19-4750; Rev 1; 7/11

DS34S132
32-Port TDM-over-Packet IC

General Description

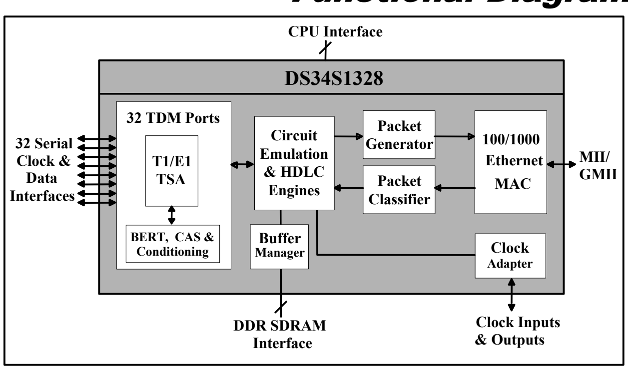
The IETF PWE3 SAToP/CESoPSN/HDLC-compliant DS34S132 provides the interworking functions that are required for translating TDM data streams into and out of TDM-over-Packet (TDMoP) data streams for L2TPv3/IP, UDP/IP, MPLS (MFA-8), and Metro Ethernet (MEF-8) networks while meeting the jitter and wander timing performance that is required by the public network (ITU G.823, G.824, and G.8261). Up to 32 TDM ports can be translated into as many as 256 individually configurable pseudowires (PWs) for transmission over a 100/1000Mbps Ethernet port. Each TDM port's bit rate can vary from 64Kbps to 2.048Mbps to support T1/E1 or slower TDM rates. PW interworking for TDM-based serial HDLC data is also supported. A built-in time-slot assignment (TSA) circuit provides the ability to combine any group of time slots (TS) from a single TDM port into a single PW. The high level of integration provides the perfect solution for high-density applications to minimize cost, board space, and time to market.

Applications

TDM Circuit Emulation Over PSN
- TDM Leased-Line Services Over PSN
- TDM Over BPON/GPON/EPON
- TDM Over Cable
- TDM Over Wireless
- Cellular Backhaul
- Multiservice Over Unified PSN

HDLC-Encapsulated Data Over PSN

Functional Diagram



Features

- **32 Independent TDM Ports with Serial Data, Clock, and Sync (Data = 64Kbps to 2.048Mbps)**
- **One 100/1000Mbps (MII/GMII) Ethernet MAC**
- **256 Total PWs, 32 PW per TDM Port, with Any Combination of TDMoP and/or HDLC PWs**
- **PSN Protocols: L2TPv3 or UDP Over IP (IPv4 or IPv6), Metro Ethernet (MEF-8), or MPLS (MFA-8)**
- **0, 1, or 2 VLAN Tags (IEEE 802.1Q)**
- **Synchronous or Asynchronous TDM Port Timing**
  - **One Clock Recovery Engine per TDM Port with One Assignable as a Global Reference**
  - **Supported Clock Recovery Techniques**
    - **Adaptive Clock Recovery**
    - **Differential Clock Recovery**
    - **Absolute and Differential Timestamps**
  - **Independent Receive and Transmit Interfaces**
  - **Two Clock Inputs for Direct Transmit Timing**
- **For Structured T1/E1, Each TDM Port Includes**
  - **DS0 TSA Block for any Time Slot to Any PW**
  - **32 HDLC/CES Engines (256 Total)**
  - **With or Without CAS Signaling**
- **For Unstructured, each TDM Port Includes**
  - **One HDLC/SAT Engine (32 Total)**
  - **Any data rate from 64Kbps to 2.048Mbps**
- **32-Bit or 16-Bit CPU Processor Bus**
- **CPU-Based OAM and Signaling**

UDP-specific	**"Special" Ethernet Type**
Inband VCCV	**ARP**
MEF OAM	**NDP/IPv6**
Broadcast DA	

- **DDR SDRAM Interface**
- **Low-Power 1.8V Core, 3.3V I/O, 2.5V SDRAM**

Ordering Information

PART	PORTS	TEMP RANGE	PIN-PACKAGE
DS34S132GNA2	32	-40°C to +85°C	676 BGA
DS34S132GNA2+	32	-40°C to +85°C	676 BGA

+Denotes a lead(Pb)-free/RoHS-compliant package.

Some revisions of this device may incorporate deviations from published specifications known as errata. Multiple revisions of any device may be simultaneously available through various sales channels. For information about device errata, go to: www.maxim-ic.com/errata. For pricing, delivery, and ordering information, please contact Maxim Direct at 1-888-629-4642, or visit Maxim's website at www.maxim-ic.com.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

The public network is in transition from a TDM Switched Network to a Packet Switched Network. A number of Pseudowire (PW) packet protocols have been standardized to enable legacy TDM services (e.g. TDM voice, TDM Leased-line and HDLC encapsulated data) to be transported and switched/routed over a single, unified PSN. The legacy service is encapsulated into a PW protocol and then transported or tunneled through the unified PSN. The PW protocols provide the addressing mechanisms that enable a PSN to switch/route the service without understanding or directly regarding the specific characteristics of the services (e.g. the PSN does not have to directly understand the timing requirements of a TDM voice service). The PW protocols have been developed for use over PSNs that utilize the L2TPv3/IP, UDP/IP, MPLS (MFA-8) or Metro Ethernet (MEF-8) protocols.

PW protocols that are used for TDM services can be categorized as TDM-over-Packet (TDMoP) PW protocols. The TDMoP protocols support all of the aspects of the TDM services (data, timing, signaling and OAM). This enables Public (WAN) and Enterprise (LAN) networks to migrate to next generation PSNs and continue supporting legacy voice and leased-line services without replacing the legacy termination equipment.

Legacy TDM services depend on constant bit rate data streams with highly accurate frequency, jitter and wander timing requirements that up until recently have not been well supported by most packet switching equipment. For public network applications the timing recovery mechanisms must achieve the jitter and wander performance that is required by the ITU-T G.823/824/8261 standards. To accomplish this, a TDMoP terminating device must incorporate innovative and complex mechanisms to recovery the TDM timing from a stream of packets.

Legacy TDM services also have numerous special features that include voice signaling and OAM systems that have been developed over many years through a long list of standardization literature to provide carrier-grade reliability and maintainability. The list of legacy functions and features is so long that today's VoIP equipment only supports a subset of what is used in the legacy TDM network. This, in part, has slowed the transition from a TDM to Packet-based network. With TDMoP technology all features and services can be supported.

The TDMoP technology is similar to VoIP technology in that both provide a means of communicating a time oriented service (e.g. voice) over a non-time oriented, packet network. TDMoP technology can be added incrementally to the network (as needed) to supplement VoIP technology to provide an alternative solution when VoIP price/performance is not optimal (e.g. where the number of supported lines does not warrant the infrastructure required of a VoIP network) and where some function/features are not supported by the VoIP protocols.

The Legacy PSTN network also supports HDLC encapsulated data that is transported over TDM lines. PWs can also be used to transport HDLC data. This form of PW could also be categorized as a TDM service since the legacy service is carried over TDM lines. However, the fundamental aspects of an HDLC service do not depend as much on TDM timing and the nature of the data can be described as "packetized" as with Ethernet, Frame Relay and ATM services. For clarity the HDLC service is categorized as "HDLC over PW". One example Legacy HDLC service is SS7 Signaling which is used to communicate voice signaling information from one TDM switch to another.

2 ACRONYMS AND GLOSSARY

- # – Number
- ACR – Adaptive Clock Recovery
- AT – Absolute Timestamps
- ATM – Asynchronous Transfer Mode
- BERT – Bit Error Rate Test
- BGA – Ball Grid Array
- BITS - Building Integrated Timing System
- Bundle – a PW with an ID that is recognized by the DS34S132
- BW – Bandwidth
- CR – Clock Recovery
- CAS – Channel Associated Signaling
- CCS – Common Channel Signaling
- CES – abbreviation for CESoPSN
- CESoPSN – Circuit Emulation Service over PSN
- CLAD – Clock Rate Adapter
- CRE – Clock Recovery Engine
- DA – Destination Address
- DCR – Differential Clock Recovery
- DCR-DT – DCR with Differential Timestamps
- DDR – Double Data Rate
- Decap –De-encapsulate
- DS0 – 64 Kb/s Timeslot within a T1 or E1 signal
- DS1 – 1.544 Mb/s TDM data stream
- E1 – 2.048 Mb/s TDM data stream
- Encap –Encapsulate
- EPON – Ethernet PON (IEEE 802.3ah)
- FCS – Frame Check Sequence
- GMII – Gigabit MII (IEEE 802.3)
- GPON – Gigabit PON (ITU-T G.984)
- GPS - Global Positioning System
- HDLC – High-level Data Link Control
- IEEE – Institute of Electrical & Electronic Engineers
- IETF – Internet Engineering Task Force
- IP – Internet Protocol
- ISDN – Integrated Services Digital Network
- ITU – International Telecommunication Union
- JB – Jitter Buffer
- L2TPv3 – Layer 2 Tunneling Protocol Version 3
- LAN – Local Area Network
- MAC – Media Access Control
- MAN – Metropolitan Area Network
- MEF – Metro Ethernet Forum
- MFA – MPLS/Frame Relay Alliance (Now called IP/MPLS Forum)
- MII – Medium Independent Interface (IEEE 802.3)
- MPLS – Multi-Protocol Label Switching
- OAM – Operations, Administration & Maintenance
- OCXO – Oven Controlled Crystal Oscillator
- OLT – Optical Line Termination
- ONU – Optical Network Unit
- PBX – Private Branch Exchange
- PDV – Packet Delay Variation
- PDVT – PDV Tolerance
- PON – Passive Optical Network
- PRBS – Pseudo-Random Bit Sequence
- PSN – Packet Switched Network
- PSTN – Public Switched Telephone Network
- PWE3 – Pseudo-Wire Edge-to-Edge Emulation
- PW – Pseudo Wire
- QoS – Quality of Service
- QRBS – Quasi-Random Bit Sequence
- RAM – Random Access Memory
- Rcv – Receive
- RXP – Receive Packet direction “from Ethernet Port to TDM Port”
- SAT – abbreviation for SAToP
- SAToP – Structure-Agnostic TDM over Packet
- SDH – Synchronous Digital Hierarchy
- SDRAM – Synchronous Dynamic RAM
- SN – Sequence Number
- SONET –Synchronous Optical Network
- SS7 – Signaling System 7
- T1 – commonly used term for DS1
- T1-ESF – T1 Extended Super-frame
- T1-SF – T1 Super-frame
- T1/E1 – T1 or E1
- TCXO – Temperature Compensated Crystal Oscillator
- TDM – Time Division Multiplexing
- TDMoIP – TDM over IP
- TDMoP – TDM over Packet
- Timeslot – 64 Kb/s channel within an E1 or T1
- TS – Timeslot
- TXP – Transmit Packet direction “from TDM Port to Ethernet Port”
- UDP – User Datagram Protocol
- VCCV – Virtual Circuit Connectivity Verification
- VoIP – Voice over IP
- WAN – Wide Area Network
- Xmt - Transmit

3 APPLICABLE STANDARDS

Table 3-1. Applicable Standards

SPECIFICATION	SPECIFICATION TITLE
ANSI	
T1.102	*Digital Hierarchy—Electrical Interfaces*, 1993
T1.107	*Digital Hierarchy—Formats Specification*, 1995
T1.403	*Network and Customer Installation Interfaces—DS1 Electrical Interface*, 1999
ETSI	
ETS 300 011	*ISDN Primary Rate User Network Interface (UNI); Part 1: Layer 1 Spec.* V1.2.2 (2000-05)
IEEE	
IEEE 802.1Q	*Virtual Bridged Local Area Networks (2003)*
IEEE 802.3	*Carrier Sense Multiple Access with Collision Detection Access Method and Physical Layer Spec.* (2005)
IEEE 1149.1	*Standard Test Access Port and Boundary-Scan Architecture*, 1990
IETF	
RFC 4553	*Structure-Agnostic Time Division Multiplexing (TDM) over Packet (SAToP)* (06/2006)
RFC 4618	*Encapsulation Methods for Transport of PPP/High-Level Data Link Control (HDLC) over MPLS Networks* (09/2006)
RFC 5086	*Structure-Aware Time Division Multiplexed (TDM) Circuit Emulation Service over Packet Switched Network (CESoPSN)* (12/2007)
RFC 5087	*Time Division Multiplexing over IP (TDMoIP)* (12/2007)
ITU-T	
G.704	*Synchronous Frame Structures at 1544, 6312, 2048, 8448 and 44736 kbit/s Levels* (10/1998)
G.732	*Characteristics of Primary PCM Multiplex Equipment Operating at 2048Kbit/s* (11/1988)
G.736	*Characteristics of Synchronous Digital Multiplex Equipment Operating at 2048Kbit/s (03/1993)*
G.823	*The Control of Jitter and Wander in Digital Networks Based on 2048kbps Hierarchy* (03/2000)
G.824	*The Control of Jitter and Wander in Digital Networks Based on 1544kbps Hierarchy* (03/2000)
G.8261/Y.1361	*Timing and Synchronization Aspects in Packet Networks* (05/2006)
G.8261/Y.1361	*Timing and Synchronization Aspects in Packet Networks* (12/2006). Corrigendum 1.
I.431	*Primary Rate User-Network Interface - Layer 1 Specification* (03/1993)
O.151	*Error Performance Measuring Equipment Operating at the Primary Rate and Above* (1992)
Y.1413	*TDM-MPLS Network Interworking – User Plane Interworking* (03/2004)
Y.1413	*TDM-MPLS Network Interworking – User Plane Interworking* (10/2005). Corrigendum 1.
Y.1414	*Voice Services–MPLS Network Interworking* (07/2004)
Y.1453	*TDM-IP Interworking – User Plane Networking* (03/2006)
MEF	
MEF 8	*Implementation Agree. for Emulation of PDH Circuits over Metro Ethernet Networks* (10/2004)
MFA	
MFA 8.0.0	*Emulation of TDM Circuits over MPLS Using Raw Encapsulation – Implement. Agree.* (11/2004)

Note: Only those sections of these standards that are affected by the DS34S132 functions are considered applicable. For example, several of the standards specify T1/E1 Framer/LIU functions (e.g. pulse shape) that are not included in the DS34S132 but also specify jitter/wander functions that are applicable.

4 HIGH LEVEL DESCRIPTION

To implement a PW (tunnel) across a PSN requires a PW termination point at each end of the PW (tunnel). Each terminating point provides the PW encapsulation functions that are required to enter the PSN (for one direction of data) and the PW de-encapsulation functions to restore the data to its original (non-PW) format (for the opposite direction). The two data directions at each termination point can be can be described as the "transmit PW packet direction" (TXP) and the "receive PW packet direction" (RXP).

The DS34S132 TDMoP device implements the complete, bi-directional PW termination point encapsulation functions for TDMoP and HDLC PWs. The DS34S132 is a high density solution that can terminate up to 256 PWs that are associated with up to 32 T1/E1 data streams and aggregate that traffic for transmission over a single 100/1000 Mb/s Ethernet data stream. The DS34S132 can encap/decap TDMoP and HDLC PWs into the following PSN protocols: L2TPv3/IPv4, L2TPv3/IPv6, UDP/IPv4, UDP/IPv6, Metro Ethernet (MEF-8) and MPLS (MFA-8).

For TDMoP PWs the DS34S132 supports the SAToP and CESoPSN payload formats. SAToP is used for Unstructured TDM transport, where an entire T1/E1 including the framing pattern (if it exists) is transferred transparently as a series of unformatted bytes of data in the PW payload without regard to any bit, byte and/or frame alignment that may exist in the TDM data stream. The DS34S132 can support Unstructured T1, E1 or slower TDM data streams (any bit rate less than or equal to 2.048 Mb/s).

CESoPSN is used for Structured TDM transport where the PW packet payload is synchronized to the T1/E1 framing. With CESoPSN the T1/E1 framing pattern is commonly not passed across the PW (removed) because the structured PW format enables the framing information to be conveyed through the PW mechanisms. The opposite end generates the T1/E1 framing pattern from the PWs payload structure. This payload format can be used when the TDM service (e.g. voice) requires the ability to interpret, and/or terminate some functional aspects of the T1/E1 signal (e.g. identify DS0s within the T1/E1). PWs with the Structured payload format can support Nx64 Kb/s, fractional T1/E1 (T1: N = 1 – 24; E1: N = 1 – 32). In some applications, a T1/E1 can be divided into multiple Nx64 blocks (M x N x 64; M = the number of fractional blocks) and the PSN can be used as a "distributed cross-connect" to implement a point to multi-point topology forwarding some Nx64 blocks to one end point and other Nx64 blocks to other end points (T1: M = 1 – 24; E1: M = 1 – 32; e.g. for E1: 32 x 1 x 64).

The CESoPSN Structured format can also convey CAS Signaling across a PW through the use of a sub-channel within the CESoPSN PW packets. The DS34S132 enables the CAS Signaling to be transparently passed, monitored by an external CPU, and/or terminated by an external CPU, all on a per Timeslot and per direction basis.

The DS34S132 allows each TDM Port to independently support asynchronous or synchronous TDM data streams. Each TDM Port has a Clock Recovery Engine to regenerate the timing from a TDMoP PW packet data stream. For applications that do not require clock recovery the DS34S132 also provides several external clocking options.

The Clock Recovery Engines support Differential Clock Recovery (DCR) and Adaptive Clock Recovery (ACR). DCR can be used when a common clock is available at both ends of the PW (e.g. BITS clock for the public network or GPS for the mobile cellular network) and requires that the PW use RTP Timestamps to convey the TDM timing information. Adaptive Clock Recovery does not use Timestamps but instead regenerates the timing based on the TDMoP PW packet transmission rate. The DS34S132 high performance clock recovery circuits enable the use of PWs in the public network by achieving the stringent jitter and wander performance requirements of ITU-T G.823/824/8261, even for networks that impose large packet delay variation (PDV) and packet loss. For far end clock recovery, the DS34S132 can generate two Timestamp formats - Absolute and Differential Timestamps.

PWs can be used to transport HDLC packet data. The DS34S132 can forward HDLC encapsulated data transparently using a TDMoP PW (as described above; idle HDLC Flags are forwarded with the data) or by first extracting the data from the HDLC coding and then only forwarding the non-idle data in an HDLC PW. The HDLC PW is useful for HDLC data streams where a significant portion of the data stream is filled with HDLC Idle Flags. For example, if a 64 Kb/s TDM Timeslot is used to carry 4 Kb/s of HDLC data then it may be more bandwidth efficient to extract the payload data from the HDLC encoding and forward the data over an HDLC PW. The DS34S132 incorporates 256 HDLC Engines so that any PW can be assigned as a TDMoP PW or an HDLC PW.

PW Termination points often must also terminate OAM and Signaling packet data streams. To support this need the DS34S132 enables an external CPU to terminate several OAM and Signaling types including: PW In-band VCCV OAM, PW UDP-specific (Out-band VCCV) OAM, MEF OAM, Ethernet Broadcast frames, ARP, IPv6 NDP and includes a user specified CPU-destination Ethernet Type. The DS34S132 can also be programmed to forward packets to the CPU that match specialized conditions for debug or other purposes (e.g. wrong IP DA).

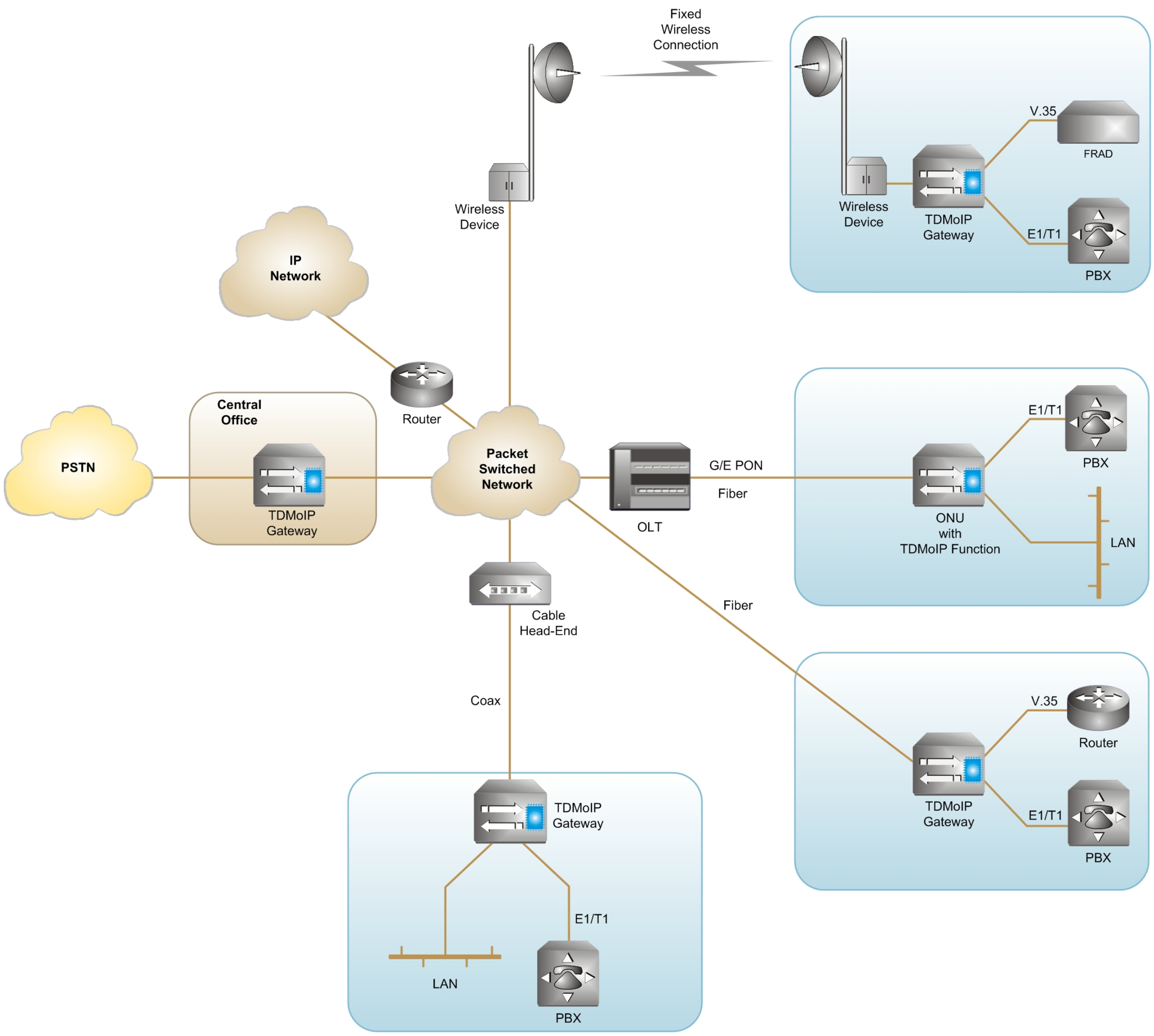
The DS34S132 uses an external DDR SDRAM device to buffer data. The large memory supplies sufficient buffer space to support a 256 ms PDV for each of the 256 PW/Bundles and to enable packet re-ordering for packets that are received out of order (the PSN may mis-order the packets). This large memory is also used to buffer the HDLC data streams and the CPU terminated OAM and Signaling packets.

TDMoP provides the perfect transition technology for next generation packet networks enabling the continued use of the vast Legacy network and at the same time supplementing new packet based technologies.

5 APPLICATION EXAMPLES

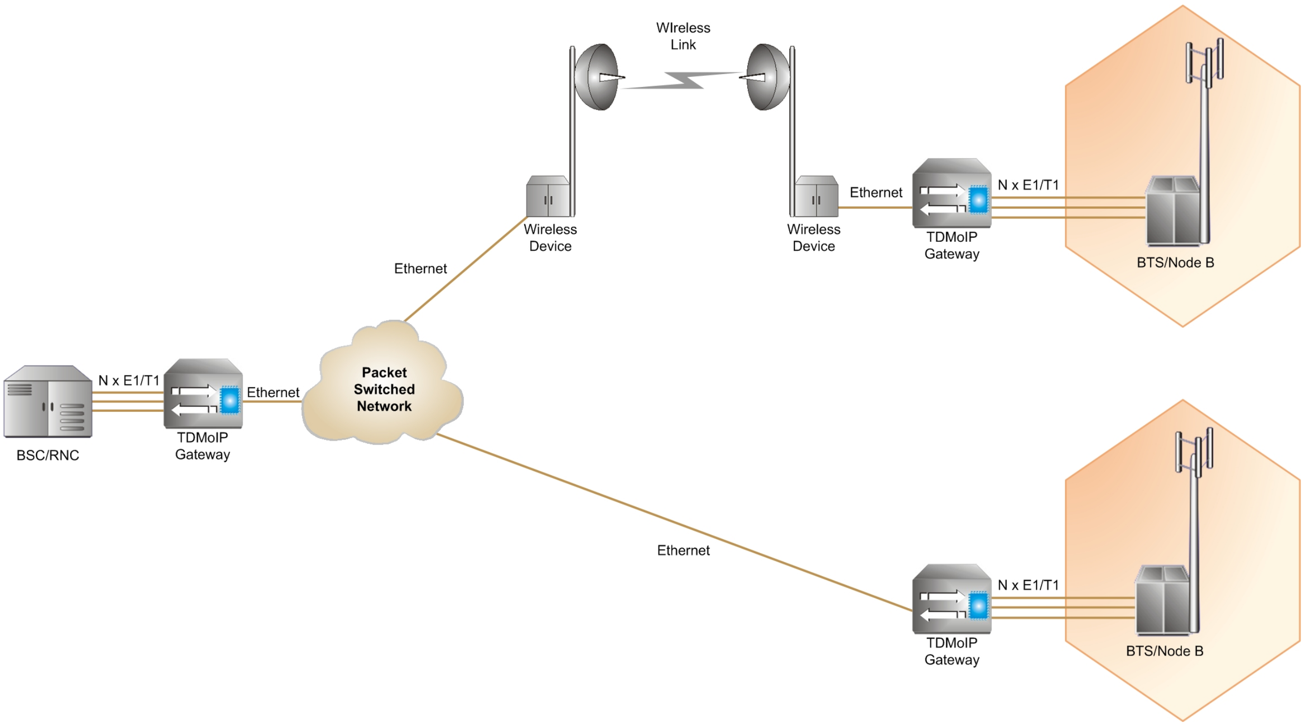
In Figure 5-1, TDMoP devices are used in gateway nodes to transport TDM services through a metropolitan PSN. The Maxim TDMoP family of devices offers a range of density solutions so that lower density solutions like the DS34T101 can be used in Service Provider Edge applications, to support a small number of T1/E1 lines, and higher density solutions like the DS34S132 can be used in Central Office applications, to terminate several Service Provider Edge nodes. PWs can be carried over fiber, wireless, SONET/SDH, G/EPON, coax, etc.

Figure 5-1. TDMoP in a Metropolitan Packet Switched Network



In Figure 5-2, DS34S132 devices are used in TDMoP gateways to enable TDM services to be transported through a Cellular Backhaul PSN.

Figure 5-2. TDMoP in Cellular Backhaul



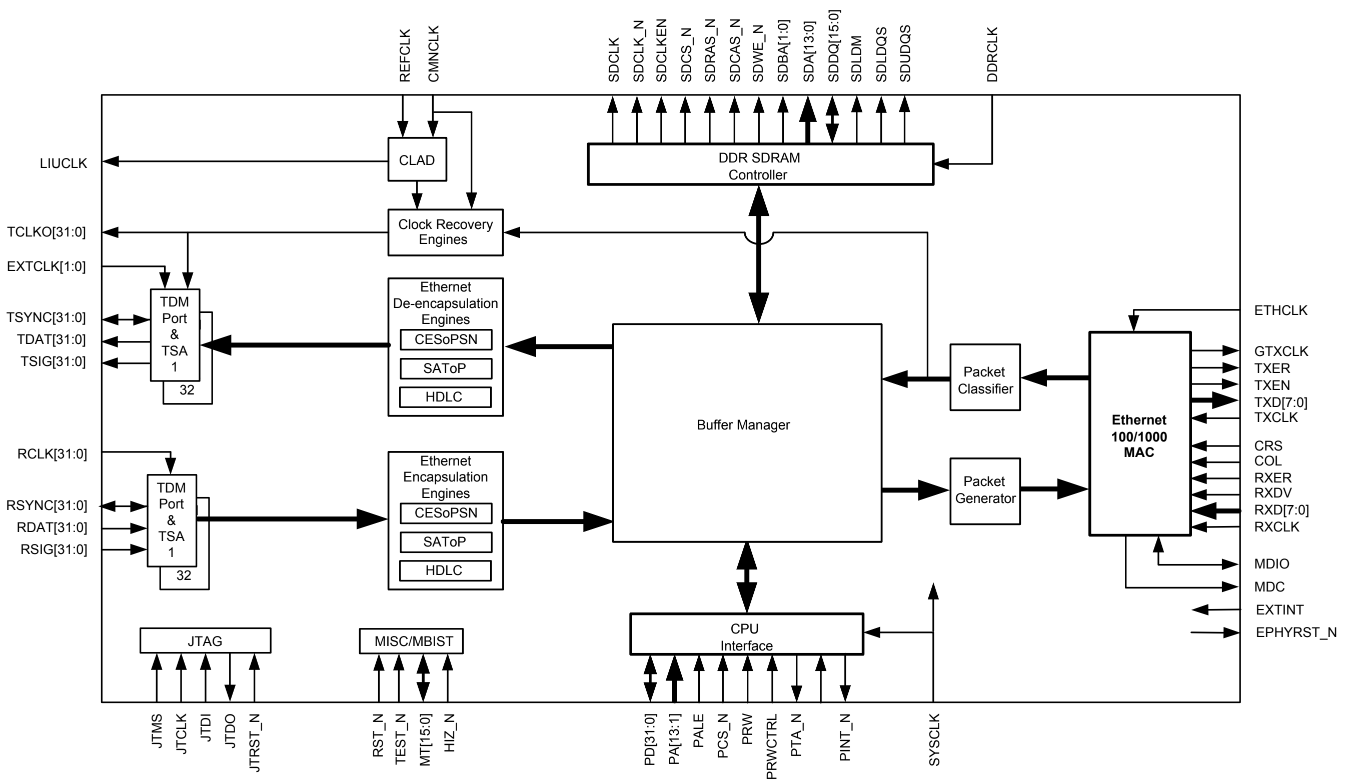
Other Possible Applications

Using a Packet Backplane for Multiservice Concentrators

Communications platforms with all/any of the above-mentioned capabilities can replace obsolete, low bandwidth TDM buses with low cost, high bandwidth Ethernet buses. The DS34S132 provides the interworking functions that are needed to packetize TDM services so that they can be multiplexed together with bursty services for transmission over a unified backplane bus. This enables a cost-effective, future-proof design with full support for both legacy and next-generation services.

6 BLOCK DIAGRAM

Figure 6-1. DS34S132 Functional Block Diagram



7 FEATURES

TDM Port Features

- TDM Ports
  - 32 TDM Ports, each with independently configured Framing Format
    - T1/E1 Structured (with T1/E1 Framing)
      - T1-SF, T1-ESF and E1 CAS Multi-frame formats
      - With and Without CAS Signaling
        CAS embedded in data bus using RDAT/TDAT pins
        Parallel CAS Interface using RSIG/TSIG pins
    - Unstructured (without Framing) - T1, E1 and slower TDM line rates (any line rate ≤ 2.048 Mb/s)
- TDM Port Timing References
  - TDM Port Clocks
    - Asynchronous or Synchronous TDM Port Timing
    - Independent Receive and Transmit Clocks
    - Transmit TDM Port Timing
      - RXP packet stream Clock Recovery
        One Clock Recovery Engine per TDM Port
        Global Clock Recovery Engine
      - EXTCLK0 or EXTCLK1 External clock reference
      - External RCLK signal (Loop timed)
    - Receive TDM Port Timing
      - External RCLK signal
      - Internally generated Transmit timing (for synchronous systems)
  - TDM Multi-frame Synchronization for CAS Signaling
    - Independent Receive and Transmit Multi-frame Synchronization for each TDM Port
    - E1, T1-SF and T1-ESF Multi-frame Synchronization
    - External input or internally generated Multi-frame synchronization
- TDM Port Clock Recovery Engines
  - Adaptive Clock Recovery or
  - Differential Clock Recovery
    - Common Clock (CMNCLK) frequency = 1MHz to 25MHz (in 8kHz increments)
    - RTP Differential Timestamp
  - Generation of Absolute Timestamps and Differential Timestamps
  - External 5.0 MHz – 155.52 MHz clock input (REFCLK) for internal Clock Recovery synthesizer
  - Fast Frequency Acquisition and Highly Accurate Phase Tracking
  - Recovered Clock Jitter and Wander per ITU-T G.823/G.824/G.8261 with Stratum 3 clock reference
  - High resilience to Packet Loss and Robust to Sudden Significant Constant Delay Changes
  - Automatic transition to hold-over during alarm/event impairments
- TDM Port Timeslot Assignment (TSA), CAS and Conditioning
  - Nx64 Kb/s – any combination of T1/E1 Timeslots from one TDM Port can be assigned to a PW/Bundle
  - T1/E1 CAS Signaling (Channel Associated Signaling)
    - Transparent CAS (forwarded from TDM to Ethernet Port and from Ethernet to TDM Port)
    - Per Timeslot CPU Controlled CAS (CPU inserts CAS; in TXP and/or RXP directions)
    - CAS Status and Change of Status for CPU Monitoring (in RXP and TXP directions)
  - Data Conditioning – can force any 8-bit pattern on any number of Timeslots (in RXP and TXP directions)

Ethernet Port Features

- Ethernet MAC Interface
  - 100/1000 Mb/s Operation using MII/GMII Interface
  - 2 programmable receive Ethernet Destination Addresses
  - Mixed Ethernet II (DIX) and IEEE 802.2 LLC/SNAP formats
  - Mixed data streams with 0, 1, or 2 VLAN Tags
    - Programmable VLAN TPID
  - Ethernet Frame Length 64 bytes to 2000 bytes

PW/Bundle Features

- RXP PW/Bundle Header
  - Up to 256 programmed PW/Bundles (32 per TDM Port)
  - PW Header Types
    - L2TPv3 / IPv6
    - L2TPv3 / IPv4
    - UDP / IPv4
    - UDP / IPv6
    - MEF (MEF-8)
    - MPLS (MFA-8)
  - Mixed MPLS data streams with 0, 1 or 2 MPLS Outer Labels
  - Mixed L2TPv3 data streams with 0, 1, or 2 L2TPv3 Cookies
  - Flexible UDP settings
    - 16-bit (standard) or 32-bit (extended) UDP PW-ID bit width
    - 16-bit UDP PW-ID selectable to be verified against UDP Source or Destination Port
    - Optional 16-bit PW-ID Mask
    - Ignore UDP Payload Protocol or verify against 2 programmable UDP Payload Protocol Values
  - Optional PW Control Word
    - Optional “In-band VCCV” Monitoring
      - Programmable 16-bit In-band VCCV value with programmable 16-bit In-band VCCV mask
  - Optional RTP Header
    - One PW/Bundle per TDM Port can be assigned to provide RTP Timestamp for Clock Recovery
  - Sequence Number
    - Selectable between Control Word or RTP Sequence Number
    - Used to initiate conditioning data when packets are missing
    - Optional re-ordering of mis-ordered packets up to the Size of the Jitter Buffer depth
  - Up to 32 UDP-Specific (Out-band VCCV) OAM PW-IDs
  - Debug settings to forward PW/Bundles with special conditions to CPU for analysis (e.g. wrong IP DA)
- TXP PW/Bundle Header
  - Store up to 256 CPU generated PW/Bundle Headers (one per PW/Bundle)
    - Maximum 122 byte header with any CPU-specified content (Layer 2/3/4 content)
      - Auto generate and insert Length and FCS functions for IP and UDP Headers
      - Optional RTP Timestamp Insertion
        Any number of TXP PW/Bundles can be assigned to include Timestamp in RTP Header
      - Optional RTP and Control Word Sequence Number Insertion
        3 HDLC Sequence Number generation modes
        - Sequence Numbers with “fixed at zero” value
        - Sequence Numbers with incremented counting using “skip zero at Rollover”
        - Sequence Numbers with incremented counting using “include zero at Rollover”
- PW/Bundle Payload Types
  - TDMoP PW/Bundles (non-HDLC) - Constant Bit Rate Services (e.g. PCM voice)
    - Unstructured PW Payload (without framing; SAToP): E1, T1 and slower TDM bit rate (≤ 2.048 Mb/s)
    - Structured PW Payload (with framing; CESoPSN)
      - E1, T1-SF and T1-ESF formats

        - Any Nx64 Kb/s bit rate from a single T1 or E1 TDM Port
        - With or without CAS Signaling
    - HDLC PW/Bundles (e.g. SS7 Signaling)
        - Unstructured PW Payload: E1, T1 and slower TDM bit rate (≤ 2.048 Mb/s)
        - Structured PW Payload: Any Nx64 Kb/s bit rate from a single T1 or E1 TDM Port (for 8-bit HDLC)

- CES/SAT Processing
    - 256 CES/SAT Engines (one per PW/Bundle)
    - Per PW/Bundle Settings
        - Any Payload Size (up to maximum 2000 byte Ethernet Packet length)
            - Optional "zero" payload size for PW/Bundles that are only used for Clock Recovery
        - RXP Jitter Buffer (to compensate for PDV and for packet re-ordering; up to 500 ms)
            - Programmable "Begin Play-out Watermark" (for PDVT)
        - TXP high or low priority queue scheduling

- HDLC Processing
    - 256 HDLC Engines (one per PW/Bundle)
    - Configurable Transmit TDM Port minimum number of Intra-frame Flags (1 to 8)
    - Per Engine Settings
        - 2-bit, 7-bit or 8-bit HDLC coding
        - 16-bit, 32-bit or "no" Trailing HDLC FCS
        - Intra-frame Flag Value (0xFF or 0x7E)
        - HDLC Transmission Bit Order using MSB first or LSB first

CPU Interface Features

- CPU Packet Interface (for CPU-based OAM and Signaling)
    - Stores up to 512 Transmit and 512 Receive Packets that can be 64 byte to 2000 byte in length
    - RXP direction
        - Provides detected packet type with each received RXP CPU packet
            - In-band VCCV OAM
            - UDP-specific (Out-band VCCV) OAM
            - MEF OAM Ethernet Type
            - Configured "CPU Ethernet Type"
            - ARP
            - Broadcast Ethernet DA
            - Several Packet Header Conditions (e.g. NDP/IPv6 & unknown IP DA)
        - Provides Local Timestamp indicating the time the packet was received (in 1 us or 100 us units)
    - TXP direction
        - Any CPU generated Header and Payload (any Layer 2/3/4 content)
            - Support for IP FCS and UDP FCS Generation
            - Optionally inserts TXP OAM Timestamps (in 1 us or 100 us units)

- DS34S132 Control Interface
    - MPC8xx or MPC83xx synchronous interface using a 50 to 80 MHz clock rate (the MPC8xx and MPC83xx are processor product families of Freescale Semiconductor, Inc.)
    - Selectable 16-bit or 32-bit data bus
    - DS34S132 device Control & Sense Registers
    - Mask-able Interrupt Hierarchy for Change of Status, Alarms and Events
    - Ethernet Port RMON Statistics

Miscellaneous

- Loopbacks
    - PW/Bundle Loopback (payload from RXP PW packets are transmitted in TXP PW packets)
    - TDM Port Line Loopback (all data from Receive TDM Port sent to Transmit TDM Port using RCLK)

    - TDM Port Timeslot Loopback (from Receive to Transmit TDM Port Timeslot using RCLK)
- TDM Port and/or Ethernet Port BERT Testing
    - Half Channel (one-way) or Full Channel (round-trip) Testing
    - Flexible PRBS, QRBS or Fixed Pattern Testing
- 16-bit DDR SDRAM Interface that does not require any glue-logic
- IEEE 1149.1 JTAG support
- MBIST (memory built-in self test)
- 1.8V Core, 2.5V DDR SDRAM and 3.3V I/O that are 5V tolerant
- 27 x 27 mm, 676-pin BGA package (1mm pitch)

8 PIN DESCRIPTIONS

8.1 Short Pin Descriptions

Table 8-1. DS34S132 Short Pin Descriptions

Name	Type*	Function
TDM Port n = 0 through 31 Ports		
TCLKO*n*	Oz	Transmit TDM Clock Output
TSYNC*n*	IO	Transmit Frame (Frame or Multi-frame Sync Pulse)
TDAT*n*	Oz	Transmit NRZ Data
TSIGn	Oz	Transmit Signaling
RCLK*n*	I	Receive Clock Input
RSYNC*n*	IO	Receive Frame (Frame or Multi-frame Sync Pulse)
RDAT*n*	I	Receive NRZ Data
RSIGn	I	Receive Signaling
100/1000 Mbps Ethernet MAC Interface (GMII/MII)		
TXCLK	Ipu	MII Transmit clock (25 MHz)
GTXCLK	Oz	GMII Transmit clock (125 MHz)
TXD[7:0]	Oz	GMII/MII Transmit data
TXEN	Oz	GMII/MII Transmit data enable
TXER	Oz	GMII/MII Transmit packet frame invalid
RXCLK	Ipu	GMII/MII Receive clock (25 MHz or 125 MHz)
RXD[7:0]	I	GMII/MII Receive data
RXDV	I	GMII/MII Receive data valid
RXER	I	GMII/MII Receive error
COL	I	MII Collision Detection (not used)
CRS	I	Carrier Sense Detection (not used)
MDC	Oz	Management Data Clock
MDIO	IO	Management Data Input/Output
CPU Interface		
PD[31:0]	IO	Data [31:0]
PA[13:1]	I	Address [13:1]
PALE	I	Address Latch Enable
PCS_N	I	Chip Select (active low)
PRW	I	Read/Write
PRWCTRL	I	Read/Write Control
PTA_N	Oz	Transfer Acknowledge (active low)
PWIDTH	I	Processor Bus Width
PINT_N	Oz	Interrupt Out (active low)
External Memory Interface – DDR SDRAM		
SDCLK, SDCLK_N	Oz	SDRAM Clock
SDCLKEN	Oz	Clock Enable
SDCS_N	Oz	Chip Select (active low)
SDRAS_N	Oz	RAS (active low)
SDCAS_N	Oz	CAS (active low)
SDWE_N	Oz	Write Enable (active low)
SDBA[1:0]	Oz	Bank Address Select
SDA[13:0]	Oz	Address
SDDQ[15:0]	IO	Bi-directional Data Bus
SDLDM	Oz	Lower Byte Data Mask
SDUDM	Oz	Upper Byte Data Mask

Name	Type*	Function
SDLDQS	Oz	Lower Byte Data Strobe
SDUDQS	Oz	Upper Byte Data Strobe
Clocks, Resets , JTAG & Miscellaneous		
CMNCLK	I	Optional Differential Clock Recovery Common Clock (8kHz to 25MHz)
EXTCLK[1:0]	I	2 Independent Optional External Clocks for TDM Port Transmit Timing
SYSCLK	I	System Clock for CPU Interface (50 MHz to 80MHz)
LIUCLK	Oz	1.544MHz or 2.048MHz
REFCLK	I	Optional Oscillator Reference for Clock Recovery (5 MHz to 155.52 MHz)
DDRCLK	I	DDR SDRAM clock (125MHz)
ETHCLK	I	Optional Clock for GMII operation & OAM Timestamps (25MHz or 125MHz)
EXTINT	I	Ethernet Phy Interrupt (if MDIO/MDC are not used)
EPHYRST_N	Oz	Ethernet Phy Reset signal
RST_N	I	Global Reset
JTCLK	I	JTAG Clock
JTMS	Ipu	JTAG Mode Select
JTDI	Ipu	JTAG Data Input
JTDO	Oz	JTAG Data Output
JTRST_N	Ipu	JTAG Reset (active low)
HIZ_N	I	High impedance test enable (active low)
TEST_N	I	Test enable (active low)
MT[15:0]	IO	Manufacturing Test
SMTI	Ipu	Manufacturing Test Input, Must be tied to VCC33.
SMTO	O	Manufacturing Test Output, Must be left unconnected (floating).
Power Supply Signals		
VDD33	pwr	Core Digital 3.3 Volt Power Supply Input
VDD18	pwr	Core Digital 1.8 Volt Power Supply Input
VSS	pwr	Ground for 3.3V and 1.8V supplies. Connect to Common Supply Ground
AVDD	pwr	SDRAM 1.8 Volt PLL Power (may be connected CVDD)
AVSS	pwr	AVDD Ground (may be connected to CVSS)
CVDD	pwr	CLAD 1.8 Volt Power (may be connected to AVDD)
CVSS	pwr	CVDD Ground (may be connected to AVSS)
VDDP	pwr	SDRAM Digital Core 2.5 Volt Power Supply Input
VDDQ	pwr	SDRAM DQ 2.5 Volt Power Supply Input
VSSQ	pwr	SDRAM Digital Ground for VDDP and VDDQ
VREF	pwr	SDRAM SSTL_2 Reference Voltage (one-half VDDQ)

Note: [1] n = 0 to 31 (port number), Ipu = input with pullup, Oz = output tri-stateable, IO = Bi-directional input/output

8.2 Detailed Pin Descriptions

Table 8-2. Detailed Pin Descriptions

Pin Name	Type	Pin Description
TDM Port n = 0 through 31 Ports		
TCLKOn	Oz	**Transmit Clock Output.** TCLKOn is derived from the clock recovery engine or from RCLKn when in loop-timed mode or from the EXTCLK signal.
TSYNCn	IO	**Transmit Sync.** TSYNCn may be a frame or multi-frame input or output signal. Each frame is a 125 us time period. The frame count for each multi-frame type is: T1-SF = 12; T1-ESF = 24; E1 = 16. If configured as an input, it is sampled by TCLKOn. If configured as an output, it is output with respect to TCLKOn.
TDATn	Oz	**Transmit Data Output.** TDATn is the TDM datastream recovered from the PSN, output with respect to TCLKOn.
TSIGn	Oz	**Transmit Signaling.** TSIGn is the transmit signaling recovered from the PSN, output with respect to TCLKOn. The CAS values are updated once every TSYNC period.
RCLKn	I	**Receive Clock.** RCLKn is input clock typically derived from a T1/E1 framer or LIU.
RSYNCn	I	**Receive Sync.** RSYNCn indicates the frame or multi-frame boundary for the T1/E1 datastream, typically derived from a T1/E1 framer or LIU and sampled by RCLKn. Each frame is a 125 us period. The frame count for each multi-frame type is: T1-SF = 12; T1-ESF = 24; E1 = 16.
RDATn	I	**Receive Data.** RDATn is the receive TDM datastream typically derived from a T1/E1 framer or LIU, sampled by RCLKn.
RSIGn	I	**Receive Signaling.** RSIGn is the receive signaling typically derived from a T1/E1 framer, sampled by RCLKn. The CAS values are updated once every RSYNC period.
100/1000 Mbps Ethernet MAC Interface (GMII/MII)		
TXCLK	Ipu	**Transmit Clock (MII).** Timing reference for TXEN and TXD[0:3]. The TXCLK frequency is 25 MHz for 100 Mbit/s operation.
GTXCLK	Oz	**GMII Transmit Clock Output.** 125MHz clock output available for GMII operation. This clock is synchronous to ETHCLK input.
TXD[0:7]	Oz	**Transmit Data 0 through 7(GMII Mode – TXD[0:7]).** TXD[0:7] is presented synchronously with the rising edge of TXCLK. TXD[0] is the least significant bit of the data. When TXEN is low the data on TXD should be ignored. **Transmit Data 0 through 3(MII Mode – TXD[0:3]).** Four bits of data TXD[0:3] presented synchronously with the rising edge of TXCLK. When MII mode is selected, TXD[4:7] pins are not used.
TXEN	Oz	**Transmit Enable (GMII).** When this signal is asserted, the data on TXD[0:7] is valid; synchronous with GTXCLK. **Transmit Enable (MII).** In MII mode, this pin is asserted high when data TXD[0:3] is being provided by the device. This signal is synchronous with the rising edge TXCLK. It is asserted with the first bit of the preamble. Synchronous with TXCLK.
TXER	Oz	**Transmit Error (GMII, MII).** When this signal is asserted, the PHY will respond by sending one or more code groups in error.
RXCLK	Ipu	**Receive Clock (GMII).** 125 MHz clock. This clock is used to sample the RXD[0:7] data. **Receive Clock (MII).** Timing reference for RXDV, RXER and RXD[0:3], which are clocked on the rising edge. RXCLK frequency is 25 MHz for 100 Mbit/s operation.

