracy	ns25(32) ns100(33) ns250(34) us1(35)* us2p5(36) us10(37) us25(38) us100(39) us250(40) ms1(41) ms2p5(42) ms10(43) ms25(44) ms100(45) ms250(46) s1(47) s10(48) s10gt(49) unknown(254)	R	Reserved
ptpClockDefaultDSQualityClass	priStdRef(6)* priStdRefHoldover(7) priAppSpecRef(13) priAppSpecRefHoldover(14) priStdRefAltA(52) priAppSpecRefAltA(58) priStdRefAltB(187) priAppSpecRefAltB(193) default(248) slaveOnly(255)	R	Reserved
ptpClockDefaultDSQualityOffset	0..65535*	R	Reserved
ptpClockDefaultDSSlaveOnly	false(0)* true(1)	R	Yes

Table 7-2 IEEE1588-MIB Objects (continued)

SNMP Object Name	Value Range	R/W	Configuration Item
ptpClockDefaultDSTwoStepFlag	false(0)* true(1)	R	No
ptpClockParentDSClockPhChRate	This MIB object is currently not supported.	R	No
ptpClockParentDSGMClockIdentity	<8-octet string>	R	No
ptpClockParentDSGMClockPriority1	0..255	R	No
ptpClockParentDSGMClockPriority2	0..255	R	No
ptpClockParentDSGMClockQualityAccuracy	ns25(32) ns100(33) ns250(34) us1(35) us2p5(36) us10(37) us25(38) us100(39) us250(40) ms1(41) ms2p5(42) ms10(43) ms25(44) ms100(45) ms250(46) s1(47) s10(48) s10gt(49) unknown(254)	R	No

Table 7-2 IEEE1588-MIB Objects (continued)

SNMP Object Name	Value Range	R/W	Configuration Item
ptpClockParentDSGMClockQualityClass	priStdRef(6) priStdRefHoldover(7) priAppSpecRef(13) priAppSpecRefHoldover(14) priStdRefAltA(52) priAppSpecRefAltA(58) priStdRefAltB(187) priAppSpecRefAltB(193) default(248) slaveOnly(255)	R	No
ptpClockParentDSGMClockQualityOffset	0..65535	R	No
ptpClockParentDSOffset	This MIB object is currently not supported.	R	No
ptpClockParentDSParentPortIdentity	<10-octet string>	R	No
ptpClockParentDSParentStats	false(0)	R	No
ptpClockPortDSAcceptablePartnerAddressAdd	Refer to the *ptpClockPortDSTable on page 180* MIB object.	R/W	No
ptpClockPortDSAcceptablePartnerAddressDel	Refer to the *ptpClockPortDSTable on page 180* MIB object.	R/W	No
ptpClockPortDSAcceptablePartnerAddressType	Refer to the *ptpClockPortDSTable on page 180* MIB object.	R/W	No
ptpClockPortDSAcceptablePartnerIsSlave	Refer to the *ptpClockPortDSTable on page 180* MIB object.	R/W	No
ptpClockPortDSAcceptablePartnerPriority1	Refer to the *ptpClockPortDSTable on page 180* MIB object.	R/W	No

Table 7-2 IEEE1588-MIB Objects (continued)

SNMP Object Name	Value Range	R/W	Configuration Item
ptpClockPortDSAcceptablePartnerPriority2	Refer to the *ptpClockPortDSTable on page 180* MIB object.	R/W	No
ptpClockPortDSAcceptablePartnerRqstUnicast	Refer to the *ptpClockPortDSTable on page 180* MIB object.	R/W	No
ptpClockPortDSAnnouncementInterval	Refer to the *ptpClockPortDSTable on page 180* MIB object.	R	No
ptpClockPortDSAnnounceRctTimeout	This MIB object is currently not supported.	R/W	No
ptpClockPortDSDelayMech	Refer to the *ptpClockPortDSTable on page 180* MIB object.	R	No
ptpClockPortDSDomain	Refer to the *ptpClockPortDSTable on page 180* MIB object.	R	No
ptpClockPortDSMinDelayReqInterval	Refer to the *ptpClockPortDSTable on page 180* MIB object.	R/W	No
ptpClockPortDSMinPeerDelayReqInterval	This MIB object is currently not supported.	R/W	No
ptpClockPortDSPeerMeanPathDelay	Refer to the *ptpClockPortDSTable on page 180* MIB object.	R	No
ptpClockPortDSPortIdentity	Refer to the *ptpClockPortDSTable on page 180* MIB object.	R	No
ptpClockPortDSPTPVersion	Refer to the *ptpClockPortDSTable on page 180* MIB object.	R	No
ptpClockPortDSRunningState	Refer to the *ptpClockPortDSTable on page 180* MIB object.	R/W	No

Table 7-2 IEEE1588-MIB Objects (continued)

SNMP Object Name	Value Range	R/W	Configuration Item
ptpClockPortDSSyncInterval	Refer to the *ptpClockPortDSTable on page 180* MIB object.	R	No
ptpClockPortDSTable	N/A (not a scalar object)	R/W	No Refer to the configuration items prefixed with ptpClockPortDS
ptpClockPortDSUseAcceptableMasters	Refer to the *ptpClockPortDSTable on page 180* MIB object.	R/W	Yes
ptpClockPortDSUseAcceptableSlaves	Refer to the *ptpClockPortDSTable on page 180* MIB object.	R/W	Yes
ptpClockTimePropertiesDSCurrentUTCOffset	`0..2147483647` seconds	R	No
ptpClockTimePropertiesDSCurrentUTCOffsetValid	false(0) true(1)	R	No
ptpClockTimePropertiesDSFreqTraceable	false(0) true(1)	R	No
ptpClockTimePropertiesDSLeap59	false(0) true(1)	R	No
ptpClockTimePropertiesDSLeap61	false(0) true(1)	R	No
ptpClockTimePropertiesDSPTPTimescale	false(0) true(1)	R	No

Table 7-2 IEEE1588-MIB Objects (continued)

SNMP Object Name	Value Range	R/W	Configuration Item
ptpClockTimePropertiesDSSource	atomic(16) gps(32) terrestrialRadio(48) ptp(64) ntp(80) handSet(96) other(144) internalOSC(160)	R	Reserved
ptpClockTimePropertiesDSTimeTraceable	false(0) true(1)	R	No
ptpCurrentMaster	<IP address>	R	No
ptpCurrentMasterAddressType	ipv4(1)	R	No
ptpVisibleMasterAddress	Refer to the *ptpVisibleMasterTable on page 185* MIB object.	R	No
ptpVisibleMasterAddressType	Refer to the *ptpVisibleMasterTable on page 185* MIB object.	R	No
ptpVisibleMasterPriority1	Refer to the *ptpVisibleMasterTable on page 185* MIB object.	R	No
ptpVisibleMasterPriority2	Refer to the *ptpVisibleMasterTable on page 185* MIB object.	R	No

7.1.1 CGM-CONTROL-MIB

The CGM-CONTROL-MIB defines the Time Division Multiplex (TDM) features of the TCM as well as other TCM features. These features are implemented as SNMP scalar, table, and trap objects.

Table 7-1 summarizes various features of the TCM SNMP objects.

7.1.1.1 bitsCurSsm

The `cgmBitsTable` column specifying the last received Synchronization Status Message (SSM).

7.1.1.2 bitsE1RxSyncC

The `cgmBitsTable` column specifying the E1 frames resynchronization criteria.

`fasError3(0)`: Resynchronize, if the Frame Alignment Signal (FAS) is received with error for three consecutive times.

`fasOrB23(1)`: Resynchronize, if the FAS or bit 2 of non-FAS is received with error for three consecutive times.

7.1.1.3 bitsEnable

The `cgmBitsTable` column specifying whether the LIU interface to the BITS/SSU is powered on or not. The LIUs cannot be powered on/off or otherwise controlled unless the `cgmBITSEnable` configuration item is enabled. Enabling the interface also applies various default parameters depending on the operation mode defined by `cgmInterfaceMode`:

Defaults for E1:

dsx1LineType: dsx1E1CRCMF(7)

`dsx1LineCode: dsx1HDB3(3)`

bitsLiuE1Lbo: o120(1)

Defaults for T1:

dsx1LineType: dsx1ESF(2)

dsx1LineCode: dsx1B8ZS(2)

bitsLiuT1Lbo: ft0to133(0)

7.1.1.4 bitsInterfaceNumber

The index variable for the `cgmBitsTable`. This is the number assigned to the front-panel BITS/SSU connector, 1 for the connector labeled TE1 and 2 for TE2. This variable is not directly accessible via SNMP queries. Some table browsers may not list the values for this variable or the table index may be listed under a different name such as Instance.

7.1.1.5 bitsLiuE1Lbo

The `cgmBitsTable` column specifying the E1 Line Build-Out (LBO) in ohms (without and with high return loss).

7.1.1.6 bitsLiuEgl

The `cgmBitsTable` column specifying the receive equalizer gain limit for T1 (-36/-15dB) and E1 (-43/-12dB).

7.1.1.7 bitsLiuJabDs

The `cgmBitsTable` column specifying the depth in bits of the jitter attenuator.

7.1.1.8 bitsLiuJatEn

The `cgmBitsTable` column specifying whether the jitter attenuator is turned on or not.

7.1.1.9 bitsLiuRxLevel

The `cgmBitsTable` column specifying the receive level in decibels.

7.1.1.10 bitsLiuT1Lbo

The `cgmBitsTable` column specifying the T1 Line Build-Out (LBO) in feet for DSX-1 applications.

7.1.1.11 bitsLiuTais

The `cgmBitsTable` column specifying the Alarm Indication Signal (AIS) transmitted in lieu of the normal signal.

7.1.1.12 bitsSsmE1SaSelect

The `cgmBitsTable` column specifying on which Sa bit the E1 Synchronization status message is expected.

7.1.1.13 bitsT1RxSyncC

The `cgmBitsTable` column specifying the T1 frames resynchronization criteria.

In D4 framing mode:

`opt1(0)`: Search for Ft pattern, then search for Fs pattern.

`opt2(1)`: Cross couple Ft and Fs pattern.

In ESF Framing mode:

`opt1(0)`: Search for FPS pattern only.

`opt2(1)`: Search for FPS and verify with CRC6.

7.1.1.14 bitsT1TaisCi

The `cgmBitsTable` column specifying whether the transmission of Alarm Indication Signal-Customer Installation (AIS-CI) is turned on or not. Setting this causes the AIS-CI code to be transmitted from the framer to the LIU, as defined in ANSI T1.403.

7.1.1.15 bitsT1TraiCi

The `cgmBitsTable` column specifying whether transmission of Remote Alarm Indication-Customer Installation (RAS-CI) is turned on or not. Setting this causes the Extended Super Frame (ESF) RAI-CI code to be transmitted in the Facility Data Link (FDL) bit position.

7.1.1.16 bitsT1TxB7zs

The `cgmBitsTable` column specifying whether transmit-side bit 7 zero-suppression is enabled or not. If enabled, bit 7 is forced to a one in channels with all zero's.

7.1.1.17 bitsT1TxFbCT1

The `cgmBitsTable` column used to cause the remote end to experience loss of synchronization. Each time this variable is set, it triggers three consecutive Ft (D4 framing mode) or FPS (ESF framing mode) bits to be corrupted. The value returned for this variable is always `trigger(0)`, but the loss of synchronization is only triggered at each instance when the variable is set to this value.

7.1.1.18 bitsT1TxYel

The `cgmBitsTable` column specifying whether transmission of Yellow Alarm is turned on or not.

7.1.1.19 bitsT3ClkForce

The `cgmBitsTable` column specifying whether forcing of the T3 clock is turned on or not. This only has an effect when the framer outputs a free-run T3 clock because there is no BITS/SSU signal. When turned on, the free-run T3 clock is sent to the CGM. Otherwise, the free-run T3 clock is squelched.

7.1.1.20 bitsT3ClkSquelch

The `cgmBitsTable` column specifying whether or not the T3 clock sent to the CGM is squelched when there is a loss of frame or an alarm condition with the BITS/SSU. When set to `enabled(0)`, the T3 clock output to the PLL is squelched when there is a loss of frame or an alarm condition. When set to `disabled(1)`, the T3 clock is not squelched when there is a loss of frame or an alarm condition. Normally this is set to `enabled(0)` and only set to `disabled(1)` for unframed E1 operation.

7.1.1.21 bitsT4Monitor

The `cgmBitsTable` column specifying the action taken if the CGM T4 PLL is not locked:

`ignore(0)`: No action taken

`disableTx(1)`: Suppress transmitter

`raiseAlarm(2)`: Raise AIS (E1) or Yellow Alarm (T1)

This setting is ignored if the BITS transmit clock is derived from the T0 PLL (refer to the *bitsTxClockDomain on page 155* MIB object). Refer to the cgmT4ClkSquelch MIB object for a related feature.

7.1.1.22 bitsTestPattern

The `cgmBitsTable` column specifying an 8-bit or 16-bit test pattern that will be serialized and repeatedly output on the T1/E1 port when the test output is enabled. This only applies to the following line-type settings:

`dsx1ESF, dsx1D4, dsx1E1, dsx1E1CRCMF, and dsx1E1Unframed`

7.1.1.23 bitsTestPatternLength

The `cgmBitsTable` column specifying the length of the test pattern specified by bitsTestPattern. This only applies to the following line-type settings:

dsx1ESF, dsx1D4, dsx1E1, dsx1E1CRCMF, and dsx1E1Unframed

7.1.1.24 bitsTestPatternType

The `cgmBitsTable` column specifying the type of a test pattern specified by bitsTestPattern. This only applies to the following line-type settings:

dsx1ESF, dsx1D4, dsx1E1, dsx1E1CRCMF, and dsx1E1Unframed

7.1.1.25 bitsTxClockDomain

The `cgmBitsTable` column specifying the CGM path used to generate the T4 clock. If T0 is specified, then the `bitsT4Monitor` object is ignored, i.e. the transmitter and the alarm state setting is independent from the state of the T0 PLL. The default is T4 path is used by default.

7.1.1.26 bitsTxEnable

The `cgmBitsTable` column specifying whether the LIU transmitter to the BITS/SSU is on (but possibly suppressed) or off. The transmitter is suppressed when the `bitsTxClockDomain` is t4(0) and `bitsT4Monitor` variable is `disableTx(1)` and the CGM T4 PLL is not locked to a Reference clock. The LIU transmitter to the BITS/SSU is enabled by default.

7.1.1.27 cgm8kTestEnable

Gets the last successful setting of this object or Set to `enable(0)`/`disable(1)` to enable or disable the FPGA 8k Test Clock, `clk8kTest`.

7.1.1.28 cgmAMCCLK3Enable

Gets the last successful setting of this object. When `cgmMasterMode` is not `amcClkGen(1)` and `cgmClockCount` is `twoClks(0)`, set to `enable(0)` to enable the AMC TCLKB to source ATCA CLK3 and set to `disable(1)` to disable the TCM from sourcing CLK3. The setting for this object is ignored when `cgmMasterMode` is `amcClkGen(1)` or `cmgClockCount` is not `twoClks(0)`. Refer to *TCM Mode Summary on page 50* for more information on how the AMC clocks are sourced for various modes.

7.1.1.29 cgmAMCEnable

Gets the last successful setting of this object. When `cgmMasterMode` is not `amcClkGen(1)`, set to `enable(0)` to enable the TCM to use various AMC clocks and set to `disable(1)` to disable the TCM from using the AMC clocks. When `cgmMasterMode` is `amcClkGen(1)`, the AMC clocks are automatically used and this variable is ignored. Refer to *TCM Mode Summary on page 50* for more information on how the AMC clocks are sourced for various modes.

7.1.1.30 cgmATCAClkDomain

Get or Set the TCM ATCA Clock Domain. The Clock Domain cannot be Set while the CGM is enabled. Refer to *Chapter 4, TCM Mode* for more information.

7.1.1.31 cgmBITSEnable

Get or Set whether the Line Interface Units for BITS/SSU devices are enabled for usage. If usage is not enabled, then the BITS ports cannot be controlled and is powered off. If usage is enabled, then the BITS ports can be powered on/off and otherwise controlled. This variable cannot be changed while `cgmEnable` is enabled.

7.1.1.32 cgmBitsTable

The CGM BITS/SSU Table allows the user to view and configure the Maxim DS26503 Line Interface Unit (LIU) interfaces to two BITS/SSU devices.

Table 7-3 is an example, where BITS/SSU Table as shown by an SNMP table browser.

Below is a summary of the *Table 7-3* columns. An (R) indicates that the associated feature of the LIU is read-only and (R/W) indicates the feature can be Set. Refer to the subsection for each column variable for more information. Also consult the BITS/SSU device documentation and the Maxim Corporation DS26503 documentation for additional information concerning these features.

`bitsInterfaceNumber` (R) - The table index. The index is the number assigned to the front-panel BITS/SSU connector, 1 for the connector labeled TE1 and 2 for TE2. Some table browsers may not list the values for this variable or the table index may be listed under a different name such as Instance.

`bitsInterfaceNumber` (R)	The table index. The index is the number assigned to the front panel BITS/SSU connector, 1 for the connector labeled TE1 and 2 for TE2. Some table browsers may not list the values for this variable or the table index may be listed under a different name such as Instance.
`bitsEnable` (R/W)	Whether or not the LIU interface to the BITS/SSU is powered on/off.
`bitsTxEnable` (R/W)	Whether or not the LIU transmitter to the BITS/SSU is on.
`dsx1LineType` (R/W)	E1/T1 Line Type.
`dsx1LineCode` (R/W)	E1/T1 Line Code.
`dsx1LoopbackConfig` (R/W)	Loopback mode.
`dsx1LineStatus` (R)	Current line status.

`bitsCurSsm` (R)	Last received Synchronization Status Message (SSM).
`bitsSsmE1SaSelect` (R/W)	Selects on which Sa bit the E1 Synchronization Status Message (SSM) is expected.
`bitsLiuJatEn` (R/W)	Whether the jitter attenuator is enabled or not.
`bitsLiuJabDs` (R/W)	Jitter attenuator depth in bits.
`bitsLiuEgl` (R/W)	Receive equalizer gain limit in decibels.
`bitsLiuE1Lbo` (R/W)	E1 Line Build-Out (LBO) in ohms.
`bitsLiuT1Lbo` (R/W)	T1 Line Build-Out (LBO) for DSX-1 applications in feet.
`bitsLiuTais` (R/W)	Alarm Indication Signal (AIS) transmitted in lieu of the normal signal.
`bitsLiuRxLevel` (R)	Receive level in decibels.
`bitsT1RxSyncC` (R/W)	T1 frame resynchronization criteria.
`bitsT1TxYel` (R/W)	Whether transmission of yellow alarm is on or not.
`bitsT1TxB7zs` (R/W)	Whether transmit-side bit 7 zero-suppression is used or not.
`bitsT1TxFbCT1` (R/W)	Trigger for causing the remote end to experience loss of synchronization.
`bitsT1TaisCi` (R/W)	Whether transmission of AIS-CI (Alarm Indication Signal-Customer Installation) is on or not.
`bitsT1TraiCi` (R/W)	Whether transmission of RAI-CI (Remote Alarm Indication-Customer Installation) is on or not.
`bitsE1RxSyncC` (R/W)	E1 frame resynchronization criteria.
`bitsT3ClkForce` (R/W)	Whether forced T3 clock transmission is on or not.
`bitsT3ClkSquelch` (R/W)	Whether T3 clock squelch is enabled or not.
`bitsTestPattern` (R/W)	Testing output pattern.
`bitsTestPatternLength` (R/W)	Testing output pattern length in bits.
`bitsTestPatternType` (R/W)	Testing output pattern type.
`bitsT4Monitor` (R/W)	Action taken if CGM T4 PLL is not locked.
`bitsTxClockDomain` (R/W)	Which CGM Path is used to generate the T4 clock.

Table 7-3 CGM BITS/SSU Table

Instance	bitsInterfaceNumber	bitsEnable	bitsTxEnable	dsx1LineType	dsx1LineCode	dsx1LoopbackConfig
1	not available	enable(1)	on(1)	dsx1E1CRCMF(7)	dsx1HDB3(3)	dsx1NoLoop(1)
2	not available	enable(1)	on(1)	dsx1E1CRCMF(7)	dsx1HDB3(3)	dsx1NoLoop(1)

Instance	dsx1LineStatus	bitsCurSsm	bitsSsmE1SaSelect	bitsLiuJatEn	bitsLiuJabDs	bitsLiuEgl
1	96	0	4	on(1)	b128(0)	db36or43(0)
2	96	0	4	on(1)	b128(0)	db36or43(0)

Instance	bitsLiuE1Lbo	bitsLiuT1Lbo	bitsLiuTais	bitsLiuRxLevel	bitsT1RxSyncC	bitsT1TxYel
1	o120(1)	notDefined(255)	transmitData(1)	db40p0(15)	opt1(0)	off(0)
2	o120(1)	notDefined(255)	transmitData(1)	db40p0(15)	opt1(0)	off(0)

Instance	bitsT1TxB7zs	bitsT1TxFbCT1	bitsT1TaisCi	bitsT1TraiCi	bitsE1RxSyncC	bitsT3ClkForce
1	noStuffing(0)	trigger(0)	off(0)	off(0)	fasError3(0)	off(0)
2	noStuffing(0)	trigger(0)	off(0)	off(0)	fasError3(0)	off(0)

Instance	bitsT3ClkSquelch	bitsTestPattern	bitsTestPatternLength	bitsTestPatternType	bitsT4Monitor	bitsTxClockDomain
1	enabled(0)	0	bit8(0)	prbs11(1)	disableTx(1)	t4(0)
2	enabled(0)	0	bit8(0)	prbs11(1)	disableTx(1)	t4(0)

7.1.1.33 cgmCLK1Src

Get the source for `clkCLK1`.