Pin Name	Type	Pin Description
RXD[7:0]	I	**Receive Data 0 through 7(GMII Mode – RXD[0:7]).** Eight bits of received data, sampled synchronously with the rising edge of RXCLK. For every clock cycle, the PHY transfers 8 bits to the device. RXD[0] is the least significant bit of the data. Data is not considered valid when RXDV is low. **Receive Data 0 through 3(MII Mode – RXD[0:3]).** Four bits of received data, sampled synchronously with RXCLK. Accepted when CRS is asserted. When MII mode is selected, RXD[4:7] pins are not used.
RXDV	I	**Receive Data Valid (GMII).** This active high signal, synchronous to RXCLK, indicates valid data from the PHY. In GMII mode the data RXD[0:7] is ignored if RXDV is not asserted high. **Receive Data Valid (MII).** This active high signal, synchronous to RXCLK, indicates valid data from the PHY. In MII mode the data RXD[0:3] is ignored if RXDV is not asserted high.
RXER	I	**Receive Error (GMII).** This signal indicates a receive error or a carrier extension in the GMII Mode. **Receive Error (MII).** Asserted by the PHY for one or more RXCLK periods indicating that an error has occurred. Active high indicates receive packet is invalid. **MII and GMII modes:** This is synchronous with RXCLK.
COL	I	**Collision Detect (MII).** Asserted by the Ethernet PHY to indicate that a collision is occurring. This signal is only valid in half duplex mode, and is ignored in full duplex mode.
CRS	I	**Receive Carrier Sense.** This signal is asserted by the PHY when either transmit or receive medium is active. This signal is not synchronous to any of the clocks.
MDC	Oz	**Management Data Clock.** A divided down SYSCLK that clocks management data to and from the PHY.
MDIO	IO	**Management Data IO.** Data path for control information between the device and the PHY. Pull to logic high externally through a 1.5K ohm resistor. The MDC and MDIO pins are used to write or read up to 32 Control and Status Registers in PHY Controllers. This port can also be used to initiate Auto-Negotiation for the PHY.
CPU Interface		
PD[31:0]	IO	**32-bit Processor Data Bus.** PD[31] is the MSB which should be mapped to D[0] of a MPC8xxx processor. **16-bit Processor Data Bus.** PD[15] is the MSB which should be mapped to D[0] of a MPC8xxx processor. PD[31:16] is not used and should be tied low. **32-bit & 16-bit Processor Data Bus.** Input signals on this bus are captured by the rising edge of SYSCLK. Output signals are updated on the rising edge of SYSCLK.
PA[13:2]	I	**Processor Address Bus.** The signals on this bus are captured by the rising edge of SYSCLK.
PA[1]	I	**32-bit Processor Address Bus Bit 1.** PA[1] is not used and should be tied low. **32-bit Processor Address Bus Bit 1.** When PA[1] = 0, PD[15:0] carries the upper 16 bits of the 32-bit word. When PA[1] = 1, PD[15:0] carries the lower 16 bits of the 32-bit word.
PALE	I	**Processor Address Latch.** PALE latches PA[13:1] on its falling edge. In non-muxed mode, tie high.
PCS_N	I	**Processor Chip Select.** Processor chip select active low. Synchronous to SYSCLK.
PRW	I	**Processor Read/Write.** The behavior of this signal is described by PRWCTRL. This signal is synchronous to SYSCLK.

Pin Name	Type	Pin Description
PRWCTRL	I	**Processor Read/Write Control.** 0 = PRW is high for a write, low for a read (PQ II Pro mode) 1 = PRW is low for a write, high for a read (PQ I mode)
PTA_N	Oz	**Processor Transfer Acknowledge.** This signal indicates to the processor on a read that data is valid on the data bus. On a write, it indicates that the DS34S132 is ready for a new transaction. This signal is synchronous to SYSCLK since the PowerQuicc I requires it. This signal requires an external pull-up. On the PowerQuicc I, the PTA_N is used as a data valid signal and therefore must be coincident with the data on read accesses (i.e. it may not be early.)
PWIDTH	I	**Processor Bus Width** 0 = 16-bit mode 1 = 32-bit mode
PINT_N	Oz	**Processor Interrupt.** When the bit configurable Interrupt Inactive Mode is '0', this pin is active low, asynchronous to SYSCLK and is high impedance when not active. When the bit configurable Interrupt Inactive Mode is '1', this pin is active low, asynchronous to SYSCLK and drives high when no interrupts are active.
External Memory Interface – DDR SDRAM		
SDCLK, SDCLK_N	Oz	**SDRAM Clock.** SDCLK and SDCLK_N are differential clock outputs. (Both pins are referenced collectively as SDCLK.) All address and control input signals are sampled on the positive edge of SDCLK and negative edge of SDCLK. Output (write) data is referenced to the rising edge and falling edge of SDCLK.
SDCLKEN	Oz	**SDRAM Clock Enable.** Active High. SDCLKEN must be active throughout DDR SDRAM READ and WRITE accesses.
SDCS_N	Oz	**SDRAM Chip Select.** All commands are masked when SDCS_N is registered high. SDCS_N provides for external bank selection on systems with multiple banks. SDCS-_N is considered part of the command code.
SDRAS_N	Oz	**SDRAM Row Address Strobe.** Active low output, used to latch the row address on rising edge of SDCLK. It is used with commands for Bank Activate, Precharge, and Mode Register Write.
SDCAS_N	Oz	**SDRAM Column Address Strobe.** Active low output, used to latch the column address on the rising edge of SDCLK. It is used with commands for Bank Activate, Precharge, and Mode Register Write.
SDWE_N	Oz	**SDRAM Write Enable.** This active low output enables write operation and auto precharge.
SDBA[1:0]	Oz	**SDRAM Bank Select.** These 2 bits select 1 of 4 banks for the read/write/precharge operations.
SDA[13:0]	Oz	**SDRAM Address.** The 14 pins of the SDRAM address bus output the row address first, followed by the column address. The row address is determined by SDA[0] to SDA[13] at the rising edge of clock. Column address is determined by SDA[0]-SDA[9] at the rising edge of the clock. SDA[10] is used as an auto-precharge signal.
SDDQ[15:0]	IO	**SDRAM Data Bus.** The 16 pins of the SDRAM data bus are inputs for read operations and outputs for write operations. At all other times, these pins are high-impedance.
SDLDM	Oz	**SDRAM Lower Data Mask.** SDLDM is an active high output mask signal for write data. SDLDM is updated on both edges of SDLDQS. SD_LDM corresponds to data on SDATA7-SDATA0.
SDUDM	Oz	**SDRAM Upper Data Mask.** SDUDM is an active high output mask signal for write data. SDUDM is updated on both edges of SDUDQS. SDUDM corresponds to data on SDATA15-SDATA8.
SDLDQS	Oz	**SDRAM Lower Data Strobe.** Output with write data, input with read data. SDLDQS corresponds to data on SDATA7-SDATA0.
SDUDQS	Oz	**SDRAM Upper Data Strobe.** Output with write data, input with read data. SDUDQS corresponds to data on SDATA15-SDATA8.

Pin Name	Type	Pin Description
Clocks, Resets , JTAG & Miscellaneous		
CMNCLK	I	**Common Clock.** This clock is used for Differential Clock Recovery. Common clock has to be a multiple of 8 kHz and in the range of 8 kHz to 25 MHz. The frequency input should not be too close to an integer multiple of the service clock frequency. Based on these criteria, the following frequencies are suggested: SONET/SDH systems: 19.44 MHz ATM systems: 9.72 MHz or 19.44 MHz GPS systems: 8.184 MHz Synchronous Ethernet systems: 25 MHz CMNCLK may also be used in lieu of REFCLK if the CMNCLK frequency matches one of the frequencies used for REFCLK and if CMNCLK is a high quality clock (Stratum 3). When CMNCLK is not used tie to ground or VDD(3.3V).
EXTCLK[1:0]	I	**External Clock.** This clock is used as an E1 or T1 Station Clock. In this mode, is used for TDATn. When this clock is not used tie to ground or VDD(3.3V).
SYSCLK	I	**System Clock.** This clock shall be in the range of 50 – 85 MHz and also synchronous with the CPU's bus clock.
LIUCLK	O	**LIU Clock.** This clock is generated by the CLAD based on either REFCLK or CMNCLK and can be selected to be 1.544 MHz or 2.048 MHz. By default, this clock drives low.
REFCLK	I	**Reference Clock.** This clock must be one of the following frequencies: 5 MHz, 5.12 MHz, 10 MHz, 10.24 MHz, 12.8 MHz, 13 MHz, 19.44 MHz, 20 MHz, 25 MHz, 30.72 MHz, 38.88 MHz, 77.76 MHz, or 155.52 MHz. This input shall be a stratum 3 quality or better. This clock is selectable by the CLAD to derive the synthesis clock for the clock recovery engine. CMNCLK can be used in lieu of REFCLK.
ETHCLK	I	**Ethernet Clock.** This clock is used as the source for the GTXCLK in GMII mode and is used as a constant reference for several internal clocks. This signal must always be provided with 125MHz clock +/- 100ppm. It may use the same oscillator as DDRCLK.
DDRCLK	I	**DDR Clock.** This clock is used as the source for SD0CLK and SDCLK. The clock frequency should be 125 MHz. It may use the same oscillator as ETHCLK.
RST_N	I	**Reset.** An active low signal on this pin resets the internal registers and logic. While this pin is held low, the microprocessor interface is kept in a high-impedance state. This pin should remain low until power is stable and then set high for normal operation.
JTCLK	I	**JTAG Clock.** This signal is used to shift data into JTDI on the rising edge and out of JTDO on the falling edge.
JTMS	Ipu	**JTAG Mode Select.** This pin is sampled on the rising edge of JTCLK and is used to place the test access port into the various defined IEEE 1149.1 states. This pin has a 10k pull up resistor.
JTDI	Ipu	**JTAG Data In.** Test instructions and data are clocked into this pin on the rising edge of JTCLK. This pin has a 10k pull up resistor.
JTDO	Oz	**JTAG Data Out.** Test instructions and data are clocked out of this pin on the falling edge of JTCLK. If not used, this pin should be left unconnected.
JTRST_N	Ipu	**JTAG Reset.** JTRST is used to asynchronously reset the test access port controller. After power up, a rising edge on JTRST will reset the test port and cause the device I/O to enter the JTAG DEVICE ID mode. Pulling JTRST low restores normal device operation. JTRST is pulled HIGH internally via a 10k resistor operation. If boundary scan is not used, this pin should be held low.
TEST_N	I	**Test Enable.** (active low)
HIZ_N	I	**High Impedance test enable.** This signal puts all digital output and bi-directional pins in the high impedance state when it is low and JTRST is low. For normal operation tie high. This is an asynchronous input.
EXTINT	I	**External PHY Interrupt.** PHY Interrupt to MAC, if MDIO and MDC are not used.
MT[15:0]	IO	**Manufacturing Test.** For normal operation leave these pins unconnected.
SMTI	Ipu	**Manufacturing Test Input**, Must be tied to VCC33.

Pin Name	Type	Pin Description
SMTO	O	**Manufacturing Test Output**, Must be left unconnected (floating).
Power Supply Signals		
VDD33	pwr	**VDD33.** Connect to 3.3 Volt Power Supply
VDD18	pwr	**VDD18.** Connect to 1.8 Volt Power Supply
VSS	pwr	**VSS.** Ground connection for 3.3V and 1.8V supplies. Connect to the Common Supply Ground
AVDD1	pwr	**Analog PLL Power 1.** Connect to a 1.8 Volt Power Supply
AVDD2	pwr	**Analog PLL Power 2.** Connect to a 1.8 Volt Power Supply
AVSS	pwr	**Analog PLL Ground.**
VDD25	pwr	**SDRAM Digital Power.** Connect to a 2.5 Volt Power Supply
VDDQ	pwr	**SDRAM Digital DQ Power.** Connect to a 2.5 Volt Power Supply
VSSQ	pwr	**SDRAM Digital Ground.**
VREF	ref	**Voltage Reference.** SDRAM SSTL_2 Reference Voltage

Notes: n=0 to 31 (port number), Ipu (input with pullup), Oz (output tri-stateable), & IO (Bi-directional input/output).

9 FUNCTIONAL DESCRIPTION

This section provides a high level, functional view of the S132. Because of the high level of integration and complexity that has been included in the S132, it is necessary to first explain the terminology and conventions that are used. This Functional Description section is further supported by the Register Guide section, which identifies common settings for specific applications, and the Register Definition section which provides a definition for each register, but without as much regard for application information.

The industry term “Pseudowire” (PW) includes the idea of a “virtual connection” (pseudo ≅ virtual; wire ≅ connection). PW data is carried in packets. The connection is not a “hardwired” connection but “virtual” in nature where the “virtual connection” is recognized by interpreting the PW packet header (e.g. the PW-ID provides the PW destination address).

There are multiple PW protocols to carry different types of data. Each is designed to support a particular service type, for example PCM, HDLC, ATM and Frame Relay. The PW protocols enable a Service Provider to transport and switch all of its services using a single unified switching/routing/forwarding technique (e.g. IPv4).

Enterprise and CPE equipment can similarly use the PW protocols. PWs can enable LAN switching/routing/forwarding using a single protocol/equipment type (e.g. Ethernet switch) to support both packet encapsulated-TDM and bursty packet data. PWs can also be used to enable the use of a single WAN interface to carry aggregated Enterprise data across the WAN/PSN. For example, if the WAN service/interface is Ethernet, then a single WAN-Ethernet interface can be used to carry “bursty” Ethernet data and packet encapsulated TDM data across the WAN. This prevents the need to pay for independent Ethernet WAN and TDM WAN services.

The DS34S132 supports three PW types. The term “TDMoP PW” is used to refer to a PW that is used to transport a constant-bit-rate TDM service. The S132 supports two types of TDMoP PWs: SAToP (SAT) and CESoPSN (CES). The term “HDLC PW” is used to refer to a PW that is used to transport the non-idle payload data from an HDLC data stream. The S132 includes the necessary functions to translate TDM constant bit rate data streams to/from TDMoP PWs and HDLC data streams to/from HDLC PWs for PSNs using the UDP/IP, L2TPv3/IP, MEF-8 or MFA-8 protocol.

PWs that are recognized by the S132 are described using the terms “Connection”, ”Packet”, “Bundle” and “Bundle ID” (BID). Each term emphasizes a different aspect of the PW. The S132 supports up to 256 programmed Bundles (numbered Bundle 0 through Bundle 255). The term “Bundle” emphasizes the recognized/programmed parameters associated with a PW (the programmed header format, payload format, PW-ID value, etc.). If a PW packet is received by an S132, but the packet format or PW-ID of the packet does not equal that of a programmed Bundle, then the packet is not recognized. The term “BID” equates to a “recognized/programmed PW-ID” (part of a Bundle).

The term “Connection” emphasizes the type of data carried by a packet (e.g. timing) and emphasizes where the data is forwarded inside the S132 (e.g. to a Clock Recovery Engine). The S132 Bundle Connection types include SAT/CES Payload, HDLC Payload, SAT/CES PW-Timing and CPU. SAT, CES and HDLC Payload Connections are used to forward the data between a TDM Port and the payload of a TDMoP PW. A SAT or CES PW-Timing Connection is used to forward the timing information between a TDM Port and the TDMoP PW. A CPU Connection is used to forward packets between the CPU and the Ethernet Port. CPU packets can be PW packets or non-PW packets. A “connection” is commonly “established” by enabling an internal S132 function. For example enabling the Clock Recovery Engine for a particular Bundle “establishes” a PW-Timing Connection for that Bundle.

The term “Bundle” can be thought of as a “small set of connections and parameters”. The packets for a Bundle can contain data/information for multiple connections, e.g. the packet for a SAT Bundle can contain data/information for a SAT Payload Connection and a SAT PW-Timing Connection.

The S132 supports a number of CPU packet types that are not Bundle/PWs. The term “CPU Connection” indicates that the data stream carries data that is forwarded to the CPU. The S132 supports specialized header field values and conditions that identify CPU Connections (e.g. the MEF OAM header).

The terms “OAM Bundle” and “OAM BID” are similar in meaning to “Bundle” and “BID” except that they are only used for CPU Connections. The S132 supports up to 32 “OAM Bundles” that are programmed independent of the 256 Bundles. “OAM Bundles” are used to support Out-band VCCV (also known as UDP-specific OAM).

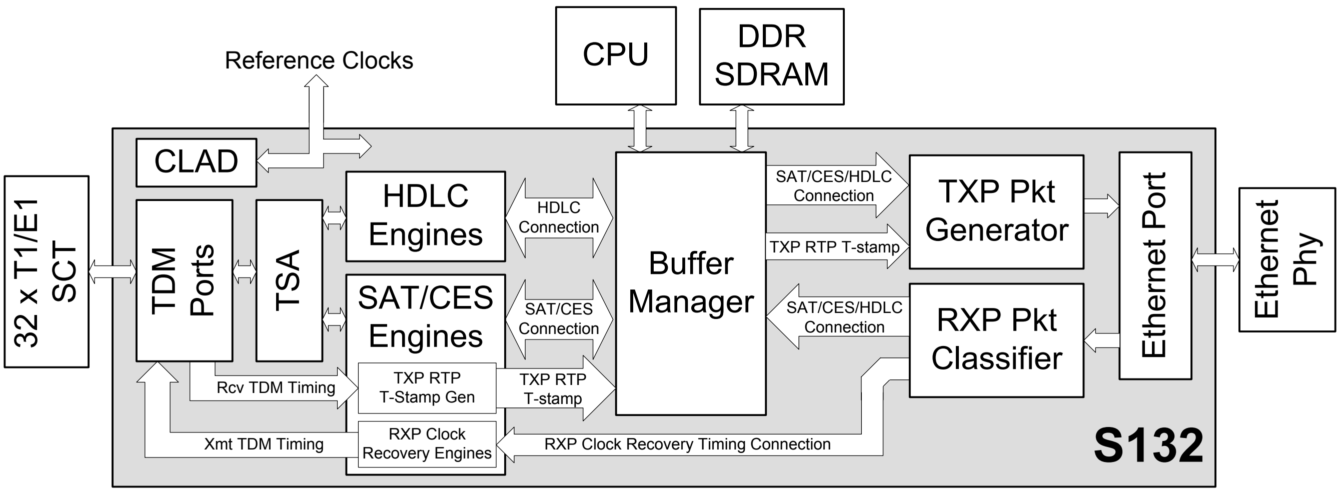
The term “Packet”, when used in combination with one of the Connection/data types emphasizes that the packet contains data/information for a particular type of connection (e.g. CES, SAT, HDLC, PW-Timing and/or CPU Packet). The terms “packet” and “frame” are loosely used interchangeably to identify a “datagram/unit” of data that is carried inside an encapsulation protocol. The term “frame” is also used to mean a “125 us TDM time period” but then can be understood to use this meaning from the TDM context of the surrounding text.

The term “HDLC” is used to mean “HDLC-encoded data that is processed by the S132 for a TDM Port that is translated to/from an HDLC PW packet stream”. The terms “CES Payload” and “SAT Payload” are used to mean “data that is processed as constant bit rate data (e.g. PCM) without HDLC encoding and translated into the payload of a TDMoP PW”. In the case of “CES Payload”, CAS Signaling may also be included through CAS timing rules. “TDM”, by itself, is used to mean any of these 3 data types (CES, SAT or HDLC; “coming from a TDM Port”).

”PW-Timing” is used to mean “the timing of a TDM Port that is communicated in a PW” and is only associated with a SAT/CES Bundles. The terms “Clock Recovery” and “RTP Timestamps” are “PW-Timing” functions.

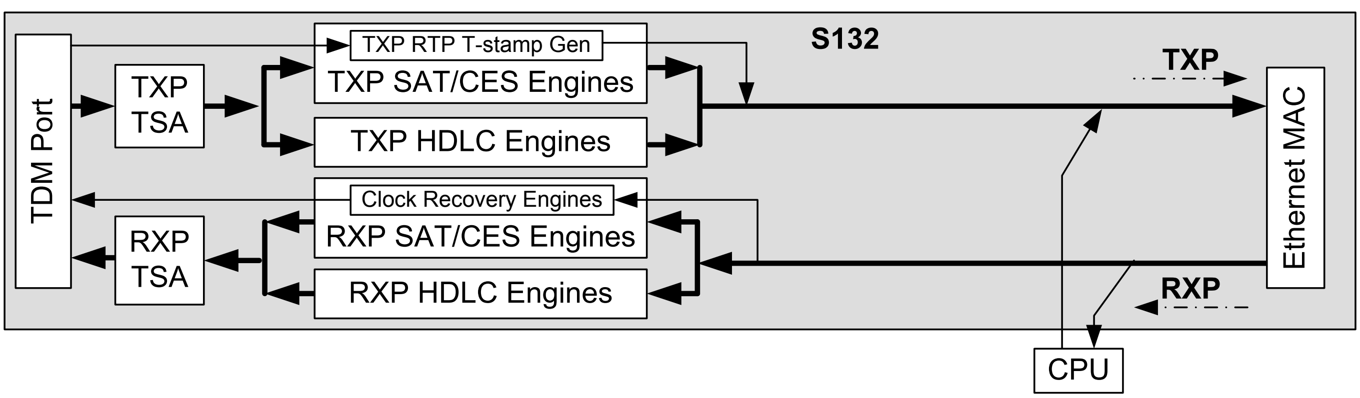
Figure 9-1 provides a high level view of the basic functional areas within the S132 device.

Figure 9-1. S132 Block Diagram



The term “RXP” is used to denote “data that is received at the Ethernet port and forwarded to a transmit TDM Port, the CPU or an RXP Clock Recovery Engine. “TXP” is used to denote “data received from a TDM Port or the CPU that is transmitted at the transmit Ethernet port”. The RXP and TXP directions are depicted in the simplified diagram in Figure 9-2. Bold lines are used to depict the “payload” connection paths (SAT/CES and HDLC). Thin lines depict the PW-Timing and CPU connection paths.

Figure 9-2. RXP/TXP Data Path Directions



The term “Port” is used with two meanings. The UDP standard uses “Port” to mean “virtual port” (e.g. UDP Source Port ID). Otherwise the term “Port” is used to mean an electrical S132 port with external pins (e.g. TDM Port).

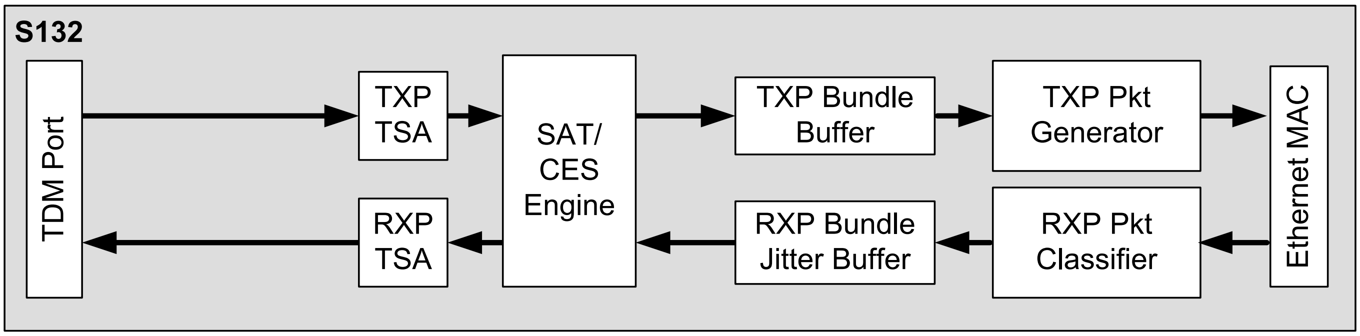
9.1 Connection Types

The following subsections describe the different connection types in more detail.

9.1.1 SAT/CES Payload Connections

The S132 can support up to 256 SAT/CES Payload Connections spread across 32 TDM Ports. Each SAT/CES Payload Connection carries constant bit rate data and is programmed as part of a Bundle. In the RXP direction, the Classifier identifies a packet for a SAT/CES Payload Connection when the received Header and PW-ID match that of a Bundle and that Bundle is programmed to forward payload data to a SAT/CES Engine. The SAT/CES Payload Connection is diagramed in Figure 9-3.

Figure 9-3. SAT/CES Payload Connection



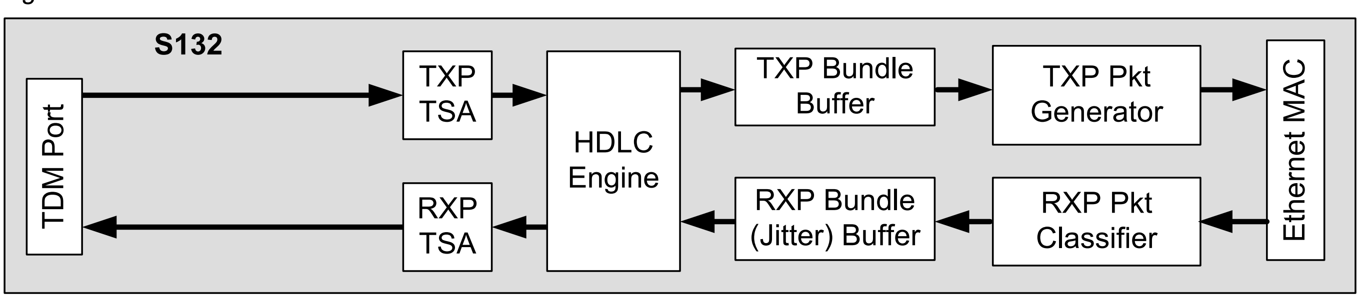
Each Bundle can be configured to support any number of DS0s up to an entire TDM Port line rate. In the RXP direction the PW Header is stripped off of the packets and the payload is stored in a Jitter Buffer to smooth the bursty transmission of the PSN. In the TXP direction, when sufficient SAT/CES Payload has been received the S132 appends a configured TXP Bundle Header to generate a PW packet. A Timeslot Assignment block provides a DS0 cross-connect function to interconnect the payload of any SAT/CES Bundle to any set of DS0 positions on a single TDM Port and to allow control and monitoring of Sub-channel CAS Signaling and Data Conditioning.

A Bundle that includes a SAT/CES Payload Connection can also include a PW-Timing Connection and an In-band VCCV (CPU) Connection (the PW-Timing and CPU Connections are described in the sections that follow).

9.1.2 HDLC Connections

The S132 supports up to 256 HDLC Connections. This connection type can be used to support T1/E1 CCS Signaling or other HDLC encoded packet streams. Each HDLC Connection is programmed as part of a Bundle. In the RXP direction, the Classifier identifies a packet for an HDLC Connection when the header and PW-ID of a received packet matches the Header protocol and BID of a Bundle and that Bundle is programmed to forward data to an HDLC Engine. The HDLC Connection is diagramed in Figure 9-4.

Figure 9-4. Bundle HDLC Connection



At the TDM Port the HDLC data appears as constant bit rate data because the HDLC packet stream, at the TDM Port, is supplemented with Idle HDLC Flags (Idle Flags are used during time periods when there are no HDLC packets). On the Ethernet/PW side the HDLC encoding does not exist. The HDLC data no longer appears as constant bit rate data since the HDLC Idle Flags are not carried by the HDLC PWs (only non-idle packet data is carried by an HDLC PW).

Each HDLC Bundle can be configured to support any number of DS0s up to the entire TDM Port line rate. In the RXP direction the PW Header is stripped off and a Buffer is used to store the complete packet so that the packet's Ethernet FCS can be verified before transmitting the payload data at the TDM Port. In the TXP direction the HDLC encoding is stripped off. When a complete HDLC frame has been received and the HDLC FCS has been verified, the S132 appends a programmed TXP Bundle Header and generates a PW packet. A Timeslot Assignment block provides a DS0 cross-connect function to interconnect the payload of any HDLC Bundle to any set of DS0 positions on a single TDM Port.

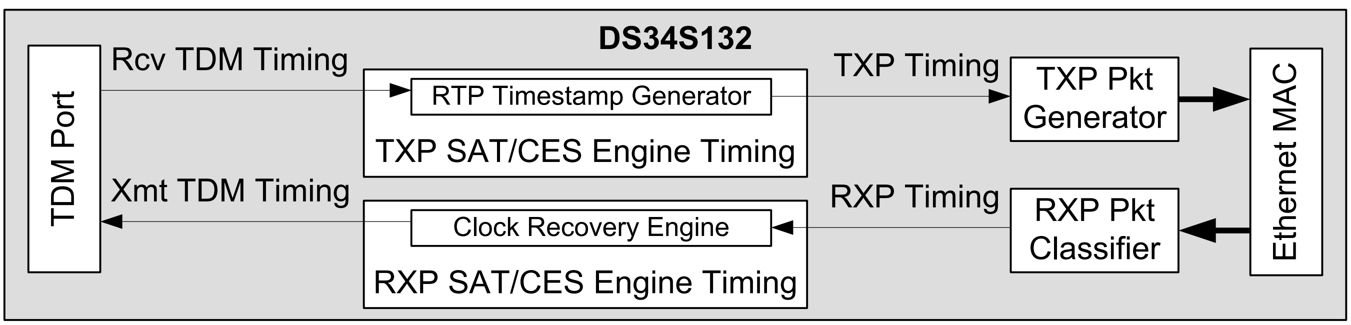
9.1.3 SAT/CES PW-Timing Connections

SAT/CES TDMoP PW packets intrinsically always carry timing information by the constant periodic transmission rate of the PW packets. The Adaptive Clock Recovery (ACR) technique takes advantage of this fact and does not require that a TDMoP PW packet include a Timestamp header field. The Differential Clock Recovery (DCR) Technique does not directly utilize the periodic transmission rate, but instead utilizes RTP Timestamps to indicate the time differences between each successive PW packet. Every TDMoP PW packet includes ACR timing information and can optionally include Timestamps.

When more than one TDMoP PWs are associated with a single TDM Port and the timing for that TDM Port uses clock recovered timing, only one Bundle/PW can be programmed to include a PW-Timing Connection (to supply the timing) and the frequency (data rate) for all of the other Bundle/PW streams assigned to that TDM Port must be identical (synchronized). Otherwise the data, at the transmit TDM Port, for a non-synchronous PW would be corrupted (the TDM Port can only transmit at one line rate). When timing information is included in PW packets, but the Bundle does not include a PW-Timing Connection, the timing information is ignored by the S132.

The DS34S132 internal PW-Timing Connections are used by the RXP Clock Recovery Engines and the TXP RTP Timestamp Generator. PW-Timing Connections can be set up in either direction or in both directions. The RXP and TXP PW-Timing Connections are diagramed in Figure 9-5.

Figure 9-5. Bundle PW-Timing Connections



The S132 supports up to 32 RXP Clock Recovery PW-Timing Connections (one for each transmit TDM Port) and up to 256 TXP, RTP Timestamp, PW-Timing Connections (one for each TXP Bundle). Each RXP PW-Timing Connection is programmed as part of the RXP Bundle parameters. Each TXP PW-Timing Connection is enabled by programming the TXP Header Descriptor to include an RTP Header.

In the RXP direction the Classifier identifies the packets for an RXP PW-Timing Connection when a received packet matches the Header protocol and BID of a Bundle and that Bundle is programmed for "Clock Recovery". The PW-Timing information from the packet is forwarded to the appropriate Clock Recovery Engine which in turn is used to drive the timing of a transmit TDM Port. The clock recovery timing information can be derived from the RXP packet rate (ACR) or RTP Differential Timestamps (DCR-DT).

In the TXP direction, the PW timing information is derived from the receive TDM Port. A TXP packet is periodically generated when the prescribed amount of SAT/CES Payload has been received from the TDM Port. The TXP PW-Timing information is conveyed through the rate at which TXP packets are transmitted (ACR) but can also be supplemented by inserting an optional TXP RTP Timestamp. The TXP PW-Timing Connection (when included/enabled in a TXP Bundle) inserts the optional TXP RTP Timestamp. The TXP PW-Timing Connection is not required if the far end clock recovery uses the ACR technique.

A Bundle that includes a PW-Timing Connection (RXP and/or TXP direction) can also include a SAT/CES Payload Connection. If the Bundle does not include a SAT/CES Connection, the Bundle/PW is called a "Clock Only" Bundle/PW. Clock Only packets do not include payload data, but instead only carry the timing information (conveyed through the packet transmission rate and/or RTP Timestamps).

The most generalized TDMoP PW application recovers TDM timing from a TDMoP PW packet stream. However, for some applications the timing/rate of the TDMoP PW payload data is synchronized to a distributed, common clock reference at both ends of the PW and clock recovery is not required (e.g. for synchronous T1/E1 data streams). For these cases the Transmit TDM Port can use an external clock signal instead of a Clock Recovery Engine (none of the Bundles associated with that TDM Port include an RXP PW-Timing Connection). This special application (synchronous T1/E1 data streams) should not be confused with the DCR mode that uses a Common Clock to drive a Clock Recovery Engine that recovers the timing of a T1/E1 data stream that may be asynchronous.

Only SAT/CES Bundle/PWs can include PW-Timing. HDLC and CPU packet streams (including those for OAM Bundles) should not be used for PW-Timing since these packet types do not provide a constant bit rate.

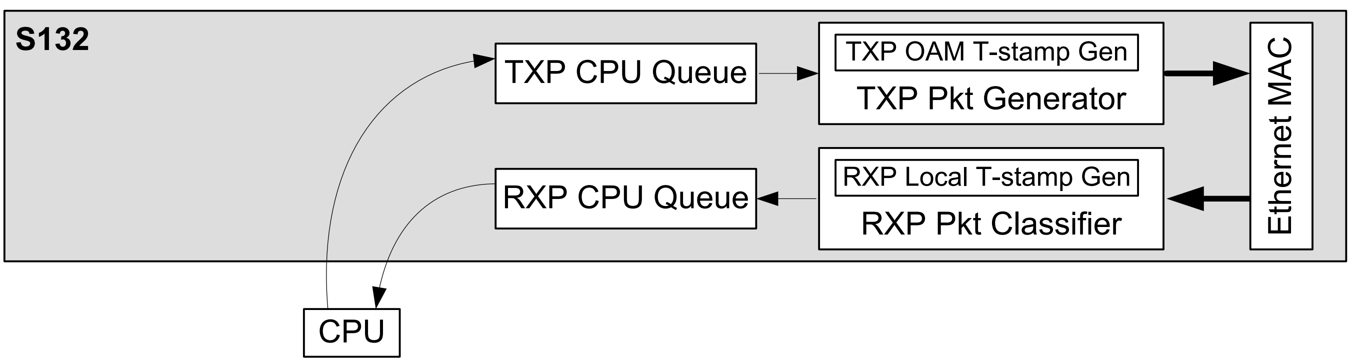
9.1.4 CPU Connections

CPU Connections provide the CPU with the ability to send and receive Ethernet packets. CPU Connections can be used for VCCV connections that are used to establish and monitor PWs, for Ethernet OAM (e.g. MEF OAM), for specialized Ethernet protocols (e.g. ARP) and for detecting unexpected or invalid packet types. The different types of CPU Connections that are supported are listed below. The CPU Connections are described in more detail in the "CPU Packet Classification" and "TXP CPU Packet Generation" sections.

Debug "Normal" Bundle	Too many MPLS Labels	ARP with known IP DA
OAM Bundle	Unknown Ethernet DA	ARP with unknown IP DA
CPU Destination Ethernet Type	Unknown PW-ID	Unknown Ethernet Type
MEF OAM Ethernet Type	Unknown UDP Protocol	Unknown IP DA
In-band VCCV OAM	Unknown IP Protocol	Ethernet Broadcast DA

The CPU Connection is diagramed in Figure 9-6.

Figure 9-6. CPU Connections



The S132 supports optional OAM Timestamps. The OAM Timestamps are independent of the RTP header Timestamps (PW-Timing Connections). In the RXP direction the S132 records when each RXP CPU packet is received and forwards an RXP Local Timestamp with each packet that is forwarded to the CPU. This RXP Local Timestamp is always enabled for all CPU packet types (the CPU can ignore the RXP Local Timestamp if the information is not relevant). In the TXP direction, the S132 can be programmed to add a TXP OAM Timestamp to any outgoing CPU packet (e.g. for TDMoIP-VCCV-OAM header according to RFC5087 Appendix D).

9.2 TDM Port Functions

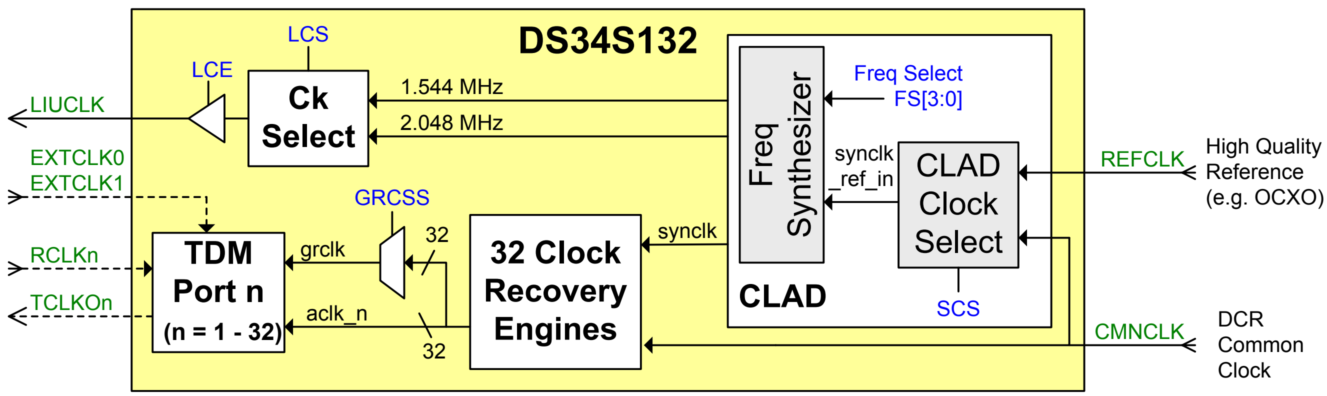
The S132 includes 32 TDM Ports. Each TDM Port can be used to support a T1, E1 or any slower TDM data stream. Each TDM Port uses a serial clock and data interface. The high level functions include:

- Structured & Unstructured Formats
- T1, E1 and slower TDM Port Line Rates
- T1SF, T1ESF and E1 Multi-frame Formats
- N x 64 Kb/s PW Packet Payload Rates
- With & without CAS Signaling
- DS0 Timeslot Assignment
- CPU Monitor and Control of CAS Signaling
- CPU Control for Data Conditioning
- TDM Port Timing
  - From Recovered or External Time References
  - Adaptive & Differential Clock Recovery
  - Generates Differential & Absolute Timestamps
- TDM Port, Timeslot and PW Loopbacks
- BERT Diagnostics

9.2.1 TDM Port Related Input and Output Clocks

The TDM Port Input and Output Clocks are identified in Figure 9-7.

Figure 9-7. TDM Port Input and Output Clock Overview



The S132 Clock Recovery Engines support "Adaptive Clock Recovery" (ACR) and "Differential Clock Recovery" (DCR). The ACR technique measures the timing of each successive RXP Packet to determine the recovered clock frequency. The DCR technique uses RTP timestamps to determine the recovered clock frequency. Two external clock recovery reference inputs (REFCLK and CMNCLK) are used to supply 1) a Frequency Synthesizer reference input and 2) to provide a DCR common clock reference.

The Frequency Synthesizer reference input (synclk_ref_in) is required to generate an internal "synclk" signal. To achieve the jitter/wander performance of ITU G.823/824/8261 the reference should be at least equal to that of a Stratum 3 clock. The reference can be input on either REFCLK or CMNCLK (selected with G.CCR.SCS). For PSTN and Cellular Mobile Phone applications, the BITS or GPS Network Timing commonly provide at least a Stratum 3 reference. For applications where a Network Timing reference is not available, then an OCXO may be used. Some specialized TCXOs can also meet these stringent requirements. Otherwise, if the jitter/wander requirements can be relaxed then the synclk reference input signal requirements can be equally relaxed.