7.1.1.34 cgmCLK2Src

Get the source for `clkCLK2`.

7.1.1.35 cgmCLK3Src

Get the source for `clkCLK3`.

7.1.1.36 cgmClockCount

Get or Set the TCM Clock Count. Refer to *Chapter 4, TCM Mode* for more information.

The Clock Count cannot be Set while the CGM is enabled. Refer to the *cgmEnable on page 159* MIB object.

7.1.1.37 cgmEnable

Get the last successful setting of this object. Set to `enable` to enable the CGM if currently disabled, or Set to `disable` to disable the CGM if currently enabled.

When the `tclk_agent` is started, the CGM is automatically disabled if it is already enabled. The CGM is automatically disabled when the `tclk_agent` exits. While the CGM is enabled, the following elements cannot be changed:

- BITS/SSU port enabled/disabled
- ATCA Clock Domain
- TCM Clock Count
- TCM Master Mode

7.1.1.38 cgmEventCode

The `cgmEventTable` column specifying the Event Code assigned to the associated Event and all other Events in the same category. Refer to *Events on page 82* for listing of all the Event Codes.

7.1.1.39 cgmEventData

The `cgmEventTable` column specifying Event Data for the associated Event. In most cases there is no Event Data other than a default value of zero. Refer to *Events* for the Event Data provided by each Event.

7.1.1.40 cgmEventDescr

The `cgmEventTable` column specifying the textual description of the associated Event. Refer to *Events* for the individual Event descriptions.

7.1.1.41 cgmEventNumber

The index variable for the `cgmEventTable`. Events are numbered in sequential order as they are generated. This variable is not directly accessible via SNMP queries. Some table browsers may not list the values for this variable or the table index may be listed under a different name such as Instance.

7.1.1.42 cgmEventSeverity

The `cgmEventTable` column specifying the Severity Code of the associated Event. When an Event gets logged, the Severity Code is used to define the priority of the log message. Refer to *Events* for the Severity Code of each Event and the mapping of Severity Codes to logging priorities.

7.1.1.43 cgmEventTable

The CGM Event Table allows the user to view the last 100 Events generated by the TCM. Events also go to the TCM log.

Table 7-4 is an example CGM Event Table as shown by an SNMP table browser. Note that some table browsers may not be able to list the values for the table index, `cgmEventNumber`, or the table index may be listed in a column with a different name such as Instance.

The `cgmEventTable` index is simply the numeric ordering number of an Event (e.g., from 1 to the number of Events generated so far). All columns of the table are read-only.

Table 7-4 CGM Event Table

cgmEvent Number	cgmEventTime	cgmEventSeverity	cgmEventCode	cgmEvent Data	cgmEventDescr
1	149 days, 15:40:48.93	info(3)	info(200)	0	Telco Clocking Module version: 2.8.2-0
2	149 days, 15:40:48.93	info(3)	assignedATCA Domain(142)	0	Assigned to ATCA Clock Domain A
3	149 days, 15:40:49.11	event(2)	cgmStatus(22)	1	ToPSync status: Ready
4	149 days, 15:40:49.11	info(3)	info(200)	0	ToPSync PTP Block configuration complete
5	149 days, 15:40:49.11	info(3)	info(200)	0	ACS85x0 Module starting
6	149 days, 15:40:49.11	info(3)	info(200)	0	Initializing ACS8520 Device
7	149 days, 15:40:49.12	info(3)	info(200)	0	Initializing Frequency Validation Controller

Table 7-4 CGM Event Table (continued)

cgmEvent Number	cgmEventTime	cgmEventSeverity	cgmEventCode	cgmEvent Data	cgmEventDescr
8	149 days, 15:40:49.12	info(3)	info(200)	0	Initializing Protection Controller
9	149 days, 15:40:49.13	info(3)	info(200)	0	Telco Clocking Ready
10	149 days, 15:40:49.13	event(2)	startupComplete(130)	0	Startup complete
11	149 days, 15:40:49.15	event(2)	nowMaster(4)	0	Now Master
12	149 days, 15:40:49.68	event(2)	validInputsChanged(6)	0	TDM Block valid inputs changed
13	149 days, 15:40:50.06	event(2)	inputStateChange(12)	10	Input 10 (M/S Sync In): New state: Working (was failed)

Each row of *Table 7-4* represents an Event and each column shows a different aspect of the Event. Refer to *Events* for a description of each column. Refer to the *cgmSysEventLogClear on page 169* MIB object for ways to remove Events from the Event Table.

7.1.1.44 cgmEventTime

The `cgmEventTable` column specifying the SNMP timestamp of the associated Event. The timestamp is based on the Linux system time when the Event was generated. The displayed value depends on how the SNMP tool interprets the timestamp.

7.1.1.45 cgmExtShelves

Get or Set which front panel Extension Shelf connections are enabled to transmit or receive clocks on their ports. The value is a string of five binary digits. A '`1`' means enabled and a '`0`' means disabled. The right-most digit is for Extension Shelf connector #1 and each subsequent digit reading right-to-left is for the next sequential Extension Shelf connector.

7.1.1.46 cgmFrameSyncMode

Get or Set the TCM Frame Sync mode. Refer to *Chapter 4, TCM Mode* for more information.

7.1.1.47 cgmFrameSyncPulse

Get or Set characteristics of the Frame Sync pulse.

7.1.1.48 cgmInputActivityMonitorEna

The `cgmInputTable` column specifying whether frequency stability monitoring of the associated Reference clock is enabled or not. By default, frequency stability monitoring is disabled. Refer to *cgmInputActivityMonitorState on page 162* for more information.

7.1.1.49 cgmInputActivityMonitorState

The `cgmInputTable` column specifying the current frequency stability state of the associated Reference clock. The frequency stability is only known when the corresponding `cgmInputActivityMonitorEna` column variable is enabled. If enabled, the input frequency of the Reference clock is verified to be within +/- 4.6 parts per million (ppm). It takes approximately a minute to qualify the frequency after the input becomes valid. The input is assigned to one of the below states.

Table 7-5 cgmInputActivityMonitor

State	Description
working(0)	Within frequency
failing(1)	The input has just been declared out of frequency. This applies only to a Slave Protection Partner.
failed(2)	Out of frequency
recovering(3)	The input has just been declared within frequency. This applies only to a Slave Protection Partner.
unknown(4)	The frequency tolerance is not being monitored or has not yet been determined.

7.1.1.50 cgmInputFrequency

The `cgmInputTable` column specifying the expected frequency of the associated Reference clock. The user may Set the frequencies for only the following Reference clocks:

- Ref A
- Ref B
- TCLKB

The frequencies of the other Reference clocks should not be Set because they are either automatically defined by the TCM or the clocks are not yet implemented. If the expected frequency of a Reference clock is set to a value not exhibited by the clock, then the clock is considered invalid and never selected as an input for the CGM.

7.1.1.51 cgmInputName

The `cgmInputTable` column specifying the name of the associated Reference clock.

7.1.1.52 cgmInputNumber

The index variable for the `cgmInputTable`. This variable is not directly accessible via SNMP queries. Some table browsers may not list the values for this variable or the table index may be listed under a different name such as Instance.

7.1.1.53 cgmInputSrc

The `cgmInputTable` column specifying the name of the source for the associated Reference clock.

7.1.1.54 cgmInputState

The `cgmInputTable` column specifying whether the state of the associated Reference clock is valid or not as determined by the CGM.

7.1.1.55 cgmInputTable

The CGM Input Table allows the user to view and configure the candidate Reference clock inputs for the CGM.

Table 7-6 is an example CGM Input Table as shown by an SNMP table browser. Note that some table browsers may not be able to list the values for the table index, `cgmInputNumber`, or the table index may be listed in a column with a different name such as Instance. The `cgmInputTable` index is simply the row number of the table (e.g., 1..12).

cgmInputNumber	cgmInputName	cgmInputSrc	cgmInputFrequency	cgmT0InputPriority	cgmT4InputPriority	cgmInputState	cgmInputActivityMonitorEna	cgmInputActivityMonitorState
1	BITS 1 Rx	notDriven	f1M544(600)	9	9	invalid(1)	disabled(1)	unknown(4)
2	BITS 2 Rx	notDriven	f1M544(600)	10	10	invalid(1)	disabled(1)	unknown(4)
3	Ref A	clkCLK3	f8k(500)	11	11	invalid(1)	disabled(1)	unknown(4)
4	Ref B	clkCLK3	f8k(500)	12	12	invalid(1)	disabled(1)	unknown(4)
5	SyncE Ref A	clkEthP1	f8k(500)	0	0	invalid(1)	disabled(1)	unknown(4)
6	SyncE Ref B	clkEthP2	f8k(500)	0	0	invalid(1)	disabled(1)	unknown(4)
7	RTM Ref	unknown	f8k(500)	0	0	invalid(1)	disabled(1)	unknown(4)
8	(reserved)	unknown	unknown(1)	0	0	unknown(2)	disabled(1)	unknown(4)
9	TCLKB	unknown	f1M544(600)	0	0	invalid(1)	disabled(1)	unknown(4)
10	M/S Sync In	clkMSRx	f8k(500)	1	0	valid(0)	disabled(1)	unknown(4)
11	(reserved)	unknown	unknown(1)	0	0	unknown(2)	disabled(1)	unknown(4)
12	SETS PTP	clk8kTest	invalid(2)	14	14	invalid(1)	disabled(1)	unknown(4)

Each row of *Table 7-6* represents a CGM Reference clock and each column shows a different aspect of the Reference clock.

The CGM Input Table MIB column objects are described below:

`cgmInputNumber` (R) - The table row number for the Reference clock. Other controls may refer to the Reference clock using this number.

`cgmInputName` (R) - A descriptive name for the input Reference clock.

`cgmInputSrc` (R) - The name of the source clock for the input. Although `cgmInputSrc` is read-only, the user may configure the sources for certain inputs at startup. Refer to the `clkSrc` configuration feature. The configurable inputs and the associated clock names are shown below.

– Ref A: `clkRefA`
– Ref B: `clkRefB`
– SyncE Ref A: `clkSyncERcvdRefA`
– SyncE Ref B: `clkSyncERcvdRefB`
– M/S Sync In: `clkMSSyncIn`

– SETS PTP: `clkSETSPTP`

`cgmInputFrequency` (R/W) - The frequency designated for the Reference clock. The user may specify the frequencies for certain inputs at startup. Refer to the `clkFreq` configuration feature. The configurable inputs and the associated clock names are shown below.

– Ref A: `clkRefA`

– Ref B: `clkRefB`

– TCLKB: `clkTCLKB`

`cgmT0InputPriority` (R/W) - The priority ranking for selecting the Reference clock for the CGM T0 Path.

`cgmT4InputPriority` (R/W) - The priority ranking for selecting the Reference clock for the CGM T4 Path.

`cgmInputState` (R) - The validity of the Reference clock as determined by the CGM. The Reference clock must have the designated frequency to be considered as a valid clock. The possible values are `valid(0)` and `invalid(1)`.

`cgmInputActivityMonitorEna` (R/W) - Whether or not frequency stability monitoring is enabled for the input. The possible values are `enabled(0)` and `disabled(1)`.

`cgmInputActivityMonitorState` (R) - The state of the frequency stability monitoring for the input. The possible values are `working(0)`, `failing(1)`, `failed(2)`, `recovering(3)`, and `unknown(4)`.

The (R) indicates that the associated aspect of the Reference clock is read-only and the (R/W) indicates the clock aspect can be Set.

7.1.1.56 cgmInterfaceMode

Get or Set the TCM Interface mode for the BITS interfaces. Refer to *Chapter 4, TCM Mode* for more information. When a Master/Slave Protection Partner Pair exists, this mode should not be set directly for a Slave. Instead, set the mode for the Master and it automatically propagates to the Slave.

7.1.1.57 cgmLogEvent

The `cgmLogEvent` is a MIB trap object. It is the only trap output by the `tclk_agent`. TCM traps are related to TCM Events. All traps are caused by Events but, depending on severity, not all Events cause traps. Refer to the *cgmSysEventTrapLevel on page 169* MIB object.

The `cgmLogEvent` trap contains the same fields as an Event. Refer to *Events on page 82* for a description of each field.

7.1.1.58 cgmMasterMode

Get or Set the TCM Master mode. Refer to *Chapter 4, TCM Mode* for more information.

The Master Mode cannot be Set while the CGM is enabled. Refer to the *cgmEnable on page 159* MIB object.

7.1.1.59 cgmOXSelect

Get or Set the oscillator used by the CGM.

7.1.1.60 cgmPathInputT0

Gets the row of the `cgmInputTable` specifying the Reference clock currently selected for the CGM T0 Path.

7.1.1.61 cgmPathInputT4

Gets the row of the `cgmInputTable` specifying the Reference clock selected for the CGM T4 Path.

7.1.1.62 cgmPathStateT0

Gets the current state of the CGM T0 Path.

7.1.1.63 cgmPathStateT0Time

Gets the elapsed time in seconds since the last state change of the T0 PLL. This will be zero if there has been no state change.

7.1.1.64 cgmPathT0LockedTime

Gets the duration in seconds of the current or most recent locked period of the T0 PLL. This will be zero if there has been no locked period.

7.1.1.65 cgmProtectionCmd

A Get returns the last Protection Command sent to the local member of a Protection Partner Pair. A Set sends the command to the local member. The command is ignored if compliance is not possible.

The following Protection Commands are supported:

`noCmd(0)`: A Set with this value has no effect. A Get of this value means no command has been sent.

`becomeMaster(1)`: A Set with this value requests that, if possible, the local Protection Partner become a Slave and the remote Protection Partner become a Master. This is the preferred command for requesting swap of Master and Slave.

`becomeSlave(2)`: A Set with this value requests that, if possible, the local Protection Partner become a Master and the remote Protection Partner become a Slave. `becomeMaster(1)` is the preferred command for requesting swap of Master and Slave.

`forceMaster(3)`: A Set with this value requests that the local Protection Partner become a Master even if it has no valid inputs. The remote Protection Partner will become Slave unless its Master/Slave sync clock is invalid. This command should not be used in a normal operational environment and is only meant for testing or maintenance purposes.

`unforceMaster(4)`: A Set with this value reverses the effect, if any, of a previous `forceMaster(3)` command.

7.1.1.66 cgmProtectionMasterToSlaveTrackDelay

Get or Set the interconnection delay for the Master/Slave clock sync signal in picoseconds. This value is used to keep the sync clocks from the two Protection Partners phase aligned. Both Protection Partners should be configured to use the same value. The value used will depend on the chosen transmission path, either a front panel cable or the backplane, and characteristics of the path such as cable length and chassis type. The default value is a nominal value for when using the backplane.

Important Information

Changing this value at runtime may cause a phase transition in the Master/Slave sync clock and the System clocks guided by this signal. Any change made will only be temporary since the value is automatically updated as the temperature of the blade fluctuates. The updated value is based on the initially configured value at TCM startup.

7.1.1.67 cgmProtectionPartnerAddress

Get the IP address of the remote Protection Partner. The remote Protection Partner uses this address to exchange UDP packets with the local Protection Partner.

Although `cgmProtectionPartnerAddress` is writable, the setting of this MIB object is not currently supported. Instead, configure this item at startup using the same-named Configuration Item.

7.1.1.68 cgmProtectionRevertiveModeState

Get or Set the state of the CGM Revertive Mode. When a CGM is in revertive operation (`enabled(0)`), it will always choose the highest priority valid input when it becomes available. When in non-revertive operation, a selected input will remain selected until it fails. The highest priority valid input is chosen at that point and remains selected until it fails. The Revertive Mode should only be set for the Master of a Protection Partner Pair. The setting for the Master is propagated to the Slave.

7.1.1.69 cgmProtectionState

Get whether this module is the Master or Slave of a Protection Partner Pair. If the remote Partner is not accessible via the network, then the local Partner will run as a Standalone Master.

7.1.1.70 cgmStratumLevel

Get which Stratum Level is currently requested for support. The desired Stratum Level can only be configured at startup using the `cgmStratumLevel` configuration item. All blades with a Telco clocking oscillator support Stratum 3. The `tclk_agent` currently supports Stratum 3 only.

7.1.1.71 cgmSyncEGenEnable

Get the last successful setting of this object. Set to `enable(0)` to enable a CGM System clock to source the F125 Synchronous Ethernet device clocks. Set to `disable(1)` to disable the CGM System clock from driving the Synchronous Ethernet circuitry.The specific CGM System clock used to drive the SyncE circuitry is not configurable. During TCM shutdown, this feature is automatically disabled if it is currently enabled.

When the blade is booted, the clock for the Ethernet circuitry comes from an onboard oscillator. This clock is suitable for the operations of ordinary Ethernet, but not Synchronous Ethernet (SyncE). Changing the source of the clock, for example, enabling or disabling `cgmSyncEGenEnable`, can cause disruptions to any ongoing Ethernet operations.

7.1.1.72 cgmSysEventLogClear

A Get returns the last Set value or `retrievedEvents(0)` if there has been no Set. A Set sends a command for removing Events from the `cgmEventTable` as follows:

`retrievedEvents(0)`: A Set with this value removes all Events that have already been retrieved via SNMP. Events that have yet to be retrieved remain in the table.

`allEvents(1)`: A Set with this value removes all Events, even if they have not been previously retrieved.

7.1.1.73 cgmSysEventLogCount

Get the current number of entries in the `cgmEventTable`.

7.1.1.74 cgmSysEventLogLevel

Get or Set the significance an Event must have to be put into the `cgmEventTable`. The settings have the follow effect:

`nothing(0)`: No Events are put into the table.

`critical(1)`: Only Events with a `cgmEventSeverity` of `critical(0)` are put into the table.

`warning(2)`: Only Events with a `cgmEventSeverity` of `warning(1)` or `critical(0)` are put into the table.

`event(3)`: Only Events with a `cgmEventSeverity` of `event(2)`, `warning(1)`, or `critical(0)` are put into the table.

`all(4)`: All Events are put into the table.

7.1.1.75 cgmSysEventLogSize

Get the maximum number of entries that can be in the `cgmEventTable`.

7.1.1.76 cgmSysEventTrapLevel

Get or Set the level of the TCM Events that also get sent as SNMP traps. All Events with a Severity Code greater than or equal to this level will get sent as SNMP traps.

7.1.1.77 cgmT0InputPriority

The `cgmInputTable` column specifying the priority of the associated Reference clock for the T0 Path of the CGM. This can be any integer from 0..15 except 1. Reference clocks with lower priority values are selected as inputs before those with higher values. Zero indicates the Reference clock should never be selected. The priority value 1 is reserved for the M/S Sync In Reference clock. When a Master/Slave Protection Partner Pair exists, the T0 Path priority for a clock should not be set directly for a Slave. Instead, set the priority for the Master and it is automatically propagated to the Slave.

7.1.1.78 cgmT4ClkSquelch

Get or Set whether outputs from the T4 Path are squelched when the T4 PLL is not locked to a Reference clock. Set to enable(0) to squelch and disable(1) to not squelch.

7.1.1.79 cgmT4InputPriority

The `cgmInputTable` column specifying the priority of the associated Reference clock for the T4 Path of the CGM. This can be any integer 0..15 except 1. Reference clocks with lower priority values are selected as inputs before those with higher values. Zero indicates the Reference clock should never be selected. The priority value 1 is reserved.

7.1.1.80 cgmTCMVersion

Get the TCM version.

7.1.1.81 cgmTSFWVersion

Get the ToPSync firmware version.

7.1.1.82 dsx1LineCode

The `cgmBitsTable` column specifying the E1/T1 Line Code.

7.1.1.83 dsx1LineStatus

The `cgmBitsTable` column specifying the current line status.

7.1.1.84 dsx1LineType

The `cgmBitsTable` column specifying the E1/T1 Line Type. Setting the Line Type also sets `bitsT3ClkSquelch` such that it is enabled for framed modes and disabled for unframed modes.

7.1.1.85 dsx1LoopbackConfig

The `cgmBitsTable` column specifying the loopback mode.

7.1.1.86 tcmCmd

This variable is reserved for SMART EC testing.