To support the DCR mode, both ends of the PW must share a common clock reference that is derived from a single timing source so that the frequency of the common clock reference at both ends of the PW are locked to each other. The CMNCLK input is used to provide the DCR common clock reference.

In public network applications that use the DCR mode, the public network broadcast Network Timing, that provides a Stratum 3 or better reference (e.g. BITS or GPS), can be used for the DCR common clock (CMNCLK) input and the synclk reference input; and the REFCLK input can be tied low to save power.

In applications that use the DCR mode, but the DCR common clock reference is not a Stratum 3 reference (e.g. private networks), the DCR common clock is connected to the CMNCLK input and a high quality reference (e.g. OCXO) is connected to the REFCLK input.

In applications that do not use the DCR mode, only a high quality reference is required that can be connected to CMNCLK or REFCLK and the unused input pin can be tied low to save power.

In general, the DCR technique provides better clock recovery performance than the ACR technique (when compared using an equal quality synclk reference input for both techniques).

For DCR applications the PW standards assume both ends of the PW use the same frequency for a DCR common clock reference. The S132 however also allows the DCR common clock frequency to differ from one end to the other (e.g. 2.5 MHz at one end and 25 MHz at the other), but with the requirement that the two are frequency locked to the same source (e.g. BITS) and the S132 is programmed to compensate for the frequencies that are used (Pn.PRCR4 and Pn.PRCR5).

To function well the DCR common clock (CMNCLK) frequency must be an integer multiple of 8 KHz and in the range of 1 MHz to 25 MHz, but not close to the T1/E1 clock frequency (1.544 MHz or 2.048 MHz). The CMNCLK frequency can be in the range from 8 KHz to 1 MHz, but with degraded MTIE (Maximum Time Interval Error) performance. The following frequencies are recommended according to equipment type. The RTP Timestamp coefficient registers (Pn.PRCR4 and Pn.PRCR5) must be set according to the CMNCLK frequency that is used.

SONET/SDH based equipment – 19.44 MHz	GPS based equipment – 8.184 MHz
ATM network equipment – 9.72 MHz or 19.44 MHz	Ethernet Equipment – 25 MHz

An internal CLAD generates the internal synclk signal from REFCLK or CMNCLK (selected with G.CCR.SCS). Any of the input frequencies listed below can be used. The input frequency is selected using G.CCR.FS.

5.000 MHz	10.24 MHz	19.44 MHz	30.72 MHz	155.52 MHz
5.120 MHz	12.80 MHz	20.00 MHz	38.80 MHz	
10.00 MHz	13.00 MHz	25.00 MHz	77.76 MHz	

The S132 includes 32 Clock Recovery Engines that are each hardwired to one of the 32 TDM Ports. One of the 32 TDM Port recovered clocks (aclk_n; n = 0 to 31) can be assigned as a Global Clock Recovery reference (grclk) using G.GCR.GRCSS. This allows the Clock Recovery Engine for one TDM Port to act as the “master timing” for other “slave timed” TDM Ports.

An LIUCLK output is generated by the CLAD to provide an optional T1/E1 clock reference for external circuits. The output is enabled with G.CCR.LCE and the frequency is set using G.CCR.LCS (1.544 MHz or 2.048 MHz).

In the TXP direction, the rate at which TXP Packets are transmitted is always directly related to the rate at which data is received at the TDM Port. In the RXP direction, there are several methods that can be used to reconstruct the transmit T1/E1 timing. The multiple timing sources provide the ability to support several different timing applications and to provide primary and secondary (backup) timing.

In the RXP direction, the TCLKOn signal can derive its timing from RCLKn (the TDM Port receive clock input), EXTCLK0, EXTCLK1, the internal aclk_n signal (the recovered clock from the Port “n” Clock Recovery Engine) or the internal grclk signal (Global Recovered Clock that is selected by G.GCR.GRCSS). Only aclk_n and grclk derive their timing from received RXP Packets. The selected timing source for a TDM Port must be equal to the payload bit rate of each of the RXP Bundles assigned to that TDM Port. If the timing of the selected clock source differs from one of its Bundles, then the internal RXP Jitter Buffer for that Bundle will overflow or underrun.

A TDM Port can be timed to an external T1/E1 reference that is input at EXTCLK0 or EXTCLK1 (e.g. for a Network Timed T1/E1). If the synclk reference input (at REFCLK or CMNCLK) is from a Network Timing source (e.g. BITS 8 KHz), then the LIUCLK output can be tied to EXTCLK0 or EXTCLK1 to provide Network Timing to the TDM Port.

RCLKn can be used as the TCLKOn timing source in applications where the TDM Port must use “Loop Timing”. “Loop Timing” can be used when the TXP data stream at any node within the network returns the TXP data back in the RXP direction (loopback). Or it can be used in applications where the local transmit T1/E1 line rate is required to use the local receive T1/E1 line rate (e.g. RCLKn provides Network Timing).

9.2.1.1 PW-Timing

The TDMoP PW standards define two PW Timing techniques: “Adaptive Clock Recovery” (ACR) and “Differential Clock Recovery using Differential Timestamps” (DCR-DT). A third technique, using “Absolute Timestamps” (AT), is supported by some companies, but is not prescribed by the TDMoP standards. The S132 is compatible with each of these PW-Timing techniques.

The ACR technique uses the (intrinsic) packet transmission rate to convey the PW-Timing (e.g. 1 packet received every 1 ms). The DCR technique uses RTP Timestamps to convey the PW-Timing information from the originating side. Differential RTP Timestamps (DCR-DT) provide a means to monitor the time period between successive packets using time units that are equalized at both ends of the PW through the use of a common clock reference signal (e.g. Timestamp = 125 might equate to 125 us). The “Absolute RTP Timestamp” (AT) indicates the amount of data that has been received at the TDM Port (e.g. 1000 bits) making the Absolute Timestamp an integer multiple of the Sequence Number. Both Timestamp types provide a measure of time, one referenced to a common clock, the other referenced to the receive TDM Port line rate.

In the TXP direction the S132 supports all 3 techniques. The ACR technique is implicit in the packet transmit rate. The DCR-DT and AT techniques are supported by transmitting Differential or Absolute Timestamps (respectively).

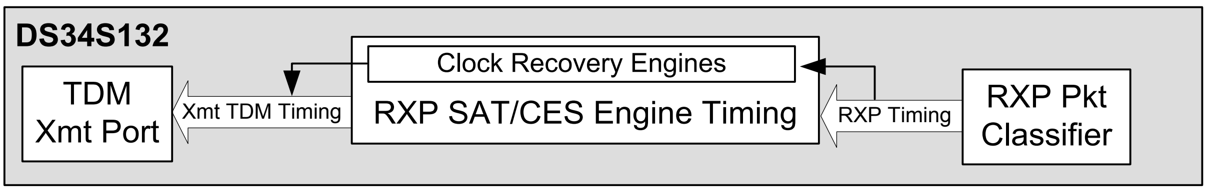
In the RXP direction the S132 directly supports the ACR and DCR-DT techniques. With the ACR technique the Clock Recovery Engine recovers timing from the rate at which packets are received. With DCR-DT the Clock Recovery Engine recovers timing from the received RTP Differential Timestamps.

In the RXP direction the S132 is compatible with (supports) the AT technique, but does not utilize the RTP Absolute Timestamps. To provide compatibility with the AT technique the Clock Recovery Engine instead recovers timing using the ACR technique (derived from the rate at which packets are received).

9.2.1.1.1 RXP Clock Recovery (RXP PW-Timing)

There are 32 RXP Clock Recovery Engines, one hardwired to each of the 32 TDM Ports. Each can be programmed to support the ACR or DCR-DT technique.

Figure 9-8. Clock Recovery Engine Environment



When a Transmit TDM Port is programmed to derive its timing from a Clock Recovery Engine, one TDMoP PW/Bundle must be programmed to include an RXP PW-Timing Connection (B.BCDR4.PCRE). No more than one PW/Bundle can be assigned to provide the RXP PW-Timing Connection for a TDM Port.

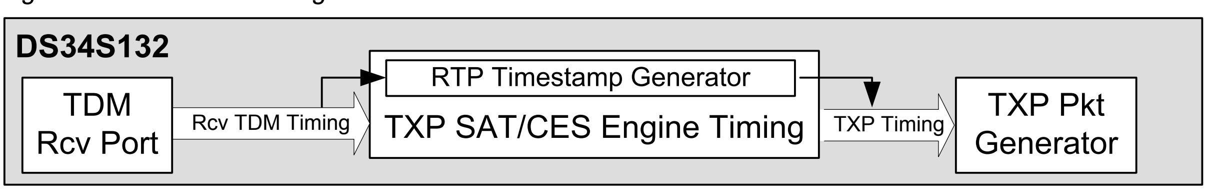
The RXP Clock Recovery technique (ACR or DCR-DT) is selected by properly programming the S132 Clock Recovery Engine DSP firmware revision (not included in this Datasheet).

9.2.1.1.2 TXP PW-Timing

In the TXP direction the TDMoP PWs communicate timing information through the transmission rate of the TXP Packets (ACR) and can optionally include an RTP Timestamp with each TXP Packet. TXP Packets are automatically transmitted when sufficient T1/E1 data has been received to fill the TXP Packet payload. A TXP PW-Timing Connection is only required if a TXP RTP Timestamp is included in the TXP packets.

The S132 appends a header to the payload of each TXP TDMoP Packet as it is transmitted. The header is programmed using a TXP Header Descriptor that is stored in a block of memory at EMI.BMCR1.TXHSO (1 TXP Header Descriptor per Bundle). A TXP PW-Timing Connection is enabled when the TXP Header Descriptor for a Bundle is programmed to insert a TXP RTP Timestamp (TXRE field = 1; see “TXP SAT/CES and HDLC PW Packet Generation” section). Any number of TXP Bundles can be programmed to include an RTP Header.

Figure 9-9. TXP PW-Timing Environment



In the TXP direction, to conform to the Clock Recovery technique that is used at the far end PW end point, the S132 allows the RTP Header to be optionally enabled with a Differential Timestamp or Absolute Timestamp, independent of the RXP RTP settings. Pn.PRCR4.TSGMS selects whether Differential or Absolute Timestamps are inserted when the RTP Header has been enabled in the TXP Header Descriptor.

RTP Differential Timestamp values are generated using the CMNCLK input and 3 coefficients that are programmed in the Pn.PRCR4.TSGMC, Pn.PRCR5.TSGN1C and Pn.PRCR5.TSGN0C registers (programmed per TDM Port).

When the RTP Absolute Timestamp is enabled, the Absolute Timestamp values are incremented according to the receive TDM Port timing (Pn.PRCR2.RSS selects the receive TDM Port timing as either RCLKn or TCLKOn).

9.2.1.2 TDM Port - One Clock and Two Clock Modes

Each TDM Port can be independently programmed to support "One Clock" or "Two Clock" operation. In the "One Clock" mode, the transmit and receive directions are both timed relative to either TCLKOn for TDM Ports that have line rates that are synchronized to a local system clock (System Timed) or relative to RCLKn for TDM Ports that are programmed to be "Loop Timed".

In the "Two Clock" mode RCLK is used to time receive data and TCLKO is used to time transmit data allowing the line rates and/or clock phases of the TXP and RXP directions to be different. This supports the most generalized case for asynchronous transmit and receive timing. Table 9-1 identifies how to select between these modes.

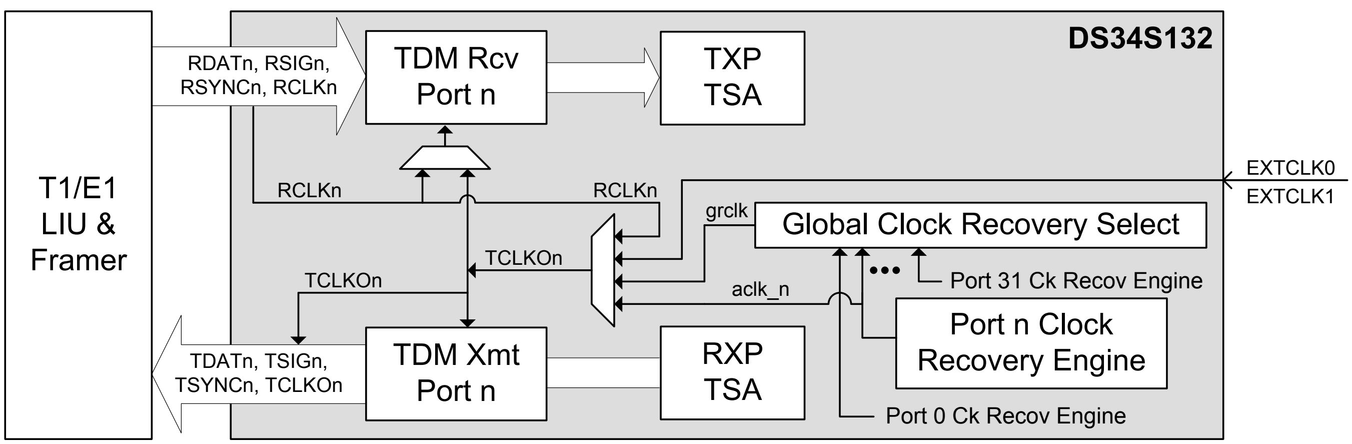
Table 9-1. One-Clock and Two-Clock Mode settings

Mode	Pn.PTCR2.TSS	Pn.PRCR2.RSS
One Clock Mode using RCLK (Loop Timed)	0	0
One Clock Mode using TCLKO (System Timed)	1, 2, 4 or 5	1
Two Clock Mode (independent receive and transmit timing)	1, 2, 4 or 5	0

9.2.2 TDM Port Interface

Each TDM Port supports independent transmit and receive NRZ data, clock, sync pulse and signaling pins. Figure 9-10 provides a high level view of the interconnections to TDM Port "n".

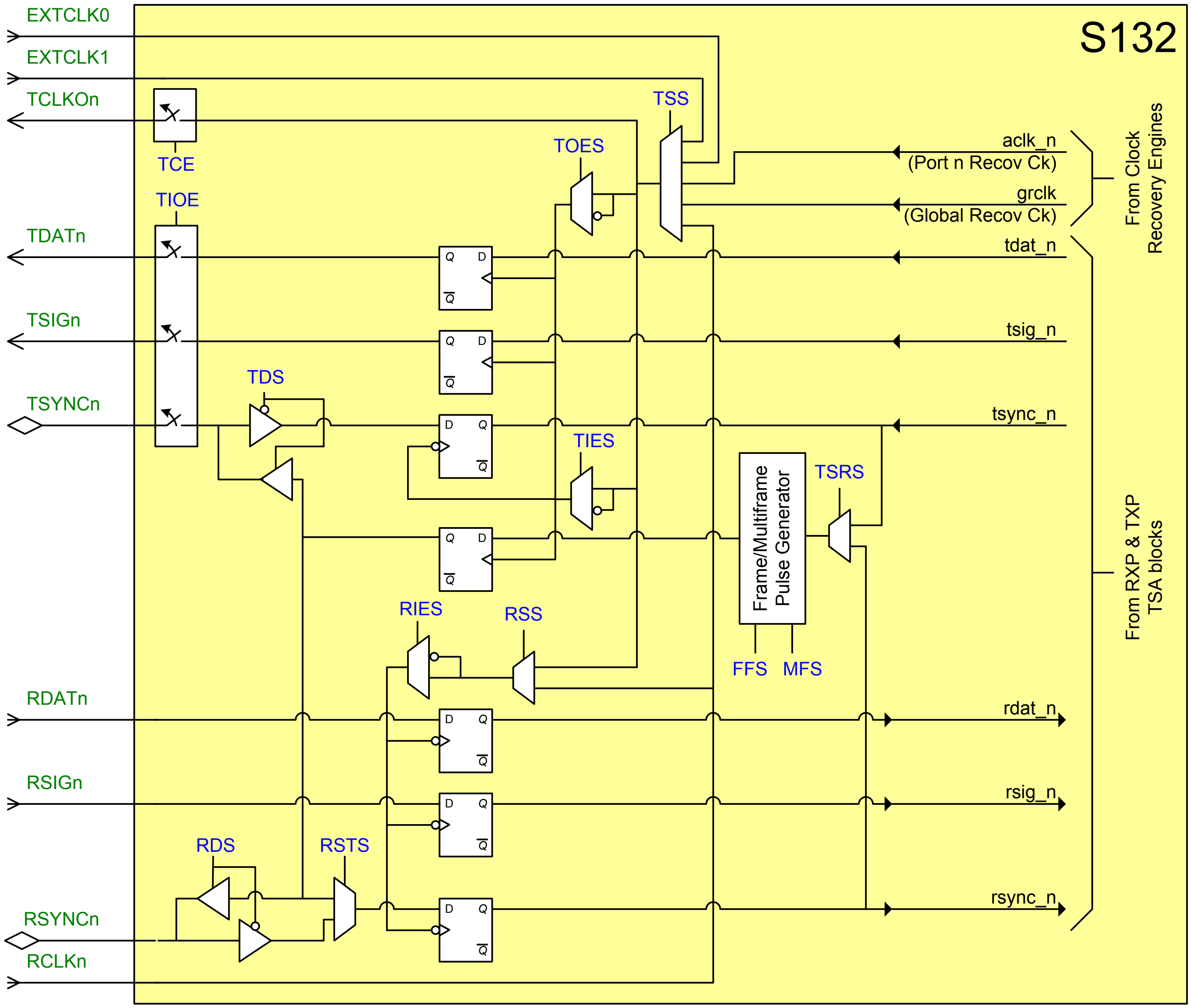
Figure 9-10. TDM Port #1 Environment



When configured for Structured (CES), the RSYNCn/TSYNCn signals are used to identify the T1/E1 frame synchronization, CAS and Timeslot positions. For Unstructured (SAT), the RSYNCn/TSYNCn signals are ignored and the entire TDM Port bandwidth is transported in the TDM-over-Packet payload without regard to framing.

The TSYNC and RSYNC signals can be programmed to be input or output signals, although they are portrayed in this diagram as unidirectional. Figure 9-11 provides a more detailed view of the TDM Port Interface.

Figure 9-11. Logic Detail for a Single TDM Port Interface



9.2.2.1 TDM Port Transmit Interface

The T1/E1 Transmit interface is controlled using the Pn.PTCR2 register to program the following functions:

- TIOE: Enable/disable the TDATn, TSIGn and TSYNCn signals
- TCE: Enable/disable TCLKOn
- TSRS: Select the transmit framing to be synchronized to TSYNCn or RSYNCn
- TDS: Select TSYNCn direction to be input or output
- TOES: Select TDATn, TSIGn and TSYNCn timed to the positive or negative TCLKOn edge
- TSS: Select TCLKOn clock source
- DOSOT: Enable CAS Signaling to be overwritten in the CAS “robbed-bit” positions on TDAT

Table 9-2. TDM Port TCLKOn Clock Source Selection

TSS	TCLKOn Clock Source	Description	Notes
0	RCLKn	Loop timed	Received Clock from LIU/Framer for TDM Port n
1	aclk_n	Port n Recovered Clock	Recovered clock from RXP PW packet stream
2	grclk	Global Recovered Clock	Selects 1 of 32 aclk_n using G.GCR.GRCSS
4	EXTCLK[1]	2nd External Clock Input	E.g. 2.048 MHz reference (or 1.544 MHz)
5	EXTCLK[0]	1st External Clock Input	E.g. 1.544 MHz reference (or 2.048 MHz)

9.2.2.2 TDM Port Receive Interface

The T1/E1 Receive interface is controlled using the Pn.PRCR1 and Pn.PRCR2 registers to program the following functions (more of the PRCR1 functions are described in the sub-sections that follow):

- RSTS: Select the receive framing to be synchronized to RSYNCn or internal transmit framing
- RDS: Select RSYNCn direction to be input or output
- RIES: Select RDATn, RSIGn and RSYNCn timed to the positive or negative RCLKn edge
- RSS: Select clock source to be RCLKn or TCLKOn
- CS: Select RDAT or RSIG for TXP direction CAS Signaling interface

9.2.3 TDM Port Structure & Frame Formats

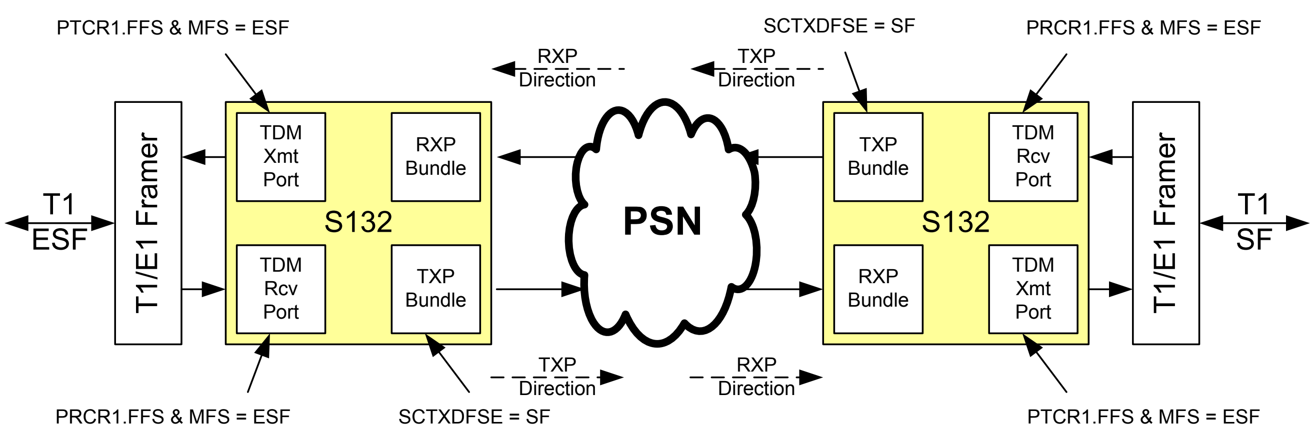
The TDM Ports support the Structured (with Framing) and Unstructured (without Framing) Formats. The Structured Format is used by T1/E1 CES applications. The Unstructured Format is used by T1/E1 SAT and non-T1/E1 SAT applications. The SAT/CES Format for each TDM Port is programmed using Pn.PTCR1.SFS and Pn.PRCR1.SFS. The programmed SAT/CES Format for all RXP and TXP Bundles (B.BCDR4.RXBTS and B.BCDR3.TXBTS) must be the same as the programmed SAT/CES Format for the TDM Port that they are assigned to.

The Structured Format (CES) is programmed to T1 or E1 Framing using Pn.PRCR1.FFS and Pn.PTCR1.FFS. The Multi-frame formats that are supported in the Structured Format (CES) are: no multi-frame, T1 SF, T1 ESF and E1 (selected using Pn.PRCR1.MFS and Pn.PTCR1.MFS). For CES with MFS = "no multi-frame" the RSYNC/TSYNC pulse period is ~125 us ("193 bits/frame" for T1 or "256 bits/frame" period for E1) and CAS is not supported. For the remaining CES settings the RSYNC/TSYNC periods are based on a "12 x 193 bit", "24 x 193 bit" or "16 x 256 bit" period (for SF, ESF or E1 respectively) and CAS Signaling is included.

SFS and MFS should be set to be the same as that of the "local" external transceiver (e.g. T1/E1 Framer). The MFS setting is a multi-frame setting to enable the CAS functions of the S132. The MFS setting may differ from the external transceiver where the multi-frame format setting determines both the T1/E1 framing pattern and the CAS Signaling format. For multi-frame, non-CAS applications the S132 MFS should be set for MFS = "no multi-frame" (meaning no CAS multi-frame) and the external transceiver should be set to T1-SF, T1-ESF or E1. This will disable the S132 CAS Signaling functions and the frame synchronization will only be used to frame align the Timeslots.

In most applications the T1/E1 format at both ends of a PW are the same (e.g. T1 SF to T1 SF). Some unusual T1 CAS applications may prefer to translate one T1 CAS format to the other (e.g. translate T1 SF CAS to T1 ESF CAS). This function is a unidirectional S132 function that is implemented in the TXP direction using the TXP Bundle Payload Multi-frame Format setting (B.BCDR1.SCTXDFSE; the RXP direction does not support translation). The SCTXDFSE setting should always match the T1 CAS Format of the far end T1 PW termination end point (this setting is not used for E1 and T1 non-CAS applications). For T1 CAS applications the SCTXDFSE setting can differ from the MFS settings to provide a T1 CAS Multi-frame Format translation (or be the same for no translation). An example T1 ESF CAS to SF CAS Translation is depicted in Figure 9-12. More CAS translation information is provided in the "TDM CAS to Packet CAS Translation" section.

Figure 9-12. T1 ESF CAS to SF CAS Translation Example



Internally, the data for SAT/CES Bundles is processed using data that is stored in short term, Staging Buffers. The buffers are filled and then forwarded. Each Staging Buffer is divided into fragments (blocks) of data. One Fragment stores the SAT/CES data for a 125 us period (TDAT or RDAT data). Pn.PRCR1.BPF and Pn.PTCR1.BPF specify the Fragment byte depth (e.g. 24 bytes for T1). Pn.PRCR1.BFD and Pn.PTCR1.BFD specify how many Fragments are used by each Staging Buffer (4 Fragments will store data for a 4 x 125 us = 500 us period).

BPF should be set to the number of bytes exchanged on the TDAT/RDAT interface in a 125 us period (e.g. for T1: 17 hex for "24 bytes"; for E1: 1F hex for "32 bytes"). For applications where TDAT and RDAT are used to support a slower, non-T1/E1 interface, the BPF can be set to any integer value to represent the TDM Port data rate (data received in a 125 us period; e.g. BPF = 1 for 64 Kb/s).

The BFD setting enables a compromise between the processing latency and the total number of Bundles supported by an S132. Smaller BFD settings enable a smaller processing latency (smaller wait period to fill the Staging Buffer), but with a smaller maximum number of Bundles. To function properly, the BFD value must also be set so that the data stored in the Staging Buffer cannot exceed the smallest Bundle payload size associated with that TDM Port (i.e. the number of bytes represented by BPF * BFD must be ≤ the number of bytes represented by B.BCDR1.PMS for all Bundles assigned to that TDM Port). Table 9-3 describes the BFD settings.

Table 9-3. TDM Port BFD Settings

BFD value	Staging Buffer Depth	Staging Buffer Latency	Maximum # of Bundles
0	TDM Port data path disabled	-	-
1	1 Fragment	125 us	64
2	2 Fragment	250 us	128
3	4 Fragment	500 us	256

For CES applications, the BFD and PMS settings can be directly compared since both are essentially specified in frames (BFD ≤ PMS; for CES applications the number of bytes stored by the Staging Buffer Fragment is equal to the number of bytes in one CES Frame, or 1 Fragment = 1 Frame). As an example, if PMS = 3 (3 frames per packet payload), then BFD should be set to 10b or 01b (1 or 2 fragments). For SAT applications the PMS setting is specified in bytes (instead of frames) and the TXP/RXP Bundle packets are programmed to carry a payload size that is not related to a frame size ("frames" are not applicable to the SAT/Unstructured application). For SAT applications the following "BFD to PMS" comparison can be used:

BFD (in Fragments) x BPF (in bytes per Fragment) ≤ PMS (in bytes)

In SAT applications, the S132 supports T1/E1 line rates and slower, non-T1/E1 rates. For all SAT applications, the Pn.PRCR1.SPL register be programmed to indicate how many bytes are included in each RXP/TXP Bundle payload. The TDM Port SPL value should be set to the same value as the Bundle PMS.

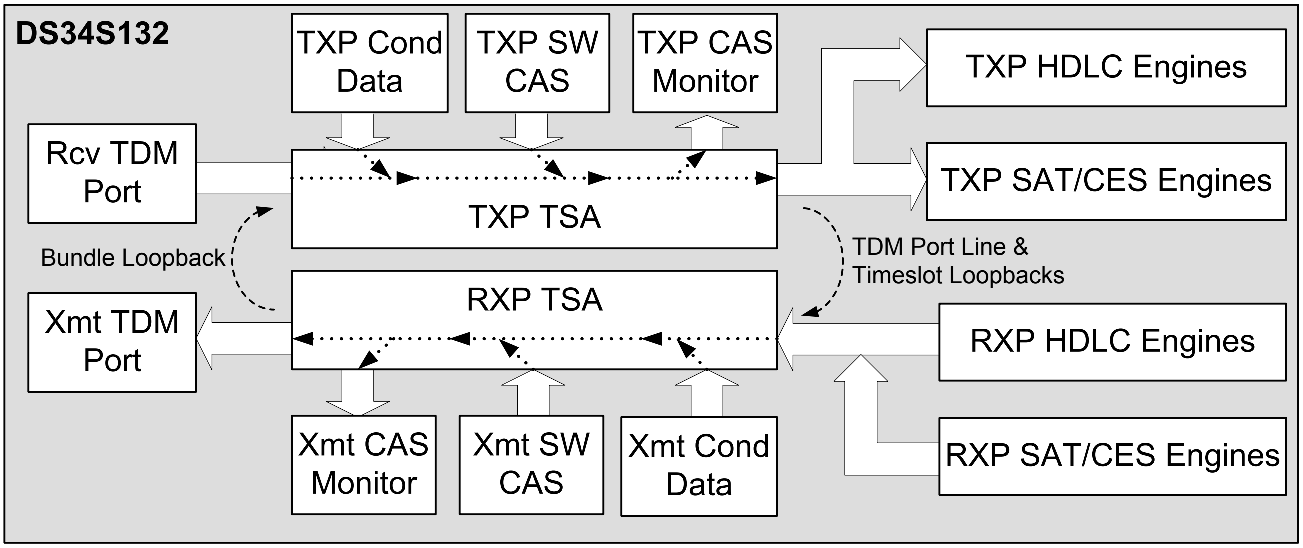
For SAT (Unstructured), non-T1/E1 applications (e.g. V.35), the TDM Port should use a line rate that is approximately equal to an integer multiple of 64 Kb/s. This might be referred to as an "Unstructured Nx64" signal. In this document it is called a "non-T1/E1" signal. Unstructured (SAT) signals usually are asynchronous signals. The term "Nx64" can also refer to a "Structured Nx64" signal that is synchronized to the public network and can be carried by a T1/E1 for transporting and switching in the public network (e.g. "Fractional T1/E1" and ISDN signals). A "Structured Nx64" signal is carried by a CES PW (the S132 only supports "Structured Nx64" with T1/E1 line rate TDM Ports).

For the best latency performance, each TDM Port BFD should be set to the lowest possible value allowed for the maximum number of Bundles that will be supported by the S132. With a selected BFD value, all Bundle PMS values associated with that TDM Port cannot be smaller than BFD. As an example, if it is necessary to support a Bundle with a PMS = 1 (1 frame per packet or one packet every 125 us) then no more than 64 Bundles can be supported by the S132 (the standards only require a maximum packet rate of one packet every 1 ms).

9.2.4 Timeslot Assignment Block

For T1/E1 applications, the S132 includes a Timeslot Assignment Block with the ability to monitor outgoing CAS and control outgoing SW CAS Conditioning, Data Conditioning, and Loopback functions (depicted in Figure 9-13).

Figure 9-13. TSA Block Environment



One Timeslot Assigner circuit is provided for each TDM Port and in each direction so that any combination of timeslots from a TDM Port can be assigned to a Bundle. The ordering of the data within a packet always follows the "chronological" order on the T1/E1 line. For example if Timeslots 0, 7, 13 and 17 are assigned to Bundle A, the data in the packet payload section will be 0, 7, 13, 17, 0, 7, 13, 17 and so on. The timeslot order cannot be programmed to provide an ordering like 7, 0, 17, 13.

For Structured TDM data streams, the association between a Bundle to its TDM Port number and T1/E1 Timeslot positions is programmed using B.BCDR4.PNS, B.BCDR2.ATSS and TSAn.m. To function properly, every Bundle must be assigned at least one TDM Port Timeslot and each TDM Port Timeslot cannot be assigned to more than one Bundle. Timeslots can be ignored by not assigning them to Bundles.

Unstructured TDM data streams do not provide a means to byte-align to the data stream. The Timeslots are viewed by the S132 as 8–bit time periods that are not synchronized to a framing pattern, but timed to a 125 us time period. Unstructured Bundles use the entire TDM Port bandwidth. The first Timeslot (TS0) of the TDM Port must be assigned to the Unstructured Bundle using the B.BCDR4.PNS, B.BCDR2.ATSS and TSAn.m registers. The first Timeslot is the only Timeslot that is assigned to that Bundle and no other Timeslots on that TDM Port should be assigned/enabled. Although the T1 line rate includes a non-integer number of bytes within a 125 us period (193 bits), there are no register settings to include/assign the 193rd bit. An Unstructured TDM Port that is programmed with Pn.PTCR1.BPF = Pn.PRCR1.BPF = 0x17 can support both 192 and 193 bits per 125 us time period.

Although the packets for Clock Only Bundles do not include packet payload (no Timeslot data), the S132 requires that Clock Only Bundles must also be assigned a fraction/portion of the TDM Port bandwidth (assigned 1 or more Timeslots). Assigning one Timeslot to a Clock Only Bundle allocates enough processing time (from the TDM Port) for the S132 to perform the Clock Only Bundle functions. A Timeslot that is assigned to a Clock Only Bundle cannot be assigned to any other Bundle even though the payload data is not used (the Timeslot processing time period can only be used by one Bundle). For E1-CES Timeslot 0 can be used for a Clock Only Bundle since the Framing Timeslot is not normally carried in the PW packets. For T1-CES, T1-SAT and E1-SAT, two TDM Ports (out of the 32 TDM Ports) can be connected in "parallel" so that one TDM Port is used for the Clock Only Bundles and the other TDM Port is used for the Bundle with payload data. The use of Clock Only Bundles is optional to provide a technique to reduce the packet latency through the use of smaller packets with high priority scheduling.

The outgoing CAS codes can be monitored in both directions. The Xmt CAS codes (RXP direction) can be read using Pn.PRSR1 – Pn.PRSR4. The TXP CAS codes can be read using Pn.PTSR1 – Pn.PTSR4. The receive TDM Port CAS codes can be sourced from RSIG or RDAT (Pn.PRCR1.CS). The CAS codes can be monitored by polling the Monitor registers (PRSRx and PTSRx) or by using an interrupt hierarchy that reports when a CAS change has been detected (G.GSR2 and G.GSR3). The interrupt method is also described in the "Interrupt Hierarchy" section.

In the RXP direction, when CAS Signaling is enabled on a Bundle (B.BCDR4.RXBTS = 2), the CAS codes received from RXP packets are forwarded to the TDM Port and transmitted in the proper Timeslot CAS code positions. When RXP CAS codes are received they are first stored in a Jitter Buffer along with the CES Bundle payload data to smooth out the irregular (bursty) receive packet rate. If the RXP packet stream is blocked (e.g. for a fault), the S132 will continue to send CAS codes until the Jitter Buffer is empty. When the Jitter Buffer empties, the S132 can be programmed to continue sending the last stored CAS code or to send the programmed Xmt SW CAS

(B.BCDR1.SCRXBCSS; Xmt SW CAS will only be sent when the Jitter Buffer is empty). These two functions are programmed on a per-Bundle basis. The Xmt SW CAS codes are programmed using RXSCn.CR1 - RXSCn.CR4 (programmed on a per-Timeslot basis).

In the TXP direction, when CAS Signaling is enabled on a Bundle (B.BCDR3.TXBTS = 2), the S132 can be programmed to "pass through" the incoming receive TDM Port CAS Signaling or to insert a programmed TXP SW CAS code (B.BCDR1.SCTXBCSS). The forced TXP SW CAS codes can be used during fault conditions (e.g. "Loss of Signal") or to force a known CAS code for idle timeslots. These two functions are programmed on a per-Bundle basis. The TXP SW CAS codes (transmitted at the Ethernet Port) are programmed using TXSCn.CR1 - TXSCn.CR4 (programmed on a per-Timeslot basis).

The S132 provides the ability to force Xmt Conditioning Data in the outgoing data stream at the transmit TDM Port (programmed on a per Bundle basis). For SAT/CES Bundles, RXP Payload that is received from the Ethernet Port is stored in a Jitter Buffer and later transmitted at the TDM Port as data is needed. For CES Payload Connections, if the Jitter Buffer runs out of data the S132 continues transmitting data at the TDM Port using either the "Last Value" or using one of eight programmed Xmt Conditioning Data values (B.BCDR4.SCLVI). For SAT Payload Bundles, the Unstructured format does not identify byte boundaries and the TDM Port should be programmed to immediately transmit Data Conditioning (SCLVI = 0). For HDLC Bundles, when the S132 runs out of HDLC data, the TDM Port transmits the selected Xmt Conditioning Data (e.g. HDLC Idle Flags). The eight Xmt Conditioning Data values are programmed using G.TCCR1 and G.TCCR2. The Conditioning Data is independently selected for each Bundle using the B.BCDR4.RXCOS register.

For SAT/CES Bundles, the S132 can force TXP Conditioning Data in the outgoing TXP Packets. This may used during incoming T1/E1 fault conditions or to send a forced PCM value like "Idle". TXP Conditioning Data can be enabled on a per Bundle basis using one of eight programmed TXP Conditioning Data values. Eight TXP Conditioning Data values can be programmed using G.ECCR1 and G.ECCR2. The Conditioning Data value is independently selected for each Bundle using BCDR1.SCTXCOS and independently enabled using BCDR1.SCTXCE. For HDLC Bundles, when the S132 runs out of received/stored HDLC packets the S132 stops transmitting TXP packets (TXP Conditioning Data is not used for HDLC Bundles).

Special considerations:

For systems that require the legacy CAS "Freeze Signaling" function (TXP and RXP directions), the Framer that interfaces to the S132 TDM Port should implement the "Freeze Signaling" function so that proper CAS codes are forwarded during fault conditions. The legacy "Freeze Signaling" function includes a CAS code de-bounce function that is not implemented in the S132 (new CAS codes are not forwarded until the CAS code is received 3 times).

For systems that need to dynamically insert the transmit TDM Port CAS codes (e.g. to continuously translate incoming RXP CAS codes into different outgoing CAS codes) the "dynamic insertion" should be implemented in the external T1/E1 Framer. The S132 CAS functions do not allow the CPU to both monitor the incoming CAS codes from RXP packets and replace the received CAS codes with Xmt SW CAS codes (the S132 function monitors the CAS output, not the input).

In the transmit TDM Port (RXP) direction, when a Timeslot and/or its CAS code is "unspecified" (e.g. for unassigned Timeslots), the data that is transmitted toward the T1/E1 Framer uses default values. The G.TCCR1.TCOA register value is transmitted for "Unspecified" Timeslot data. "Unspecified" Timeslot CAS positions are filled with the RXSCn.CTSx register value ("x" is equal to the Timeslot number).

In the transmit TDM Port (RXP) direction, when CAS is enabled on a TDM Port, CAS data is inserted in all Timeslots (24 for T1, 30 for E1) regardless of whether all Timeslots are intended to include CAS. For T1 applications that use CAS in some Timeslots and "no CAS" in other Timeslots, TSIG should be used to transmit the CAS codes to the external Framer, the S132 "Overwrite TDAT with CAS" function should be disabled (Pn.PTCR2.DOSOT) and the external Framer should be programmed to insert the CAS codes (from TSIG) in the appropriate Timeslots (the "Overwrite TDAT with CAS" function overwrites "with CAS" and "no CAS" Timeslots).

When the "CAS Change Interrupt" function for a TDM Port is enabled (G.GSR2 and G.GSR3), even non-CAS Timeslots can generate an interrupt since all Timeslots are monitored. If the T1/E1 includes non-CAS Timeslots, frequent interrupts may occur (once per multi-frame) because the data in the 8^{th} bit position (CAS position) may be constantly changing.

9.2.4.1 TDM CAS to Packet CAS Translation

When "pass through" CAS Signaling is enabled, the S132 translates the T1/E1 CAS timing at the TDM Port into PW CAS Sub-channel signaling used by the Bundles. In the TXP and RXP directions the S132 stores and forwards 16 frames of received CAS Signaling for the E1 format and 24 frames of received CAS Signaling for the T1 SF and T1

ESF formats. For the T1 SF format this means that two successive 2-bit CAS Codes from 2 consecutive SF frames are stored.

For T1 ESF and E1, the 4-bit ABCD CAS-codes received at a T1/E1 Port are stored and forwarded unmodified. When the T1 SF format is used the S132 "extends" the T1-SF, 2-bit AB CAS-codes into a 4-bit CAS field since the PW Sub-channel CAS Signaling requires a 4-bit field (regardless of whether it is T1-ESF, T1-SF or E1). Two dummy bits are appended to and removed from the T1-SF, AB CAS-code to fill the packet's 4-bit CAS field.

Most applications will use the same T1/E1 framing on both ends of the PW (e.g. T1-ESF to T1-ESF). For T1 applications, the S132 can be programmed to provide a translation between the 2-bit SF CAS Codes and 4-bit ESF CAS Codes. This is a unidirectional function that can be enabled in the TXP direction (see Figure 9-12, "ESF to SF Translation Example"). When translating 4-bit, ESF, "ABCD" CAS into 2-bit, SF, "AB" CAS the "CD" bits are discarded. When translating 2-bit, SF, "AB" CAS into 4-bit, ESF, "ABCD" CAS the S132 generates an ABCD code by appending a programmed, 2-bit "CD" value to the received, 2-bit "AB" code (the "CD" insertion bits are programmed using Pn.PRCR1.CBVSE and Pn.PRCR1.DBVSE).

Table 9-4 describes how to program each of the various translation functions and how the 4-bit fields are interpreted when using RSIG and TSIG. The table should be read from left to right. The "TXP Direction", "SCTXDFSE & PRCR1.MFS Format" column identifies each programmed translation function (e.g. ESF to SF). For example, for "ESF to SF", Pn.PRCR1.FFS and Pn.PRCR1.MFS are programmed for ESF; B.BCDR1.SCTXDFSE is programmed for SF; Pn.PTCR1.FFS and Pn.PTCR1.MFS are programmed for SF.

The RSIG column includes two sub-columns that provide an example of CAS data that might be received in frames 1 - 24. The "TXP Packet Out" columns indicate how the CAS codes received from the RSIG pin would be transmitted in the TXP Packets. The "RXP Packet In" column is identical to the "TXP Packet Out" column to represent the process on the opposite end of the PW (as though the TXP Out is connected to the RXP In). The TSIG column indicates how the CAS code (received from the RXP Packet) would be transmitted on the TSIG pin.

The "Format" settings determine whether CAS is sent once every 12 T1 frames or once every 24 T1 frames. SCTXDFSE specifies the RSIG frame rate. PRCR1.MFS specifies the "TXP Packet Out" frame rate. PTCR1.MFS specifies the TSIG frame rate.

The protocols for the RSIG and TSIG pins always include a 4-bit field for the CAS Code (even for the SF format). In the SF format only 2-bits of the 4-bit field are regarded as valid by the protocol. In the "RSIG" column, "XY" is used to indicate that the values of the two "extra" bits are unknown. An external T1 SF Framer will ignore the last two TSIG dummy bits. The "SF to SF", "ESF to ESF" and "E1 to E1" translation functions are included in the table to show how the CAS codes are handled for all combinations.

Table 9-4. CAS Translation using RSIG and TSIG

TXP Direction				RXP Direction			
SCTXDFSE to PRCR1.MFS	TDM Port RSIG		TXP Packet Out	RXP Packet In	TDM Port TSIG		PTCR1.MFS
Format	Frm 1-12	Frm 13-24	Frm 1-24	Frm 1-24	Frm 1-12	Frm 13-24	Format
ESF to SF	$A_1B_1C_1D_1$		$A_1B_1A_1B_1$	$A_1B_1A_1B_1$	$A_1B_1A_1B_1$	$A_1B_1A_1B_1$	**SF**
SF to ESF	$A_1B_1X_1Y_1$	$A_2B_2X_2Y_2$	A_1B_1CD	A_1B_1CD	A_1B_1CD		**ESF**
SF to SF	$A_1B_1X_1Y_1$	$A_2B_2X_2Y_2$	$A_1B_1A_2B_2$	$A_1B_1A_2B_2$	$A_1B_1A_1B_1$	$A_2B_2A_2B_2$	**SF**
ESF to ESF	$A_1B_1C_1D_1$		$A_1B_1C_1D_1$	$A_1B_1C_1D_1$	$A_1B_1C_1D_1$		**ESF**
E1 to E1	$A_1B_1C_1D_1$		$A_1B_1C_1D_1$	$A_1B_1C_1D_1$	$A_1B_1C_1D_1$		**E1**

Notes: The "X" and "Y" values mean "any value", these values doesn't matter since these bit positions are ignored."