7.1.2 IEEE1588-MIB

The following subsections describe each of the MIB objects summarized in *Table 7-2*.

7.1.2.1 ptpAcceptablePartnerAddress

The `ptpAcceptablePartnerTable` column specifying the IP address of an Acceptable Partner in dotted-decimal format.

7.1.2.2 ptpAcceptablePartnerAddressType

The `ptpAcceptablePartnerTable` column specifying the type of the IP address of an Acceptable Partner. Only `ipv4(1)` is supported.

7.1.2.3 ptpAcceptablePartnerIsSlave

The `ptpAcceptablePartnerTable` column specifying whether an Acceptable Partner is a Slave or Master. `true(1)` indicates an Acceptable Slave, otherwise an Acceptable Master.

7.1.2.4 ptpAcceptablePartnerPortNumber

The `ptpAcceptablePartnerTable` column specifying the TCM PTP port associated with an Acceptable Partner.

7.1.2.5 ptpAcceptablePartnerPriority1

The ptpAcceptablePartnerTable column specifying the PTP *alternatePriority1* of an Acceptable Master. Refer to *Section 17.6 in IEEE1588v2*. The default value is 0.

7.1.2.6 ptpAcceptablePartnerPriority2

The `ptpAcceptablePartnerTable` column specifying an alternate PTP *priority2* for an Acceptable Master. This is a TCM extension of the PTP standard that serves a function corresponding to PTP *alternatePriority1*, but applying to PTP *priority2*. The default value is 0. Refer to *Section 17.6 in IEEE1588v2*.

7.1.2.7 ptpAcceptablePartnerRqstUnicast

The `ptpAcceptablePartnerTable` column specifying whether an Acceptable Partner is requested to use unicast Announce messages. `true(1)` indicates an Acceptable Partner is requested to use unicast Announce messages, otherwise the Partner may use choose to use multicast. The TCM currently supports only unicast Announce messages. This variable pertains to an Acceptable Master only.

7.1.2.8 ptpAcceptablePartnerTable

The PTP Acceptable Partner Table supported by the TCM is an extension to the PTP Acceptable Master Table defined by *IEEE1588v2*. Refer to section *17.6 in IEEE1588v2*.

The PTP Acceptable Partner Table allows the user to view the Acceptable Partners associated with the TCM PTP ports. The Acceptable Partners for a port are the PTP nodes with which the port may interact, either as the possible Slaves of a Master port or the possible Masters of a Slave port. If a PTP port is not restricted to Acceptable Partners, then the port may interact with any PTP node in the same PTP Domain that establishes contact.

Note that Acceptable Partners are required for all PTP nodes in a network that, like the TCM, use only unicast PTP message addressing. Otherwise the nodes will not establish contact with each other.

An Acceptable Partner is either an Acceptable Master or Acceptable Slave. An Acceptable Master is a PTP Master allowed to master the associated TCM PTP port when it is a PTP Slave. An Acceptable Slave is a PTP Slave allowed to be mastered by the associated TCM PTP port when it is a PTP Master.

Acceptable Masters defined for a port are ignored unless the `ptpClockPortDSUseAcceptableMasters` configuration/MIB variable is `true(1)`. Similarly, Acceptable Slaves defined for a port are ignored unless the `ptpClockPortDSUseAcceptableSlaves` configuration/MIB variable is `true(1)`.

If `ptpClockPortDSUseAcceptableMasters is true(1)` and no Acceptable Masters are defined for a port, then the port will not be able to become a PTP Slave. Similarly, if `ptpClockPortDSUseAcceptableSlaves` is `true(1)` and no Acceptable Slaves are defined for a port, then the port will not be able to Master any PTP nodes.

The Acceptable Partner Table can be populated at startup via the TCM configuration file. Refer to *PTP Acceptable Partner Configuration on page 105*. Runtime additions and deletions can be made to the Acceptable Partner Table by using the `ptpClockPortDSAcceptablePartnerAddressAdd` and `ptpClockPortDSAcceptablePartnerAddressDel` columns, respectively, of the `ptpClockPortDSTable`. For more information, refer to *ptpClockPortDSTable on page 180*.

The number of Acceptable Partners for a single port is restricted to no more than 10 Acceptable Masters and 64 Acceptable Slaves. There is also a restriction of no more than 74 Acceptable Partners for all ports combined.

The below table is an example Acceptable Partner Table as shown by an SNMP table browser. Each row of the table represents an Acceptable Partner and each column shows a different aspect of the Partner. Note that some table browsers may not be able to list the values for the table index, `ptpAcceptablePartnerEntryIndex`, or the table index may be listed in a column with a different name such as Instance.

Below is a summary of the columns. An (R) indicates that the column variable is read-only and (R/W) indicates the variable can be Set. Refer to the subsection for each column variable for more information. Note that some columns only apply to an Acceptable Master and are ignored for an Acceptable Slave.

`ptpAcceptablePartnerEntryIndex` (R)	The table row number of the Acceptable Partner.
`ptpAcceptablePartnerPortNumber` (R)	The TCM PTP port for the Acceptable Partner.
`ptpAcceptablePartnerIsSlave` (R)	Whether the Acceptable Partner is a Slave or Master.
`ptpAcceptablePartnerAddressType` (R)	The type of the Acceptable Partner IP address.
`ptpAcceptablePartnerAddress` (R)	The IP address of the Acceptable Partner.
`ptpAcceptablePartnerPriority1` (R)	The PTP *alternatePriority1* of the Acceptable Master.
`ptpAcceptablePartnerPriority2` (R)	Similar to `ptpAcceptablePartnerPriority1` except applying to PTP *priority2* of a Master.
`ptpAcceptablePartnerRqstUnicast` (R)	Whether an Acceptable Master is requested to use unicast Announce messages

ptpAcceptablePartnerEntryIndex	ptpAcceptablePartnerPortNumber	ptpAcceptablePartnerIsSlave	ptpAcceptablePartnerAddressType	ptpAcceptablePartnerAddress	ptpAcceptablePartnerPriority1	ptpAcceptablePartnerPriority2	ptpAcceptablePartnerRqstUnicast
1	1	false(0)	ipv4(1)	10.206.159.90	128	128	true(1)
2	1	false(0)	ipv4(1)	10.206.159.91	128	128	true(1)
3	2	true(1)	ipv4(1)	10.206.159.92	128	128	false(0)

7.1.2.9 ptpClockCurrentDSMeanPathDelay

This variable represents the meanPathDelay member of the PTP Current Data Set. The value is the current mean propagation time in nanoseconds between the TCM PTP Slave clock and its Master. Refer to *Section 8.2.2* in *IEEE1588v2.*

7.1.2.10 ptpClockCurrentDSOffsetFromMaster

This variable represents the offsetFromMaster member of the PTP Current Data Set. The value is the current time difference in nanoseconds between the TCM PTP Slave clock and its Master. Refer to *Section 8.2.2 in IEEE1588v2.*

7.1.2.11 ptpClockCurrentDSStepsRemoved

This variable represents the stepsRemoved member of the PTP Current Data Set. The value is the number of steps on the network that the TCM is from the current PTP Grandmaster. The value is 1 if the TCM and Grandmaster are on the same communication path. Refer to *Section 8.2.2 in IEEE1588v2.*

7.1.2.12 ptpClockDefaultDSClockIdentity

This variable represents the clockIdentity member of the PTP Default Data Set. The value is the Clock Identity PTP device attribute of the TCM. The Identity is derived from the ToPSync MAC address. Refer to *Sections 7.5.2.2, 7.6.2, and 8.2.1 in IEEE1588v2* and *PTP Default Data Set Configuration on page 103*.

7.1.2.13 ptpClockDefaultDSDomain

This variable represents the domainNumber member of the PTP Default Data Set. The value is the PTP Domain for a TCM PTP Slave port. Refer to Section 8.2.1 in *IEEE1588v2* and *PTP Default Data Set Configuration on page 103*.

7.1.2.14 ptpClockDefaultDSNumOfPorts

This variable represents the numberPorts member of the PTP Default Data Set. The value is the number of PTP ports for the TCM. Refer to *Sections 7.6.2 and 8.2.1 in IEEE1588v2* and *PTP Default Data Set Configuration on page 103*.

7.1.2.15 ptpClockDefaultDSPriority1

This variable represents the priority1 member of the PTP Default Data Set. The value is the *Priority 1* PTP device attribute for the TCM. Refer to *Sections 7.6.2 and 8.2.1 in IEEE1588v2* and *PTP Default Data Set Configuration on page 103*.

7.1.2.16 ptpClockDefaultDSPriority2

This variable represents the priority2 member of the PTP Default Data Set. The value is the Priority 2 PTP device attribute for the TCM. Refer to *Sections 7.6.2 and 8.2.1 in IEEE1588v2* and *PTP Default Data Set Configuration on page 103*.

7.1.2.17 ptpClockDefaultDSQualityAccuracy

This variable represents the clockQuality.clockAccuracy member of the PTP Default Data Set. The value is the Clock Quality Accuracy PTP device attribute for the TCM. Refer to Sections 7.6.2 and 8.2.1 in IEEE1588v2 and *PTP Default Data Set Configuration on page 103*.

7.1.2.18 ptpClockDefaultDSQualityClass

This variable represents the clockQuality.clockClass member of the PTP Default Data Set. The value is the Clock Quality Class PTP device attribute for the TCM. Refer to Sections 7.6.2 and 8.2.1 in IEEE1588v2 and *PTP Default Data Set Configuration on page 103*.

7.1.2.19 ptpClockDefaultDSQualityOffset

This variable represents the clockQuality.offsetScaledLogVariance member of the PTP Default Data Set. The value is the Clock Quality Offset PTP device attribute for the TCM. Refer to *Section 8.2.1 in IEEE1588v2* and *PTP Default Data Set Configuration on page 103*.

7.1.2.20 ptpClockDefaultDSSlaveOnly

This variable represents the slaveOnly member of the PTP Default Data Set. If PTP operations are enabled and this variable is true(1), then the TCM operates as a Slave-Only Clock and the ptpBoundaryClockEnable variable is ignored. Refer to Section 8.2.1 in IEEE1588v2, *PTP Default Data Set Configuration on page 103*, and *PTP Operations Overview on page 59*.

7.1.2.21 ptpClockDefaultDSTwoStepFlag

This variable represents the twoStepFlag member of the PTP Default Data Set. The TCM supports One-Step Clock operations only. Refer to *Section 8.2.1 in IEEE1588v2* and *PTP Default Data Set Configuration on page 103*.

7.1.2.22 ptpClockParentDSClockPhChRate

This variable represents the observedParentClockPhaseChangeRate member of the PTP Parent Data Set. The TCM does not support reporting this or other Parent Clock Statistics. Refer to *Section 8.2.3 in IEEE1588v2.*

7.1.2.23 ptpClockParentDSGMClockIdentity

This variable represents the grandmasterIdentity member of the PTP Parent Data Set. Refer to *Section 8.2.3 in IEEE1588v2.*

7.1.2.24 ptpClockParentDSGMClockPriority1

This variable represents the grandmasterPriority1 member of the PTP Parent Data Set. Refer to *Section 8.2.3 in IEEE1588v2.*

7.1.2.25 ptpClockParentDSGMClockPriority2

This variable represents the grandmasterPriority2 member of the PTP Parent Data Set. Refer to *Section 8.2.3 in IEEE1588v2.*

7.1.2.26 ptpClockParentDSGMClockQualityAccuracy

This variable represents the grandmasterClockQuality.clockAccuracy member of the PTP Parent Data Set. Refer to *Section 8.2.3 in IEEE1588v2.*

7.1.2.27 ptpClockParentDSGMClockQualityClass

This variable represents the grandmasterClockQuality.clockClass member of the PTP Parent Data Set. Refer to *Section 8.2.3 in IEEE1588v2.*

7.1.2.28 ptpClockParentDSGMClockQualityOffset

This variable represents the grandmasterClockQuality.offsetScaledLogVariance member of the PTP Parent Data Set. Refer to *Section 8.2.3 in IEEE1588v2.*

7.1.2.29 ptpClockParentDSOffset

This variable represents the observedParentOffsetScaledLogVariance member of the PTP Parent Data Set. The TCM does not support reporting this or other Parent Clock Statistics. Refer to *Section 8.2.3 in IEEE1588v2.*

7.1.2.30 ptpClockParentDSParentPortIdentity

This variable represents the parentPortIdentity member of the PTP Parent Data Set. Refer to *Section 8.2.3 in IEEE1588v2.*

7.1.2.31 ptpClockParentDSParentStats

This variable represents the parentStats member of the PTP Parent Data Set. The TCM does not support reporting Parent Clock Statistics and the value of this variable is always false(0). Refer to *Section 8.2.3 in IEEE1588v2.*

7.1.2.32 ptpClockPortDSAcceptablePartnerAddressAdd

The `ptpClockPortDSTable` column for adding Acceptable Partners to the `ptpAcceptablePartnerTable`. Set this column to the IP address of an Acceptable Partner. The Acceptable Partner is assigned to the TCM port corresponding to the row of the `ptpClockPortDSTable`. The attributes of the Acceptable Partner are defined by the current values of the following columns of the `ptpClockPortDSTable` row:

`ptpClockPortDSAcceptablePartnerIsSlave`

`ptpClockPortDSAcceptablePartnerPriority1`

`ptpClockPortDSAcceptablePartnerPriority2`

`ptpClockPortDSAcceptablePartnerRqstUnicast`

`ptpClockPortDSAcceptablePartnerAddressType`

A Get returns the last value Set or NULL if never Set. An attempt to add an Acceptable Partner already of the same type (Master or Slave) with the same address to the same port will fail.

The number of Acceptable Partners for a single port is restricted to no more than 10 Acceptable Masters and 64 Acceptable Slaves. There is also a restriction of no more than 74 Acceptable Partners for all ports combined. An attempt to add an Acceptable Partner that will cause the total to go beyond these maximums will fail.

7.1.2.33 ptpClockPortDSAcceptablePartnerAddressDel

The `ptpClockPortDSTable` column for removing Acceptable Partners from the `ptpAcceptablePartnerTable`. Set this column to the IP address of an Acceptable Partner. The Acceptable Partner is deleted for the TCM port corresponding to the row of the `ptpClockPortDSTable`. The attributes of the Acceptable Partner deleted are defined by the current value of the following column of the `ptpClockPortDSTable` row:

`ptpClockPortDSAcceptablePartnerIsSlave`

An attempt to delete an Acceptable Partner not present in the table will fail.

A Get returns the last value Set or NULL if never Set.

7.1.2.34 ptpClockPortDSAcceptablePartnerAddressType

The `ptpClockPortDSTable` column variable for specifying the IP address type of an Acceptable Partner added via `ptpClockPortDSAcceptablePartnerAddressAdd`.

The TCM supports only the `ipv4(1)` address type and the value of this variable is ignored.

7.1.2.35 ptpClockPortDSAcceptablePartnerIsSlave

The `ptpClockPortDSTable column` variable for specifying whether an Acceptable Partner added via `ptpClockPortDSAcceptablePartnerAddressAdd` or removed via `ptpClockPortDSAcceptablePartnerAddressDel` is an Acceptable Slave or an Acceptable Master. Set to `true(1)` for an Acceptable Slave, otherwise for an Acceptable Master.

7.1.2.36 ptpClockPortDSAcceptablePartnerPriority1

The `ptpClockPortDSTable` column variable for specifying the PTP *alternatePriority1* of an Acceptable Master added via `ptpClockPortDSAcceptablePartnerAddressAdd`. Refer to *Section 17.6 in IEEE1588v2*.

7.1.2.37 ptpClockPortDSAcceptablePartnerPriority2

The `ptpClockPortDSTable` column variable for specifying the alternate PTP *priority2* of an Acceptable Master added via `ptpClockPortDSAcceptablePartnerAddressAdd`. This is a TCM extension of the PTP standard that serves a function corresponding to PTP *alternatePriority1*, but applying to PTP *priority2*. Refer to *Section 17.6 in IEEE1588v2*.

7.1.2.38 ptpClockPortDSAcceptablePartnerRqstUnicast

The `ptpClockPortDSTable` column variable for specifying whether an Acceptable Master added via `ptpClockPortDSAcceptablePartnerAddressAdd` is requested to use unicast Announce messages, otherwise the Partner may choose to use multicast. The TCM supports only unicast Announce messages.

7.1.2.39 ptpClockPortDSAnnouncementInterval

This `ptpClockPortDSTable` column variable represents the *logAnnounceInterval* member of the PTP Port Data Set. The value is the log Announce Interval for the corresponding TCM PTP port. Refer to *Section 8.2.5 in IEEE1588v2*.

7.1.2.40 ptpClockPortDSAnnounceRctTimeout

This `ptpClockPortDSTable` column variable represents the `announceReceiptTimeout` member of the PTP Port Data Set. The TCM does not support this variable. Refer to *Section 8.2.5 in IEEE1588v2.*

7.1.2.41 ptpClockPortDSDelayMech

This `ptpClockPortDSTable` column variable represents the *delayMechanism* member of the PTP Port Data Set. The value is the Delay Mechanism for the corresponding TCM PTP port. The TCM supports only the `e2e(1)` mechanism. Changing this variable is not supported. Refer to *Section 8.2.5 in IEEE1588v2.*

7.1.2.42 ptpClockPortDSDomain

This `ptpClockPortDSTable` column variable represents the PTP *domainNumber* of a port. The value is the PTP Domain for the corresponding TCM PTP port. Refer to Section 7.1 in IEEE1588v2.

7.1.2.43 ptpClockPortDSMinDelayReqInterval

This `ptpClockPortDSTable` column variable represents the *logMinDelayReqInterval* member of the PTP Port Data Set. The value is the log Minimum Delay Request Interval for the corresponding TCM PTP port. Refer to *Section 8.2.5 in IEEE1588v2*.

7.1.2.44 ptpClockPortDSMinPeerDelayReqInterval

This `ptpClockPortDSTable` column variable represents the *logMinPdelayReqInterval* member of the PTP Port Data Set. This column is not supported. Refer to *Section 8.2.5 in IEEE1588v2*.

7.1.2.45 ptpClockPortDSPeerMeanPathDelay

This `ptpClockPortDSTable` column variable represents the *peerMeanPathDelay* member of the PTP Port Data Set. The value is the Peer Mean Path Delay for the corresponding TCM PTP port. Refer to *Section 8.2.5 in IEEE1588v2*.

7.1.2.46 ptpClockPortDSPortIdentity

This ptpClockPortDSTable column variable represents the portIdentity member of the PTP Port Data Set. The value is the Port Identity for the corresponding TCM PTP port. Both ports use the same Clock Identity as part of the Port Identity. Refer to *Section 8.2.5 in IEEE1588v2*.

7.1.2.47 ptpClockPortDSPTPVersion

This `ptpClockPortDSTable` column variable represents the *versionNumber* member of the PTP Port Data Set. The value is the PTP Version for the corresponding TCM PTP port. The TCM supports only version 2. Refer to *Section 8.2.5 in IEEE1588v2*.

7.1.2.48 ptpClockPortDSRunningState

This ptp`ClockPortDSTable` column variable represents the *portState* member of the PTP Port Data Set. The value is the PTP State for the corresponding TCM PTP port. Refer to *Section 8.2.5 in IEEE1588v2*.

7.1.2.49 ptpClockPortDSSyncInterval

This `ptpClockPortDSTable` column variable represents the *logSyncInterval* member of the PTP Port Data Set. The value is the log Sync Interval for the corresponding TCM PTP port. Refer to *Section 8.2.5 in IEEE1588v2*.

7.1.2.50 ptpClockPortDSTable

The PTP Clock Port Data Set Table allows the user to view and update properties of the two TCM PTP ports. The table also allows the user to make runtime additions or deletions to the PTP Acceptable Partner Table for each port.

The port properties of the PTP Clock Port Data Set Table are a superset of the PTP Port Data Set defined by IEEE1588v2 Section 8.2.5. This data set specifies values for PTP protocol decisions and message fields for each PTP port. The following table maps PTP Port Data Set member names to TCM Clock Port Data Set Table column names.

PTP Port Data Set Member	`ptpClockPortDSTable` column name
portIdentity	ptpClockPortDSPortIdentity
portState	ptpClockPortDSRunningState
logMinDelayReqInterval	ptpClockPortDSMinDelayReqInterval
peerMeanPathDelay	ptpClockPortDSPeerMeanPathDelay
logAnnounceInterval	ptpClockPortDSAnnouncementInterval
announceReceiptTimeout	ptpClockPortDSAnnounceRctTimeout
logSyncInterval	ptpClockPortDSSyncInterval
delayMechanism	ptpClockPortDSDelayMech
logMinPdelayReqInterval	ptpClockPortDSMinPeerDelayReqInterval
versionNumber	ptpClockPortDSPTPVersion

The TCM extends the PTP Port Data Set with additional port characteristics. Most of these characteristics are related to usage of PTP Acceptable Partners for a port.