Table 9-5 describes the same information for applications that use RDAT and TDAT instead of RSIG and TSIG. For T1-SF, TDAT and RDAT only exchange a 2-bit CAS field for each 12- frame, SF multi-frame.

Table 9-5. CAS Translation using RDAT and TDAT

TXP Direction				RXP Direction			
SCTXDFSE to PRCR1.MFS	TDM Port RDAT		TXP Packet Out	RXP Packet In	TDM Port TDAT		PTCR1.MFS
Format	Frm 1-12	Frm 13-24	Frm 1-12	Frm 1-12	Frm 1-12	Frm 13-24	Format
ESF to SF	$A_1B_1C_1D_1$		$A_1B_1A_1B_1$	$A_1B_1A_1B_1$	A_1B_1	A_1B_1	**SF**
SF to ESF	A_1B_1	A_2B_2	A_1B_1CD	A_1B_1CD	A_1B_1CD		**ESF**
SF to SF	A_1B_1	A_2B_2	$A_1B_1A_2B_2$	$A_1B_1A_2B_2$	A_1B_1	A_2B_2	**SF**
ESF to ESF	$A_1B_1C_1D_1$		$A_1B_1C_1D_1$	$A_1B_1C_1D_1$	$A_1B_1C_1D_1$		**ESF**
E1 to E1	$A_1B_1C_1D_1$		$A_1B_1C_1D_1$	$A_1B_1C_1D_1$	$A_1B_1C_1D_1$		**E1**

Special considerations:

Each system should be analyzed to determine whether the 2-bit to 4-bit translation function is appropriate. The method of appending a fixed, programmed "CD" value in one direction (SF to ESF) and discarding the "CD" bits in the other direction (ESF to SF) may not be valid.

In applications where T1-SF CAS Signaling is carried in RXP packets, because the S132 stores 24 frames of T1-SF CAS Signaling, it is possible (during a loss of RXP packet condition) that a constantly alternating 2-bit CAS code ($A_1B_1 \neq A_2B_2$) is transmitted at the TDM Port if the last 2 received AB CAS-codes are 2 different values, the Jitter Buffer underruns (e.g. RXP packet fault) and the CAS "Last Value" function is enabled (B.BCDR1.SCRXBCSS). This can occur, for example, if the far end of the PW transmitted an On-hook to Off-hook CAS Code transition in the last received RXP packet. If this condition occurs ($A_1B_1 \neq A_2B_2$), the TDM Port transmitted CAS codes will alternate between these two values every 12 frames. Each system should be evaluated to determine whether this condition is acceptable (an external T1 Framer with CAS debounce function should filter out the alternating pattern).

The support of CAS Signaling in a system that allows the use of multiple Nx64 PWs with a single T1/E1 may require the system to be compliant with the defect and alarm requirements of a Digital Cross-Connect. When a T1/E1 is divided into multiple segment/paths, the segments are unable to use the T1/E1 framing as an indication of the state of the connection. For example if 2 PWs are merged into a single T1/E1, a far end T1/E1 fault in the RXP direction of PW #1 (e.g. LOS) cannot be directly communicated over the local, T1/E1 transmit port since that would imply that PW #2 is experiencing the same fault (i.e. the local T1/E1 transmit Port cannot forward the T1/E1-AIS, Alarm Indication Signal, for PW #1 without indicating the same for PW #2; some systems allow DS0-AIS). Each system should be analyzed to determine whether Digital Cross-Connect defect and alarm conditioning is required. If these functions are required, they should be implemented external to the S132 (e.g. in the T1/E1 Framer).

9.2.4.2 TSA Block Loopbacks

The TSA Timeslots can be programmed to loopback data using a Bundle Loopback, TDM Port Line Loopback or TDM Port Timeslot Loopback (see Figure 9-13). Any number of Timeslots, Bundles and/or TDM Ports can be in Loopback at the same time.

The Bundle Loopback sends packet payload data that has been received for an RXP Bundle back toward the Ethernet Port in TXP packets for the same Bundle. When the Bundle Loopback is enabled (Pn.PTCR3.RXTXTSL), the RXP packet payload data is processed as though it will be transmitted at a TDM Port. But when the payload data reaches the TSA block the data is looped back in the TXP direction and processed as though the data was received from the TDM port. To work properly all Timeslots associated with the Bundle should be programmed into the Bundle Loopback state.

The TDM Port Line Loopback and TDM Port Timeslot Loopback send payload data from the receive TDM Port back toward the transmit TDM Port without any packet processing functions. The TDM Port Timeslot Loopback (Pn.PTCR3.PRPTTSL) allows loopback selection on a per-Timeslot basis while the TDM Port Line Loopback (Pn.PTCR2.PRPTLL) provides a loopback of all Receive TDM Port Data.

9.2.5 TDM Port Data Processing Engines

A TDM Port is assigned to a Bundle using B.BCDR4.PNS. The format of the TDM Port data streams can be Unstructured, Structured T1/E1 without CAS or Structured T1/E1 with CAS and are processed using 3 engine types: HDLC, SAT/CES and Clock Recovery. The combination of B.BCDR1.PMT (Payload Engine Type), B.BCDR3.TXBTS (TXP Bundle Structure), B.BCDR4.RXBTS (RXP Bundle Structure) and B.BCDR4.PCRE (Clock Recovery Enable) select the payload format and engine type for each Bundle. Enabling a particular Engine Type for a Bundle is equivalent to enabling a "Connection" for that Engine (e.g. enabling SAT/CES Engine = enable SAT/CES Connection). B.BCDR1.RXBDS further supplements these selections by providing the ability to instead forward the packets for a Bundle to the CPU (for debug; CPU Debug RXP PW Bundle) or to discard the packet payload for Clock Only Bundles (to reduce the S132 payload processing functions). The following types of Bundles can be programmed using these registers. The "Register Definition" and the "Register Guide" sections provide more information on how to set these registers for each Bundle type.

SAT	HDLC for Unstructured TDM Port	SAT Clock Only
Nx64 CES without CAS	Structured Nx64 HDLC	CES Clock Only
Nx64 CES with CAS	Structured 56 Kb/s or 16 Kb/s HDLC	

The SAT and CES Bundles can include an RXP PW-Timing Connection by enabling B.BCDR4.PCRE and/or a TXP PW-Timing Connection by including the RTP Timestamp in the TXP Header Descriptor for that Bundle (see the "TXP SAT/CES and HDLC PW Packet Generation" section). The Clock Recovery functions are described in more detail in the "PW-Timing" section.

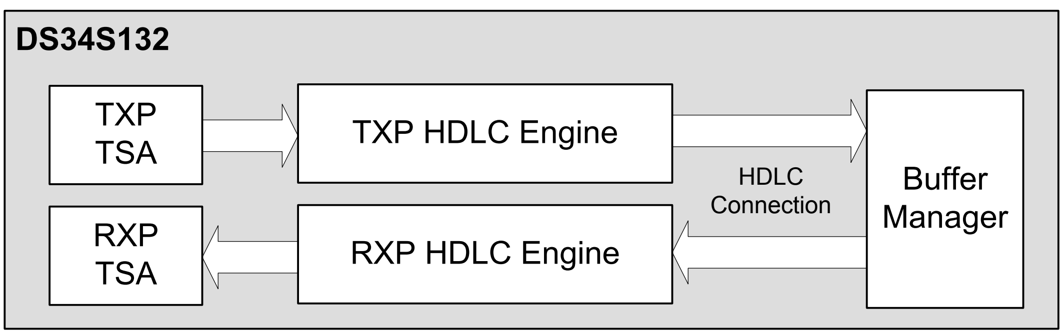
All of the Bundle types can include support for In-band VCCV (In-band VCCV CPU Connection). When a packet is received for a recognized Bundle, the received packet header matches the In-band VCCV Control Word (PC.CR5.VOV and PC.CR5.VOM) and In-band VCCV has been enabled for that Bundle (B.BCDR4.RXCWE and B.BCDR4.RXOICWE) the S132 forwards the In-band VCCV packet to the CPU or discards the packet according to the PC.CR1.DPS7 setting (OAM Packet Discard switch; this switch also affects other OAM types).

In the TXP direction, Receive TDM Port data that is ready for transmission is buffered in one of two priority queues so that the packets can be scheduled according to their importance when congestion occurs. TXP Packets that are buffered in the higher priority queue are processed before TXP Packets in the lower priority queue. For example, Bundles with PW-Timing Connections can be assigned to the higher priority queue. The TXP priority is selected for each TXP Bundle using B.BCDR3.TXBPS.

9.2.5.1 HDLC Engine

The S132 includes 256 HDLC Engines, one each for up to 256 Bundles. Several HDLC Bundle types are supported including Unstructured HDLC (full TDM Port bandwidth), Structured Nx64 Kb/s HDLC, Structured 56 Kb/s or Structured 16 Kb/s. With HDLC Bundles the terms "Unstructured" and "Structured" refer to the format of the TDM Port. These terms do not have any direct relevance to the packet format of an HDLC Bundle. A single Structured TDM Port can support any combination of Structured HDLC Bundles and CES Bundles since each Bundle can be assigned to independent Timeslots on a Structured TDM Port.

Figure 9-14. HDLC Engine Environment



An Unstructured HDLC Bundle uses the entire bandwidth of its assigned TDM Port. The HDLC coding/decoding is performed using the entire data stream without regard for T1/E1 framing or Timeslot positions.

Structured HDLC Bundles can be programmed to use 2-bit, 7-bit or 8-bit HDLC coding (for 16 Kb/s, 56 Kb/s and "Nx64 Kb/s" channels respectively; B.BCDR1.SCTXCOS). The bit-width setting identifies how many bits are used in the assigned Timeslot. For 8-bit, all 8 bits of the timeslot are HDLC coded. For 7-bit coding, only the 7 MSbits are HDLC coded (the LSbit is unused). For 2-bit coding, the two MSbits or two LSbits can be selected for HDLC coding (the remaining 6-bits are unused). Unstructured HDLC Bundles always use 8-bit coding.

The "8-bit" format allows an HDLC Bundle to combine the data from multiple 8-bit Timeslots of a single Structured T1/E1 to support bandwidths like 384 Kb/s (using six 8-bit Timeslots). Any number of 8-bit Timeslots can be combined (up to 24 for Structured T1 or 31 for Structured E1).

Only one 2-bit or 7-bit HDLC coded Timeslot from a Structured T1/E1 can be assigned to an HDLC Bundle.

Each HDLC Engine can be programmed to use MSbit or LSbit first transmission (BCDR1.SCSNRE). This function does not specify which bits of the Timeslot are used (previous paragraphs), but instead specifies whether the MSbit or LSbit of each HDLC coded byte is transmitted first (the byte order is always MSByte first). For example, if LSbit transmission and 8-bit coding are selected, then the LSbit of each byte is transmitted first (in the “bit 8” position of the Timeslot). If instead MSbit transmission and 8-bit coding are selected then the MSbit of each byte is transmitted first (in “bit 8”). Most T1/E1 applications use MSbit first.

Each HDLC Bundle can be programmed to include a 16-bit, 32-bit or “no” FCS (B.BCDR1SCRXBCSS and SCTXBCSS).

In the TXP direction the HDLC Engine receives data from a TDM Port and removes the HDLC encoding (HDLC Flags and HDLC Control Characters). The de-encoded packet data is buffered until a complete packet has been received. After the HDLC FCS has been verified to be correct, the packet is queued for transmission as the payload of a TXP HDLC Bundle packet.

TXP HDLC Bundles can optionally include RTP and Control Word Headers (enabled using the TXP Header Descriptor). For RTP and/or Control Word headers can use Sequence Numbers that are always “zero”, or are constantly incremented by one with each successive packet (B.BCDR4.SCTXCE and B.BCDR1.SCTXDFSE). When incremented Sequence Numbers are used the S132 can be programmed to skip or include the Sequence Number = “zero” value when the Sequence Number reaches roll-over.

In the RXP direction, when the RXP Classifier identifies an error-free packet for an HDLC Bundle, the PW packet header and FCS are removed and the PW packet payload is stored for later processing by the RXP HDLC Engine. The HDLC Engine inserts a Flag (Packet Delimiter = 0x7E) in between each successive RXP Packet to identify the start and stop of each packet. The G.GCR.RXHMFIS register specifies the minimum number of Flags that are inserted in between 2 HDLC packets where RXHMFIS + 1 = minimum number of flags (e.g. RXHMFIS = 0 for 1 flag). When the HDLC Engine no longer has a packet to forward and the minimum number of flags have been transmitted the HDLC engine inserts “Inter-frame Fill” into the outgoing HDLC data stream. The Inter-frame Fill value can be programmed to 0x7E or 0xFF (B.BCDR4.SCLVI).

In the RXP direction the S132 does not provide re-ordering of mis-ordered HDLC packets, so the optional RTP and/or Control Word Sequence Numbers received in packets for RXP HDLC Bundles are ignored.

The B.BCDR1.PMS register is used to define the largest Ethernet packet that is accepted for an RXP HDLC Bundle. Packets with a size greater than PMS are discarded.

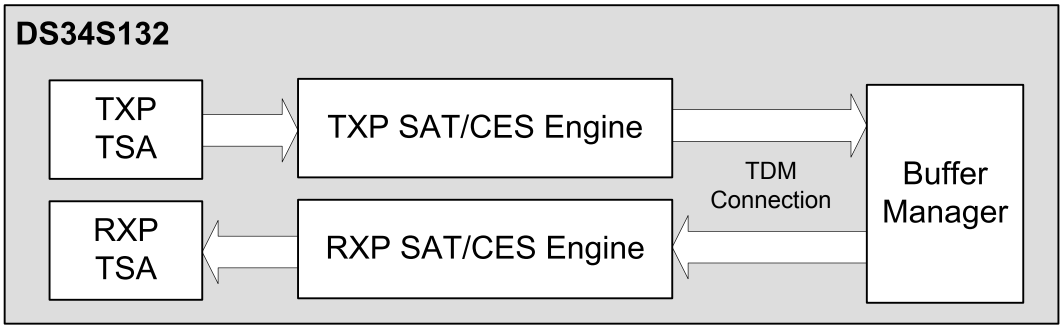
Special Considerations

The S132 does not provide special handling for CAS Signaling when a T1/E1 Port includes an RXP HDLC Bundle. If CAS Signaling is enabled for the T1/E1 Port and if the “overwrite CAS on TDAT” is enabled, CAS values will be written in Timeslot positions assigned to HDLC Bundles. To prevent this, the HDLC 7-bit Sampling format can be used, or else TSIG can be used to provide the CAS values (disable the “Overwrite CAS on TDAT”). In the TXP direction, the RSIG value and the SW TXP CAS functions are ignored by TXP HDLC Bundles.

9.2.5.1.1 SAT/CES Engine

The S132 includes 256 SAT/CES Engines, one for each of the 256 possible SAT/CES Bundles.

Figure 9-15. SAT/CES Engine Environment



In the RXP direction, B.BCDR1.PMS specifies the expected packet payload size for each RXP Bundle (not including the optional CAS bytes). For SAT applications, PMS specifies how many bytes; for CES applications, how many frames. In the TXP direction, the PMS setting determines the amount of data that is included in the payload of each TXP Bundle packet.

For RXP Bundles, the S132 monitors the received Control Word L-bit field. If the RXP Bundle is programmed with B.BCDR1.SCSCFPD = 1 (verify packet size) and the received L-bit = "0" ("PW payload is valid"), the S132 discards the packet if the PW packet payload size does not match the PMS setting. If SCSCFPD = 1, the received L-bit = 1 (packet payload invalid) and B.BCDR1.LBCAI = 1 (conditioning for L-bit = 1) the PMS setting is ignored.

B.BCDR1.SCSNRE selects whether the packets for RXP SAT/CES Bundles are re-ordered when they are received out of order. B.BCDR1.RSNS is used to specify whether the Sequence Number in the Control Word or RTP header is used by this re-ordering function.

A packet is only accepted as a SAT/CES Bundle if the first 4 bits of the Control Word equal 0h. Packet payload data for RXP SAT/CES Bundles is stored in a Jitter Buffer according to its Sequence Number. When a packet for a Bundle is "missing" (Sequence Number not received) or the Jitter Buffer underruns, the S132 replaces the missing data at the transmit TDM Port according to the B.BCDR4.SCLVI setting.

For CES Bundles when B.BCDR4.SCLVI is enabled, the S132 uses the last received byte (Last Value) for each Timeslot of the Bundle to replace the missing data for up to 375 us. After 375 us, the Conditioning Data selected by B.BCDR4.RXCOS is inserted. If SCLVI is disabled, the missing data is immediately replaced by Conditioning Data.

For SAT Bundles, the Unstructured format does not identify byte boundaries so the SCLVI function must be disabled so that Conditioning Data is always used to replace SAT missing data.

9.2.5.2 TDM Port Priority

Each Port can be assigned as "high" or "low" priority, using Pn.PTCR1.DP (TXP direction) and Pn.PRCR1.EP (RXP direction) so that the SAT/CES/HDLC Engines process some TDM Ports before others. In most applications all TDM Ports should be assigned the same priority level.

9.2.5.3 Jitter Buffer Settings

The Jitter Buffer provides a means of transitioning TDM data between the PW and TDM domains (RXP direction). There are 3 fundamental issues when reconstructing a TDM data stream from a stream of packetized data: data content, delay and frequency. The S132 Jitter Buffer settings are complex so it is important to understand the parameters that are affected by these settings.

For TDM services, all 3 issues are important. For example, if voice data is delivered error-free, but with 1 second of delay, then the conversation can be confusing (each person does not know how long to wait to keep from talking over the other person). A voice connection that adds more than 150 ms of delay is considered a poor connection, although in unusual cases, up to 400 ms of delay may be accepted. As another example, for PCM voice switching (e.g. PBX or Class 5 switch), if the reconstructed TDM data is error-free, but the TDM line frequency is not synchronized to the voice switch, the TDM switching process will corrupt the data. A TDM voice connection is not significantly affected by a small amount of data corruption, whereas a computer data connection, generally, depends on almost error-free transmission to minimize the need for re-transmission. All 3 issues are important.

The S132 transmit TDM Port Jitter/Wander performance is affected by the clocking technique that is used. If an external clock is used, then the S132 Jitter/Wander is primarily determined by the Jitter/Wander of the external reference. If an internal Clock Recovery Engine is used, then the Jitter (high frequency variation) is determined from an internal S132 frequency synthesizer that is designed to comply with the TDM Jitter requirements in all Clock Recovery settings and conditions. The Wander (low frequency variation) is determined by how well the Clock Recovery Engine can reconstruct the timing of the incoming packet stream. The performance of the Clock Recovery Wander depends on the maximum excursion and nature of the packet stream PDV, and on the packet transmission error rate (high packet loss may affect the performance). When it is possible, PWs that are used to carry Clock Recovery information should be assigned a high priority on the originating PW end point to minimize the PDV. B.BCDR3.TXBPS can be used to select S132 internal high priority TXP processing and the TXP VLAN Header P-bits (programmed in the TXP Header Descriptor) can be used to indicate high priority to the network.

The S132 Jitter Buffer smoothes the irregular (bursty) RXP packet rate. The Jitter Buffer stores and then supplies data as needed according to the transmit TDM Port timing. Because the TDM Port line rate is nearly constant (with only small variations), the TDM Port cannot significantly slow down or speed up to compensate for too much or too little stored data. To compensate for the irregular packet rate (burstiness), an infinite depth Jitter Buffer would insure that data is never lost/discarded, but would also potentially store so much data that the forwarding delay is too long (potentially making a conversation impossible). A very shallow Jitter Buffer would minimize the delay, but may not store enough data to prevent a data under-run event (missing data is replaced with dummy data). Each PW system must determine how to balance these conditions (discard, delay and under-run).

The delay of data at the input of the Jitter Buffer is caused by fixed and PDV (variable) delay parameters according to the equation below.

Maximum Jitter Buffer Input Delay = PCT + fixed transmission and circuit processing delay + Total PDV

PCT (Packet Creation Time) is a fixed delay that is equal to the amount of time that it takes to receive enough data from a TDM Port to fill the Payload section of a PW Bundle. The B.BCDR1.PMS (Packet Payload Size) setting is programmed according to the desired PCT value using the equations below. For example it may be desirable for a CES payload to carry 8 frames of data (from the equations below, PCT = 1 ms and PMS = 8).

T1/E1 Nx64 CES: B.BCDR1.PMS = "# T1/E1 Frames per Packet Payload" = PCT ÷ 125us

T1 SAT: B.BCDR1.PMS = "# bytes per Packet Payload" = PCT ÷ 5.2us

E1 SAT: B.BCDR1.PMS = "# bytes per Packet Payload" = PCT ÷ 3.9us

"Slow rate" SAT: B.BCDR1.PMS = "# bytes per Packet Payload" = PCT ÷ ($8/f_{TDM}$)
(where "f_{TDM}" is equal to the data bit rate at the TDM Port).

The fixed transmission delay will differ for each PW connection according to the distance between the end points (e.g. a signal may take 500 us to travel 100 km). The fixed circuit processing delay varies according to the type and number of network nodes (e.g. routers) and the S132 fixed circuit delays. These fixed delays do not affect the Clock Recovery performance or Jitter Buffer depth (unless they change, e.g. when switching to a backup/protection line).

PDV is caused when congestion occurs at a port that has more than one packet waiting to be transmitted and can be caused by circuits that process data in "blocks" (delayed waiting to finish a block).

There are several PDV parameters that are identified in the equation below and described in Table 9-6.

Total PDV = Network PDV + S132 Ether Media PDV + S132 Schedule PDV + S132 BFD PDV + S132 MTIE PDV

Table 9-6. PDV Parameters that affect the latency of a PW packet

PDV Type	Description
Network PDV:	Network PDV is generated by the packet switches between the two PW End Points. Each packet switch becomes congested when more than one incoming switch port has a packet to send to the same outgoing switch Port (one incoming packet must wait for the other). For example some networks may assume that each packet switch might introduce up to 1 ms of PDV.
S132 Ethernet Media PDV:	S132 Ethernet Media PDV is generated when the Ethernet Port Line Rate (100 Mb/s or 1000 Mb/s) delays the delivery of the packet because the line rate is unable to transmit infinitely fast. For example if 32 packets that are 64 bytes in length, are waiting to be transmitted at the S132 Ethernet Port, the last packet will not be transmitted until after the 31 other packets are transmitted. The Ethernet Media PDV can be a large number. For this reason, the 1000 Mb/s line rate should be used whenever possible to minimize this parameter. The Ethernet Media PDV is dependent on the Ethernet line rate, the number of Bundles, the size of the Ethernet packets that are being transmitted and includes the Ethernet 20-byte Inter-packet Gap (IPG). The equation below assumes all of the Bundles use the same packet size. Table 9-7 provides 6 examples that use this equation. *S132 Ethernet Media PDV = [# Bundles * (# pkt bytes + 20 byte IPG) * 8 bits/byte] ÷ line rate*
S132 RXP & TXP Scheduling PDV:	The S132 RXP and TXP Scheduling PDV values are caused by the limited rate at which data can be transferred to/from the SDRAM. Similar to the Ethernet Media PDV, if 32 packets are ready (in the SDRAM) to be sent, the S132 Buffer Manager can only retrieve one packet at a time and the last packet is delayed waiting for the other 31 packets. This PDV parameter increases the Total PDV only if the Ethernet Media is able to forward packets faster than the S132 Buffer Manager can retrieve the packets from the SDRAM (i.e. the Scheduling PDV is "hidden" by the Ethernet Media PDV as long as the Buffer Manager can keep up with the Ethernet Port transmission rate).
S132 RXP & TXP BFD PDV:	The S132 RXP and TXP BFD PDV values are caused as the S132 waits for sufficient data to fill the SDRAM Staging Buffers. The depths of these buffers are programmed using the BFD registers to determine the data block size that is used to store and retrieve data from the SDRAM. This PDV parameter can vary from 125 us to 500 us according to the BFD setting (one in each direction).
S132 RXP MTIE PDV:	The S132 MTIE PDV is generated by the varying output frequency of the Clock Recovery Engine. Before the Clock Recovery Engine has locked to the incoming RXP packet rate, the transmit TDM Port line rate can vary (slightly) adding to the Total PDV. After the Clock Recovery Engine is locked to the RXP data rate, this parameter becomes insignificant. This parameter is difficult to characterize,

	but is generally not important to consider since its impact is only during the "start-up" of a TDM Line.

Table 9-7. Maximum S132 Ethernet Media PDV

Maximum Number of Bundles	Packet Size for 100 Mb/s Interface			Packet Size for 1000 Mb/s Interface		
	64 Bytes	193 Bytes	1500 Bytes	64 Bytes	193 Bytes	1500 Bytes
32 Bundles	0.215 ms	0.663 ms	3.89 ms	0.0215 ms	0.0663 ms	0.389 ms
256 Bundles	1.73 ms	5.30 ms	31.1 ms	0.172 ms	0.530 ms	3.11 ms

As depicted in Table 9-7, the S132 Ethernet media interface (MII/GMII) can introduce a high PDV level with systems that have a high number of Bundles when using the 100 Mb/s interface and a large packet size. Most applications will want to minimize delay. The S132 Ethernet Media PDV parameter can be minimized by using smaller packet sizes and the 1000 Mb/s Interface. A TXP Bundle that is used for Clock Recovery at the far PW End Point should be programmed for S132 high priority TXP processing (B.BCDR3.TXBPS). If only one TXP Bundle from each receive TDM port is programmed for high priority, then each high priority TXP Bundle will not be delayed by more than 31 other Bundles (no more than one Bundle for each of the other enabled TDM Ports).

The following provides an example set of assumptions for a T1-SAT PW:

Ethernet Media PDV

Packet Payload size = 193 bytes (PCT = 1 ms)

MPLS Header size with 2 MPLS Labels, Control Word, RTP Headers and 4-byte Ethernet FCS = 46 bytes

Ethernet Media Type = 100 Mb/s

Maximum PW Bundles (not including OAM Bundles) = 32

*Ethernet Media PDV = [# Bundles * (# bytes per pkt + 20 byte IPG) * 8 bits/byte] ÷ line rate*

Ethernet Media PDV = [32 * (193 + 46 + 20) * 8b] ÷ 100 Mb/s = [32 * (259 * 8b)] ÷ 100 Mb/s = 660 us

Scheduling PDV

The scheduling PDV is assumed to be "hidden" by the Ethernet Media PDV and can be ignored.

BFD PDV

RXP & TXP BFD settings = 125 us (in each direction)

MTIE PDV

The startup MTIE is assumed to be insignificant except at startup.

Total PDV

Total PDV = Network PDV + Ethernet Media PDV + Scheduling PDV + TXP & RXP BFD PDV + MTIE PDV

Total PDV = Network PDV + 660 us + (125 us * 2) = Network PDV + 910 us

The S132 can support up to 500 ms of packet Jitter (PDV) for up to 256 Bundles (one Jitter Buffer is provided for each Bundle). In most cases, however, the PDV of a network will be limited to a much smaller value like 10 ms. The Jitter Buffers for all Bundles are located in a single block of memory that begins at the SDRAM address specified by G.BMCR2.JBSO. The Jitter Buffer memory block is divided into equal sized Jitter Buffer FIFOs according to the G.GCR.JBMD setting (one FIFO per Bundle; JBMD sets the depth for all Jitter Buffer FIFOs to 32 Kbyte, 64 Kbyte, 128 Kbyte or 256 Kbyte).

The Maximum PDV that each Jitter Buffer can support can be determined according to the equation below. The "Register Guide", "SDRAM" subsection includes a table (based on this equation) that describes the "Maximum PDV" each Jitter Buffer can store for various combinations of PCT, JBMD and "maximum Timeslots in a Bundle".

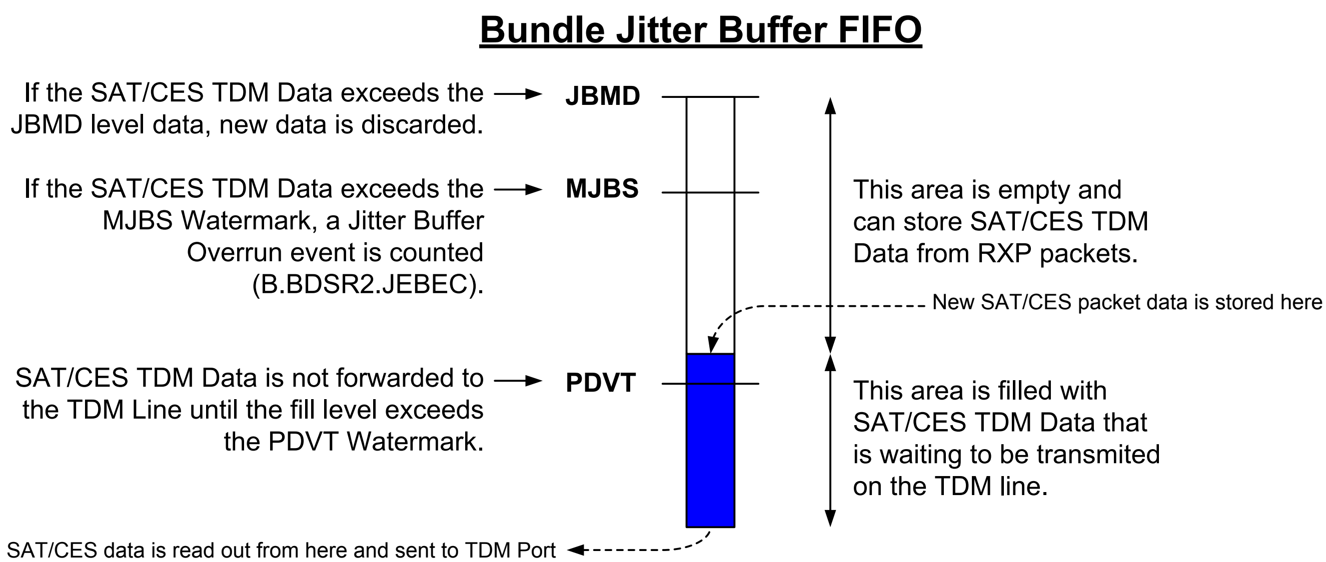
Max PDV in ms = Integer(([Roundup((JBMD in bytes) – 2048) ÷ ((PCT in ms / 0.125) + 4)) + 1] * PCT in ms) ÷ 2)
where the "Roundup" function provides the next higher integer value for non-integer numbers

In addition to the global Jitter Buffer settings (JBSO and JBMD) there are two Jitter Buffer settings for each Bundle to program the Bundle's Jitter Buffer Playout Watermark (B.BCDR5.PDVT) and Jitter Buffer Overrun Watermark (B.BCDR5.MJBS).

The PDVT setting specifies how much data must be stored by the Jitter Buffer before "play-out" (FIFO read) begins. After "play-out" begins the Jitter Buffer will continue to supply data until the Jitter Buffer is empty. If the Jitter Buffer empties, then the Jitter Buffer must again fill to the PDVT level before data will again be forwarded.

The MJBS watermark can be used to indicate when the level of stored data exceeds an "expected" maximum (overrun) level. This can be used to monitor for "unexpected" fill levels (e.g. too much data accumulated because of improperly configured input or output clocks or if the MJBS setting does not allow for the maximum PDV). MJBS can be monitored to implement a discard process that prevents each Bundle's Jitter Buffer from over-filling and adding to the latency of the data (some Clock Recovery Engine firmware revisions may include a function to discard Jitter Buffer data when MJBS indicates the Jitter Buffer has too much data). The MJBS register should be programmed to a level that is lower than the JBMD level so that an MJBS Overrun condition can be detected before JBMD discarding begins. Figure 9-16 depicts the relationship between the JBMD, MJBS and PDVT settings (the blue area depicts data that is stored in the Jitter Buffer FIFO).

Figure 9-16. Bundle Jitter Buffer FIFO



The purpose of the Jitter Buffer is to store data that can be transmitted during time periods when the S132 must wait for a packet that has been "delayed". At the receiving end of a PW, when a packet is received the PW end point cannot know whether the PDV for that packet was "zero", the maximum PDV value or any value in between.

If the receiving PW end point knew that the PDV for a received packet was zero, then the best situation would be to begin storing data and not forward that data until a time period equal to the maximum PDV. Or, if the PW end point instead knew that a packet was received with the maximum PDV, then the best situation would be to immediately forward the data (data will never come later than the maximum PDV; storing would add unnecessary delay). However the PW end point does not know the PDV level for each packet and thereby must make an assumption.

There are three approaches for setting the PDVT and MJBS values. Each system should be analyzed to determine which approach is preferred. In each of these approaches the minimum Jitter Buffer delay is equal to the PDVT setting, while the maximum Jitter Buffer delay (maximum fill level) is either equal to the MJBS or JBMD setting (MJBS is the maximum if MJBS is monitored as a watermark for discarding; otherwise the maximum is JBMD).

The first approach assumes that it is important to never discard data. This approach results in "2 * Total PDV" ≤ "Jitter Buffer Delay" ≤ "MJBS or JBMD". This may be the most commonly used setting for existing/installed TDM over PW services. The settings for this approach are specified by the following equations:

$PDVT_1$ (in ms) = 2 * Total PDV (in ms)

$MJBS_1$ (in ms) = PCT (in ms) + 2 * Total PDV (in ms)

The PCT value is included as part of the MJBS setting to provide a watermark condition that is slightly higher than the PDVT (playout) watermark and because the originating and terminating ends of the PW cannot be perfectly phase synchronized together. When the PCT is included as part of the MJBS value, in most cases, the S132 fixed circuit processing delays can be disregarded (included as part of the PCT value, e.g. BFD PDV).

The second approach assumes that delay must be minimized and only a small amount of discarding should be allowed. This approach results in a temporary, maximum latency = "2 PDV + PCT". But as the PDV varies from its minimum to its maximum value, a small number of packets are discarded and the latency is reduced to "PDV + PCT". You could say that this approach assumes the PDV = 0 for the first packet. The maximum number of packets that will be discarded during the life of the connection will be the Integer value of (PDV ÷ PCT) + 1 (e.g. if PDV = 10 ms and PCT = 2 ms, then up to 6 packets may be discarded). The discard timing is not predictable since the discarding only occurs when the PDV extremes are reached. The settings for this approach are specified by:

$PDVT_2$ (in ms) = Total PDV (in ms)

$MJBS_2$ (in ms) = PCT (in ms) + Total PDV (in ms)

The third approach also assumes that the delay and data errors must be minimized but also prevents the latency from exceeding "PDV + PCT". Instead of allowing for a small number of packet discards, this approach allows for a small amount of dummy data insertion. The Jitter Buffer immediately forwards the first data is received, as though the packet is assumed to be received with maximum PDV. Since this will not normally be the case, a Jitter Buffer underrun will be expected. However, the amount of dummy data that is inserted (to stabilize the Jitter Buffer fill level) is limited by the Total PDV value. For example if PDV = 10 ms and PCT = 2 ms, then ≤ 12 ms of dummy data may be transmitted. The timing of the dummy data is not predictable since the insertion of dummy data depends on when the PDV extremes are reached. The settings for this approach are specified by the following equations:

$PDVT_3$ = 0x0001 (minimum setting > 0)

$MJBS_3$ (in ms) = PCT (in ms) + Total PDV (in ms)

The PDVT and MJBS values are programmed using the equations below and should be rounded up to the nearest integer setting. The units used by these registers vary according to the application:

PDVT setting units for T1/E1 CES: 125, 250 or 500 us (according to the Pn.PTCR1.BFD setting)
PDVT setting units for SAT: 32 ÷ "TDM Port bit rate" (e.g. the T1 SAT PDVT setting is in 20.7 us steps)
MJBS setting units for T1/E1 CES: 500 us
MJBS setting units for SAT: 1024 ÷ "TDM Port bit rate" (e.g. the E1 SAT PDVT setting is in 500 us steps)

The Jitter Buffer Fill Level impacts the total delay of the reconstructed TDM data stream. The fill level of the Jitter Buffer is constantly changing according to the bursty nature of the RXP packets. So the delay of a TDM data stream is not referenced to when an RXP packet is received but is instead viewed as the delay from the receive TDM Port at the far PW End Point to the transmit TDM Port at the near/local end.

If the Jitter Buffer can store enough data to equal (or exceed) the Total PDV, then the Total PDV can be viewed as being included in the Maximum Jitter Buffer Fill Level. Because the Jitter Buffer fill level is constantly changing, it is not easy to define an independent Jitter Buffer delay parameter (to calculate the total delay). But in general the "highest" Jitter Buffer fill level can be equated to the "Jitter Buffer + Total PDV" delay (assuming Maximum Fill Level ≥ Total PDV). The term "highest" is used, because it is possible that the Jitter Buffer fill level will stabilize at a level that is lower than the programmed Maximum Fill Level (e.g. the Jitter Buffer "highest" fill level may stabilize at a 6 ms level, while MJBS may be programmed to 8 ms). Although the Jitter Buffer for a PW may stabilize below the Maximum Fill Level, the total delay is most commonly estimated with the equation below:

Max Total Delay ≅ PCT + fixed transmission delay + TXP BFD + Max Jitter Buffer Fill Level

For a T1 SAT PW and assuming PCT = 1 ms, fixed transmission delay = 2.5 ms (e.g. 500 km fiber), Network PDV = 3 ms and the remaining PDV = 910 us (from the previous Total PDV example), the 3 approaches will result in:

Approach #1 (No Data Discard)

$PDVT_1$ (in ms) = 2 * 3.91 ms = 7.82 ms ($PDVT_1$ register = 0x017A or 378 decimal which equates to 7.82 ms)

$MJBS_1$ (in ms) = 1 ms + 7.82 ms = 8.82 ms ($MJBS_1$ register = 0x0012 or 18 decimal which equates to 9 ms)

Max Total $Delay_1$ = 1 ms + 2.5 ms + 9 ms = 12.5 ms (assuming MJBS is used to discard data)

Approach #2 (Minimize Delay With Limited Overrun)

$PDVT_2$ (in ms) = 3.91 ms ($PDVT_2$ register = 0x00BD or 189 decimal which equates to 3.91 ms)

$MJBS_2$ (in ms) = 1 ms + 3.91 ms = 4.91 ms ($MJBS_2$ register = 0x000A or 10 decimal which equates to 5 ms)

Max Total $Delay_2$ = 1 ms + 2.5 ms + 5 ms = 8.5 ms (assuming MJBS is used to discard data)

For this approach the initial Max Total Delay may be as much as 1 + 2.5 + 2 * 5 = 13.5 ms, but will drop to Max Total Delay = 8.5 ms after packets have been discarded due to Jitter Buffer overrun events.

Approach #3 (Minimize Delay With Limited Underrun)

$PDVT_3$ (in ms) = 0 ms ($PDVT_3$ register = 0x0001 which equates to 20.7 us)

$MJBS_3$ (in ms) = 1 ms + 3.91 ms = 4.91 ms ($MJBS_3$ register = 0x000A or 10 decimal which equates to 5 ms)

Max Total $Delay_3$ = 1 ms + 2.5 ms + 5 ms = 8.5 ms (assuming MJBS is used to discard data)

The Jitter Buffer Maximum Fill Level generally determines the maximum delay. Although the fill level will initially stabilize at a level just high enough to support the Total PDV, when anomalies occur (e.g. temporary line failures and RXP PW protection switching) the Jitter Buffer can fill beyond the "Total PDV" level. If the Jitter Buffer fill level is not "corrected" after an anomaly, because of the near constant rate of the transmit TDM Port, the "extra" data will not dissipate and will increase the total delay. For example if the Maximum Fill Level is programmed to 1 second (MJBS or JBMD), and the Total PDV is 10 ms, initially the Jitter Buffer may stabilize at a 10 ms level. But anomalies could cause the Jitter Buffer to fill beyond the 10 ms level (e.g. equipment programming changes) and as more anomalies occur, the fill level could accumulate to any level up to 1 second.

There are several registers that the CPU can use to monitor the Jitter Buffer Fill level. Monitoring can be implemented by polling the Jitter Buffer Maximum and Minimum fill levels or by monitoring for Overrun/Underrun event indications (data discarded or dummy data inserted). The Jitter Buffer Fill Levels can help to identify setup errors. Other Jitter Buffer functions that can be enabled include Packet Reordering (for packets received out of order), packet discard monitoring for too early, too late and duplicate packet Sequence Number. The registers that support these Jitter Buffer functions include: G.GCR.IPSE, G.GCR.RDPC, G.GSR1.JBS, G.GSRIE1.JBUIE, G.GSR6.JBGS, PC.CR1.DPDE, B.BCDR1.SCSNRE, B.BDSRL1.JBLPDSL, B.BDSR2 - B.BDSR3, B.BDSR5 - B.BDSR7, B.GxSRL, and JB.GxSRL.

A Jitter Buffer overflow can occur for three reasons: the selected Transmit TDM Port clock is not the same rate as that used by the RXP packets (i.e. the wrong clock was selected); clock recovery is selected but has not yet fully converged to the RXP Packet data rate and is running too slow; the Jitter Buffer depth is too small to handle the maximum incoming PDV.

The Jitter Buffer is also used by HDLC Connections. However, HDLC Connections, in general, do not transport constant bit rate data streams (unlike SAT/CES Payload Connections), so the Jitter Buffer is instead used as a more simplistic FIFO. The Jitter Buffer PDVT and MJBS settings, and the Packet Reordering, Early/Late and Duplicate Discard functions do not have any meaning with HDLC Connections. HDLC data is forwarded as soon as it is available. JBMD defines the depth of the FIFO.

9.2.6 TDM Diagnostic Functions

The S132 supports TDM Loopback and TDM BERT Functions for diagnostic testing of the TDM Ports.

9.2.6.1 TDM Loopback

The S132 supports 3 types of Loopbacks for the TDM Ports: TDM Port Line Loopback, TDM Port Timeslot Loopback and Bundle Loopback. Any number of TDM Ports can be in loopback at the same time.

The TDM Port Line Loopback is enabled using Pn.PTCR2.PRPTLL. This loopback takes data from RDAT and re-transmits that data on TDAT. All data that is received on RDAT is looped back to TDAT.

The TDM Port Timeslot Loopback is enabled using Pn.PTCR3.PRPTTSL (32 bits, one for each TDM Port Timeslot). This loopback also takes data from RDAT and re-transmits that data on TDAT, but only for those Timeslots that have the loopback function enabled. Timeslots that do not have the loopback function enabled continue to pass data (from Receive TDM Port to TXP Packet and from RXP packet to transmit TDM Port).

For either of these loopbacks to function properly the programmed Transmit TDM Port clock and synchronization sources (when applicable) must be set to be the same as that of the Receive TDM Port.

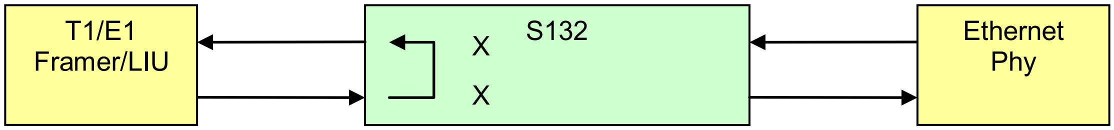
When either loopback is enabled, the data for receive TDM Timeslots, that are in loopback, will continue to be transmitted in TXP packets if TXP Bundles are assigned to the Receive TDM Port and enabled. The TXP packet stream can be disabled by de-activating the Bundle or by disabling TXP Bundle transmission (B.BCDR3.TXPMS).