The below table is an example Clock Port Data Set Table as shown by an SNMP table browser. Each row of the table represents one of the TCM PTP ports and each column shows a different aspect of the port. Note that some table browsers may not be able to list the values for the table index, ptpPortNumber, or the table index may be listed in a column with a different name such as Instance.

Below is a summary of the table columns. An (R) indicates that the column variable is read-only and (R/W) indicates the variable can also be Set. Refer to the subsection for each column variable for more information.

`ptpPortNumber` (R)	The TCM PTP port number, 1 or 2.
`ptpClockPortDSPortIdentity` (R)	The PTP portIdentity of the port.
`ptpClockPortDSDomain` (R)	The PTP domainNumber of the port.
`ptpClockPortDSRunningState` (R/W)	The PTP portState of the port.
`ptpClockPortDSMinDelayReqInterval` (R/W)	The PTP *logMinDelayReqInterval* for the port.
`ptpClockPortDSMinPeerDelayReqInterval` (R/W)	The PTP `logMinPdelayReqInterval` for the port. This column is not supported.
`ptpClockPortDSPeerMeanPathDelay` (R)	The PTP *peerMeanPathDelay* for the port.
`ptpClockPortDSAnnouncementInterval` (R)	The PTP *logAnnounceInterval* for the port.
`ptpClockPortDSAnnounceRctTimeout` (R/W)	The PTP *announceReceiptTimeout* for the port. This column is not supported.
`ptpClockPortDSSyncInterval` (R)	The PTP logSyncInterval for the port.
`ptpClockPortDSDelayMech` (R/W)	The PTP *delayMechanism* for the port. Changing this variable is not supported.
`ptpClockPortDSPTPVersion` (R)	The PTP versionNumber for the port.
`ptpClockPortDSUseAcceptableMasters` (R/W)	Whether the port is restricted to the Acceptable Partner Table Masters defined for the port.
`ptpClockPortDSUseAcceptableSlaves` (R/W)	Whether the port is restricted to the Acceptable Partner Table Slaves defined for the port.
`ptpClockPortDSAcceptablePartnerIsSlave` (R/W)	Whether a Partner added to the Acceptable Partners Table for the port via `ptpClockPortDSAcceptablePartnerAddressAdd` is a PTP Slave.
`ptpClockPortDSAcceptablePartnerPriority1` (R/W)	Specifies the PTP *priority1* of a Master added to the Acceptable Partner Table via `ptpClockPortDSAcceptablePartnerAddressAdd`.
`ptpClockPortDSAcceptablePartnerPriority2` (R/W)	Specifies the PTP *priority2* of a Master added to the Acceptable Partner Table via `ptpClockPortDSAcceptablePartnerAddressAdd`.

`ptpClockPortDSAcceptablePartnerRqstUnicast` (R/W)	Whether a Master added to the Acceptable Partners Table for the port via `ptpClockPortDSAcceptablePartnerAddressAdd` is requested to use unicast Announce messages.
`ptpClockPortDSAcceptablePartnerAddressType` (R/W)	Specifies the address type of a Partner added to the Acceptable Partner Table via `ptpClockPortDSAcceptablePartnerAddressAdd`. Only ipv4(1) is supported and the value of this variable is ignored.
`ptpClockPortDSAcceptablePartnerAddressAdd` (R/W)	The address of a Partner to be added to the Acceptable Partner Table for the port.
`ptpClockPortDSAcceptablePartnerAddressDel` (R/W)	The address of a Partner to be deleted from the Acceptable Partner Table for the port

.

ptpPortNumber	ptpClockPortDSPortIdentity	ptpClockPortDSDomain	ptpClockPortDSRunningState	ptpClockPortDSMinDelayReqInterval	ptpClockPortDSMinPeerDelayReqInterval	ptpClockPortDSPeerMeanPathDelay	ptpClockPortDSAnnouncementInterval
1	00 16 c0 ff fe 00 fc de	alt1(1)	passive(7)	-5	not available	0	1
2	00 16 c0 ff fe 00 fc de	default(0)	master(6)	-5	not available	0	1

ptpClockPortDSAnnounceRctTimeout	ptpClockPortDSSyncInterval	ptpClockPortDSDelayMech	ptpClockPortDSPTPVersion	ptpClockPortDSUseAcceptableMasters	ptpClockPortDSUseAcceptableSlaves	ptpClockPortDSAcceptablePartnerIsSlave	ptpClockPortDSAcceptablePartnerPriority1
not available	-5	e2e(1)	2	true(1)	false(0)	false(0)	128
not available	-5	e2e(1)	2	false(0)	true(1)	true(1)	128

ptpClockPortDSAcceptablePartnerPriority2	ptpClockPortDSAcceptablePartnerRqstUnicast	ptpClockPortDSAcceptablePartnerAddressType	ptpClockPortDSAcceptablePartnerAddressAdd	ptpClockPortDSAcceptablePartnerAddressDel
128	true(1)	ipv4(1)		
128	true(1)	ipv4(1)		

7.1.2.51 ptpClockPortDSUseAcceptableMasters

This `ptpClockPortDSTable` column variable specifies whether the corresponding port, when a PTP Slave, is restricted to interacting with only the Masters defined for the port in the Acceptable Partner Table. If `true(1)`, the port is restricted. If `false(0)`, the port may interact with any PTP Master that establishes contact and is in the same PTP Domain as the port.

7.1.2.52 ptpClockPortDSUseAcceptableSlaves

This `ptpClockPortDSTable` column variable specifies whether the corresponding port, when a Master, is restricted to interacting with only the Slaves defined for the port in the Acceptable Partner Table. If `true(1)`, the port is restricted. If `false(0)`, the port may interact with any PTP Slave that establishes contact and is in the same PTP Domain as the port.

7.1.2.53 ptpClockTimePropertiesDSCurrentUTCOffset

This variable represents the *currentUtcOffset* member of the PTP Time Properties Data Set. Use this column to view the offset in seconds between TAI and UTC time. The value only has meaning when *ptpClockTimePropertiesDSCurrentUTCOffsetValid* is `true(1)`. Refer to *Section 8.2.4 in IEEE1588v2*.

7.1.2.54 ptpClockTimePropertiesDSCurrentUTCOffsetValid

This variable represents the `currentUtcOffsetValid` member of the PTP Time Properties Data Set. If `true(1)`, then `ptpClockTimePropertiesDSCurrentUTCOffset` is known to have a proper value, otherwise not. Refer to *Section 8.2.4 in IEEE1588v2*.

7.1.2.55 ptpClockTimePropertiesDSFreqTraceable

This variable represents the `frequencyTraceable` member of the PTP Time Properties Data Set. If `true(1)`, then the frequency determining the timescale is traceable to a primary standard, otherwise not. Refer to *Section 8.2.4 in IEEE1588v2*.

7.1.2.56 ptpClockTimePropertiesDSLeap59

This variable represents the leap59 member of the PTP Time Properties Data Set. If this variable and `ptpClockTimePropertiesDSPTPTimescale` are both `true(1)`, then the last minute of the current UTC day will contain 59 seconds, otherwise 60 seconds. Refer to *Section 8.2.4 in IEEE1588v2*.

7.1.2.57 ptpClockTimePropertiesDSLeap61

This variable represents the leap61 member of the PTP Time Properties Data Set. If this variable and `ptpClockTimePropertiesDSPTPTimescale` are both `true(1)`, then the last minute of the current UTC day will contain 61 seconds, otherwise 60 seconds. Refer to *Section 8.2.4 in IEEE1588v2*.

7.1.2.58 ptpClockTimePropertiesDSPTPTimescale

This variable represents the *ptpTimescale* member of the PTP Time Properties Data Set. If `true(1)`, then the clock timescale of the Grandmaster clock is PTP, otherwise it is unknown. Refer to Section *8.2.4 in IEEE1588v2.*

7.1.2.59 ptpClockTimePropertiesDSSource

This variable represents the timeSource member of the PTP Time Properties Data Set. Use this column to view the source of time used by the Grandmaster clock. Refer to *Section 8.2.4 in IEEE1588v2.*

7.1.2.60 ptpClockTimePropertiesDSTimeTraceable

This variable represents the timeTraceable member of the PTP Time Properties Data Set. If `true(1)`, then the timescale and the value of `ptpClockTimePropertiesDSCurrentUTCOffset` are traceable to a primary standard, otherwise the traceability is unknown. Refer to *Section 8.2.4 in IEEE1588v2*.

7.1.2.61 ptpCurrentMaster

This variable specifies the IP address of the PTP Master of the TCM.

7.1.2.62 ptpCurrentMasterAddressType

This variable specifies the type of the IP address of the PTP Master of the TCM. The TCM supports only the `ipv4(1)` address type.

7.1.2.63 ptpVisibleMasterAddress

The `ptpVisibleMasterTable` column variable specifying the IP address of a visible PTP Master for a TCM PTP Slave port.

7.1.2.64 ptpVisibleMasterAddressType

The `ptpVisibleMasterTable` column variable specifying the address type of a visible PTP Master for a TCM PTP Slave port.

7.1.2.65 ptpVisibleMasterPriority1

The `ptpVisibleMasterTable` column variable specifying the PTP priority1 of a visible PTP Master for a TCM PTP Slave port.

7.1.2.66 ptpVisibleMasterPriority2

The `ptpVisibleMasterTable` column variable specifying the PTP priority2 of a visible PTP Master for a TCM PTP Slave port.

7.1.2.67 ptpVisibleMasterTable

The PTP Visible Master Table allows the user to view details about the PTP Masters with which a TCM PTP Slave port has established contact. Only a Visible Master can become a Master of the Slave port. Which Visible Master becomes the current Master for the port depends on any restrictions based on the usage of Acceptable Masters and the PTP Best Master algorithm.

The below table is an example Visible Master Table as shown by an SNMP table browser. Each row of the table represents one Visible Master and each column shows a different aspect of the Master. Note that some table browsers may not be able to list the values for the table index, `ptpVisibleMasterEntryIndex,` or the table index may be listed in a column with a different name such as “Instance”.

Below is a summary of the table columns. An (R) indicates that the column variable is read-only and (R/W) indicates the variable can also be set. Refer to the subsection for each column variable for more information.