RXP Packet data that is received for Timeslots that are in loopback is still forwarded to the Jitter Buffer and is still used for Clock Recovery. When the loopback is removed, any data that is waiting in the Jitter Buffer is forwarded to the TDM Port. To prevent the Jitter Buffer from filling with data during a loopback, the payload data for a Bundle can be discarded (B.BCDR4.RXBDS). Clock Recovery will continue to function for an RXP Bundle that is in one of these 2 loopbacks as long as the Bundle is selected for Clock Recovery (B.BCDR4.PCRE).

The Transmit TDM Port can only use one timing source, so caution must be exercised when enabling loopbacks for some Timeslots while other Timeslots are not in loopback. A frequency difference between the looped back RDAT data and the (non-looped) RXP Packet data will result in occasional slips (corrupted data).

These 2 loopbacks are depicted in Figure 9-17 using a T1/E1 example. The arrow depicts the loopback direction. The diagram does not depict how "normal" data continues to be forwarded to/from the Ethernet Phy.

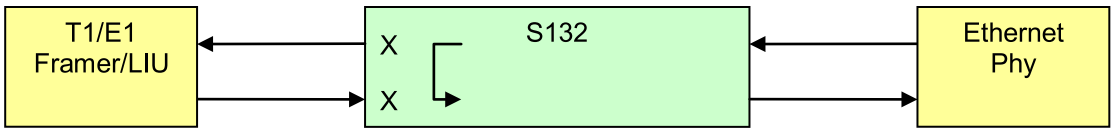
Figure 9-17. T1/E1 Port Line Loopback and TDM Port Timeslot Loopback Diagram



The TDM Bundle Loopback is enabled using Pn.PRCR3.PTPRTSL (32 bits, one for each TDM Port Timeslot). This loopback takes received RXP packet data and re-transmits that data in TXP packets. To work properly, when this loopback is used, all Timeslots for an RXP Bundle should be enabled for loopback; the TXP and RXP Bundles should be programmed to use the same number of Timeslots and the same functions (e.g. if the RXP Bundle is Structured, the TXP Bundle should also be Structured); and the Receive TDM Port timing source should be equal to the data rate of the RXP Packet data (the Receive TDM Port timing determines the fill rate of the TXP Packet).

In the RXP direction, data received from RXP packets is also transmitted at the transmit TDM Port. In the TXP direction, data that is received at the TDM Port for Timeslots that are in loopback is discarded. This loopback is depicted in Figure 9-18 using a T1/E1 example. The arrow in the figure shows the direction of the looped back data. The diagram does not depict how "normal" data continues to be forwarded to and from the Ethernet Phy.

Figure 9-18. T1/E1 Port Bundle Loopback Diagram



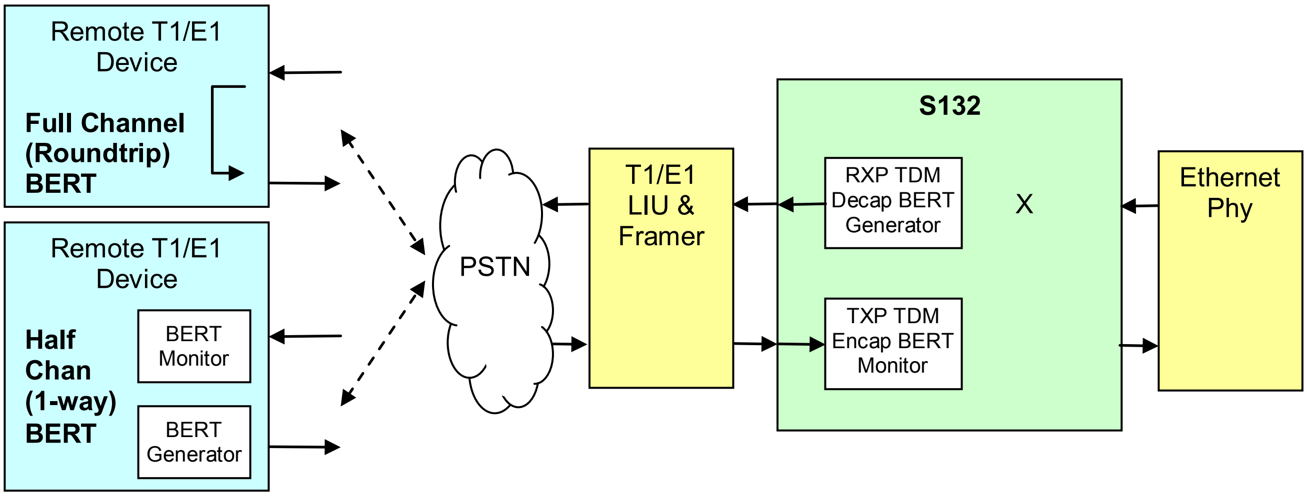
The TDM Bundle Loopback de-encapsulates the payload data from RXP packets, sends the RXP payload data back in the TXP direction and then re-encapsulates the data into a TXP packet. The data for the TDM Line and TDM Line Timeslot Loopbacks is not "packetized" (encapsulated/de-encapsulated) before loopback.

Each TDM Loopback type can be enabled for both Structured and Unstructured data streams.

9.2.6.2 TDM BERT

A TDM Port can be tested using a BERT test pattern. The S132 supports "Full Channel" (bidirectional) and "Half Channel" (unidirectional) TDM BERT Testing. Only one TDM BERT Test can be enabled on an S132 device at a time. The "Full Channel" and "Half Channel" BERT Tests are depicted in Figure 9-19 using a T1/E1 Example.

Figure 9-19. TDM Port BERT Diagram



The Full Channel (Roundtrip) Test requires a loopback at the far end (left side of diagram). The S132 Decap BERT Pattern Generator sends a BERT Pattern to the S132 Transmit TDM Port. The Encap BERT Monitor verifies that data, returned at the Receive TDM Port, is error free.

The Half Channel (one-way) Test requires an equivalent BERT Tester at the far end (on the left side of the diagram). The S132 BERT Pattern Generator sends a BERT Pattern to the S132 Transmit TDM Port. The far end uses a BERT Monitor to verify that the data is received error free. Similarly, the far end can transmit a BERT Pattern in the opposite direction and the S132 BERT Monitor can be used to verify the received data is error free.

There is also a Packet BERT that is described in the “Packet BERT” section. The TDM BERT Engine can be enabled at the same time as the Packet BERT. However, the two BERT Engines share several register settings, so the TDM and Packet BERT tests do not function independent of each other. For Half Channel TDM BERT Testing the Generator and Monitor must be programmed to match what is expected at the far end (left side of Figure 9-19). There is no register setting to program the BERT Test Engine to “Full” or “Half” Channel Testing. The connections that are external to the S132 determine the Full vs. Half Channel application.

The S132 TDM BERT Engine uses an Encap BERT Monitor and a Decap BERT Generator. The MD.EBCR.ERBE enable/disables the TDM Encap BERT Monitor and MD.EBCR.ERBBS selects the TXP Bundle that is to be monitored. Programming ERBBS with a TXP Bundle number identifies the TDM Port and Timeslots that are tested (from the B.BCDR4.PNS and B.BCDR2.ATSS that are assigned to that TXP Bundle). The MD.DBCR.DTBE enable/disables the TDM Decap BERT Generator and MD.DBCR.DTBBS selects the RXP Bundle that is replaced with the generated pattern (from the B.BCDR4.PNS and B.BCDR2.ATSS that are assigned to that RXP Bundle).

The TDM BERT Engine supports 3 Test Pattern Types: Pseudo-Random Bit Sequence (PRBS), Quasi-Random Bit Sequence (QRSS) and Repetitive Patterns. The TDM BERT Generator Test Pattern Type is programmed using DB.BPCR.PTS and DB.BPCR.QRSS. The TDM BERT Monitor Test Pattern Type is programmed using EB.BPCR.PTS and EB.BPCR.QRSS. For Full Channel testing these should be programmed to the same settings.

For the Pseudo-Random pattern, the “z” coefficient, “y” coefficient and Seed for the $X + X^y +1$ PRBS pattern is selected for the Generator using DB.BPCR.PTF, DB.BPCR.PLF and DB.BPCR.BPS; and for the Monitor using EB.BPCR.PTF, EB.BPCR.PLF and EB.BPCR.BPS.

For the Quasi-Random pattern the PTF, PLF and BPS registers are ignored and the $X^{20} + X^{17} +1$ QRBS pattern is used. The Quasi-Random pattern is similar to a PRBS pattern but with the number of “consecutive zeros" in the pattern limited to 14.

For the Repetitive pattern, the pattern length and pattern value are selected for the Generator using, DB.BPCR.PLF, DB.BPCR.BPS; and for the Monitor using EB.BPCR.PLF and EB.BPCR.BPS. The EB.BPCR.PTF and DB.BPCR.PTF settings are ignored.

The DB.BCR.TNPL is used to initiate the TDM BERT Generator with a New Test Pattern Load and TPIC is used to enable Test Pattern Inversion.

The EB.BCR.RNPL is used to initiate the TDM BERT Monitor with a New Test Pattern Load, RPIC enables Test Pattern Inversion, MPR enables Manual Resynchronization and APRD Disables the automatic “Pattern Resynchronization” function (the APRD = “0” setting enables auto-resynchronization when test pattern lock is lost).

The EB.BSR, EB.BSRL, EB.BSRIE, EB.RBECR, EB.RBCR are used to Monitor the status of the TDM BERT Test and measure the bit error performance.

The TDM BERT Generator can be programmed to insert errors in the BERT Test Pattern using the DB.TEICR register. This can be used to demonstrate that the monitoring function (local or far end) is functioning properly.

RXP and TXP Packet functions, for Bundles that have been assigned to a TDM BERT Test, continue to function when a BERT Test has been enabled (e.g. Clock Recovery) except that the RXP Packet payload is replaced by the TDM BERT Test Pattern in the transmit TDM Port Timeslots. For most applications the TXP and RXP Bundles should be disabled during a TDM BERT Test.

Special Consideration

CAS Signaling functions should be disabled for a Bundle that is used for TDM BERT Testing. In some applications the BERT Test Pattern may be over-written with CAS Signaling.

9.3 Packet Processing Functions

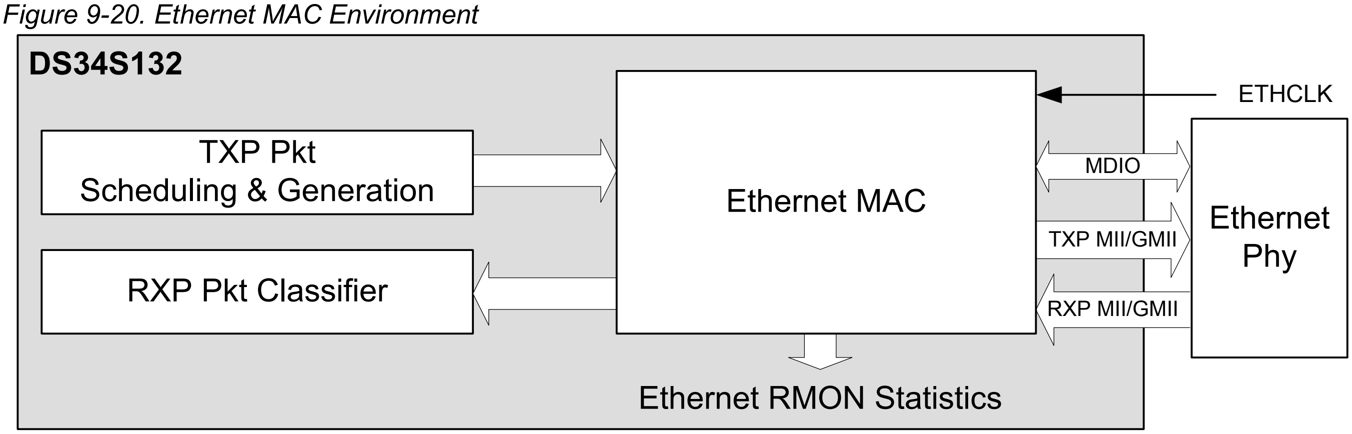
The S132 includes one Ethernet Port to receive and transmit Ethernet packets. The high level functions include:

- 100 Mbps MII or 1000 Mbps GMII Interface
- Ethernet II and IEEE 802.2 LLC/SNAP formats
- 0, 1 or 2 VLAN Tags with programmed TPID values
- 2000 byte Maximum Ethernet Frame Length
- 2 programmable Ethernet DAs
- Broadcast Ethernet DA
- L2TPv3, UDP, MEF-8 or MFA-8 PW protocol
- 0, 1 or 2 Outer MPLS Labels
- 0, 1, or 2 L2TPv3 Cookies
- Up to 256 PW Bundles
    - Any mix of SAT, CES, HDLC and Clock Only
    - T1, E1 or slower payload data rates
    - CES with/without Sub-channel CAS Signaling
- "IPv4-only", "IPv6-only" or "IPv4 and IPv6"
    - 3 programmed IPv4 DAs
    - 2 programmed IPv6 IP DAs
- UDP
    - 2 programmed UDP Protocol Types (or ignore)
    - Selectable 16-bit or 32-bit PW-ID
    - Optional 16-bit PW-ID Mask
- Verify and generate FCS for IPv4 and UDP
- Optional Control Word and RTP Headers
    - Flexible PW Sequence Numbering functions
        - Missing Packet Detection
        - Packet Re-ordering
- RXP CPU packet monitoring
    - In-band VCCV
    - 32 Out-band VCCV BIDs (UDP-specific OAM)
    - Several programmed "send to CPU" Conditions
        - Special Ethernet Type
        - Detected Packet Error Conditions
        - PW Bundle Debug
- TXP Packet Generation
    - 256 programmed TXP Bundle Headers
    - Flexible CPU generated TXP packet format
    - CES/SAT packets with/without RTP Timestamp
    - CPU packets with/without OAM Timestamps
    - High and Low Priority TXP PW Queues
- Ethernet Port RMON Statistics
- Ethernet Port Loopback
- Ethernet Port BERT Testing
- MDIO Interface for Phy device Management

9.3.1 Ethernet MAC

The Ethernet MAC/port can support 100 Mbps using an MII interface or 1000 Mbps using a GMII interface to transmit and receive data with an external Ethernet Phy device. The MAC also provides RMON statistics and an MDIO interface for communicating with the Phy device. Figure 9-20 provides a high level view of the Ethernet MAC environment.

Figure 9-20. Ethernet MAC Environment



The Ethernet Line rate is selected using M.NET_CONFIG.GIG_MODE_EN and G.GCR.GMMS.

For 100 Mbps the S132 uses an MII interface with two 4-bit, unidirectional data-buses. Transmit data (TXD [3:0]) and Receive data (RXD [3:0]) are timed using the RXCLK and TXCLK inputs from the Phy device. The ETHCLK input must be 25 MHz and G.GCR.EC25 = 1.

For 1000 Mbps, the S132 uses a GMII interface with two 8-bit, unidirectional data-buses. Transmit data (TXD [7:0]) is timed using the GTXCLK output. Receive data (RXD [7:0]) is timed using the RXCLK input from the Phy device. Both GTXCLK and RXCLK are 125 MHz signals. The GTXCLK signal is derived from a 125 MHz ETHCLK input

(G.GCR.EC25 = 0). In some cases ETHCLK can be tied to RXCLK to have the Phy device drive both inputs at one time (as long as the RXCLK output from the Phy is a constant, non-gapped 125 MHz signal).

M.NET_CONTROL.TXP_HALT, START_TXP, TXP_EN and RXP_EN enable/disable the flow of RXP and TXP data at the Ethernet MAC/Port.

The MAC must be programmed to operate in the Full-duplex mode (M.NET_CONFIG.EN_FRMS_UDUP = 0 and M.NET_CONFIG.FULL_DUPLEX = 1). The Half-duplex mode and Pause Control are not supported because they can adversely affect the delay/latency of the PW packets.

The standard maximum Ethernet Frame size is 1518 bytes. The MAC can be programmed to accept RXP Ethernet frames with byte lengths of 1518 bytes or 1536 bytes using M.NET_CONFIG.RXP_1536FRMS or up to 2000 bytes using M.NET_CONFIG.JUMBO_FRMS.

The MAC can be programmed to accept or discard all non-VLAN frames using M.NET_CONFIG.DISC_NOVLAN.

The MAC, when programmed as prescribed in the "Register Guide", "Global Ethernet MAC" section, checks each received RXP packet for valid Ethernet preamble, FCS, alignment and length. Packets with errors are discarded. In the TXP direction the MAC appends an Ethernet FCS and adds padding to packets that are < 64-bytes in length.

The MDIO interface can be enabled using M.MAN_PORT_EN, and programmed using the M.PHY_MAN and M.NET_STATUS registers.

MDC (MDIO Clock) is divided down from the SYSCLK input. MDC_CLK_DIV sets the "divided by" value and should be set such that MDC frequency = SYSCLK ÷ (selected MDC_CLK_DIV divider value) ≤ 2.5 MHz. For example if SYSCLK = 50 MHz and MDC_CLK_DIV = 010b (selects divide by 32), then the MDC frequency will be 1.56 MHz.

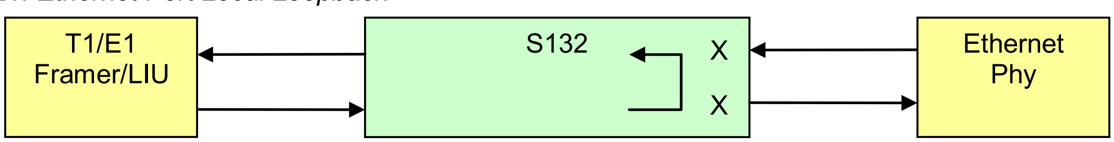
9.3.1.1 Ethernet Port Diagnostic Functions

The S132 supports Ethernet Loopback and Packet BERT Functions for diagnostic testing of the Ethernet Port.

9.3.1.1.1 Ethernet Loopback

The M.NET_CONTROL.LB_LOCAL = 1 enables the Ethernet Port Loopback that sends all receive TXP packet data back in the RXP direction. CES, SAT, HDLC and Clock data/information that is received at a TDM Port is encapsulated into TXP packets using the programmed Bundle settings. TXP packets that are initiated by the CPU are also encapsulated into TXP CPU Packets. The combination of all TXP packet types is looped back in the RXP direction. The RXP packets are forwarded according to the programmed RXP Bundle settings (forwarded to the TDM Ports and/or CPU). No data is transmitted toward the Ethernet Phy and no data is received from the Ethernet Phy while the Ethernet loopback is active. This loopback is depicted in Figure 9-21 using a T1/E1 example (the loopback of TXP CPU packets to the CPU is not depicted).

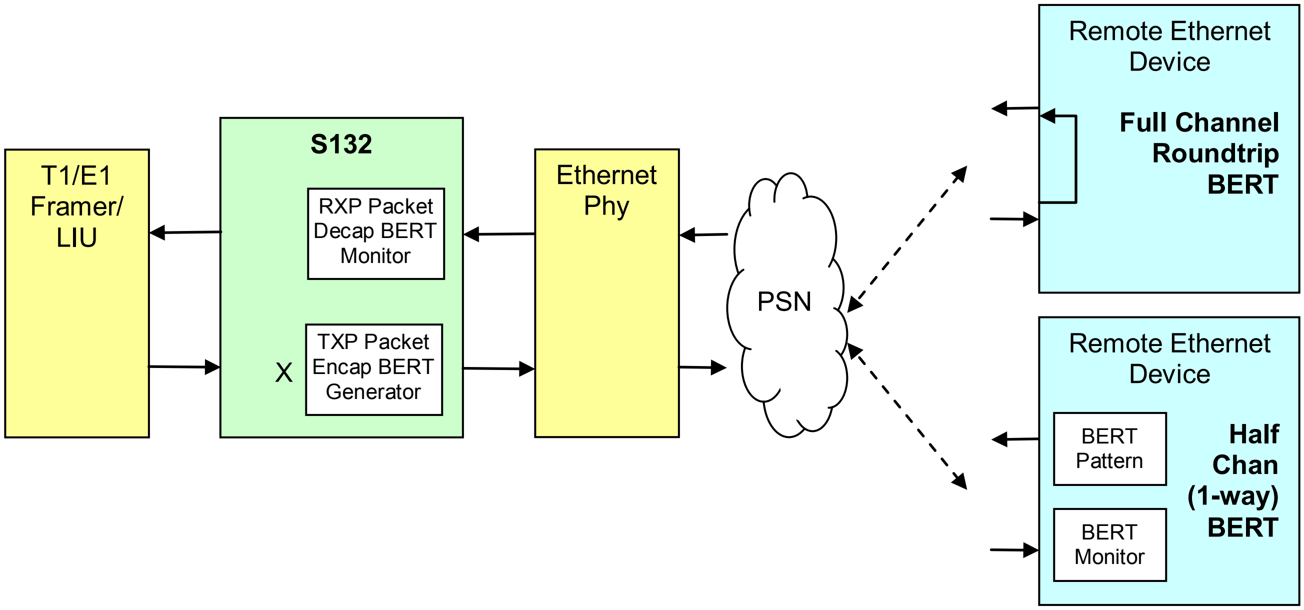
Figure 9-21. Ethernet Port Local Loopback



9.3.1.1.2 Packet BERT

An Ethernet path can be tested using a BERT Test Pattern. The S132 supports "Full Channel" and "Half Channel" Packet BERT Testing. Only one Packet BERT Test can be enabled on an S132 device at a time. The "Full Channel" and "Half Channel" BERT Tests are depicted in Figure 9-22.

Figure 9-22. Ethernet Port BERT Diagram



The Full Channel (Roundtrip) Test requires a loopback at the far end (on the right side of the diagram). The S132 Packet BERT Pattern Generator sends a BERT Pattern to the S132 Transmit Ethernet Port. The BERT Monitor verifies that the data returned at the Receive Ethernet Port is error free.

The Half Channel (one-way) Test requires an equivalent BERT Tester at the far end (on the right side of the diagram). The S132 Packet BERT Pattern Generator sends a BERT Pattern to the S132 Transmit Ethernet Port. The far end must use a BERT Pattern Monitor to verify that the data is received error free. Similarly, the far end can transmit a BERT Pattern in the opposite direction. The S132 BERT Monitor can be used to verify that the data is received error free.

The Packet BERT Engine can be enabled at the same time as the TDM BERT. The two BERT Engines share several register settings, so the TDM and Packet BERT tests are not independent of each other. For Half Channel Packet BERT Testing the Generator and Monitor must be programmed to match what is expected at the far end (right side of Figure 9-22). There is no register setting to program the BERT Test Engine to “Full” or “Half” Channel Testing. The connections that are external to the S132 determine the Full vs. Half Channel application.

The S132 Packet BERT Engine uses an Encap BERT Generator and a Decap BERT Monitor. The MD.EBCR.ETBE enable/disables the Packet BERT Generator and MD.EBCR.ETBBS selects the TXP Bundle that the generated BERT Test Pattern is to be inserted into. The BERT Test Pattern is placed in the Payload section. If a Bundle that is programmed to support sub-channel CAS Signaling is assigned to a Packet BERT Test, the sub-channel CAS Signaling is unaffected (not tested) by the BERT Test. The MD.DBCR.DRBE enable/disables the Packet BERT Monitor and MD.DBCR.DRBBS selects the RXP Bundle that is to be monitored.

The Packet BERT Engine supports 3 Test Pattern Types: Pseudo-Random Bit Sequence (PRBS), Quasi-Random Bit Sequence (QRSS) and Repetitive Patterns. The Packet BERT Generator Test Pattern Type is programmed using EB.BPCR.PTS and EB.BPCR.QRSS. The Packet BERT Monitor Test Pattern Type is programmed using DB.BPCR.PTS and DB.BPCR.QRSS.

For the Pseudo-Random pattern, the “z” coefficient, “y” coefficient and Seed for the $X + X^y + 1$ PRBS pattern is selected for the Generator using EB.BPCR.PTF, EB.BPCR.PLF and EB.BPCR.BPS; and for the Monitor using DB.BPCR.PTF, DB.BPCR.PLF and DB.BPCR.BPS.

For the Quasi-Random pattern the PTF, PLF and BPS registers are ignored and the $X^{20} + X^{17} + 1$ QRBS pattern is used. The Quasi-Random pattern is similar to a PRBS pattern but with the number of “consecutive zeros" in the pattern limited to 14.

For the Repetitive pattern, the pattern length and pattern value are selected for the Generator using EB.BPCR.PLF and EB.BPCR.BPS; and for the Monitor using DB.BPCR.PLF and DB.BPCR.BPS. The PTF settings are ignored.

The EB.BCR register is used to program the Packet BERT Generator for New Test Pattern Load (TNPL; initiate generation of the test pattern) and Test Pattern Inversion (TPIC).

The DB.BCR register is used to program the Packet BERT Monitor for New Test Pattern Load (RNPL), Test Pattern Inversion (RPIC), Manual Resynchronization (MPR) and Pattern Resynchronization Disable (APRD).

The DB.BSR, DB.BSRL, DB.BSRIE, DB.RBECR, DB.RBCR are used to Monitor the Packet BERT Test status.

The Packet BERT Generator can be programmed to insert errors in the BERT Test Pattern using the EB.TEICR register. This can be used to demonstrate that the monitoring function (local or far end) is functioning properly.

Receive and transmit TDM Port Timeslot functions, for Bundles that have been assigned to a Packet BERT Test, continue to function when a Packet BERT Test has been enabled except that the received TDM Port Timeslot data for the TXP Bundle that is assigned to the Packet BERT Test is replaced by the Packet BERT Test Pattern. For most applications the TDM Port Timeslots should be disabled during a Packet BERT Test.

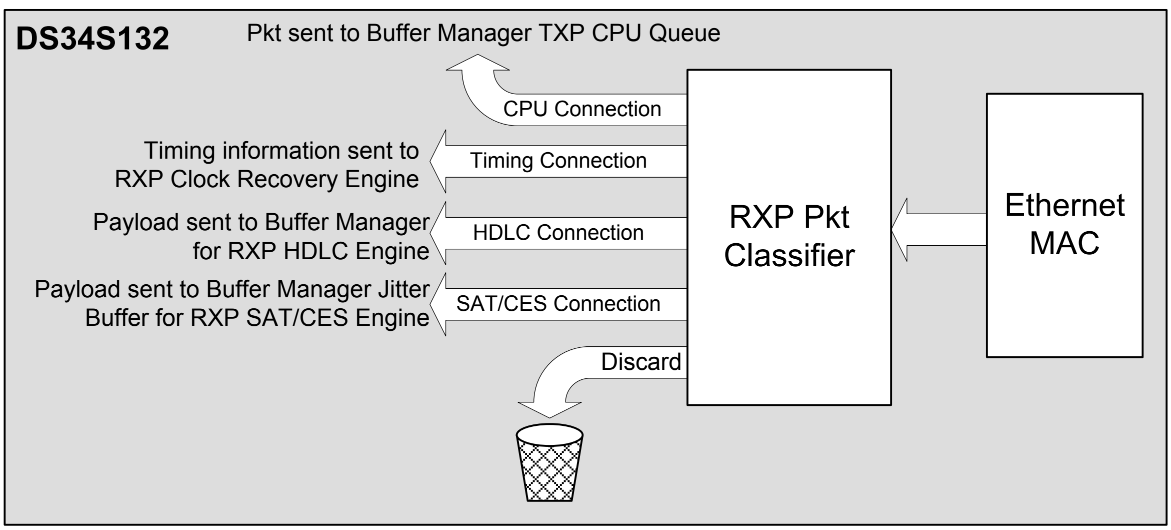
Special Consideration

CAS Signaling functions should be disabled for a Bundle that is used for Ethernet BERT Testing.

9.3.2 RXP Packet Classification

The header for a packet commonly contains several different header fields. The Classifier iteratively steps through each field of the header, looking for recognized formats and values. When the Classifier detects a recognized format/value, the Classifier either continues the classification process or has sufficient information to forward the packet to the next internal circuit block. Programmed settings determine the outcome for each interpretive step and are described in this section at a functional level. Figure 9-23 depicts the various destinations for RXP Packets.

Figure 9-23. RXP Packet Classifier Environment



When the Classifier determines that the format of a received RXP Packet header is not recognized, the packet may be discarded or forwarded to the CPU depending on the packet format and programmed settings. The program settings that determine Discard vs. CPU for unrecognized format/values are referred to as Discard Switches. These are described in the CPU Packet Classification section.

9.3.2.1 Generalized Packet Classification

The Classifier can be programmed to recognize 2 Ethernet DAs (PC.CR17 – PC-CR19) and the Ethernet Broadcast Address. If a received Ethernet DA is not equal to one of these values the packet is either forwarded to the CPU or discarded (PC.CR1.DPS9).

To be accepted an RXP Packet must use the DIX/Ethernet II or IEEE 802 LLC/SNAP format and can include 0, 1, or 2 VLAN tags. If VLAN tags are included, the inner VLAN tag TPID must equal PC.CR3.VITPID (normally 0x8100 for CVLAN). When a packet includes 2 VLAN tags the outer VLAN TPID must equal PC.CR3.VOTPID (for SVLAN).

The next packet header field that is tested is the Ethernet Type. The Classifier tests the Ethernet Type to determine if the packet uses a recognized PW Header. Six PW Headers can be recognized: MEF-8, MFA-8, UDP/IPv4, UDP/IPv6, L2TPv3/IPv4 and L2TPv3/IPv6.

For UDP and L2TPv3 applications, the Ethernet Type field must either be equal to IPv4 or IPv6. The S132 can be programmed to only recognize IPv4, only recognize IPv6 or to recognize both IPv4 and IPv6 (PC.CR1.RXPIVS and

PC.CR1.RXPDSD). If a packet is received that includes an Ethernet Type field that is equal to an IP version that is not enabled, the packet is discarded. According to the enabled IP version(s), the S132 can recognize up to 3 IPv4 DAs (PC.CR6 – PC.CR8) and up to 2 IPv6 DAs (PC.CR9 – PC.CR16). If the packet matches the enabled IP version(s), but does not match one of the programmed IP DAs, the packet is either forwarded to the CPU or discarded (PC.CR1.DPS1).

For MEF-8 applications the received Ethernet Type is compared against the programmed PC.CR4.MET value. For MFA-8 the Ethernet Type field is compared against the Unicast and Multicast MPLS Ethernet Type values. Table 9-8 identifies each of the recognized PW Ethernet Types.

Table 9-8. Recognized PW Ethernet Types

Ethernet Type	Ethernet Type value	Comment
MEF-8	PC.CR4.MET	Should be programmed to 0x88D8
MFA-8 Unicast MPLS	0x8847	Hardwired value in the S132.
MFA-8 Multicast MPLS	0x8848	Hardwired value in the S132.
IPv4	0x0800	Hardwired value in the S132.
IPv6	0x86DD	Hardwired value in the S132.

The information for identifying UDP and L2TPv3 headers is hardwired in the Classifier, without any enable settings.

If a received packet header matches one of the 6 PW Header Types, the packet is further processed as a PW packet. If the packet does not include one of the recognized PW Header Types the packet is further analyzed to determine whether it is a CPU packet (see "CPU Packet Classification section").

A packet with a recognized PW Header that includes the Ethernet Broadcast Address can be further processed or discarded (PC.CR1.DBTP).

9.3.2.2 PW (BID and OAM BID) Packet Classification

When one of the 6 PW Header Types has been detected, the Classifier next interprets the packet to find its PW-ID and then tests the PW-ID to see if it matches a recognized Bundle or OAM Bundle. The S132 can recognize up to 256 PW/Bundles and up to 32 OAM PW/Bundles. "Bundles" can be programmed to include CES, SAT, HDLC, PW-Timing (Clock Recovery) and/or CPU connections. The 256 Bundles are referred to as "Bundle 0" through "Bundle 255". "OAM Bundles" are similar to "Bundles", but restricted in their use.

"OAM Bundles" are commonly used in UDP applications to provide CPU, Out-band VCCV connections (such as "UDP-specific OAM"). The "OAM Bundles" are referred to as "OAM Bundle 0" through "OAM Bundle 31". Each OAM Bundle is usually associated with one or more of the 256 Bundles (to provide OAM for those Bundles). The use of OAM Bundles is optional and the association between "normal" Bundles and OAM Bundles must be made outside of the S132 (there are no internal S132 association settings or interactions). They are all treated independently by the S132.

The BID and OAM BID values must be programmed for each Bundle and OAM Bundle. The B.BACR register is used to select which of the 256 Bundles or 32 OAM Bundles is to be programmed, the B.BADR1 register to select the Active or Inactive state and the B.BADR2 register to specify the BID or OAM BID value.

For each "normal" Bundle there is a wide range of settings that can be programmed. The B.BCCR register is used to select which of the 256 Bundles is to be programmed and the B.BCDR1 – B.BCDR5 registers are used to specify the Bundle parameters. The OAM Bundles do not support other programmable "per-Bundle" parameters.

When the Classifier has determined that a received packet includes a recognized PW Header, the received PW-ID is compared against each of the active BIDs and OAM BIDs. The BID bit-width varies according to the PW Header type. The Classifier is hard-wired to support a 20-bit comparison for MEF- and MFA-8 and a 32-bit comparison for L2TPv3. For UDP the Classifier can be programmed to support a 16-bit or 32-bit comparison (G.GCR.UBIDLS).

To find a matching BID/OAM BID, the Classifier initially compares all of its active BIDs and OAM BIDs against each received PW-ID without verifying that the received Header Type is also correct. The received PW-ID field is identified (according to the received PW header type) and then compared against 256 BIDs and 32 OAM BIDs. If a received PW-ID does not match any of the active BIDs or OAM BIDs the packet is either forwarded to the CPU or discarded (PC.CR1.DPS6).

The S132 includes several BID/OAM BID settings and tests for UDP applications that are not used by the non-UDP applications. These are explained in the “UDP Settings” section. For non-UDP applications, the “UDP Settings” section can be skipped/ignored.

9.3.2.2.1 UDP Settings

For UDP packets the S132 can be programmed to perform the following BID comparison rules using PC.UBIDLS, PC.UBIDLCE and B.BCDR4.RXUBIDLS.

A) Always 16-bit and accepted in either the UDP Source or Destination position (automatic position detection)
B) Always 16-bit and only accepted in the UDP Port position according to B.BCDR4.RXUBIDLS (set per Bundle)
C) Always 16-bit in the UDP Destination Port position
D) Always 16-bit in the UDP Source Port position
E) Always 32-bit in the combined Source and Destination Port positions

The UDP OAM BID comparison rules follow the BID comparison rules except when the “per Bundle setting” using B.BCDR4.RXUBIDLS (rule “B” above) is enabled. When this rule has been enabled, the S132 tests for 16-bit OAM BIDs in either UDP Port position (rule “A” above; auto detected).

If a matching BID or OAM BID is not found in the location specified by these registers, the UDP packet is either forwarded to the CPU or discarded (PC.CR1.DPS6).

The “16-bit Auto-detected” and “Per-Bundle 16-bit using RXUBIDLS” settings are designed to allow a mixture of PWs with the BID in the UDP Source Port location and other PWs with the BID in the UDP Destination Port location. The “16-bit Per-Bundle using RXUBIDLS” setting requires that the BID location is programmed for all Bundles (RXUBIDLS). The “16-bit Auto-detected” does not use a location setting, but rather tests both locations accepting a match in either location.

When the S132 is programmed to use 32-bit BIDs, the UDP 16-bit Source and 16-bit Destination Port values are combined into a single 32-bit value in the same order in which they are received (the Source Port becomes the 16 MSbits for the 32-bit BID). The 32-bit setting is also applied to the OAM BIDs so that there is only one accepted bit-width for all BIDs and OAM BIDs (either all are 16-bit or all are 32-bit).

The S132 can ignore any of the UDP PW-ID bit positions from bit-0 to bit-15 using PC.CR20.UBIDM. This can be used to support a smaller UDP BID bit-width (e.g. bits 0 – 11), or to mask particular bit positions. As an example, with UBIDM = 0xF0FF the Classifier will match any received PW-ID = 0xCZ00 (where Z = any hex value 0 to F) with BID = 0xC000 (bits 8 – 11 are ignored). When using the 32-bit BIDs, bits 16 – 31 cannot be masked.

The S132 can verify each received UDP Protocol Type field against either of two programmed values (PC.CR2.UPVC1 and PC.CR2.UPVC2) or can ignore the UDP Protocol Type (PC.CR1.UPVCE). When enabled, the UDP Protocol Type is tested in the UDP Source or Destination Port location, whichever location is not used by the BID/OAM BID (e.g. if the BID is tested in the Source location, the Protocol Type is tested in the Destination location). For UDP packets that match a BID but do not include the correct UDP Protocol Type (when UDP Protocol testing is enabled), PC.CR1.DPS5 determines whether the packet is discarded or sent to the CPU.

The UDP Protocol Type is ignored (not tested), regardless of the UPVCE and DPS5 settings, for 3 conditions: when no matching BID/OAM BID is found, when a matching OAM BID is found and when using the 32-bit BID mode.

PC.SRL.UPVCSL and PC.SRL.UBIDLCSL can be used as debug tools to monitor the UDP BID location and Protocol Type value are correct. These status indications are available with all of the UDP BID test modes. However, the UBIDLCSL status only indicates whether the UDP BID was found in the location specified by the RXUBIDLS for each Bundle (regardless of the BID Test Mode setting). This means that for BID Test modes A, C, D and E the UBIDLCSL status may not agree with the results of the enabled BID Test mode (BID Test Modes A, C, D and E do not use the RXUBIDLS settings to determine where to look for the BID).

9.3.2.2.2 Handling of Packets with a Matching BID or OAM BID

When a packet matches an OAM BID (any PW Header type), then the packet is either forwarded to the CPU or discarded (PC.CR1.DPS7). The Classifier does not regard the remaining header fields.

If a BID match is found, then the PW Header Type that is programmed for that Bundle is verified (B.BCDR4.RXHTS). If the PW Header Type does not match, the packet is discarded.

When the PW-ID and PW Header Type match that of a programmed Bundle the Classifier can optionally verify the functions identified in Table 9-9. If the packet passes these tests the Classification process continues.

Table 9-9. Malformed PW Header Handling (not including the UDP specific settings)

Test Description	Functional Settings	Special "Fail" Setting	"Fail" Result
Wrong Payload Size	B.BCDR1.PMS, B.BCDR1.SCSCFPD	-	Discard
RTP Header existence	B.BCDR4.RE	-	Discard
Control Word Header existence	B.BCDR4.CWE	-	Discard
Wrong # L2TPv3 Cookies or MPLS Labels	B.BCDR4.RXLCS	PC.CR1.DPS10	Discard/CPU

The first nibble of the Control Word of a PW packet is used to identify whether the packet payload carries data that is destined for a TDM Port or that is destined for the CPU (In-band VCCV OAM). When the first nibble is equal to 0x0 the payload is destined for the TDM Port. The Classifier can be programmed to monitor for In-band VCCV packets by enabling the Control Word monitoring function (B.BCDR4.RXOICWE) and by specifying the Control Word value that is expected. The commonly used nibble value for (CPU) In-band VCCV packets is 0x1. When RXOICWE is enabled, the first byte of each received Control Word is compared against PC.CR5.VOV using PC.CR5.VOM to specify how many bits are to be Masked/Ignored. If the first byte matches VOV and VOM, then the packet is forwarded to the CPU. If the first byte does not match VOV and VOM, and the first nibble is not 0x0 the packet is discarded. In-band VCCV can be enabled (per Bundle) for SAT, CES, HDLC and Clock Only Bundles.

If a packet matches all of the Bundle test conditions so far described and has not already been discarded or forwarded to the CPU, the packet payload/information is forwarded to the HDLC Engine, SAT/CES Engine, Clock Recovery or CPU. Table 9-10 identifies the register settings that are required for each Connection/Bundle Type that is listed.

Table 9-10. Bundle Forwarding Options

Connection/Bundle Type	Destination setting B.BCDR4.RXBDS	Engine Type setting B.BCDR1.PMT	Clock Recovery setting B.BCDR4.PCRE
SAT/CES Payload Only	TDM Port	SAT/CES	disable
SAT/CES Payload & PW-Timing	TDM Port	SAT/CES	enable
Clock Only (PW-Timing only)	Discard	SAT/CES	enable
HDLC	TDM Port	HDLC	disable
CPU Debug RXP PW Bundle[1]	CPU	HDLC or SAT/CES	enable/disable

Note: [1] The SAT/CES and HDLC PW packets can be diverted from their "normal destination" and forwarded to the CPU for debug purposes by setting the RXBDS to send the packet to the CPU.

The forwarding of payload data for RXP SAT/CES Payload Connections and HDLC Connections can be disabled using B.BCDR4.RXBDS = 11. When a Payload Connection is disabled and the Jitter Buffer for that Bundle is empty, the data that is transmitted at the TDM Port is filled with Conditioning Data. The PW-Timing Connection is disabled using B.BCDR4.PCRE (it is not disabled using B.BCDR4.RXBDS).

Special considerations

Each programmed BID and OAM BID value must be unique across all PW Header types. For example one Bundle cannot use BID = "17" with UDP and another Bundle use BID = "17" with L2TPv3 (OAM BID = "17" would also not be allowed). In systems that are unable to co-ordinate the assignment of PW-IDs across all supported PW Header Types, only one PW Header Type should be used by the S132 to insure unique BIDs and OAM BIDs.

9.3.2.2.3 L-bit Signaling for RXP PWs

The L-bit in the Control Word of a PW packet can be used to indicate across the PSN when a T1/E1 fault has been detected. The CPU can monitor for received L-bit changes for each RXP Bundle using the G.GCR.LBCDE, B.G0SRL.CWCDSL - B.G31SRL.CWCDSL, B.G0SRL.CWCDIE - B.G31SRL.CWCDIE, G.GSR5.BGS, G.GSR1.BS and G.GSRIE1.BIE Registers. This is explained in more detail in the Monitor & Interrupt section.

The S132 can also be programmed to automatically discard the RXP Packet payload when the received L-bit = 1 (invalid payload) and B.BCDR1.LBCAI = 1 (conditioning for L-bit = 1).

9.3.2.3 CPU Packet Classification

Packets that are not identified as PW packets are further processed according to the rules described in this section to determine whether they are to be sent to the CPU (or discarded). The previously described "send to CPU" conditions in the "Generalized Packet Classification" and "PW Packet Classification" sections are also repeated in this section so that all “send to CPU conditions” are described together in one section (e.g. OAM BIDs, In-band VCCV, “CPU Debug RXP PW Bundle” and error condition Discard Switches).

9.3.2.3.1 Packets with Broadcast Ethernet DA (DPC.CR1.DPBTP and DPC.CR1.DPBCP)

When an Ethernet packet is received with the Ethernet Broadcast Destination Address (DA) and the packet includes one of the PW Header Types, the PC.CR1.DPBTP setting determines whether the packet is sent to the CPU (0) or Discarded (1).

When an Ethernet packet is received with the Ethernet Broadcast DA and the packet doe not include one of the PW Header Types, the PC.CR1.DPBCP setting determines whether the packet is sent to the CPU (0) or Discarded (1).

9.3.2.3.2 Packets with Unknown Ethernet DA (PC.CR7 – PC.CR19 and DPC.CR1.DPS9)

When an Ethernet packet is received and the packet includes an Ethernet DA that is not recognized (not equal to PC.CR7 – PC.CR12, PC.CR13 – PC.CR19 or the Ethernet Broadcast Address), the PC.CR1.DPS9 setting is used to determine whether the packet is forwarded to the CPU (0) or Discarded (1). Packets with the Ethernet Broadcast DA are regarded as having a “known” address and are not affected by the DPS9 setting.

If the Ethernet DA registers are not programmed (PC.CR7 – PC.CR19; DA values in their default state = “0”) the combined settings of PC.CR1.DPS9, PC.CR1.DPBTP and PC.CR1.DPBCP can be used to specify that all valid Ethernet packets that do not use the “0” DA value are forwarded to the CPU (0) or Discarded (1).

9.3.2.3.3 PW Packets with Unknown PW-ID (DPS6)

When a packet is received with a recognized PW Header (MEF-8, MFA-8, UDP or L2TPv3) but the received PW-ID does not match any of the programmed BIDs or OAM BIDs, the PC.CR1.DPS6 setting determines whether the packet is forwarded to the CPU (0) or Discarded (1).

9.3.2.3.4 MEF OAM Ethernet Type Packets (MOET)

MEF OAM Ethernet Type packets are recognized when the received Ethernet Type field is equal to the programmed PC.CR4.MOET. The PC.CR1.DPS7 setting determines whether the packet is forwarded to the CPU (0) or Discarded (1).

9.3.2.3.5 CPU Destination Ethernet Type Packets (CDET and DPS8)

When an Ethernet packet is received with an Ethernet Type field that is equal to PC.CR20.CDET, the PC.CR1.DPS8 setting determines whether the packet is forwarded to the CPU (0) or Discarded (1).

9.3.2.3.6 ARP Packet with Known IP Destination Address (PC.CR6 – PC.CR8 and DPS3)

When an ARP packet is received (Ethernet Type = 0x0806) with an IP Destination Address that equals one of the IPv4 addresses programmed at PC.CR6 – PC.CR8, the PC.CR1.DPS3 setting determines whether the packet is forwarded to the CPU (0) or Discarded (1).

9.3.2.3.7 ARP Packet with Unknown IP Destination Address (PC.CR6 – PC.CR8 and DPS0)

When an ARP packet is received (Ethernet Type = 0x0806) with an IP Destination Address that is not equal to one of the IPv4 addresses programmed at PC.CR6 – PC.CR8, the PC.CR1.DPS0 setting determines whether the packet is forwarded to the CPU (0) or Discarded (1).

9.3.2.3.8 Packet with Unknown Ethernet Type (DPS2)

When an Ethernet packet is received with an Ethernet Type field that is not recognized (not equal to ARP, Unicast MPLS, Multicast MPLS, IPv4, IPv6, PC.CR20.CDET, PC.CR4.MET or PC.CR4.MOET), the PC.CR1.DPS2 setting determines whether the packet is forwarded to the CPU (0) or Discarded (1).

9.3.2.3.9 IP Packets with Unknown IP Protocol (DPS4)

When an IP packet is received with an IP Protocol field that is not UDP or L2TPV3, the PC.CR1.DPS4 setting determines whether the packet is forwarded to the CPU (0) or Discarded (1).

9.3.2.3.10 IP Packet with Unknown IP Destination Address (PC.CR6 – PC.CR16 and DPS1)

When an IP packet is received with an IP DA that is not equal to one of the IP addresses programmed at PC.CR6 – PC.CR16, the PC.CR1.DPS1 setting determines whether the packet is forwarded to the CPU (0) or Discarded (1). The IP version that is recognized is selected with PC.CR1.RXPIVS and PC.CR1.RXPDSD.

9.3.2.3.11 "CPU Debug RXP PW Bundle" Setting (RXBDS)

PW Bundles (not including OAM Bundles) are normally used for SAT, CES, HDLC or PW-Timing Connections, but can be programmed to instead send packets to the CPU for debug. When the CPU Debug setting is enabled (B.BCDR4.RXBDS), the received packets for the RXP Bundle are redirected to the CPU (instead of sending the data to the SAT/CES/HDLC/Clock Recovery Engines). The RXP Bundle parameters can be fully programmed or partially programmed. A received packet is identified as a "CPU Debug RXP PW Bundle" packet when the packet includes any of the PW Header Types, the PW-ID of the packet matches a BID and the Bundle that uses that BID is programmed to "CPU Debug" (RXBDS). The other Bundle register settings are ignored.

9.3.2.3.12 PW Bundle with Unknown UDP Protocol Type (UPVCE and DPS5)

When the Classifier is programmed to verify the UDP Payload Protocol (PC.CR1.UPVCE) and a UDP packet is received with a recognized BID, but with a UDP Payload Protocol value that is not equal to PC.CR2.UPVC1 or UPVC2, PC.CR1.DPS5 selects whether the packet is sent to the CPU (0) or Discarded (1). The DPS5 setting does not affect packets that are otherwise identified as CPU packets.

9.3.2.3.13 PW Bundle In-band VCCV OAM (RXOICWE and DPS7)

In-band VCCV CPU Connections can be thought of as "secondary" connections that are used to support the "primary" SAT/CES/HDLC/Clock Only PW for functions like setup, configuration and monitoring. An In-band VCCV connection can be established before the primary connections have been established. The In-band VCCV may be used, e.g., to negotiate the configuration settings of the primary connection before enabling the primary connection. The Classifier monitors for In-band VCCV packets for a Bundle when B.BCDR4.RXOICWE = 1. The PC.CR1.DPS7 setting determines whether In-band VCCV packets are forwarded to the CPU (0) or Discarded (1).

9.3.2.3.14 PW Bundle with Too Many MPLS Labels (DPS10)

When an MFA-8 (MPLS) packet is received with a recognized BID and the packet includes more than 2 MPLS Labels, PC.CR1.DPS10 determines whether the packet is forwarded to the CPU (0) or Discarded (1).

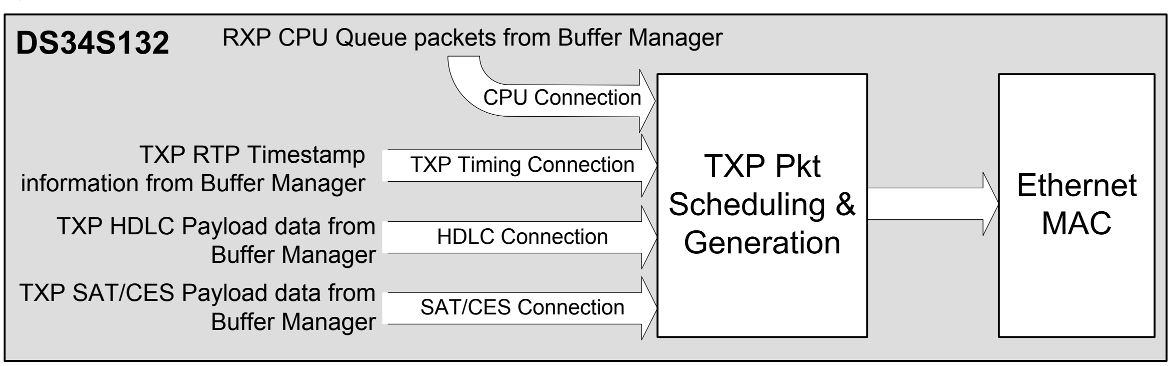
9.3.2.3.15 PW OAM Bundle - Out-band VCCV OAM Packets (DPS7)

Up to 32 Out-band VCCV OAM Connections can be programmed using OAM BIDs. OAM BIDs are used to support what the standards call "UDP-specific OAM", "Out-band VCCV" or "OAM using Separate PW-ID" (meaning OAM PW-IDs that are separate/unique from the PW-IDs used by the primary PW connection). The UDP application commonly uses this OAM form instead of the "In-band VCCV" form. This OAM format is not commonly used with L2TPv3, MEF-8 or MFA-8. A packet is recognized as an OAM Bundle when the received packet includes a one of the PW Header Types and the received PW-ID matches one of the 32 programmed OAM BIDs. The PC.CR1.DPS7 setting determines whether this packet type is forwarded to the CPU (0) or Discarded (1).

9.3.3 TXP Packet Generation

The TXP Packet Generator schedules the packet data for CPU, PW-Timing, HDLC and SAT/CES Payload Connections and appends the TXP Header (including FCS field values when required) and TXP Timestamp (when required). The Ethernet FCS is appended outside this block in the Ethernet MAC block.

Figure 9-24. TXP Packet Generation Environment



9.3.3.1 TXP SAT/CES/HDLC/Clock Only PW Packet Generation

A TXP Header Descriptor is programmed for each activated SAT, CES, HDLC and Clock Only Bundle (up to 256). The TXP Header Descriptors are retrieved from memory as they are needed for each outgoing TXP PW Packet.

Figure 9-25 depicts the format of the data that is programmed in the TXP Header Descriptor. The Header Control is used to identify the number of bytes included in the transmitted TXP Header and where the TXP Local Timestamp, Length and FCS fields are located in the header so that the S132 can modify these fields "on-the-fly" when required. The Header Control field values are not included in the transmitted Ethernet packets.

Figure 9-25. SAT/CES/HDLC/Clock Only PW TXP Header Descriptor

31–16	15–0
Header Control	
2-bytes of Dummy Fill = 0x00.00	Ethernet Header including …
… DA, SA, optional VLAN Tags, optional LLC/SNAP and Type/Length fields	
Application Protocol Header (e.g. UDP/IP, L2TPv3/IP, MFA-8, MEF-8)	
Optional RTP and Control Word Headers	

The 256 TXP Header Descriptors are programmed in an SDRAM memory block that begins at address EMI.BMCR1.TXHSO. The TXP Header Descriptor for each Bundle is stored in a 128-byte SDRAM slot that is addressed/indexed at the location = TXHXSO + (Bundle Number * 128 bytes).

The TXP Header Control for SAT/CES and HDLC PW packets is a 32-bit Dword (depicted in Table 9-11.

Table 9-11. TXP SAT/CES/HDLC/Clock Only PW Header Control

Field	**Bit [x:y]**	Description
RSVD	[31:26]	**Reserved.**
TXELEN	[25:21]	**TXP Header Length** specifies how many Dwords are in the packet header including the Ethernet, Application, RTP and Control Word Headers (when applicable).
TXCWE	[20]	**TXP Control Word Exists.** 1 = included; 0 = not included.
TXRE	[19]	**TXP RTP Exists.** 1 = included; 0 = not included.
RSVD	[18:16]	**Reserved.**
TXALEN	[15:11]	**TXP Application Header Length** specifies how many Dwords are included in the Application Protocol Header beginning just after the Ethernet Header and including the Control Word and RTP Headers (when applicable).
TXAOFF	[10:6]	**TXP Application Header Offset** = Dword offset of Application Header in Ethernet packet. TXAOFF = ("Application Header starting byte position in Ethernet packet" - 2) ÷ 4
TXUDPE	[5]	**TXP UDP Header Exists.** 1 = included; 0 = not included.
TXIPV6E	[4]	**TXP IPv6 Header Exists.** 1 = included; 0 = not included.
TXIPV4E	[3]	**TXP IPv4 Header Exists.** 1 = included; 0 = not included.
TXVLTC	[2:1]	**TXP VLAN Tag Count** specifies # VLAN tags in the header (valid values = 0, 1 or 2).
TXETHF	[0]	**TXP Ethernet Header Format.** 0 = DIX/Ethernet II format; 1 = IEEE 802.2 LLC/SNAP.

The S132 automatically generates TXP SAT/CES/Clock Only packets when sufficient data has been received from the TDM Port to satisfy the B.BCDR1.PMS (effective payload size) and B.BCDR3.TXBTS (payload type) settings. All SAT/CES/Clock Only Bundles must be assigned at least one Timeslot (B.BCDR2.ATSS and TSAn.m). B.BCDR3.TXPMS selects whether the packet stream for the TXP Bundle is disabled, transmitted without payload (Clock Only) or transmitted with payload (normal).

TXP SAT/CES/Clock Only Packets can optionally include an RTP Timestamp using the RTP Exists field in the TXP Header Descriptor Header Control (RTP is not commonly used with HDLC Bundles).

A TXP HDLC packet is generated for each HDLC packet that is received from the TDM Port with a correct HDLC FCS value. The HDLC packet encoding is removed before the TXP Header is appended. HDLC packets that are received with bad HDLC FCS values are discarded. HDLC TXP Packet transmission can be enable/disabled using B.BCDR3.TXPMS.

The IP Length, IP FCS, UDP Length and UDP FCS fields are auto generated for SAT, CES, HDLC and Clock Only packets when these fields are enabled by the TXP Header Descriptor Header Control.

9.3.3.1.1 L-bit Signaling

The L-bit in the Control Word of a PW packet can be used to indicate, across the PSN, when the data contained in a TDMoP PW payload may be corrupted (e.g. for a T1/E1 LOS condition, L-bit = 1). The Pn.PRCR1.LBSS register selects whether the L-bit in each TXP Packet is controlled on a "per-Bundle basis" using the TXP Header Descriptor or on a "per-TDM Port basis" using Pn.PRCR1.LB. When the "per-Bundle" method is selected, the CPU must modify all of the programmed TXP Header Descriptors that are associated with a TDM Port that requires an L-bit change. When the "per-TDM Port" method is selected, changing Pn.PRCR1.LB changes the L-bit value in all TXP Packets for that TDM Port.

The standards allow TXP SAT/CES PW packets, to optionally truncate/remove the payload section when the TXP L-bit = 1 to save network bandwidth during receive TDM fault conditions (detected by the external TDM Port Framer/LIU). B.BCDR3.TXPMS can be programmed to "transmit without payload", so that the TXP Bundle packet transmit rate does not change but with a smaller packet size (like that of a Clock Only packet).

9.3.3.2 TXP CPU Packet Generation

The generation of TXP Bundle packets is described in the "TXP CPU Packet Interface" section.

9.3.3.3 TXP Packet Scheduling

The transmit PDV for Bundles that are used for clock recovery can be minimized to improve the clock recovery performance at the far end by programming the TXP Bundle with the high scheduling priority (B.BCDR3.TXBPS) and, for networks that support VLAN CoS, by assigning a high P-bit priority in the VLAN tag (the P-bit value is provided by the CPU in the TXP Header Descriptor; high priority packets are processed before low priority packets). Bundles that can be used for Clock Recovery include SAT/CES Bundles with payload and Clock Only Bundles without payload. The TXP Clock Only Bundle is designed to provide the best possible transmit PDV and latency by suppressing the payload. HDLC Bundles should normally be assigned low priority (B.BCDR3.TXBPS).

9.3.3.4 TXP Packet Queue Monitoring

The TXP Packet Queue fill levels can be monitored using the G.TPISR1 (TXP Bundle High Priority Queue), G.TPISR2 (TXP Bundle Low Priority Queue) and G.TPISR3 (TXP CPU Queue) registers. Each of these queues also provides a maskable interrupt using G.TPISRL.HPQOSL, G.TPISRL.LPQOSL and EMA.WSRL1.WFOSL.

9.4 CPU Packet Interface

- Up to 512 stored RXP CPU packets
- RXP CPU packet size up to 2000 bytes
- RXP Local Timestamp
- RXP Packet Classification Results
- Up to 512 stored TXP CPU packets
- TXP CPU packet size up to 2000 bytes
- TXP RTP (OAM) Timestamp generation

RXP CPU Packets that are received from the Ethernet Port are stored in an SDRAM RXP CPU Queue for the CPU to Read. The CPU Writes TXP CPU Packets into an SDRAM TXP CPU Queue that are later transmitted at the Ethernet Port. The depth of the RXP CPU FIFO and TXP CPU Queues are programmed at EMI.BMCR3.PRSO and EMI.BMCR3.PRSO.

9.4.1 RXP CPU Packet Interface

RXP CPU Packets that are received at the Ethernet Port are stored in 2 Kbyte slots in the SDRAM RXP CPU Queue. The S132 stores an RXP CPU Header Descriptor with each RXP CPU packet to provide information about the CPU Packet. The format of the packet and Header Descriptor are provided in Figure 9-26 and Table 9-12 through Table 9-14.

Figure 9-26. Stored RXP CPU Packet

31 … 16	15 … 0
3-Dword RXP Header Descriptor	
2-bytes of Dummy Fill = 0x00.00	Entire RXP CPU packet from …
… the Ethernet Destination Address to the end of Ethernet Payload but not including the Ethernet FCS	

Table 9-12. RXP CPU Header Descriptor – 1st Dword

Field	**Bit [x:y]**	Description
RXPLEN	[31:21]	**RXP Packet Length.** The length (in bytes) of the complete RXP CPU Packet from the Ethernet DA to the end of the Ethernet Payload (not including the Ethernet FCS).
RXNBP	[20]	**RXP Non-Bundle Packet.** 0 = packet matches a Bundle; 1 = not a packet for a Bundle.
RSVD	[19:11]	**Reserved.**
RXRE	[10]	**RXP RTP Exists.** 1 = RTP Header is included.
RSVD	[9:8]	**Reserved.**
TBN	[7:0]	**TDM Bundle Number.** When RXNBP = 0, these bits identify the Bundle Number.

Table 9-13. RXP CPU Header Descriptor – 2nd Dword

Field	Bit [x:y]	Description
RSVD	[31:30]	**Reserved.**
RXIPV6	[29]	**RXP IPv6 packet.** 1 = IPv6 Header; 0 = not IPv6.
RXIPV4	[28]	**RXP IPv4 packet.** 1 = IPv4 Header; 0 = not IPv4.
RXMO	[27]	**RXP MEF OAM packet.** 1 = MEF OAM Header; 0 = not MEF OAM.
RXIVO	[26]	**RXP In-band VCCV OAM packet.** 1 = In-band VCCV Header; 0 = not In-band VCCV.
RXMLC	[25:24]	**RXP MPLS Label Count.** 1 = number of outer MPLS labels.
RXLSF	[23]	**RXP LLC/SNAP Format.** 1 = IEEE 802.2 LLC/SNAP Header; 0 = not LLC/SNAP.
RXDF	[22]	**RXP DIX Format.** 1 = DIX Header; 0 = not DIX.
RSVD	[21]	**Reserved.**
RXL2TP	[20]	**RXP L2TPv3 packet.** 1 = L2TPv3 Header; 0 = not L2TPv3.
RX2VT	[19]	**RXP 2 VLAN Tagged packet.** 1 = 2 VLAN tags in header; 0 = does not have 2 tags.
RXVT	[18]	**RXP VLAN Tagged packet.** 1 = 1 or 2 VLAN tags in header; 0 = no VLAN tags.
RXUDP	[17]	**RXP UDP packet.** 1 = UDP Header; 0 = not UDP.
RXIP	[16]	**RXP IP packet.** 1 = IPv4 or IPv6 Header; 0 = not IPv4 or IPv6.
RXMEF	[15]	**RXP MEF packet.** 1 = MEF Header; 0 = not MEF.
RXMPLS	[14]	**RXP MPLS packet.** 1 = MPLS Header; 0 = not MPLS.
RSVD	[13:11]	**Reserved.**
RXMLE	[10]	**RXP MPLS Label Error.** 1 = more than 3 MPLS labels; 0 = not more than 3 labels.
RXUEDA	[9]	**RXP Unknown Ethernet DA.** 1 = unknown Ethernet DA; 0 = recognized Ethernet DA.
RXCET	[8]	**RXP CPU Ethernet Type.** 1 = "CPU Destination" Ethernet Type.
RXOVO	[7]	**RXP Out-band VCCV OAM.** 1 = Out-band VCCV Header (matches OAM BID).
RXUPW	[6]	**RXP Unknown PW.** 1 = packet with a PW Header Type but with unknown PW-ID.
RXUUP	[5]	**RXP Unknown UDP Protocol.** 1 = UDP with unknown UDP protocol.
RXUIPP	[4]	**RXP Unknown IP Protocol.** 1 = IP with unknown IP protocol.
RXRARP	[3]	**RXP Recognized ARP packet.** 1 = ARP Ethernet Type with recognized IP DA.
RXUET	[2]	**RXP Unknown Ethernet Type.** 1 = unknown Ethernet Type
RXUIPA	[1]	**RXP Unknown IP DA.** 1 = IP with unknown DA
RXUARP	[0]	**RXP Unknown ARP packet.** 1 = ARP Ethernet Type with unknown IP DA.

Table 9-14. RXP CPU Header Descriptor – 3rd Dword

Field	Bit [x:y]	Description
RXLTS	[31:0]	**RXP Local Timestamp.** 32-bit Timestamp with 100 us or 1 us resolution (G.GCR.OTRS), latched at the time the packet is received by the Packet Classifier.

The RXP Local Timestamp may be used by the CPU for OAM Timestamp purposes (not for clock recovery).

RXP CPU Packets are first stored in the SDRAM RXP CPU Queue. The CPU controls the transfer of each RXP CPU packet to an internal staging RXP CPU FIFO that the CPU can read from. The FIFO holds one RXP CPU Packet at a time. The RXP CPU Queue can hold up to 512 packets (each 2Kbyte slot of the RXP CPU Queue is reserved for one packet).

The CPU process of Reading the RXP CPU packets can be polling based using the EMA.RSR1.RFRS status bit or interrupt driven using the EMA.RSRL1.RFRSL (latched status) and EMA.RSRIE1.RFRIE (Interrupt enable) register bits. When the CPU detects that a packet is waiting in the RXP CPU FIFO, the CPU must specify the read operation (EMA.RCR.RPCRC = 110b), specify the read transfer length in Dwords (EMA.RCR.TL) and then begin reading the data at EMA.RDR.EMRD. The EMA.RCR.TL value specifies how many Dwords are transferred from the RXP CPU Queue to the RXP CPU FIFO.

The smallest possible RXP Packet Read is 19 Dwords for a 64-byte Ethernet Packet with the 4-byte FCS removed, 3-Dword Header Descriptor and 2-byte Dummy Fill appended to the beginning of the packet. The initial Transfer Length for each packet can be any value from 1 to 18. The first Dword of the Header Descriptor that is Read by the CPU identifies the length of the RXP CPU Packet. This is used to determine how many remaining Dwords must be transferred from the RXP CPU Queue to the RXP CPU FIFO and then Read by the CPU. Each successive Read Transfer at EMA.RDR.EMRD causes the S132 to update the register with the next Dword in the RXP CPU FIFO.

The EMA.RSR1, EMA.RSR2, EMA.RSRL1 and EMA.RSRIE1 registers provide other control and status bits for the SDRAM RXP CPU Queue and the RXP CPU FIFO.

9.4.2 TXP CPU Packet Interface

The CPU writes each TXP CPU packet into an S132 staging TXP CPU FIFO and then controls the Writing (transfer) of the packet to the TXP CPU Queue in the SDRAM. The TXP CPU FIFO can hold 1 packet. The TXP CPU Queue can hold up to 512 packets. The S132 transmits each packet in the TXP CPU Queue when the Ethernet Port is not busy transmitting PW packets.

The TXP CPU packets from the CPU must include all of the fields that will be transmitted at the Ethernet Port including the Ethernet and Application Headers, but not including the Ethernet FCS. Each TXP CPU packet can be 2 Kbytes in length. The CPU must also append a TXP Header Descriptor to the beginning of each packet with information about the packet. The format of the packet and TXP Header Descriptor are provided in Figure 9-27.

Figure 9-27. Stored TXP CPU Packet and Header Descriptor

31 … 16	15 … 0
1 Dword - Header Control	
2-bytes of Dummy Fill = 0x00.00	Entire TXP CPU packet from …
… the Ethernet Destination Address to the end of Ethernet Payload but not including the Ethernet FCS	

The TXP CPU Header Control is a single 32-bit Dword as depicted in Table 9-15.

Table 9-15. TXP CPU Header Control

Field	**Bit [x:y]**	Description
TXPLEN	[31:21]	**TXP Packet Length.** The length (in bytes) of the complete TXP CPU Packet from the Ethernet DA to the end of the Ethernet Payload (not including the Ethernet FCS).
TXOTSO	[20:12]	**TXP OAM Timestamp Offset** = Dword position for TXP OAM Timestamp in Ethernet packet. TXOTSO = ("Timestamp starting byte position in Ethernet packet" - 2) ÷ 4
TXOTSE	[11]	**TXP OAM Timestamp Enable.** 1 = insert TXP OAM Timestamp; 0 = disabled.
TXAOFF	[10:6]	**TXP UDP/IP Application Offset** = Dword position of IP Header in Ethernet packet. TXAOFF = ("IP Header starting byte position in Ethernet packet" - 2) ÷ 4
TXUDP	[5]	**TXP UDP Header FCS Modify Enable.** 1 = insert UDP FCS (only valid if TXIPV4 = 1 or TXIPV6 = 1).
TXIPV6	[4]	**TXP IPv6 Header Exists.** 1 = header includes IPv6; 0 = not IPv6.
TXIPV4	[3]	**TXP IPv4 Header Exists.** 1 = header includes IPv4 (S132 will insert IP FCS); 0 = not IPv4.
RSVD	[2:0]	**Reserved.**

The S132 can be programmed to add a 32-bit TXP OAM Timestamp to a TXP CPU Packet. One example use for the TXP OAM Timestamp is described in RFC5087 Appendix D (TDMoIP Performance Monitoring Mechanisms).

When TXOTSE is enabled, TXOTSO identifies the OAM Timestamp Dword position in the packet. The CPU must make the initial OAM Timestamp value 0x0000 in the packet stored in the TXP CPU FIFO. The S132 overwrites that position with the current OAM Timestamp value as the packet is being transmitted at the Ethernet Port.

The S132 uses the ETHCLK signal to generate the TXP OAM Timestamps. The TXP OAM Timestamp Resolution can be programmed for 1 us or 100 us using G.GCR.OTRS.

The OAM Timestamp can only be positioned on Dword boundaries within the Application Header. Because Ethernet Headers do not commonly include an integer number of Dwords, the Application Header is commonly offset by 2 bytes in the Ethernet packet, as depicted in Figure 9-25 (Ethernet packets do not include the “dummy” bytes depicted in this figure). If the OAM Timestamp position is referenced instead to the beginning of the Ethernet packet, some example OAM Timestamp byte positions are 46, 50, 54 (etc.), which would equate to TXOTSO Dword = 11, 12 , 13 (etc.; respectively).

When a TXP CPU Packet uses the IPv4 protocol, the S132 can be programmed to assist with the generation of the IPv4 Header FCS. The CPU must pre-calculate and include an IP FCS for all of the fields of the IP Header but the IP Total Length field. The S132 modifies the IP FCS on-the-fly to include the IP Total Length. This allows the CPU to store a “generic” pre-calculated IP FCS with each stored IP Header in CPU memory. The CPU pre-calculates the IP FCS for the IP Header beginning with the IP Version field and ending with the IP Destination Address, but using “0” values in the IP Total Length and IP Header FCS fields. The IPv6 Header does not include an FCS.

When a TXP CPU Packet uses the UDP/IP protocol, the S132 can be programmed to assist with the generation of the FCS in the UDP Header. The CPU must include a UDP FCS for all but the IP length and UDP length fields. The CPU pre-calculates the UDP FCS with a Pseudo IP Header that is added to the beginning of the UDP packet (added only for this UDP FCS calculation) and including the entire UDP packet (from UDP Source Port to the end of the UDP Payload; per RFC768), but calculates with “0” values in the “UDP Length” field of the Pseudo IP Header, the ”Length” field of the UDP Header and the “Checksum” field of the UDP Header. The S132 modifies the pre-calculated UDP FCS on-the-fly to include the lengths. This function can be used when the S132 is programmed to add a TXP OAM Timestamp.

If the S132 FCS functions are not needed, then the CPU should not identify the packet as IPv4 or UDP (so that the S132 does not modify the FCS values) and the CPU must include the IP and UDP FCS values for transmission.

All of these functions can be enabled at the same time or in various combinations as identified in Table 9-16. When the CPU is ready the CPU writes the entire TXP CPU Packet including the pre-calculated FCS values, the “real” packet length values and TXP OAM Timestamp = “0” (when applicable). For example, when TXOTSO = 1, TXUDP = 1 and TXIPV4 = 1 (Add TXP OAM Timestamp & Re-calculate UDP FCS & IPv4 FCS), the CPU provides the entire CPU packet (from Ethernet DA to the end of the Ethernet Payload) including the IP Total Length field (to indicate the size of the IP packet), the pre-calculated IP FCS, the UDP Length field (to indicate the size of the UDP packet), the pre-calculated UDP FCS and the “0” value in the TXP OAM Timestamp position. The S132 then overwrites with the TXP OAM Timestamp and corrects the IP FCS and UDP FCS values.

Table 9-16. Modify FCS and Add TXP OAM Timestamp Functions

Application	TXOTSE	TXUDP	TXIPV6	TXIPV4
Re-calculate IPv4 FCS to include Length	0	0	0	1
For IPv4: Re-calculate UDP FCS & IPv4 FCS to include Length	0	1	0	1
For IPv6 packets: Re-calculate UDP FCS to include Length	0	1	1	0
Add TXP OAM Timestamp to any protocol (with no FCS modifications)	1	0	0	0
For IPv4: Add TXP OAM Timestamp & Re-calculate UDP FCS & IPv4 FCS	1	1	0	1
For IPv6: Add TXP OAM Timestamp & Re-calculate UDP FCS	1	1	1	0

The Write TXP CPU Packet process can be polling based using the EMA.WSR1.WFES status bit or interrupt driven using the EMA.WSRL1.WFESL (latched status) and EMA.WSRIE1.WFEIE (Interrupt enable) register bits. When the CPU is ready to Write a TXP CPU Packet into the TXP CPU FIFO, the CPU should begin by verifying that the TXP CPU FIFO is empty (read the FIFO status at EMA.WSR1.WFES or flush the FIFO with EMA.WCR.TPCWC = 3). The CPU then Writes the Dwords for the packet at EMA.WDR.EMWD. Each successive Write at EMA.WDR.EMWD fills the next Dword position in the FIFO. When the entire packet has been stored in the TXP CPU FIFO the CPU must indicate the length of the packet (EMA.WCR.TL), how many packet bytes are included in the last Dword (EMA.WCR.TLBE) and indicate that the packet should be transferred to the TXP CPU Queue in the SDRAM (EMA.WCR.TPCWC = 6). When the complete packet has been transferred to the TXP CPU Queue, EMA.WSR1.WFES will indicate that the FIFO is empty.

The EMA.WSR1, EMA.WSR2, EMA.WSRL1 and EMA.WSRIE1 registers provide other control and status bits for the TXP CPU FIFO and the SDRAM TXP CPU Queue.

9.5 Clock Recovery Functions

The S132 includes a DSP to implement its Clock Recovery functions. The Clock Recovery functions include the RXP and TXP PW-Timing functions. The DSP is controlled by firmware code. The firmware code must be downloaded to the S132 each time a global reset is initiated (e.g. after power up). In addition to the firmware code, the Clock Recovery functions must be programmed using the CR. Registers. The CR. Registers enable the PW-Timing functions to be configured according to each PW application (e.g. DCR-DT vs. ACR). The functionality of the firmware and its configuration registers is defined in an independent S132 Firmware Definition document.

9.6 Miscellaneous Global Functions

9.6.1 Global Resets

A Global Reset can be implemented using G.GRCR.RST or the RST_N pin.

9.6.2 Latched Status and Counter Register Reset

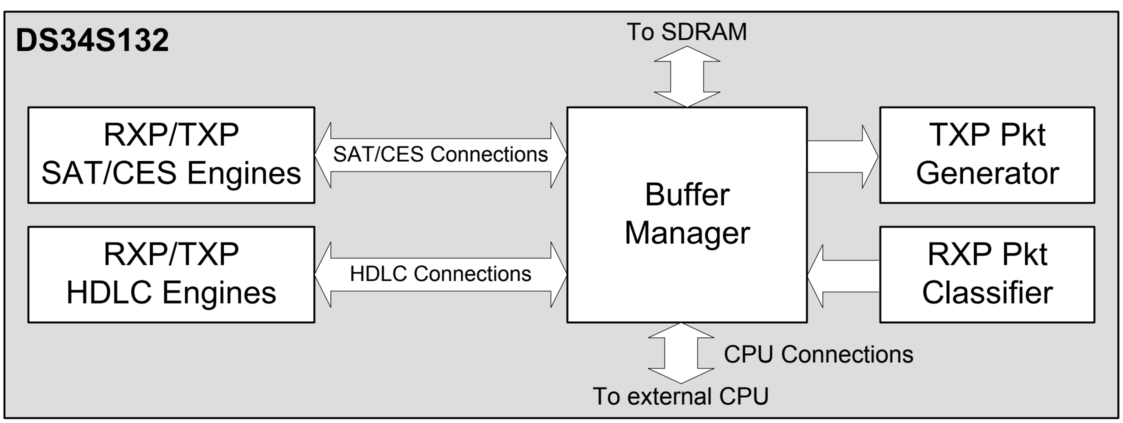
The S132 provides Latched Status register bits so that the CPU can discover transient events that might otherwise be missed by a simple "real-time" status register. Programming the G.GCR.LSBCRE register selects whether to clear (restore to the default value) the Latched Status bits automatically when the CPU Reads the Latched Status register, or to wait until the CPU performs an explicit Write operation to over-write the Latched Status value.

The G.GCR.CCOR bit selects whether the "Clear on Read" function is enabled for the RXP Bundle Counts, TXP Bundle Counts and Packet Classifier Counts or whether the "Clear on Read" function for these registers is disabled (the counters roll over after they reach their maximum value).

9.6.3 Buffer Manager

The Buffer Manager controls and monitors the SDRAM that stores the Bundle and RXP/TXP CPU Queues and the TXP Header Descriptors. The Buffer Manager environment is depicted in Figure 9-28.

Figure 9-28. Buffer Manager Environment



The starting addresses for the Queues and TXP Header Descriptor section are programmed using the EMI registers. Each address is a 16-bit address that indexes a 2 Kbyte segment of SDRAM memory (2^{16} x 2 Kbyte = 1 Gbit). For a smaller SDRAM size the address bit-width is reduced (e.g. a 512 Kbit SDRAM uses 15-bit addressing).

The programmed starting addresses are programmed using the following queue depth equations. The "Register Guide" section provides example settings that can be used in most applications.

RXP CPU Queue:	16384 * maximum # of RXP CPU Packets
TXP CPU Queue:	16384 * maximum # of TXP CPU Packets
TXP Header Descriptors:	1024 * maximum # of BIDs
TXP Bundle Payload Queues:	131072 * maximum # of BIDs
RXP Bundle Jitter Buffer Queues:	G.GCR.JBMD setting in Kbytes * maximum # BIDs
Total SDRAM storage area:	sum of all of the above

9.6.3.1 SDRAM Interface

The S132 has been designed to work with DDR-SDRAM devices that are compatible with the JEDEC JESD79C standard and that support 2-2-2 timing with a clock rate of at least 125 MHz. Table 9-17 identifies several SDRAM device sizes that can be used to support different Jitter Buffer Depth/Bundle Count combinations.

Table 9-17. SDRAM Device Selection Table

Single SDRAM Device Description			Qty SDRAM Devices per DS34S132	Total SDRAM bits per DS34S132	Max Configurable Values	
Array Size	Data Width	Targeted Vendor Part #[1]			Bundle Count	Jitter Buffer Depth (JBMD)
512 Mb	16 bit	Micron MT46V32M16P-6T or Samsung K4H511638B-TCB3	2	1 Gbit	256	256 Kbyte
512 Mb	16 bit	Micron MT46V32M16P-6T or Samsung K4H511638B-TCB3	1	512 Mbit	128	256 Kbyte
					256	128 Kbyte
256 Mb	16 bit	Micron MT46V16M16P-75E	1	256 Mbit	64	256 Kbyte
					128	128 Kbyte
					256	64 Kbyte
128 Mb	16 bit	Micron MT46V8M16P-75E	1	128 Mbit	32	256 Kbyte
					64	128 Kbyte
					128	64 Kbyte
					256	32 Kbyte

Note: [1] These SDRAM vendor parts are targeted for use with the S132. Compatibility with these parts has not yet been fully verified.

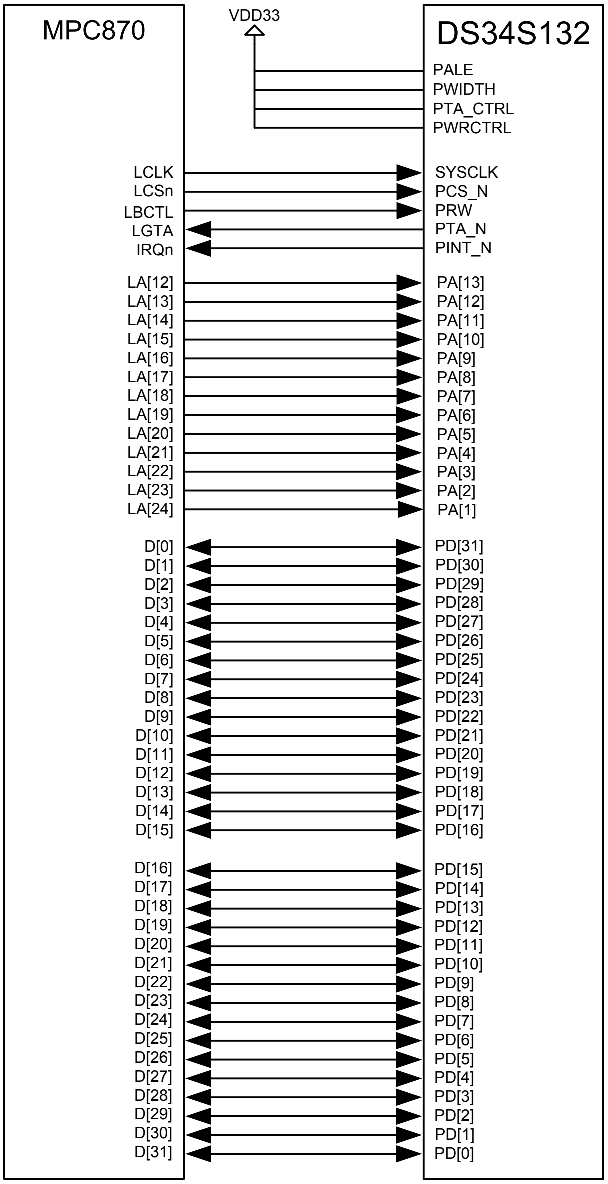
The external SDRAM is used by many of the S132 processes so the SDRAM interface should be configured before any of the TDM or Ethernet Ports are enabled. The SDRAM column width, memory size and control functions must be programmed (EMI.DCR2 and EMI.DCR3) to match to the DDR SDRAM that is used. EMI.DCR1.DIR should be used to reset the SDRAM after changing the EMI.DCR2 and EMI.DCR3 settings.

The SDRAM clock must be 125 MHz and can be derived from the ETHCLK or DDRCLK (G.GCR.ECDC).

9.6.4 CPU Electrical Interconnect

The CPU interface is used to program the S132, to transmit and receive CPU Packets (to/from the Ethernet Port) and to monitor the various status and interrupt signals from the S132. The CPU interface supports two processor interface types, one to work with processors like the Freescale MPC870 (depicted in Figure 9-29) and the other to work with processors like the Freescale MPC8313 (depicted in Figure 9-30 and Figure 9-31). The MPC8313 style processor supports a non-multiplexed and multiplexed mode, which determines whether the S132 PA[13:10] signals are connected to the processor address or data bus.

Figure 9-29. MPC870 32-bit Bus Interface



The MPC870 and MPC8313 are processor products of Freescale Semiconductor, Inc.

Figure 9-30. MPC8313, Non-multiplexed Bus Interface

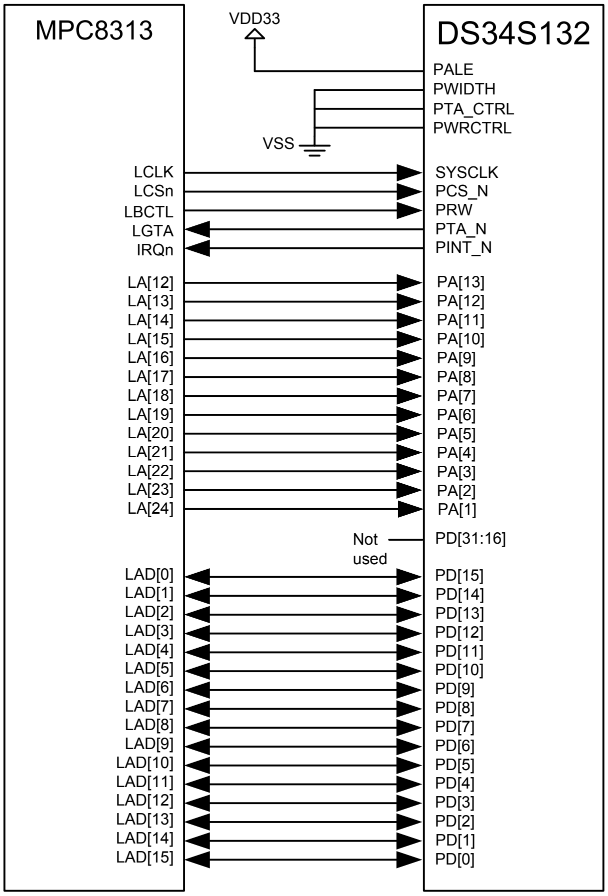
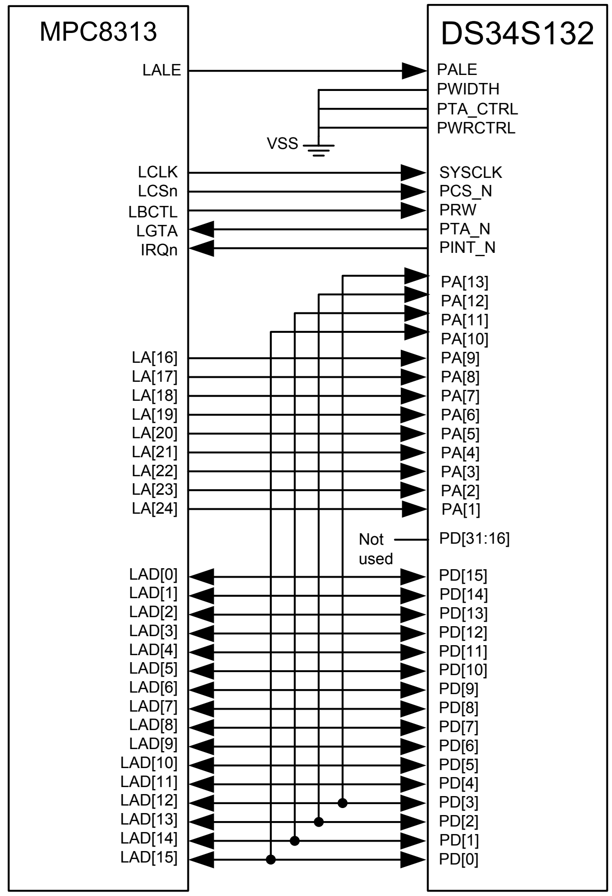


Figure 9-31. MPC8313, Multiplexed Bus Interface



9.6.5 Interrupt Hierarchy

The S132 includes a 3-level hierarchical interrupt system for interrupting the CPU. There are more than 700 conditions that can generate an interrupt on PINT_N. The 3-level hierarchy enables the CPU to discover any active interrupt condition with no more than 3 register reads.

The Level 3 Latched Status registers are the lowest level registers in the hierarchy and indicate when an interrupt condition has been detected. The latched bits insure that the CPU does not “miss” transient interrupt conditions. Real-time Status Register indications are also provided for some of the Level 3 Interrupt Conditions.

The Level 2 Status registers (G.GSR4, G.GSR5 and G.GSR6) are used to combine 640 latched active Level 3 interrupt conditions into Level 2 group status indications. The Level 3 registers that are combined are B.GxSRL, JB.GxSRL, G.PTSRL and G.PRSRL. Each Level 2 bit indicates if any of its “group member” (Level 3 Latched register) bits are enabled and are indicating an active interrupt event has been detected.

The Level 1 Interrupt register, G.GSR1, combines the remaining Level 3 Latched register indications with the Level 2 group status indications so that that the CPU can read one register (G.GSR1) to monitor all latched, active Level 3 interrupt conditions. These Level 1 and Level 2 register bits are real-time (non-latched) bits to indicate when any enabled Level 3 latched interrupt condition is active.

The Level 1 interrupt register, G.TPISRL, provides latched indications for each of its interrupt conditions. There are no Level 3 or Level 2 registers associated with these interrupt conditions.

One Interrupt Enable bit is provided for each of the latched interrupt register bits and for each of the Level 1, real-time G.GSR1 register indications so that any number or combination of the interrupt conditions can be disabled from generating an interrupt toward the CPU. When any latched register bit indicates that an active interrupt was detected (1), that latched bit is enabled, and its associated Level 1 register bit is enabled, the S132 will generate an active Interrupt signal (0) toward the CPU on PINT_N. The inactive state for PINT_N signal can be programmed to be high impedance or logic 1 using G.GCR1.IIM. Table 9-18 identifies the interrupt functions and how they relate to each other.