`ptpVisibleMasterEntryIndex` (R)	The table row number.
`ptpVisibleMasterAddressType` (R)	The address type of the Visible Master address.
`ptpVisibleMasterAddress` (R)	The address of the Visible Master.
`ptpVisibleMasterPriority1` (R)	The PTP priority1 of the Visible Master.
`ptpVisibleMasterPriority2` (R)	The PTP priority2 of the Visible Master

Table 7-6 PTP Visible Master Table

ptpVisibleMasterEntryIndex	ptpVisibleMasterAddressType	ptpVisibleMasterAddress	ptpVisibleMasterPriority1	ptpVisibleMasterPriority2
1	ipv4(1)	10.206.159.97	128	128
2	ipv4(1)	10.206.159.98	128	129

TCM Configuration Example

8.1 Example TCM Configuration

The below subsections contain listings of example TCM configuration files. The listings are meant only to illustrate usage of various TCM features. The configurations are not intended to meet any specific real-world operational needs.

The example configurations may explicitly configure some items to their default values. Although this is an unnecessary configuration step, the intent is to make apparent the availability of configurable options for the features being illustrated. On the other hand, features not being illustrated are left unconfigured if their default values are consistent with the intended configuration.

The configuration is shown for only one member of a Protection Partner Pair. The configuration of the other member should be the same except that the Local and Partner addresses are flipped. Refer to *cgmProtectionLocalAddress on page 122* and *cgmProtectionPartnerAddress on page 168*.

The proper operation of the TCM depends not only on the associated configuration file contents, but also areas such as cabling, activation of external producers and consumers of clocks, activation of networking interfaces and ports, configuration of VLANs, as well as other areas.

8.1.1 BITS/SSU Example Configuration

```
#####################################
#####################################
#  Configuration Summary
#
#  TDM System Clocks from single BITS/SSU Reference
#
#  TDM Reference Clock (frequency): Source
# -BITS1 Rx (2.048 MHz): BITS port #1
#
#  TDM System Clock (frequency): Source
#  - CLK1 ( 8    kHz): T0 (Protection Partner
#    Master/Slave Sync Clock)
#  - CLK2 (19.44 MHz): T0
#####################################
#####################################
```

```
##################################
#  TDM Reference Clocks setup  #
##################################
clkT0Priority clkBITS1Rx  2
################################
#  Protection Partner setup  #
################################
# Flip addresses for configuration of other Protection Partner
cgmProtectionLocalAddress   192.168.25.1
cgmProtectionPartnerAddress 192.168.25.2
#####################################################
#  Subsystems activation   #
#  (BITS/PTP/SyncE Generation/AMC/Extension Shelf)  #
#####################################################
# BITS/SSU Line Interface Units in E1/SDH mode
cgmBITSEnable enable
cgmInterfaceMode sdhE1
##############################
#  Operational Enviroment  #
##############################
# Optional TCM Watchdog
TCMWatchdogEnable enable

# Automatically determine ATCA Domain
autoATCAClkDomainEnable enable

# Boot ToPSync at startup
cgmEnable enable

# Use CLK1 for Protection Partner
# Master/Slave Sync clock so
# phase alignment can be maintained.
# Defaults to 8 kHz.
clkSrc clkMSSyncIn clkCLK1
################
#  Messaging  #
################
# Log all levels of Events
```

```
cgmSysEventLogLevel all

# Generate SNMP traps from Event-level Events
cgmSysEventTrapLevel event
```

8.1.2 Pass Thru Example Configuration

```
#####################################
#####################################
#  Configuration Summary
#
#  Distribute ATCA Backplane Clocks
#  from local Line Cards to
#  remote Extension Shelf
#
#  TDM Reference Clock (frequency): Source
#  - [none]
#
#  TDM System Clock (frequency): Source
#  - [none]
#
#  Extension Shelf Clock Distribution
#  - ATCA CLK1 (local Domain) to
#    Extension Shelf Connector #1 Port 1
#  - ATCA CLK2 (local Domain) to
#    Extension Shelf Connector #1 Port 3
#  - ATCA CLK3A to
#    Extension Shelf Connector #1 Port 4
#  - ATCA CLK3B to
#    Extension Shelf Connector #1 Port 2
#####################################
#####################################

#####################
#  Pass Thru setup  #
#####################
cgmMasterMode passThru
cgmClockCount fourClks
```

```
##############################
#  Protection Partner setup  #
##############################
# Flip addresses for configuration of other Protection Partner
cgmProtectionLocalAddress   192.168.25.1
cgmProtectionPartnerAddress 192.168.25.2
#####################################################
#  Subsystems activation    #
#  (BITS/PTP/SyncE Generation/AMC/Extension Shelf)  #
#####################################################
# Extension Shelf bit-field to activate port #1
cgmExtShelves 00001
############################
#  Operational Enviroment  #
############################
# Optional TCM Watchdog
TCMWatchdogEnable enable
# Automatically determine ATCA Domain
autoATCAClkDomainEnable enable
# Boot ToPSync at startup
cgmEnable enable
# Use CLK1 for Protection Partner
# Master/Slave Sync clock so
# phase alignment can be maintained
# Defaults to 8 kHz
clkSrc clkMSSyncIn clkCLK1
###############
#  Messaging  #
###############
# Log all levels of Events
cgmSysEventLogLevel all
# Generate SNMP traps from Event-level Events
cgmSysEventTrapLevel event
```

8.1.3 PTP Boundary Clock Example Configuration

```
#####################################
#####################################
#  Configuration Summary
#
#  PTP Boundary Clock + TDM System Clocks
#
#  PTP Acceptable Master(s)
#  - 10.206.159.89
#  - 10.206.159.90
#
#  PTP Acceptable Slave(s)
#  - 10.206.159.98
#
#  TDM Reference Clock (frequency): Source
#  -   RefA (8 kHz): CLK3 (Domain A)
#  -  RefB (8 kHz): CLK3 (Domain B)
#  - SyncE RefA (8 kHz): Front Panel
#     Ethernet Port #3 (10G connection)
#
#  TDM System Clock (frequency): Source
#  - CLK1 ( 8    kHz): T0 (Protection Partner
#     Master/Slave Sync Clock)
#  - CLK2 (19.44 MHz): T0
#  - CLK3 ( 8    kHz): PTP
#####################################
#####################################
################################
#  ToPSync Ethernet interface  #
################################
cgmTSIPAddress 10.206.159.97
cgmTSNetMask   255.255.0.0
cgmTSDfltGW    10.206.159.254
################################
#  TDM Reference Clocks setup  #
################################
```

```
clkSrc clkRefA clkCLK3
clkFreq clkRefA f8k
clkSrc clkRefB clkCLK3
clkFreq clkRefB f8k
# 10G connection means 161.13 MHz recovered clock.
# (automatically scaled to 8 kHz for clkSyncERefA)
clkSrc clkSyncERcvdRefA clkEthP3
clkFreq clkSyncERcvdRefA f161M13
clkT0Priority clkSyncERefA  2
clkT0Priority clkRefA       3
clkT0Priority clkRefB       4
##############################
#  Protection Partner setup  #
##############################
# Flip addresses for configuration of other Protection Partner
cgmProtectionLocalAddress   192.168.25.1
cgmProtectionPartnerAddress 192.168.25.2
#############################
#  Acceptable PTP Partners  #
#############################
# Masters port 1
ptpAcceptableMasterAddress 1 10.206.159.89
ptpAcceptableMasterAddress 1 10.206.159.90
# Slaves for port 2
ptpAcceptableSlaveAddress  2 10.206.159.98
#####################################################
#  Subsystems activation                            #
#  (BITS/PTP/SyncE Generation/AMC/Extension Shelf)  #
#####################################################
# PTP Clocks
ptpEnable enable
############################################
#  Standard PTP routing and distribution   #
############################################
# Use the Local Generation Master mode as the general
# clock distribution scheme.
```

```
cgmMasterMode localClkGen
# Use the Three Clocks Clock Count mode to access
# all three ATCA clocks.
cgmClockCount threeClks
# Use our CLK3 System Clock also as a Reference Clock.
cgmRefLoopbackEnable enable
# Source both Reference Clocks RefA and RefB from CLK3.
# One reference will be from the local ATCA Domain and
# the other from the remote.
clkSrc clkRefA clkCLK3
clkSrc clkRefB clkCLK3
# Use the PTP TCM Clock (the source for OPCK0)
# as the source for the Usr1 System Clock.
# Usr1 is the source for CLK3 when in
# Local Generation Mode + Three Clocks Mode.
clkSrc clkUsr1 clkOPCLK0
############################
#  Operational Enviroment  #
############################
# Optional TCM Watchdog
TCMWatchdogEnable enable
# Automatically determine ATCA Domain
autoATCAClkDomainEnable enable
# Boot ToPSync at startup
cgmEnable enable
# Use CLK1 for Protection Partner
# Master/Slave Sync clock so
# phase alignment can be maintained.
# Defaults to 8 kHz.
clkSrc clkMSSyncIn clkCLK1
##########################
#  PTP Operational Mode  #
##########################
# Operate as a PTP Boundary Clock
ptpBoundaryClockEnable enable
# Boundary Clock Master port is
```

```
# PTP default Domain (== zero).
ptpBoundaryMasterDomain default
# Boundary Clock Slave port is
# PTP alt1 Domain (== 1).
ptpClockDefaultDSDomain alt1
###############
#  Messaging  #
###############
# Log all levels of Events
cgmSysEventLogLevel all
# Generate SNMP traps from Event-level Events
cgmSysEventTrapLevel event
```

Related Documentation

A.1 SMART Embedded Computing Documentation

The publications listed below are referenced in this manual. You can obtain electronic copies of SMART EC publications by contacting your local SMART EC sales office. For released products, you can also visit our web site for the latest copies of our product documentation.

Go to www.artesyn.com/computing/support/product/technical-documentation.php.

Under **FILTER OPTIONS**, click the Document types drop-down list box to select the type of document you are looking for.

In the **Search** text box, type the product name and click **GO**.

Table A-1 SMART Embedded Computing Publications

Document Title	Publication Number
BBS on ATCA-F125 with SRstackware Programmer's Reference	6806800L83
SRstackware Intelligent Network Software Protocol Demo Guide	6806800L35
SRstackware Intelligent Network Software VRRP Command Reference	6806800L36
SRstackware Intelligent Network Software RIP Command Reference	6806800L37
SRstackware Intelligent Network Software Layer 2 Configuration Guide	6806800L39
SRstackware Intelligent Network Software OSPF Command Reference	6806800L40
SRstackware Intelligent Network Software Layer 2 Command Reference	6806800L41
SRstackware Intelligent Network Software Unicast Configuration Guide	6806800L42
SRstackware Intelligent Network Software Troubleshooting Guide	6806800L34
SRstackware FAQ	6806800L44
SRstackware Intelligent Network Software Layer3 Command	6806800L87
SRstackware Intelligent Network Software Switch Configuration Command Reference	6806800L88
SRstackware Application Programming Interface Developer Guide	6806800L43