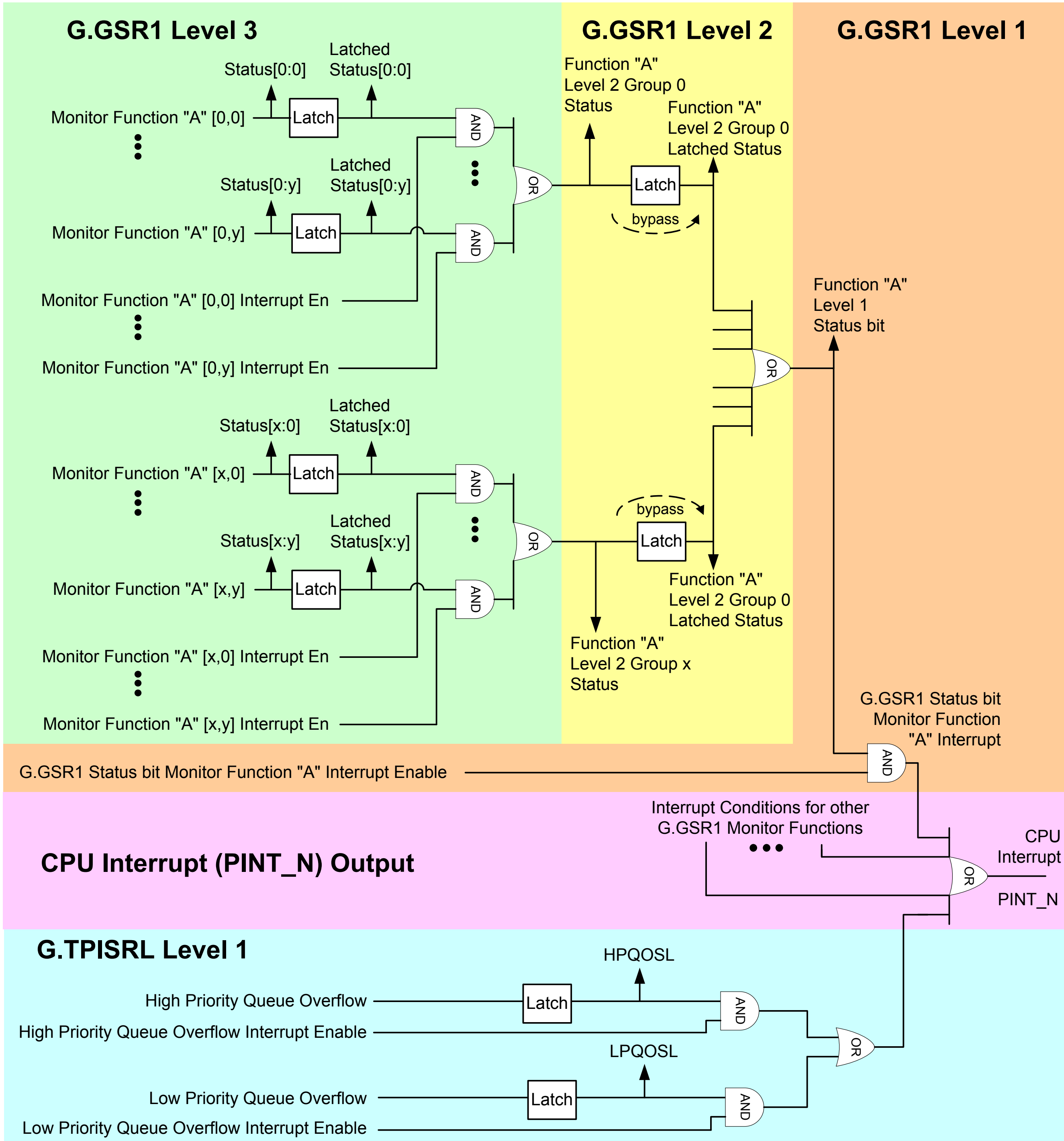
Table 9-18. Interrupt Hierarchy

Monitor Function	**Level 3 Interrupt Condition Registers**			**Level 2 Group Registers**		**Level 1 – Global Register bits**	
G.GSR1 Interrupts	**Status**	**Latched Status**	**Interrupt Enable**	**Status**	**Latched Status**	**G.GSR1 Status**	**G.GSRIE Enable**
Ethernet Port BERT	EB.BSR	EB.BSRL	EB.BSRIE	NA	NA	EBS	EBIE
TDM Port BERT	DB.BSR	DB.BSRL	DB.BSRIE	NA	NA	DBS	DBIE
TXP packet CAS	NA	G.GSR2	G.GSRIE2	NA	NA	PTCS	PTCIE
Xmt TDM Port CAS	NA	G.GSR3	G.GSRIE3	NA	NA	PRCS	PRCIE
Ethernet MAC	NA	M.IRQ_STATUS	M.IRQ_ENABLE M.IRQ_DISABLE	NA	NA	MIRS	MIRIE
Clock Recovery Engines	(note 1)	(note 1)	(note 1)	(note 1)	(note 1)	CRHS	CRHIE
Control Word	NA	B.GxSRL	B.GxSRIE	G.GSR5	NA	BS	BIE
Jitter Buffer Underrun	NA	JB.GxSRL	JB.GxSRIE	G.GSR6	NA	JBS	JBIE
Underrun/ Frame Align	NA	G.PTSRL	G.PTSRIE	NA	G.GSR4	PS	PIE
Overrun/ Frame Align	NA	G.PRSRL	G.PRSRIE				
Packet Classifier	NA	PC.SRL	PC.SRIE	NA	NA	PCS	PCIE
SDRAM Queue Error	NA	EMI.BMSRL	EMI.BMSRIE	NA	NA	EMIS	EMIIE
TXP CPU FIFO & Queue	EMA.WSR1	EMA.WSRL1	EMA.WSRIE1	NA	NA	EMAWS	EMAWIE
RXP CPU FIFO & Queue	EMA.RSR1	EMA.RSRL1	EMA.RSRIE1	NA	NA	EMARS	EMARIE
G.TIPSRL Interrupts	**Status**	**Latched Status**	**Interrupt Enable**	**Status**	**Latched Status**	**G.TPISRL**	**G.TPISRIE**
High Priority Overflow	NA	NA	NA	NA	NA	HPQOSL	HPQOSIE
Low Priority Overflow	NA	NA	NA	NA	NA	LPQOSL	LPQOSIE

Notes: [1] The Clock Recovery Engine interrupts are specified by the DSP firmware load (not included here).

Figure 9-32 depicts the interrupt hierarchy using an example "Monitor Function A" (e.g. Monitor Function "A" = "Rcv TDM Port CAS Change"). The "[x:y]" notation means "[Group:Member]". Some Monitor Functions have only one "group" so the Level 3 "OR" output would be connected directly to the Level 1 "AND" input (the Level 2 Status is "NA = Not Applicable" and there is no Level 2 OR gate). Some of the Status signals are latched (Latched Status) and others are not as indicated in Table 9-18. When a Status is provided, but without a "Latched Status" signal, the non-latched, Status bypasses the "latch" function in Figure 9-32. In this case the Status connects directly to the next logic element (OR gate or AND gate) in the interrupt hierarchy (e.g. the Level 2, G.GSR5 Status register bits are ORed together bypassing the latch function in the diagram). The G.TIPSRL interrupts are not driven by any lower level conditions. All G.TIPSRL conditions are Latched Status and connect directly to the Level 1 "OR".

Figure 9-32. Interrupt Hierarchy Diagram



10 DEVICE REGISTERS

10.1 Register Block Address Ranges

Table 10-1. Register Block Address Ranges

Registers	Address Range
Global Registers	
Global Configuration Registers (G.)	0000h – 002Fh
Global Status Registers (G.)	0030h – 005Fh
Global Status Register Interrupt Enables (G.)	0060h – 007Fh
Individual Bundle and Jitter Buffer Registers	
Global Bundle Reset Registers (B.)	0080h – 008Fh
Global Bundle Data Control Registers (B.)	0094h – 00A3h
Global Bundle Data Registers (B.)	00A4h – 00EFh
Bundle Group Status Latch Registers (B.)	0100h – 017Fh
Bundle Group Status Register Interrupt Enables (B.)	0180h – 01FFh
Jitter Buffer Status Registers (JB.)	0200h – 027Fh
Jitter Buffer Status Register Interrupt Enables (JB.)	0280h – 02FFh
Packet Classifier Registers	
Packet Classifier Configuration Registers (PC.)	0300h – 035Fh
Packet Classifier Status Register Latches (PC.)	0360h – 0367h
Packet Classifier Status Register Interrupt Enables (PC.)	0368h – 036Fh
Packet Classifier Counter Registers (PC.)	0370h – 037Fh
SDRAM Interface and Access Registers	
External Memory Interface Configuration Registers (EMI.)	0380h – 039Fh
External Memory Interface Status Registers (EMI.)	03A0h – 03AFh
External Memory Interface Status Register Interrupt Enables (EMI.)	03B0h – 03B7h
External Memory DLL/PLL Test Registers (EMI.)	03B8h – 03BFh
Write Registers (EMA.)	03C0h – 03DFh
Read Registers (EMA.)	03E0h – 03FFh
Test, Diagnostics and Clock Registers	
Encap BERT Registers (EB.)	0400h – 043Fh
Decap BERT Registers (DB.)	0440h – 047Fh
Miscellaneous Diagnostic Registers (MD.)	0480h – 04AFh
Test Registers (TST.)	0600h – 067Fh
Clock Recovery Registers (CR.)	0800h – 0BFFh
Ethernet MAC Registers	
MAC Registers (M.)	0C00h – 0DBFh
Per Port Registers	
Port n TXP SW CAS Registers (TXSCn.; n = 0 to 31)	1000h – 11FFh
Port n Xmt (RXP) SW CAS Registers (RXSCn. ; n = 0 to 31)	1200h – 13FFh
Port n Transmit Configuration Registers (Pn.; n = 0 to 31)	2000h – 2FFFh
Port n Transmit Status Registers (Pn.; n = 0 to 31)	
Port n Transmit Status Register Latches (Pn.; n = 0 to 31)	
Port n Transmit Status Register Interrupt Enables (Pn.; n = 0 to 31)	
Port n Receive Configuration Registers (Pn.; n = 0 to 31)	
Port n Receive Status Registers (Pn.; n = 0 to 31)	
Port n Receive Status Register Latches (Pn.; n = 0 to 31)	
Port n Receive Status Register Interrupt Enables (Pn.; n = 0 to 31)	
Time Slot Assignment Registers (TSAn.m.; n = 0 – 31; m = 0 – 31)	3000h – 4000h

10.2 Register Address Reference List

Table 10-2. Register Address Reference List

Register Name	Addr (hex)	Description
Global Configuration Registers (G.)		
IDR.	0000	ID Register
GCR	0004	Global Configuration Register
GRCR	0008	Global Reset Control Register
CCR	000C	CLAD Control Register
ECCR1	0010	Ethernet Conditioning Configuration Register 1
ECCR2	0014	Ethernet Conditioning Configuration Register 2
TCCR1	0018	TDM Conditioning Configuration Register 1
TCCR2	001C	TDM Conditioning Configuration Register 2
Global Status Registers (G.)		
GSR1	0030	Global Status Register 1
GSR2	0034	Global Status Register 2
GSR3	0038	Global Status Register 3
GSR4	003C	Global Status Register 4
GSR5	0040	Global Status Register 5
GSR6	0044	Global Status Register 6
TPISR1	0048	Transmit Packet Interface Status Register 1
TPISR2	004C	Transmit Packet Interface Status Register 2
TPISR3	0050	Transmit Packet Interface Status Register 3
TPISRL	0054	Transmit Packet Interface Status Register Latches
TPISRIE	0058	Transmit Packet Interface Status Register Interrupt Enable
Global Status Register Interrupt Enable Registers (G.)		
GSRIE1	0060	Global Status Register Interrupt Enable 1
GSRIE2	0064	Global Status Register Interrupt Enable 2
GSRIE3	0068	Global Status Register Interrupt Enable 3
Bundle Reset Registers (B.)		
BRCR1	0080	Bundle Reset Control Register 1.
BRCR2	0084	Bundle Reset Control Register 2.
BRSR	0088	Bundle Reset Status Register.
Bundle Data Control Registers (B.)		
BACR	0094	Bundle Activation Control Register
BCCR	0098	Bundle Configuration Control Register
BESCR	009C	Bundle Encap Status Control Register
BDSCR	00A0	Bundle Decap Status Control Register
Bundle Data Registers (B.)		
BADR1	00A4	Bundle Activation Data Register 1
BADR2	00A8	Bundle Activation Data Register 2
BCDR1	00AC	Bundle Configuration Data Register 1
BCDR2	00B0	Bundle Configuration Data Register 2
BCDR3	00B4	Bundle Configuration Data Register 3
BCDR4	00B8	Bundle Configuration Data Register 4
BCDR5	00BC	Bundle Configuration Data Register 5
BESR1	00C0	Bundle Encap Status Register 1
BESR2	00C4	Bundle Encap Status Register 2
BESR3	00C8	Bundle Encap Status Register 3
BDSR1	00D0	Bundle Decap Status Register 1
BDSR2	00D4	Bundle Decap Status Register 2
BDSR3	00D8	Bundle Decap Status Register 3
BDSR4	00DC	Bundle Decap Status Register 4
BDSR5	00E0	Bundle Decap Status Register 5
BDSR6	00E4	Bundle Decap Status Register 6

Register Name	Addr (hex)	Description
BDSR7	00E8	Bundle Decap Status Register 7
BDSR8	00EC	Bundle Decap Status Register 8
BDSR9	00F0	Bundle Decap Status Register 9
Bundle Group Status Latch Registers (B.)		
G0SRL	0100	Group 0 Status Register Latch
G1SRL	0104	Group 1 Status Register Latch
G2SRL	0108	Group 2 Status Register Latch
G3SRL	010C	Group 3 Status Register Latch
G4SRL	0110	Group 4 Status Register Latch
G5SRL	0114	Group 5 Status Register Latch
G6SRL	0118	Group 6 Status Register Latch
G7SRL	011C	Group 7 Status Register Latch
G8SRL	0120	Group 8 Status Register Latch
G9SRL	0124	Group 9 Status Register Latch
G10SRL	0128	Group 10 Status Register Latch
G11SRL	012C	Group 11 Status Register Latch
G12SRL	0130	Group 12 Status Register Latch
G13SRL	0134	Group 13 Status Register Latch
G14SRL	0138	Group 14 Status Register Latch
G15SRL	013C	Group 15 Status Register Latch
G16SRL	0140	Group 16 Status Register Latch
G17SRL	0144	Group 17 Status Register Latch
G18SRL	0148	Group 18 Status Register Latch
G19SRL	014C	Group 19 Status Register Latch
G20SRL	0150	Group 20 Status Register Latch
G21SRL	0154	Group 21 Status Register Latch
G22SRL	0158	Group 22 Status Register Latch
G23SRL	015C	Group 23 Status Register Latch
G24SRL	0160	Group 24 Status Register Latch
G25SRL	0164	Group 25 Status Register Latch
G26SRL	0168	Group 26 Status Register Latch
G27SRL	016C	Group 27 Status Register Latch
G28SRL	0170	Group 28 Status Register Latch
G29SRL	0174	Group 29 Status Register Latch
G30SRL	0178	Group 30 Status Register Latch
G31SRL	017C	Group 31 Status Register Latch
Bundle Group Status Register Interrupt Enable Registers (B.)		
G0SRIE	0180	Group 0 Status Register Interrupt Enable
G1SRIE	0184	Group 1 Status Register Interrupt Enable
G2SRIE	0188	Group 2 Status Register Interrupt Enable
G3SRIE	018C	Group 3 Status Register Interrupt Enable
G4SRIE	0190	Group 4 Status Register Interrupt Enable
G5SRIE	0194	Group 5 Status Register Interrupt Enable
G6SRIE	0198	Group 6 Status Register Interrupt Enable
G7SRIE	019C	Group 7 Status Register Interrupt Enable
G8SRIE	01A0	Group 8 Status Register Interrupt Enable
G9SRIE	01A4	Group 9 Status Register Interrupt Enable
G10SRIE	01A8	Group 10 Status Register Interrupt Enable
G11SRIE	01AC	Group 11 Status Register Interrupt Enable
G12SRIE	01B0	Group 12 Status Register Interrupt Enable
G13SRIE	01B4	Group 13 Status Register Interrupt Enable
G14SRIE	01B8	Group 14 Status Register Interrupt Enable
G15SRIE	01BC	Group 15 Status Register Interrupt Enable
G16SRIE	01C0	Group 16 Status Register Interrupt Enable
G17SRIE	01C4	Group 17 Status Register Interrupt Enable

Register Name	Addr (hex)	Description
G18SRIE	01C8	Group 18 Status Register Interrupt Enable
G19SRIE	01CC	Group 19 Status Register Interrupt Enable
G20SRIE	01D0	Group 20 Status Register Interrupt Enable
G21SRIE	01D4	Group 21 Status Register Interrupt Enable
G22SRIE	01D8	Group 22 Status Register Interrupt Enable
G23SRIE	01DC	Group 23 Status Register Interrupt Enable
G24SRIE	01E0	Group 24 Status Register Interrupt Enable
G25SRIE	01E4	Group 25 Status Register Interrupt Enable
G26SRIE	01E8	Group 26 Status Register Interrupt Enable
G27SRIE	01EC	Group 27 Status Register Interrupt Enable
G28SRIE	01F0	Group 28 Status Register Interrupt Enable
G29SRIE	01F4	Group 29 Status Register Interrupt Enable
G30SRIE	01F8	Group 30 Status Register Interrupt Enable
G31SRIE	01FC	Group 31 Status Register Interrupt Enable
Jitter Buffer Status Registers (JB.)		
G0SRL	0200	Group 0 Status Register Latch
G1SRL	0204	Group 1 Status Register Latch
G2SRL	0208	Group 2 Status Register Latch
G3SRL	020C	Group 3 Status Register Latch
G4SRL	0210	Group 4 Status Register Latch
G5SRL	0214	Group 5 Status Register Latch
G6SRL	0218	Group 6 Status Register Latch
G7SRL	021C	Group 7 Status Register Latch
G8SRL	0220	Group 8 Status Register Latch
G9SRL	0224	Group 9 Status Register Latch
G10SRL	0228	Group 10 Status Register Latch
G11SRL	022C	Group 11 Status Register Latch
G12SRL	0230	Group 12 Status Register Latch
G13SRL	0234	Group 13 Status Register Latch
G14SRL	0238	Group 14 Status Register Latch
G15SRL	023C	Group 15 Status Register Latch
G16SRL	0240	Group 16 Status Register Latch
G17SRL	0244	Group 17 Status Register Latch
G18SRL	0248	Group 18 Status Register Latch
G19SRL	024C	Group 19 Status Register Latch
G20SRL	0250	Group 20 Status Register Latch
G21SRL	0254	Group 21 Status Register Latch
G22SRL	0258	Group 22 Status Register Latch
G23SRL	025C	Group 23 Status Register Latch
G24SRL	0260	Group 24 Status Register Latch
G25SRL	0264	Group 25 Status Register Latch
G26SRL	0268	Group 26 Status Register Latch
G27SRL	026C	Group 27 Status Register Latch
G28SRL	0270	Group 28 Status Register Latch
G29SRL	0274	Group 29 Status Register Latch
G30SRL	0278	Group 30 Status Register Latch
G31SRL	027C	Group 31 Status Register Latch
Jitter Buffer Status Register Interrupt Enable Registers (JB.)		
G0SRIE	0280	Group 0 Status Register Interrupt Enable
G1SRIE	0284	Group 1 Status Register Interrupt Enable
G2SRIE	0288	Group 2 Status Register Interrupt Enable
G3SRIE	028C	Group 3 Status Register Interrupt Enable
G4SRIE	0290	Group 4 Status Register Interrupt Enable
G5SRIE	0294	Group 5 Status Register Interrupt Enable
G6SRIE	0298	Group 6 Status Register Interrupt Enable

Register Name	Addr (hex)	Description
G7SRIE	029C	Group 7 Status Register Interrupt Enable
G8SRIE	02A0	Group 8 Status Register Interrupt Enable
G9SRIE	02A4	Group 9 Status Register Interrupt Enable
G10SRIE	02A8	Group 10 Status Register Interrupt Enable
G11SRIE	02AC	Group 11 Status Register Interrupt Enable
G12SRIE	02B0	Group 12 Status Register Interrupt Enable
G13SRIE	02B4	Group 13 Status Register Interrupt Enable
G14SRIE	02B8	Group 14 Status Register Interrupt Enable
G15SRIE	02BC	Group 15 Status Register Interrupt Enable
G16SRIE	02C0	Group 16 Status Register Interrupt Enable
G17SRIE	02C4	Group 17 Status Register Interrupt Enable
G18SRIE	02C8	Group 18 Status Register Interrupt Enable
G19SRIE	02CC	Group 19 Status Register Interrupt Enable
G20SRIE	02D0	Group 20 Status Register Interrupt Enable
G21SRIE	02D4	Group 21 Status Register Interrupt Enable
G22SRIE	02D8	Group 22 Status Register Interrupt Enable
G23SRIE	02DC	Group 23 Status Register Interrupt Enable
G24SRIE	02E0	Group 24 Status Register Interrupt Enable
G25SRIE	02E4	Group 25 Status Register Interrupt Enable
G26SRIE	02E8	Group 26 Status Register Interrupt Enable
G27SRIE	02EC	Group 27 Status Register Interrupt Enable
G28SRIE	02F0	Group 28 Status Register Interrupt Enable
G29SRIE	02F4	Group 29 Status Register Interrupt Enable
G30SRIE	02F8	Group 30 Status Register Interrupt Enable
G31SRIE	02FC	Group 31 Status Register Interrupt Enable
Packet Classifier Configuration Registers (PC.)		
CR1	0300	Configuration Register 1
CR2	0304	Configuration Register 2
CR3	0308	Configuration Register 3
CR4	030C	Configuration Register 4
CR5	0310	Configuration Register 5
CR6	0314	Configuration Register 6
CR7	0318	Configuration Register 7
CR8	031C	Configuration Register 8
CR9	0320	Configuration Register 9
CR10	0324	Configuration Register 10
CR11	0328	Configuration Register 11
CR12	032C	Configuration Register 12
CR13	0330	Configuration Register 13
CR14	0334	Configuration Register 14
CR15	0338	Configuration Register 15
CR16	033C	Configuration Register 16
CR17	0340	Configuration Register 17
CR18	0344	Configuration Register 18
CR19	0348	Configuration Register 19
CR20	034C	Configuration Register 20
CR21	0350	Configuration Register 21
Packet Classifier Status Latch Registers (PC.)		
SRL	0360	Status Register Latch
Packet Classifier Status Interrupt Enable Registers (PC.)		
SRIE	0368	Status Register Interrupt Enable
Packet Classifier Counter Registers (PC.)		
CPCR	0370	Classified Packet Counter Register
PCECR	0374	IP/UDP Packet Checksum Error Counter Register
SPCR	0378	Stray Counter Register

Register Name	Addr (hex)	Description
FOCR	037C	FIFO Overflow Counter Register
External Memory Interface Configuration Registers (EMI.)		
BMCR1	0380	Buffer Manager Configuration Register 1
BMCR2	0384	Buffer Manager Configuration Register 2
BMCR3	0388	Buffer Manager Configuration Register 3
DCR1	0390	DDR SDRAM Configuration Register 1
DCR2	0394	DDR SDRAM Configuration Register 2
DCR3	0398	DDR SDRAM Configuration Register 3
External Memory Interface Status Registers (EMI.)		
BMSRL	03A0	Buffer Manager Status Register Latch
External Memory Interface Status Interrupt Enable Registers (EMI.)		
BMSRIE	03B0	Buffer Manager Status Register Interrupt Enable
External Memory Interface Test Status Registers (EMI.)		
TSRL	03B4	Test Status Register Latched
External Memory DLL/PLL Test Registers (EMI.)		
TCR1	03B8	Test Configuration Register 1
TCR2	03BC	Test Configuration Register 2
Write Registers (EMA.)		
WCR	03C0	Write Control Register
WAR	03C4	Write Address Register
WDR	03C8	Write Data Register
WSR1	03CC	Write Status Register 1
WSR2	03D0	Write Status Register 2
WSRL1	03D4	Write Status Register Latch 1
WSRIE1	03D8	Write Status Register Interrupt Enable 1
Read Registers (EMA.)		
RCR	03E0	Read Control Register
RAR	03E4	Read Address Register
RDR	03E8	Read Data Register
RSR1	03EC	Read Status Register 1
RSR2	03F0	Read Status Register 2
RSRL1	03F4	Read Status Register Latch 1
RSRIE1	03F8	Read Status Register Interrupt Enable 1
Encap BERT Registers (EB.)		
BCR	0400	BERT Control Register
BPCR	0404	BERT Pattern Configuration Register
BSPR	0408	BERT Seed / Pattern Register
TEICR	0410	Transmit Error Insertion Control Register
BSR	0414	BERT Status Register
BSRL	0418	BERT Status Register Latch
BSRIE	041C	BERT Status Register Interrupt Enable
RBECR	0420	Receive Bit Error Count Register
RBCR	0424	Receive Bit Count Register
TSTCR	0430	Test Control Register
Decap BERT Registers (DB.)		
BCR	0440	BERT Control Register
BPCR	0444	BERT Pattern Configuration Register
BSPR1	0448	BERT Seed / Pattern Register
TEICR	0450	Transmit Error Insertion Control Register
BSR	0454	BERT Status Register
BSRL	0458	BERT Status Register Latch
BSRIE	045C	BERT Status Register Interrupt Enable
RBECR	0460	Receive Bit Error Count Register
RBCR	0464	Receive Bit Count Register

Register Name	Addr (hex)	Description
TSTCR	0470	Test Control Register
Miscellaneous Diagnostic Registers (MD.)		
DCR	0480	Diagnostic Control Register
EBCR	0484	Encap BERT Control Register
DBCR	0488	Decap BERT Control Register
MBSR1	04A0	Memory BIST Status Register 1
MBSR2	04A4	Memory BIST Status Register 2
MBSR3	04A8	Memory BIST Status Register 3
MBSR4	04AC	Memory BIST Status Register 4
MBSR5	04B0	Memory BIST Status Register 5
Test Registers (TST.)		
GTR1	0600	Global Test Control Register 1
BTCR1	0604	Block Test Control Register 1
BTCR2	0608	Block Test Control Register 2
BTCR3	060C	Block Test Control Register 3
BTCR4	0610	Block Test Control Register 4
BTCR5	0614	Block Test Control Register 5
BTCR6	0618	Block Test Control Register 6
CRJBT	061C	Clock Recovery Jitter Buffer Test
BTSR1	0624	Block Test Status Register 1
BTSR2	0628	Block Test Status Register 2
BTSR3	062C	Block Test Status Register 3
BTSR4	0630	Block Test Status Register 4
BTSR5	0634	Block Test Status Register 5
BTSR6	0638	Block Test Status Register 6
CTCR1	0640	CLAD Test Control Register 1
CTCR2	0644	CLAD Test Control Register 2
CTCR3	0648	CLAD Test Control Register 3
CTCR4	064C	CLAD Test Control Register 4
EDTCR	0660	Encap/Decap Test Control Register
EDTSR1	0664	Encap/Decap Test Status Register 1
EDTSR2	0668	Encap/Decap Test Status Register 2
EDTSR3	066C	Encap/Decap Test Status Register 3
EDTSR4	0670	Encap/Decap Test Status Register 4
EDTSR5	0674	Encap/Decap Test Status Register 5
FID	06FC	Block Test Control Register 6
Clock Recovery Registers (CR.)		
CRCR	0800	Clock Recovery Control Register
MAC Registers (M.)		
NET_CONTROL.	0C00	Network Control Register
NET_CONFIG	0C04	Network Configuration Register
NET_STATUS	0C08	Network Status Register
RSVD	0C0C	Reserved
USER_IO	0C10	User Input/Output Register
TX_STATUS	0C14	Transmit Status Register
RX_QPTR	0C18	Receive Buffer Queue Base Address
TX_QPTR	0C1C	Transmit Queue Base Address
RX_STATUS	0C20	Receive Status Register
IRQ_STATUS	0C24	Interrupt Status Register
IRQ_ENABLE	0C28	Interrupt Enable Register
IRQ_DISABLE	0C2C	Interrupt Disable Register
IRQ_MASK	0C30	Interrupt Mask Register
PHY_MAN	0C34	Phy Maintenance Register
RX_PAUSE_TIME	0C38	Received Pause Quantum Register
TX_PAUSE_QUANT	0C3C	Transmit Pause Quantum Register

Register Name	Addr (hex)	Description
HASH_BOT	0C80	Hash Register Bottom
HASH_TOP	0C84	Hash Register Top
LADDR1_BOT	0C88	Specific Address 1 Bottom
LADDR1_TOP	0C8C	Specific Address 1 Top
LADDR2_BOT	0C90	Specific Address 2 Bottom
LADDR2_TOP	0C94	Specific Address 2 Top
LADDR3_BOT	0C98	Specific Address 3 Bottom
LADDR3_TOP	0C9C	Specific Address 3 Top
LADDR4_BOT	0CA0	Specific Address 4 Bottom
LADDR4_TOP	0CA4	Specific Address 4 Top
ID_CHECK1	0CA8	Type ID Match 1
ID_CHECK2	0CAC	Type ID Match 2
ID_CHECK3	0CB0	Type ID Match 3
ID_CHECK4	0CB4	Type ID Match 4
RSVD	0CB8	Reserved
IPG_STRETCH	0CBC	IPG Stretch Register
MOD_ID	0CFC	Module Revision ID Register
OCT_TX_BOT	0D00	Octet Transmitted Bottom
OCT_TX_TOP	0D04	Octet Transmitted Top
STATS_FRAMES_TX	0D08	Frames Transmitted Top
BROADCAST_TX	0D0C	Broadcast Frames Transmitted
MULTICAST_TX	0D10	Multicast Frames Transmitted
STATS_PAUSE_TX	0D14	Pause Frames Transmitted
FRAME64_TX	0D18	64 Byte Frames Transmitted
FRAME65_TX	0D1C	65 to 127 Byte Frames Transmitted
FRAME128_TX	0D20	128 to 255 Byte Frames Transmitted
FRAME256_TX	0D24	256 to 511 Byte Frames Transmitted
FRAME512_TX	0D28	512 to 1023 Byte Frames Transmitted
FRAME1024_TX	0D2C	1024 to 1518 Byte Frames Transmitted
FRAME1519_TX	0D30	Greater Than 1518 Byte Frames Transmitted
STATS_TX_URUN	0D34	Transmit Under Runs
STATS_SINGLE_COL	0D38	Single Collision Frames
STATS_MULTI_COL	0D3C	Multiple Collision Frames
STATS_LATE_COL	0D44	Late Collisions
STATS_DEF_TX	0D48	Deferred Transmission Frames
STATS_CRS_ERRORS	0D4C	Carrier Sense Errors
OCT_RX_BOT	0D50	Octets Received Bottom
OCT_RX_TOP	0D54	Octets Received Top
STATS_FRAMES_RX	0D58	Frames Received
BROADCAST_RX	0D5C	Broadcast Frames Received
MULTICAST_RX	0D60	Multicast Frames Received
STATS_PAUSE_RX	0D64	Pause Frames Received
FRAME64_RX	0D68	64 Byte Frames Received
FRAME65_RX	0D6C	65 to 127 Byte Frames Received
FRAME128_RX	0D70	128 to 255 Byte Frames Received
FRAME256_RX	0D74	256 to 511 Byte Frames Received
FRAME512_RX	0D78	512 to 1023 Byte Frames Received
FRAME1024_RX	0D7C	1024 to 1518 Byte Frames Received
FRAME1519_RX	0D80	1519 to Maximum Byte Frames Received
STATS_USIZE_FRAMES	0D84	Undersized Frames Received
STATS_EXCESS_LEN	0D88	Oversized Frames Received
STATS_JABBERS	0D8C	Jabbers Received
STATS_FCS_ERRORS	0D90	Frame Check Sequence Errors
STATS_LENGTH_ERRORS	0D94	Length Field Frame Errors
STATS_RX_SYM_ERR	0D98	Receive Symbol Errors

Register Name	Addr (hex)	Description
STATS_ALIGN_ERRORS	0D9C	Alignment Errors
STATS_RX_RES_ERR	0DA0	Receive Resource Errors
STATS_RX_ORUN	0DA4	Receive Overruns
IP_HDR_CHK	0DA8	IP Header Checksum Errors
TCP_CHK	0DAC	TCP Checksum Errors
UDP_CHK	0DB0	UDP Checksum Errors
RSVD	0E00	Reserved
REG_TOP	0E3C	Reserved
Port n TXP SW CAS Registers (TXSCn.; n = 0 to 31)		
CR1	1000 + n*0010	Port n Configuration Register 1
CR2	1004 + n*0010	Port n Configuration Register 2
CR3	1008 + n*0010	Port n Configuration Register 3
CR4	100C + n*0010	Port n Configuration Register 4
Port n Xmt (RXP) SW CAS Registers (RXSCn. ; n = 0 to 31)		
CR1	1200 + n*0010	Port n Configuration Register 1
CR2	1204 + n*0010	Port n Configuration Register 2
CR3	1208 + n*0010	Port n Configuration Register 3
CR4	120C + n*0010	Port n Configuration Register 4
Port n Transmit Configuration Registers (Pn.; n = 0 to 31)		
PTCR1	2000 + n*0080	Port n Transmit Configuration Register 1
PTCR2	2004 + n*0080	Port n Transmit Configuration Register 2
PTCR3	2008 + n*0080	Port n Transmit Configuration Register 3
Port n Transmit Status Registers (Pn.; n = 0 to 31)		
PTSR1	2020 + n*0080	Port n Transmit Status Register 1
PTSR2	2024 + n*0080	Port n Transmit Status Register 2
PTSR3	2028 + n*0080	Port n Transmit Status Register 3
PTSR4	202C + n*0080	Port n Transmit Status Register 4
Port n Transmit Status Latch Registers (Pn.; n = 0 to 31)		
PTSRL	2030 + n*0080	Port n Transmit Status Register Latch
Port n Transmit Status Interrupt Enable Registers (Pn.; n = 0 to 31)		
PTSRIE	2038 + n*0080	Port n Transmit Status Register Interrupt Enable
Port n Receive Configuration Registers (Pn.; n = 0 to 31)		
PRCR1	2040 + n*0080	Port n Receive Configuration Register 1
PRCR2	2044 + n*0080	Port n Receive Configuration Register 2
PRCR3	2048 + n*0080	Port n Receive Configuration Register 3
PRCR4	204C + n*0080	Port n Receive Configuration Register 4
PRCR5	2050 + n*0080	Port n Receive Configuration Register 5
Port n Receive Status Registers (Pn.; n = 0 to 31)		
PRSR1	2060 + n*0080	Port n Receive Status Register 1
PRSR2	2064 + n*0080	Port n Receive Status Register 2
PRSR3	2068 + n*0080	Port n Receive Status Register 3
PRSR4	206C + n*0080	Port n Receive Status Register 4
Port n Receive Status Latch Registers (Pn.; n = 0 to 31)		
PRSRL	2070 + n*0080	Port n Receive Status Register Latch
Port n Receive Status Interrupt Enable Registers (Pn.; n = 0 to 31)		
PRSRIE	2078 + n*0080	Port n Receive Status Register Interrupt Enable
Time Slot Assignment Registers (TSAn.m.; port "n" = 0 to 31 and Timeslot "m" = 0 to 31)		
CR	3000 + m*0004	Configuration Register

10.3 Register Definitions

In the sub-sections that follow each register definition includes a Register Type definition with 3 Type Categories: Signal Type, Clear Type and Misc Type. The Type definition uses the form "a-b-c" where a = Signal Type, b = Clear Type and c = Misc Type. If one of these categories is not applicable to a register bit, then an underscore, "_", is used (e.g. ros-cor-_).

Signal Type
- ros: Read Only Status
- rls: Read Latched Status
- rcs: Read Count Status
- woc: Write Only Control
- rwc: Read/Write Control
- rod: "ros" Delayed
- rld: "rls" Delayed
- rcd: "rcs" Delayed
- rwd: "rwc" Delayed

Clear Type
- cor: Clear On Read
- cow: Clear On Write
- crw: Clear on Read or Write (G.GCR.LSBCRE selected)
- cnr: Clear on None or Read (G.GCR.CCOR selected)

Misc Type
- ix: Interrupt level "x" (x = 1, 2 or 3)
- sc: Saturating Counter
- nc: Non-saturating Counter

The term "Delayed" means that the Read or Write operation does not complete within one clock cycle and the external CPU must provide sufficient time for the operation to complete. These are RAM-based registers that do not support immediate read/write operations. The data in this type of register is not valid until after the first Write to the register (the data is invalid/unknown after a reset).

The term "Clear" indicates how a latch or counter is returned to its reset state. "Clear on Read" means the signal is reset by a Read operation. For "Clear on Write", a Write with any register value resets the register. "Clear On None" is used by some counters to mean that the count is not reset by any action. For registers with the clear option "crw", the global G.GCR.LSBCRE bit selects between "Clear On Read" and "Clear On Write". For registers with the clear option "cnr", the global G.GCR.CCOR bit selects between "Clear On Read" and "Clear On None".

Saturating Counters stop incrementing at their maximum count. Non-saturating counters roll-over back to "zero" after they reach their maximum count.

The "x" that is used in the "ix" Type means that the interrupt level may be any of x = 1 to 3, where 1 is lowest level interrupt in the S132 interrupt hierarchy (e.g. roi1). All interrupt generating registers have an associated register that is used to enable or disable (mask) the interrupt.

The "Description" term "Reserved" means that this bit has only one valid setting. The bit name in the far left column may be "RSVD" or some other name (e.g. "CCRSTDP"). In most cases, the only valid setting is the default value. In a few cases (as noted) they use a non-default value that is indicated in the Description column (e.g. Reserved**.** This must be programmed to "1".)

Numbers are written in decimal notation unless a "b" suffix is used for binary (e.g. 010b) or a "0x" prefix or "h" suffix is used for hexadecimal (e.g. "0x4F" or "0800h"; the "0x" and 'h' notation have the same meaning).

Yellow shading is used to identify the 32-bit register name and characteristics. White (non-shaded) rows are used to define the bit field s within each 32-bit register.

10.3.1 Global Registers (G.)

10.3.1.1 Global Configuration Registers (G.)

Table 10-3. Global Configuration Registers

G. Field Name	Addr (A:) Bit [x:y]	Type	Description
IDR.	**A:0000h**		**ID Register. Default: 00.0J.JJh where J = JTAG ID**
ID	[31:20]	ros-_-_	**ID.** Reserved
ID	[19:4]	ros-_-_	**ID.** Same information as the lower 16 bits of JTAG CODE ID portion of the JTAG ID register. JTAG ID[27:12].
ID	[3:0]	ros-_-_	**Originial Rev ID.** Was not modified to reflect Rev A2 ID. Still reads 4'b0000.
GCR.	**A:0004h**		**Global Configuration Register. Default: 0x00.00.08.00**
RSVD	[31:28]		**Reserved.**

G. Field Name	Addr (A:) Bit [x:y]	Type	Description
EC25	[27]	rwc-_-_	**Ethernet Clock 25 MHz** selects the rate of the ETHCLK input. 0 = ETHCLK is being driven by 125MHz source 1 = ETHCLK is being driven by 25MHz source
ECDC	[26]	rwc-_-_	**Ethernet Clock DDR SDRAM Clock** selects the SDCLK source. 0 = SDCLK is sourced from ETHCLK (125 MHz only; tie DDRCLK low) 1 = SDCLK is sourced from DDRCLK (independent of ETHCLK)
IIM	[25]	rwc-_-_	**Interrupt Inactive Mode** determines the inactive mode of the INT_N pin. The INT_N pin always drives low when an enabled interrupt source is active. 0 = Pin is high impedance when all enabled interrupts are inactive 1 = Pin drives high when all enabled interrupts are inactive
RDPC	[24]	rwc-_-_	**Reorder or Duplicate Packet Counters** selects which condition increments the Reorder Counters (see B.BDSR6.SCRPC). 0 = Count the number of reordered good packets 1 = Count the number of duplicate packets
JLPC	[23]	rwc-_-_	**Jump or Lost Packet Counters** selects which condition increments the Jumped Packet Counters (see B.BDSR5.SCJPC) 0 = Count the jump size for good packets 1 = Count the number of lost packets not received before playout.
IPSE	[22]	rwc-_-_	**Indicate Playout Start Enable** selects which conditions are indicated by the Jitter Buffer Underrun Status bits (see GxSRL.JBU). 0 = Detect Jitter Buffer Underrun only 1 = Detect Jitter Buffer Underrun and “Start of Playout” changes (monitor B.BDSR3.JBLL to determine if change is Underrun or Playout)
JBMD	[21:20]	rwc-_-_	**Jitter Buffer Max Depth** = the byte depth for all Jitter Buffers. 0 = 256KB per Jitter Buffer 1 = 128KB per Jitter Buffer 2 = 64KB per Jitter Buffer 3 = 32KB per Jitter Buffer
GRCSS	[19:15]	rwc-_-_	**Global Recovered Clock Source Select** selects which Clock Recovery Engine (0 – 31) generates the Global Recovered Clock. 0x00 = Clock Recovery Engine 0.
GMMS	[14:12]	rwc-_-_	**GMII - MII Mode Select** selects the Ethernet port mode and interface type. 0 = Ethernet port disabled 2 = Ethernet port enabled using MII interface 3 = Ethernet port enabled using GMII interface
CCOR	[11]	rwc-_-_	**Clear Counter On Read** selects the clear function for the RX Bundle, TX Bundle and Packet Classifier counters (affects registers with “-cnr-” in the “Type” column). The counters will roll over after the maximum value. 0 = Counters do not clear 1 = Each Read operation clears the counter
RXHMFIS	[10:8]	rwc-_-_	**RXP HDLC Minimum Flag Insertion** selects minimum number of HDLC flags that are inserted between HDLC frames at the Transmit TDM Ports. The number of inserted flags is 1 more than this programmed value (i.e. 0 setting = 1 flag).
OTRS	[7]	rwc-_-_	**OAM Timestamp Resolution Select** selects OAM Timestamps resolution. 0 = 1us OAM Timestamp resolution 1 = 100us OAM Timestamp resolution
LSBCRE	[6]	rwc-_-_	**Latch Status Bit Clear on Read Enable** selects when the latched status register bits are cleared, but does not apply to the Clock Recovery Status Registers or the Ethernet MAC Status Registers. 0 = Latched status register bits are cleared when the CPU writes to the register 1 = Latched status register bits are cleared when the CPU reads the register
LBCDE	[5]	rwc-_-_	**L Bit Change Detect Enable** = “1” enables L-bit change detection for all Bundles.
RBCDE	[4]	rwc-_-_	**R Bit Change Detect Enable** = “1” enables R-bit change detection for all Bundles.

G. Field Name	Addr (A:) Bit [x:y]	Type	Description
MBCDE	[3:2]	rwc-_-_	**M Bit Change Detect Enable** = "1" enables M-bit change detection for all Bundles. Each M-bit can be enabled individually.
FBCDE	[1:0]	rwc-_-_	**Fragmentation Bit Change Detect Enable** = "1" enables F-bit change detection for all Bundles. Each F-bit can be enabled individually.
GRCR.	**A:0008h**		**Global Reset Control Register. Default: 00.00.07.FEh**
RSVD	[31:11]		**Reserved.**
CCRSTDP	[10]		**Reserved.**
DRSTDP	[9]		**Reserved.**
SCRSTDP	[8]		**Reserved.**
MRSTDP	[7]		**Reserved.**
EMARSTDP	[6]		**Reserved.**
TERSTDP	[5]		**Reserved.**
TDRSTDP	[4]		**Reserved.**
TPIRSTDP	[3]		**Reserved.**
RPIRSTDP	[2]		**Reserved.**
RSTDP	[1]	rwc-_-_	**Global Datapath Reset** selects the internal global datapath reset state (CPU programmed control registers are not reset by this function, but should be re-programmed to insure S132 functions properly after performing this reset). 0 = Normal operation 1 = Force Data Path to default state (must be "1" > 100ns; similar to RST_N = 0)
RST	[0]	rwc-_-_	**Global Reset** selects the reset state for the internal global datapath, status and control registers. The Bundle (B.), Timeslot Assignment (TSAn.m.), TXP SW CAS (TXSCn.) and Xmt SW CAS (RXSCn.) registers are not reset by this. However, these registers should be considered to be reset and reloaded after de-assertion of this reset. This reset function is similar to the reset for the RST_N pin. 0 = Normal operation 1 = Force internal registers to their default values (must be high > 100ns)
CCR.	**A:000Ch**		**CLAD Control Register. Default: 0x00.00.00.78**
RSVD	[31:7]		**Reserved**.
FS	[6:3]	rwc-_-_	**Frequency Select** selects the CLAD input clock rate. The CLAD input clock can be sourced from the REFCLK or CMNCLK pins (selected using G.CCR.SCS). 0000b = 5 MHz 0101b = 13 MHz 1001b = 25 MHz 0001b = 5.12 MHz 0110b = 19.44 MHz 1010b = 38.88 MHz 0010b = 10 MHz 0111b = 20 MHz 1011b = 77.76 MHz 0011b = 10.24 MHz 1000b = 20.48 MHz 11xxb = 155.52 MHz 0100b = 12.8 MHz
LCE	[2]	rwc-_-_	**LIU Clock Enable** = "1" enables LIUCLK. "0" disables LIUCLK PLL and output.
LCS	[1]	rwc-_-_	**LIU Clock Select** selects the LIUCLK output rate. 0 = 1.544 MHz output clock 1 = 2.048 MHz output clock
SCS	[0]	rwc-_-_	**Synthesis Clock Select** selects the CLAD input clock source. 0 = REFCLK input 1 = CMNCLK input
ECCR1.	**A:0010h**		**Ethernet Conditioning Configuration Register 1. Default: 0x00.00.00.00**
ECOA	[31:24]	rwc-_-_	**Ethernet Conditioning Octet A.** TXP Ethernet Conditioning Octet A
ECOB	[23:16]	rwc-_-_	**Ethernet Conditioning Octet B.** TXP Ethernet Conditioning Octet B
ECOC	[15:8]	rwc-_-_	**Ethernet Conditioning Octet C.** TXP Ethernet Conditioning Octet C
ECOD	[7:0]	rwc-_-_	**Ethernet Conditioning Octet D.** TXP Ethernet Conditioning Octet D

G. Field Name	Addr (A:) Bit [x:y]	Type	Description
ECCR2.	**A:0014h**		**Ethernet Conditioning Configuration Register 2. Default: 0x00.00.00.00**
ECOE	[31:24]	rwc-_-_	**Ethernet Conditioning Octet E.** TXP Ethernet Conditioning Octet E
ECOF	[23:16]	rwc-_-_	**Ethernet Conditioning Octet F.** TXP Ethernet Conditioning Octet F
ECOG	[15:8]	rwc-_-_	**Ethernet Conditioning Octet G.** TXP Ethernet Conditioning Octet G
ECOH	[7:0]	rwc-_-_	**Ethernet Conditioning Octet H.** TXP Ethernet Conditioning Octet H
TCCR1.	**A:0018h**		**TDM Conditioning Configuration Register 1. Default: 0x00.00.00.00**
TCOA	[31:24]	rwc-_-_	**TDM Conditioning Octet A.** RXP TDM Conditioning Octet A.
TCOB	[23:16]	rwc-_-_	**TDM Conditioning Octet B.** RXP TDM Conditioning Octet B
TCOC	[15:8]	rwc-_-_	**TDM Conditioning Octet C.** RXP TDM Conditioning Octet C
TCOD	[7:0]	rwc-_-_	**TDM Conditioning Octet D.** RXP TDM Conditioning Octet D
TCCR2.	**A:001Ch**		**TDM Conditioning Configuration Register 2. Default: 0x00.00.00.00**
ETCOE	[31:24]	rwc-_-_	**TDM Conditioning Octet E.** RXP TDM Conditioning Octet E
TCOF	[23:16]	rwc-_-_	**TDM Conditioning Octet F.** RXP TDM Conditioning Octet F
TCOG	[15:8]	rwc-_-_	**TDM Conditioning Octet G.** RXP TDM Conditioning Octet G
TCOH	[7:0]	rwc-_-_	**TDM Conditioning Octet H.** RXP TDM Conditioning Octet H

10.3.1.2 Global Status Registers (G.)

Table 10-4. Global Status Registers (G.)

G. Field Name	Addr (A:) Bit [x:y]	Type	Description
GSR1.	**A:0030h**		**Global Status Register 1. Default: 0x00.00.00.00**
RSVD	[31:18]		**Reserved.**
EBS	[17]	ros-_-i1	**Encap (Ethernet) BERT Status** = "1" indicates one or more Packet BERT Status Latch bits = "1" (EB.BSRL) and are enabled (EB.BSIE). The combination of EBS = 1 and G.GSRIE1.EBIE = 1 forces an interrupt on INT_N.
DBS	[16]	ros-_-i1	**Decap (TDM Port) BERT Status** = "1" indicates one or more TDM BERT Status Latch bits = "1" (DB.BSRL) and are enabled (DB.BSIE). The combination of DBS = 1 and G.GSRIE1.DBIE = 1 forces an interrupt on INT_N.
PTCS	[15]	ros-_-i1	**Port Transmit CAS Status** = "1" indicates one or more Transmit (RXP) CAS Status Latch bits = "1" (G.GSR2) and are enabled (G.GSRIE2). The combination of PTCS = 1 and G.GSRIE1.PTCIE = 1 forces an interrupt on INT_N.
PRCS	[14]	ros-_-i1	**Port Receive CAS Status** = "1" indicates one or more Receive (TXP) CAS Status Latch bits = "1" (G.GSR3) and are enabled (G.GSRIE3). The combination of PRCS = 1 and G.GSRIE1.PRCIE = 1 forces an interrupt on INT_N.
MIRS	[13]	ros-_-i1	**MAC Interrupt Register Status** = "1" indicates one or more M.IRQ_STATUS Status Latch bits = "1" and are enabled (M.IRQ_ENABLE and M.IRQ_DISABLE). The combination of MIRS = 1 and G.GSRIE1.MIRIE = 1 forces an interrupt on INT_N.
CRHS	[12:8]	ros-_-i1	**Clock Recovery Hardware Status** = "1" indicates one or more Clock Recovery Engine Status Latch bits = "1" (the Clock Recovery Status is defined by the DSP Firmware load). The combination of any CRHS[x] = 1 (x = 8 to 12) and G.GSRIE1.CRHIE[x] = 1 forces an interrupt on INT_N.
BS	[7]	ros-_-i1	**Bundle Status** = "1" indicates one or more Group Bundle Status bits are "1" (G.GSR5). The combination of BS = 1 and G.GSRIE1.BIE = 1 forces an interrupt on INT_N.

G. Field Name	Addr (A:) Bit [x:y]	Type	Description
JBS	[6]	ros-_-i1	**Jitter Buffer Status** = "1" indicates one or more Jitter Buffer Status bits are "1" (G.GSR6). The combination of JBS = 1 and G.GSRIE1.JBIE = 1 forces an interrupt on INT_N.
PS	[5]	ros-_-i1	**Port Status** = "1" indicates one or more TDM Per Port Status bits are "1" (G.GSR4). The combination of PS = 1 and G.GSRIE1.PIE = 1 forces an interrupt on INT_N.
PCS	[4]	ros-_-i1	**Packet Classifier Status** = "1" indicates one or more Packet Classifier special event/errors have been detected (PC.SRL) and enabled (PC. SRIE). An interrupt is generated on INT_N when PCS = 1 and G.GSRIE1.PCIE = 1.
EMIS	[3]	ros-_-i1	**External Memory Interface Status** = "1" indicates one or more SDRAM Queue Errors have been detected (EMI.BMSRL) and enabled (EMI.BMSRIE). An interrupt is generated on INT_N when EMIS = 1 and G.GSRIE1.EMIIE = 1.
RSVD	[2]		**Reserved.**
EMAWS	[1]	ros-_-i1	**External Memory Access Write Status** = "1" indicates one or more TXP CPU Packet Write Status Latch bits = "1" (EMA.WSRL1) and enabled (EMA.WSRIE1). The combination of EMAWS = 1 and G.GSRIE1.EMAWIE = 1 forces an interrupt on INT_N.
EMARS	[0]	ros-_-i1	**External Memory Access Read Status** = "1" indicates one or more RXP CPU Packet Read Status Latch bits = "1" (EMA.RSRL1) and enabled (EMA.RSRIE1). The combination of EMARS = 1 and G.GSRIE1.EMARIE = 1 forces an interrupt on INT_N.
GSR2.	**A:0034h**		**Global Status Register 2. Default: 0x00.00.00.00**
PPTCSL	[31:0]	rls-crw-i3	**Per-Port Transmit (RXP) CAS Latched Status** = "1" in PPTCSL bit position "x" (x = 0 to 31) indicates one or more received RXP CAS Codes for Transmit TDM Port "x" have changed. The combination of any PPTCSL[x] = 1 and its associated G.GSRIE2.PPTCSIE[x] = 1 will make G.GSR1.PTCS = 1.
GSR3.	**A:0038h**		**Global Status Register 3. Default: 0x00.00.00.00**
PPRCSL	[31:0]	rls-crw-i3	**Per-Port Receive (TXP) CAS Latched Status** = "1" in PPRCSL bit position "x" (x = 0 to 31) indicates one or more CAS Codes received from TDM Port "x" have changed (TXP direction). The combination of any PPRCSL[x] = 1 and its associated G.GSRIE3.PPRCSIE[x] = 1 will make G.GSR1.PRCS = 1.
GSR4.	**A:003Ch**		**Global Status Register 4. Default: 0x00.00.00.00**
PPS	[31:0]	ros-crw-i2	**Per-Port Latched Status** = "1" in PPS bit "x" (x = 0 to 31) indicates one or more Frame Alignment or Over/underrun errors have been detected at TDM Port "x" (any "Pn.PTSRL[z] and Pn.PTSRIE[z]" = 1 or any "Pn.PRSRL[z] and Pn.PTSRIE[z]" = 1; where "Pn" = "Port x" and z = bit 0 or bit 1). This is a latched status register, which means a 0 to 1 transition on any associated PTSRL[z]/PRSRL[z] forces a latched PPS=1. The G.GCR.LSBCRE register selects whether a Read or Write operation to GSR4 clears the register (-crw-; even if all PTSRL[z]/PRSRL[z] transition back to "0", a PPS[x] = 1 value will not clear until GSR4 is cleared by a Read or Write operation). Any PPS[x] = 1 will force G.GSR1.PS = 1.
GSR5.	**A:0040h**		**Global Status Register 5. Default: 0x00.00.00.00**
BGS	[31:0]	ros-_-i2	**Bundle Group Status** = "1" in BGS bit position "x" (x = 0 to 31) indicates one or more PW Control Word changes have been detected in Bundle Group "x" (any B.GxSRL[z] = 1 and B.GxSRIE[z] = 1 for z = 0 to 7). Bundles with a detected change can be identified from: Bundle # = BGS "x" bit position x 8 + B.GxSRL "z" bit position. Any BGS[x] = 1 (x = 0 to 31) will force G.GSR1.BS = 1.

G. Field Name	Addr (A:) Bit [x:y]	Type	Description
GSR6.	**A:0044h**		**Global Status Register 6. Default: 0x00.00.00.00**
JBGS	[31:0]	ros-_-i2	**Jitter Buffer Group Status** = "1" in JBGS bit position "x" (x = 0 - 31) indicates one or more Jitter Buffer Underruns have been detected in Bundle Group "x" (any JB.GxSRL[z] = 1 and JB.GxSRIE[z] = 1 for z = 0 to 7). Bundles with a detected change can be identified from: Bundle # = JBGS "x" bit position x 8 + JB.GxSRL "z" bit position. Any JBGS[x] = 1 (x = 0 to 31) will force G.GSR1.JBS = 1.
TPISR1.	**A:0048h**		**Transmit Packet Interface Status Register 1. Default: 0x00.00.00.00**
RSVD	[31:11]		**Reserved.**
TXHPQL	[10:0]	ros-_-_	**TXP High Priority Queue Level** indicates # packets in the queue (0 – 1024).
TPISR2.	**A:004Ch**		**Transmit Packet Interface Status Register 2. Default: 0x00.00.00.00**
RSVD	[31:11]		**Reserved.**
TXLPQL	[10:0]	ros-_-_	**TXP Low Priority Queue Level** indicates # packets in the queue (0 – 1024).
TPISR3.	**A:0050h**		**Transmit Packet Interface Status Register 3. Default: 0x00.00.00.00**
RSVD	[31:10]		**Reserved.**
TXCQL	[9:0]	ros-_-_	**TXP CPU Queue Level** indicates # packets in the queue (0 – 512).
TPISRL.	**A:0054h**		**Transmit Packet Interface Status Register Latches. Default: 0x00.00.00.00**
RSVD	[31:3]		**Reserved.**
HPQOSL	[2]	rls-crw-i1	**High Priority Queue Overflow Status Latch** = "1" indicates an overflow of the TXP TDM High Priority Queue (data discarded). The combination of HPQOSL = 1 and G.TPISRIE.HPQOSIE = 1 forces an interrupt on INT_N.
LPQOSL	[1]	rls-crw-i1	**Low Priority Queue Overflow Status Latch** = "1" indicates an overflow of the TXP TDM Low Priority Queue (data discarded). The combination of LPQOSL = 1 and G.TPISRIE.LPQOSIE = 1 forces an interrupt on INT_N.
RSVD	[0]		**Reserved.**
TPISRIE.	**A:0058h**		**Transmit Packet Interface Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:3]		**Reserved.**
HPQOSIE	[2]	rwc-_-i1	**High Priority Queue Overflow Status Interrupt Enable.** (see TPISRL.HPQOSL)
LPQOSIE	[1]	rwc-_-i1	**Low Priority Queue Overflow Status Interrupt Enable.** (see TPISRL.LPQOSL)
RSVD	[0]		**Reserved.**

10.3.1.3 Global Status Register Interrupt Enables (G.)

Table 10-5. Global Status Register Interrupt Enables (G.)

G. Field Name	Addr (A:) Bit [x:y]	Type	Description
GSRIE1.	**A:0060h**		**Global Status Register Interrupt Enable 1. Default: 0x00.00.00.00**
RSVD	[31:18]		**Reserved.**
EBSIE	[17]	rwc-_-i1	**Encap (Ethernet) BERT Status Interrupt Enable.** (see G.GSR1.EBS)
DBSIE	[16]	rwc-_-i1	**Decap (TDM Port) BERT Status Interrupt Enable.** (see G.GSR1.DBS)
PTCIE	[15]	rwc-_-i1	**Port Transmit (RXP) CAS Interrupt Enable.** (see G.GSR1.PTCS)
PRCIE	[14]	rwc-_-i1	**Port Receive (TXP) CAS Interrupt Enable.** (see G.GSR1.PRCS)
MIRIE	[13]	rwc-_-i1	**MAC Interrupt Register Interrupt Enable.** (see G.GSR1.MIRS)
CRHIE	[12:8]	rwc-_-i1	**Clock Recovery Hardware Interrupt Enable.** (see G.GSR1.CRHS)
BIE	[7]	rwc-_-i1	**Bundle Interrupt Enable.** (see G.GSR1.BS)

G. Field Name	Addr (A:) Bit [x:y]	Type	Description
JBUIE	[6]	rwc-_-i1	**Jitter Buffer Interrupt Enable.** (see G.GSR1.JBS)
PIE	[5]	rwc-_-i1	**Port Interrupt Enable.** (see G.GSR1.PS)
PCIE	[4]	rwc-_-i1	**Packet Classifier Interrupt Enable.** (see G.GSR1.PCS)
EMIIE	[3]	rwc-_-i1	**External Memory Interface Interrupt Enable.** (see G.GSR1.EMIS)
RSVD	[2]		**Reserved.**
EMAWIE	[1]	rwc-_-i1	**External Memory Access Write Interrupt Enable.** (see G.GSR1.EMAWS)
EMARIE	[0]	rwc-_-i1	**External Memory Access Read Interrupt Enable.** (see G.GSR1.EMARS)
GSRIE2.	**A:0064h**		**Global Status Register Interrupt Enable 2. Default: 0x00.00.00.00**
PPTCIE	[31:0]	rwc-_-i3	**Per Port Transmit (RXP) CAS Interrupt Enable.** (see G.GSR2.PPTCSL)
GSRIE3.	**A:0068h**		**Global Status Register Interrupt Enable 3. Default: 0x00.00.00.00**
PPRCIE	[31:0]	rwc-_-i3	**Per Port Receive (TXP) CAS Interrupt Enable.** (see G.GSR3.PPRCSL)

10.3.2 Bundle Registers (B.)

10.3.2.1 Bundle Reset Registers (B.)

Table 10-6. Bundle Reset Registers (G.)

B. Field Name	Addr (A:) Bit [x:y]	Type	Description
BRCR1.	**A:0080h**		**Bundle Reset Control Register 1. Default: 0x00.00.00.00**
SNS	[31:16]	rwc-_-_	**Sequence Number Seed** = Sequence # seed used in the next TXP packet for the Bundle number specified by RXTXBS when the TXP direction is released from reset (B.BRCR2.TXBRE). SNS should be a random/unpredictable value.
RSVD	[15:8]		**Reserved.**
RXTXBS	[7:0]	rwc-_-_	**RXP or TXP Bundle Select** specifies the Bundle Number that is used in the next Bundle Reset (B.BRCR2) or Bundle Reset Status (B.BRSR) operation. To change a Bundle Data Path Reset State, B.BRCR2 must be programmed first to specify the new RXP and TXP Data Path Reset States. Next a write to BRCR1 initiates the B.BRCR2 reset command to the Bundle specified by RXTXBS (and initiates a new TXP Sequence Number). To read the status of a Bundle Data Path Reset State, RXTXBS must be programmed first to specify the Bundle number. Next a read to B.BRSR will provide the status of the TXP and RXP Reset States for the Bundle specified by RXTXBS.
BRCR2.	**A:0084h**		**Bundle Reset Control Register 2. Default: 0x00.00.00.00**
RSVD	[31:2]		**Reserved.**
RXBRE	[1]	rwc-_-_	**RXP Bundle Reset Enable** selects the Reset State for the RXP Payload Data Path of the Bundle identified by B.BRCR1. RXBRE does not affect RXP Clock Recovery for SAT/CES Bundles with payload or SAT/CES Clock Only Bundles. 0 = Release Bundle Reset to forward payload data and reset Bundle Status 1 = Hold Bundle Data Path in reset (does not reset Bundle Status value)
TXBRE	[0]	rwc-_-_	**TXP Bundle Reset Enable** selects the Reset State for the TXP Bundle Payload Data Path identified by B.BRCR1. TXBRE disables transmission of TXP Bundles (it blocks the receive TDM Port data and disables TXP Bundle Status registers). 0 = Release Bundle Reset to forward payload data and reset Bundle Status 1 = Hold Bundle Data Path in reset (does not reset Bundle Status values)
BRSR.	**A:0088h**		**Bundle Reset Status Register. Default: 0x00.00.00.00**
RSVD	[31:2]		**Reserved.**

B. Field Name	Addr (A:) Bit [x:y]	Type	Description
RXBRS	[1]	ros-_-_	**RXP Bundle Reset Status** indicates whether the RXP packet payload data path for the selected Bundle (B.BRCR1.RXTXBS) is released from reset. 0 = The RXP side of the Bundle is in reset 1 = The RXP side of the Bundle is released from reset
TXBRS	[0]	ros-_-_	**TXP Bundle Reset Status** indicates whether the TXP packet payload path for the selected Bundle (B.BRCR1.RXTXBS) is released from reset. 0 = The TXP side of the Bundle is in reset 1 = The TXP side of the Bundle is released from reset

10.3.2.2 Bundle Data Control Registers (B.)

Table 10-7. Bundle Data Control Registers (B.)

B. Field Name	Addr (A:) Bit [x:y]	Type	Description
BACR.	**A:0094h**		**Bundle Activation Control Register. Default: 0x00.00.00.00**
RSVD	[31:13]		**Reserved.**
OBS	[12]	rwc-_-_	**OAM Bundle Select** selects whether B.BACR.BS is for a Bundle or OAM Bundle. 0 = BS is for a Bundle ID 1 = BS is for an OAM Bundle ID
WE	[11]	rwc-_-_	**Write Enable**, on a transition from zero to one, writes the B.BADR1 and B.BADR2 register values to the Bundle selected by B.BACR.OBS and BS.
RE	[10]	rwc-_-_	**Read Enable**, on a transition from zero to one, loads the B.BADR1 and B.BADR2 registers with values from the Bundle selected by B.BACR.OBS and BS. The B.BADR1 and B.BADR2 read operations may take more than one CPU access time, so the CPU should perform a no-op before reading the BADRx values.
RSVD	[9:8]	rwc-_-_	**Reserved.**
BS	[7:0]	rwc-_-_	**Bundle Select** specifies the Bundle or OAM Bundle Number that is used when accessing B.BADR1 and B.BADR2. When OBS = 0, the valid BS values are 0 to 255. When OBS = 1, the valid BS values are 0 to 31.
BCCR.	**A:0098h**		**Bundle Configuration Control Register. Default: 0x00.00.00.00**
RSVD	[31:12]		**Reserved.**
WE	[11]	rwc-_-_	**Write Enable**, on a transition from zero to one, writes the programmed settings in B.BCDR1 - B.BCDR5 registers to the Bundle selected by B.BCCR.BS.
RE	[10]	rwc-_-_	**Read Enable**, on a transition from zero to one, loads the B.BCDR1 - B.BCDR5 registers with values from the Bundle selected by B.BCCR.BS. The B.BCDR1 - B.BCDR5 read operations may take more than one CPU access time, so the CPU should perform a no-op before reading the BCDRx values.
RSVD	[9:8]	rwc-_-_	**Reserved.**
BS	[7:0]	rwc-_-_	**Bundle Select** selects the Bundle Number (0 – 255) that is used when accessing the B.BCDR1 - B.BCDR5 registers.
BESCR.	**A:009Ch**		**TXP Bundle Encap Status Control Register. Default: 0x00.00.00.00**
RSVD	[31:11]		**Reserved.**
ESRE	[10]	rwc-_-_	**Encap Status Read Enable**, on a transition from zero to one, loads the B.BESR1 – B.BESR3 registers with values from the Bundle selected by B.BESCR.ESBS. The B.BESR1 - B.BESR3 read operations may take more than one CPU access time, so the CPU should perform a no-op before reading the BESRx values.
RSVD	[9:8]	rwc-_-_	**Reserved.**
ESBS	[7:0]	rwc-_-_	**Encap Status Bundle Select** selects the Bundle Number (0 – 255) that is used when accessing the B.BESR1 – B.BESR3 registers.

B. Field Name	Addr (A:) Bit [x:y]	Type	Description
BDSCR.	**A:00A0h**		**RXP Bundle Decap Status Control Register. Default: 0x00.00.00.00**
RSVD	[31:11]		**Reserved.**
DSRE	[10]	rwc-_-_	**Decap Status Read Enable**, on a transition from zero to one, loads B.BDSR1 – B.BDSR9 with values from the Bundle selected by B.BDSCR.DSBS. The B.BDSR1 - B.BDSR9 read operations may take more than one CPU access time, so the CPU should perform a no-op before reading the BDSRx values.
RSVD	[9:8]	rwc-_-_	**Reserved.**
DSBS	[7:0]	rwc-_-_	**Decap Status Bundle Select** selects the Bundle Number (0 – 255) that is used when accessing the B.BDSR1 – B.BDSR9 registers.

10.3.2.3 Bundle Data Registers (B.)

Table 10-8. Bundle Data Registers (B.)

B. Field Name	Addr (A:) Bit [x:y]	Type	Description
BADR1.	**A:00A4h**		**Bundle Activation Data Register 1. Default: 0x00.00.00.00**
RSVD	[31:1]		**Reserved.**
ABE	[0]	rwd-_-_	**Active Bundle Enable** = "1" indicates the RXP Bundle selected by B.BACR is enabled. When "0" the RXP Bundle is disabled/ignored. This bit does not affect the Bundle's TXP direction. The chip reset functions disable all 256 Bundles (G.GRCR.RST and RST_N pin).
BADR2.	**A:00A8h**		**Bundle Activation Data Register 2. Default: 0x00.00.00.00**
BIDV	[31:0]	rwd-_-_	**Bundle ID Value** is the BID or OAM BID value for the Bundle Number or OAM Bundle Number selected by B.BACR. The bit width of BIDV varies as indicated below. When BIDV bit width <32, the unused MSbits of the BIDV must be "0". 32 bits - L2TPv3 and UDP when 32-bit width is selected by PC.PCCR1.UBIDLS 20 bits - MPLS and MEF 16 bits - UDP when 16-bit width is selected by PC.PCCR1.UBIDLS
BCDR1.	**A:00ACh**		**Bundle Configuration Data Register 1. Default: 0x00.00.00.00**
RSVD	[31:24]		**Reserved.**
LBCAI	[23]	rwd-_-_	**L Bit Conditioning Auto Insert** determines how the RXP packet payload is handled when L-bit = 1. This setting does not affect the Clock Recovery functions or the Jumped Packet Count. 0 = L-bit is ignored, payload is processed normally (no special handling). 1 = Discard RXP packet payload (if it exists). Note: If LBCAI = 1 and L-bit = 1, the packet is not counted as lost.
PMT	[22:21]	rwd-_-_	**Payload Machine Type.** 0 = HDLC Payload Machine Type 1 = Reserved 2 = Reserved 3 = SAT/CES Machine Type

B. Field Name	Addr (A:) Bit [x:y]	Type	Description
PMS	[20:10]	rwd-_-_	**Payload Size.** The Range of valid values is 1 – 2047 CES: # frames of assigned TDM Port data associated with each RXP/TXP packet (for CES with CAS, the PMS value does not include the CAS Signaling bytes) SAT: # bytes of assigned TDM Port data associated with each RXP/TXP packet. HDLC: Max # bytes in received HDLC packets (TXP direction only; not incl. FCS). Note: For SAT/CES packets that include payload data, PMS specifies # bytes or frames of TDM Port Data are carried in the packet payload. Packets for SAT/CES Clock Only Bundles do not include payload data, so PMS specifies # of assigned bytes/frames that exist on the TDM Port Line (but are not carried in the packet) for each RXP/TXP packet. The PMS setting has the same meaning for both, but the payload is deleted (does not exist) in the RXP/TXP packets. For HDLC, if a received packet size exceeds PMS, the packet is discarded.
SCSCFPD	[9]	rwd-_-_	**SAT/CES Sanity Check Fail Packet Discard.** 0 = Disable RXP Sanity Check 1 = Discard RXP packet if T1/E1 payload in RXP packet does not equal PMS. Note: For SAT/CES Bundle packets, if B.BCDR1.LBCAI = 1 and L-bit = 1 (“Invalid Payload” indication), the Sanity Check is auto-disabled for that packet. For HDLC and Clock Only Bundles the only valid setting is SCSCFPD = 0.
SCSNRE	[8]	rwd-_-_	**SAT/CES Sequence Number/HDLC Transmission Reordering Enable.** SAT/CES Bundles 0 = Disable RXP Sequence Number reordering 1 = Enable RXP Sequence Number reordering HDLC Bundles 0 = use HDLC MSB first transmission (transmit and receive directions) 1 = use HDLC LSB first transmission (transmit and receive directions)
SCRXBCSS	[7]	rwd-_-_	**SAT/CES RXP Bundle CAS Source Select / HDLC FCS Processing Disable.** SAT/CES Bundles 0 = When Jitter Buffer empties send last stored RXP CAS codes to TSIG/TDAT 1 = When Jitter Buffer empties send Xmt SW CAS codes (RXSCn) to TSIG/TDAT HDLC Bundles 0 = FCS processing is enabled (RXP & TXP directions). 1 = FCS processing is disabled (RXP & TXP directions).
SCTXBCSS	[6]	rwd-_-_	**SAT/CES TXP Bundle CAS Source Select / HDLC FCS 32 Bit Width Select.** SAT/CES Bundles 0 = Use CAS data received at TDM Port for TXP Bundle 1 = Use CAS data from TXP SW CAS codes (TXSCn) in TXP packets HDLC Bundles 0 = Use 16-bit FCS (RXP & TXP directions) 1 = Use 32-bit FCS (RXP & TXP directions)
RSNS	[5]	rwd-_-_	**Reorder Sequence Number Select.** 0 = Use the Control Word Sequence Number for reordering RXP packets 1 = Use the RTP Sequence Number for reordering RXP packets
SCTXCE	[4]	rwd-_-_	**SAT/CES TXP Conditioning Enable / HDLC Packet Sequence # Select 1.** SAT/CES Bundles When this bit is set the selected condition data (See SCTXCOS bits) will be sent in the packet to the PSN. When reset, normal operation is active. HDLC Bundles - see SCTXDFSE bit description.

B. Field Name	Addr (A:) Bit [x:y]	Type	Description
SCTXDFSE	[3]	rwd-_-_	**CES TXP Destination Framing SF or ESF / HDLC Packet Seq # Select 0.** CES Bundles (T1 Only; for TXP direction) 0 = Framer at far end PW termination point uses T1-SF framing. 1 = Framer at far end PW termination point uses T1-ESF framing. HDLC Bundles: Combined SCTXCE/SCTXDFSE bits (TXP & RXP directions); 0/0b = Sequence Number is always 0 0/1b = Sequence Number is auto-incremented and wrap-around uses zero 1/0b = Reserved 1/1b = Sequence Number is auto-incremented and wrap-around skips zero
SCTXCOS	[2:0]	rwd-_-_	**SAT/CES TXP Conditioning Octet Select / HDLC Time Slot Width Select.** SAT/CES Bundles – selects TXP packet Conditioning Data value 0 = Ethernet Conditioning Octet A (G.ECCR1.ECOA) 1 = Ethernet Conditioning Octet B (G.ECCR1.ECOB) 2 = Ethernet Conditioning Octet C (G.ECCR1.ECOC) 3 = Ethernet Conditioning Octet D (G.ECCR1.ECOD) 4 = Ethernet Conditioning Octet E (G.ECCR2.ECOE) 5 = Ethernet Conditioning Octet F (G.ECCR2.ECOF) 6 = Ethernet Conditioning Octet G (G.ECCR2.ECOG) 7 = Ethernet Conditioning Octet H (G.ECCR2.ECOH) HDLC Bundles – HDLC Encapsulation bit-width 0 = Use Nx8-bit HDLC encapsulation (for Unstructured and Nx64 Kb/s HDLC) 1 = Use Structured 7-bit HDLC encapsulation + 1 unassigned bit 2 = Use Structured 2-bit HDLC encapsulation (2 MSbits) + 6 unassigned LSbits 3 = Use Structured 2-bit HDLC encapsulation (2 LSbits) + 6 unassigned MSbits 4 to 7 reserved
BCDR2.	**A:00B0h**		**Bundle Configuration Data Register 2. Default: 0x00.00.00.00**
ATSS	[31:0]	rwd-_-_	**Active Time Slot Select** selects which TDM Port Timeslots are used by this Bundle (TXP and RXP directions). One bit for each Timeslot (E1: 0 – 31; T1: 0 – 23). ATSS[x] = 0 = Timeslot “x” disabled. 1 = Timeslot “x” enabled. For an Unstructured Bundle (SAT or HDLC), ATSS = 0x0000.0001.
BCDR3.	**A:00B4h**		**Bundle Configuration Data Register 3. Default: 0x00.00.00.00**
RSVD	[31:5]		**Reserved.**
TXPMS	[4:3]	rwd-_-_	**TXP Packet Mode Select.** 0 = Stop Transmission of TXP packets (CES, SAT, HDLC and Clock Only) 1 = Transmit TXP packets with payload (CES, SAT and HDLC) 2 = Transmit TXP packets without payload (Clock Only) 3 = reserved
TXBTS	[2:1]	rwd-_-_	**TXP Bundle Structure Type Select.** 0 = SAT or HDLC for Unstructured TDM Port 1 = CES without CAS or HDLC for Structured T1/E1 Port 2 = CES with CAS or HDLC for Structured T1/E1 Port 3 = Reserved
TXBPS	[0]	rwd-_-_	**TXP Bundle Priority Select** selects transmit priority for SAT/CES TXP packets. 0 = Low priority (“normal” for Bundles not used for far end Clock Recovery) 1 = High priority (“normal” for Bundles used for far end Clock Recovery)
BCDR4.	**A:00B8h**		**Bundle Configuration Data Register 4. Default: 0x00.00.00.00**
RSVD	[31:22]		**Reserved.**
RXRE	[21]	rwd-_-_	**RXP RTP Enable.** 0 = RTP header is not accepted in RXP packets. 1 = RTP header is required

B. Field Name	Addr (A:) Bit [x:y]	Type	Description
RXCWE	[20]	rwd-_-_	**RXP Control Word Enable.** 0 = Control Word is not accepted in RXP packets (optional for HDLC) 1 = Control Word is required (only valid setting for CES/SAT; optional for HDLC)
RXHTS	[19:18]	rwd-_-_	**RXP Header Type Select** selects the PW Header Type that is required. 0 = MPLS (MFA-8) 1 = UDP over IP (IPv4 or IPv6) 2 = L2TPv3 over IP (IPv4 or IPv6) 3 = MEF-8
RXBTS	[17:16]	rwd-_-_	**RXP Bundle Structure Type Select.** 0 = SAT or HDLC for Unstructured TDM Port 1 = CES without CAS or HDLC for Structured T1/E1 Port 2 = CES with CAS or HDLC for Structured T1/E1 Port 3 = Reserved
RXLCS	[15:14]	rwd-_-_	**RXP Labels Cookie Select** selects maximum # of Labels or Cookies allowed. MPLS: 0 = Reserved 1 = One label in the RXP MPLS Header (1 Inner Label) 2 = Two labels in the RXP MPLS Header (1 Inner and 1 Outer Label) 3 = Three labels in the RXP MPLS Header (1 Inner and 2 Outer Labels) L2TPv3: 0 = No Cookies in the RXP L2TPv3 Header 1 = One Cookie in the RXP L2TPv3 Header 2 = Two Cookies in the RXP L2TPv3 Header 3 = Reserved
RXUBIDLS	[13]	rwd-_-_	**RXP UDP Bundle ID Location Select** selects UDP BID location when PC.CR1.UBIDLS= 0 and PC.CR1.UBIDLCE = 1 (otherwise RXUBIDLS is ignored) 0 = Test UDP Source Port for BID match 1 = Test UDP Destination Port for BID match
SCLVI	[12]	rwd-_-_	**CES Last Value Insertion.** CES Bundles – selects type of data transmitted in place of missing RXP packets 0 = Last Value Insertion disabled, use Conditioning Data (B.BCDR1.SCTXCOS) 1 = Repeat Timeslot data for up to 3 TDM frames then use Conditioning Data HDLC Bundles - selects Inter-frame Fill used between transmit HDLC packets. 0 = Use 0x7E for Inter-frame Fill 1 = Use all ones for Inter-frame Fill
RXCOS	[11:9]	rwd-_-_	**Xmt Conditioning Octet Select** selects Xmt Conditioning Data transmitted at TDM Port for unassigned timeslots, missing packets and empty Jitter Buffer. 0 = TDM Conditioning Octet A (G.TCCR1.TCOA) 1 = TDM Conditioning Octet B (G.TCCR1.TCOB) 2 = TDM Conditioning Octet C (G.TCCR1.TCOC) 3 = TDM Conditioning Octet D (G.TCCR1.TCOD) 4 = TDM Conditioning Octet E (G.TCCR2.TCOE) 5 = TDM Conditioning Octet F (G.TCCR2.TCOF) 6 = TDM Conditioning Octet G (G.TCCR2.TCOG) 7 = TDM Conditioning Octet H (G.TCCR2.TCOH)
RXOICWE	[8]	rwd-_-_	**RXP OAM In Control Word Enable** enables processing of In-band VCCV packets when Control Word matches PC.CR5.VOV and PC.CR5.VOM. 0 = Do not look for In-band VCCV indication in Control Word 1 = Send "In-band VCCV" packet to CPU or discard according to PC.CR1.DPS7
RXBDS	[7:6]	rwd-_-_	**RXP Bundle Destination Select.** 0 = Send packet to SAT/CES Jitter Buffer or HDLC Buffer 1 = Send packet to CPU ("CPU Debug RXP PW Bundle" setting) 2 = reserved 3 = Discard the packet (timing information is still available for Clock Recovery)

B. Field Name	Addr (A:) Bit [x:y]	Type	Description
PNS	[5:1]	rwd-_-_	**Port Number Select** selects TDM Port associated with Bundle (PNS = 0 = TDM Port #1, PNS = 31 = TDM Port #32).
PCRE	[0]	rwd-_-_	**Port Clock Recovery Enable.** 0 = Do not use this Bundle for Clock Recovery 1 = Use RXP Bundle for Clock Recovery (enable Clock Connection)
BCDR5.	**A:00BCh**		**Bundle Configuration Data Register 5. Default: 0x00.00.00.00**
RSVD	[31:25]		**Reserved.**
PDVT	[24:10]	rwd-_-_	**Packet Delay Variation Time** selects minimum Jitter Buffer fill level required before RXP payload data is forwarded to the transmit TDM Port. This function is not used with Clock Only Bundles. SAT Bundles with payload and SAT Clock Only Bundles Forward data when fill level = PDVT * 32 TDM Port bit periods CES Bundles with payload and CES Clock Only Bundles (set unused bits = "0") For Pn.PTCR1.BFD = 1: Forward when fill level = PDVT * 125 us; use bits [19:10] For Pn.PTCR1.BFD = 2: Forward when fill level = PDVT * 250 us; use bits [20:10] For Pn.PTCR1.BFD = 3: Forward when fill level = PDVT * 500 us; use bits [21:10]
MJBS	[9:0]	rwd-_-_	**Maximum Jitter Buffer Sense** selects Jitter Buffer Overrun Fill level that increments the Overrun Event Count (JEBEC). This function is not used with Clock Only Bundles. This function does not generate an interrupt. SAT Bundles: Overrun Fill Level = MJBS * 1024 TDM Port bit periods CES Bundles: Overrun Fill Level = MJBS * 500 us
BESR1.	**A:00C0h**		**Bundle Encap Status Register 1. Default: 0x00.00.00.00**
PRHASL	[31]	rld-cor-_	**Port Receive HDLC Abort Status Latch** = "1" indicates one or more HDLC Abort codes have been detected on one or more receive TDM Ports.
RSVD	[30:20]		**Reserved.**
PRHEFC	[19:0]	rld-cnr-nc	**Port Receive HDLC Error Frame Count** = number of receive TDM Port HDLC frames with an error (including FCS, alignment, abort, too short or too long).
BESR2.	**A:00C4h**		**Bundle Encap Status Register 2. Default: 0x00.00.00.00**
GPTXC	[31:0]	rld-cnr-nc	**Good Packet TXP Count** = # transmitted TXP packets (all Bundle types)
BESR3.	**A:00C8h**		**Bundle Encap Status Register 3. Default: 0x00.00.00.00**
RSVD	[31:5]		**Reserved.**
SHFSL	[4]	rld-cor-_	**Short HDLC Frame Status Latch** = "1" indicates the size for one or more receive TDM Port HDLC frames was < 4 bytes (including FCS bytes).
LHFSL	[3]	rld-cor-_	**Long HDLC Frame Status Latch** = "1" indicates the size for one or more received HDLC frames was > maximum size (including FCS; B.BCDR1.PMS)
AESL	[2]	rld-cor-_	**Alignment Error Status Latch** = "1" indicates one or more receive TDM Port HDLC frames had an alignment error.
CESL	[1]	rld-cor-_	**CRC Error Status Latch** = "1" indicates one or more receive TDM Port HDLC frames had a CRC (FCS) Error.
TXPSFSL	[0]	rld-cor-_	**TXP Packet Space Full Status Latch** = "1" indicates one or more receive TDM Port HDLC frames were discarded due to TXP packet buffer overflow in SDRAM.
BDSR1.	**A:00D0h**		**Bundle Decap Status Register 1. Default: 0x00.00.00.00**
JBLPDSL	[31]	rld-cor-_	**Jitter Buffer Late Packet Discard Status Latch** = "1" indicates one or more RXP packets discarded due to late arrival (Sequence # already passed; SAT/CES).
RSVD	[30:20]		**Reserved.**
PDC	[19:0]	rld-cnr-nc	**Packet stream Defect Count** = # SAT/CES RXP packet stream defect events. PC.CR21.PDCC selects which defect conditions are counted. BDSR1.PDC and BDSR2.JBEC can be programmed to count the same or different conditions. Not valid for Clock Only Bundles. For HDLC Bundles only Overruns can be counted.

B. Field Name	Addr (A:) Bit [x:y]	Type	Description
BDSR2.	**A:00D4h**		**Bundle Decap Status Register 2. Default: 0x00.00.00.00**
RSVD	[31:20]		**Reserved.**
JBEC	[19:0]	rld-cnr-nc	**Jitter Buffer Event Count** = # SAT/CES RXP packet stream defect events. PC.CR21.JBECC selects which defect conditions are counted. BDSR1.PDC and BDSR2.JBEC can be programmed to count the same or different conditions. Not valid for Clock Only Bundles. For HDLC Bundles only Overruns can be counted.
BDSR3.	**A:00D8h**		**Bundle Decap Status Register 3. Default: 0x00.00.00.00**
RSVD	[31]		**Reserved.**
JBLL	[30:16]	rld-cor-_	**Jitter Buffer Low Level** = lowest Jitter Buffer fill level since last read. A read operation forces JBLL = “all ones” until next Jitter Buffer current level available. When Underrun is reached, the value remains zero until it is read by the CPU. The # JBLL bits is equal to the # JBCL bits (not valid for HDLC or Clock Only).
RSVD	[15]		**Reserved.**
JBHL	[14:0]	rld-cor-_	**Jitter Buffer High Level** = highest Jitter Buffer fill level since last read (not valid for HDLC or Clock Only). A read operation forces JBLL = “all zeros” until next Jitter Buffer current level available. When Overrun is reached, JBHL = B.BCDR5.MJBS until read by CPU. The # JBHL bits is equal to the # JBCL bits.
BDSR4.	**A:00DCh**		**Bundle Decap Status Register 4. Default: 0x00.00.00.00**
GPRXC	[31:0]	rld-cnr-nc	**Good Packet RXP Count** = # received good RXP packets (all Bundle types)
BDSR5.	**A:00E0h**		**Bundle Decap Status Register 5. Default: 0x00.00.00.00**
SCJPC	[31:0]	rld-cnr-nc	**SAT/CES Jumped/Lost Packet Count** indicates how many Jumped or Lost Sequence # conditions have been detected (according to G.GCR.JLPC). SAT/CES Bundles – accumulated difference between expected and received packet Sequence #. Total Missing Packets can be calculated with: Total Missing Packets = Jumped Packets – Re-ordered Packets = (B.BSDR5.SCJPC – B.BSDR6.SCRPC) HDLC Bundles (Jumped Count only) – accumulated difference between expected and received packet Sequence # for difference < 32,768 (see B.BSDR6.SCRPC).
BDSR6.	**A:00E4h**		**Bundle Decap Status Register 6. Default: 0x00.00.00.00**
RSVD	[31:20]		**Reserved.**
SCRPC	[19:0]	rld-cnr-nc	**SAT/CES Reordered/Duplicate Packet Count** indicates how many Re-ordered or Duplicate packet conditions have been detected (according to G.GCR.RDPC). SAT/CES Bundles - # successfully Re-ordered or Duplicate packet events. HDLC Bundles - # RXP packet with a Sequence Number Jump > 32,767.
BDSR7.	**A:00E8h**		**Bundle Decap Status Register 7. Default: 0x00.00.00.00**
RSVD	[31:24]		**Reserved.**
SCPSES L	[23]	rld-cor-_	**SAT/CES Payload Size/Sequence Error Status Latch.** SAT Bundles: “1” = 1 or more RXP packets with payload size ≠ B.BCDR1.PMS CES Bundles: “1” = 1 or more RXP packets with payload size ≠ B.BCDR1.PMS (for CES with CAS this test function includes the expected CAS Signaling bytes). HDLC Bundles “1” = 1 or more RXP packets with late or early Sequence Number Jump > 32,768.

B. Field Name	Addr (A:) Bit [x:y]	Type	Description
PLESL	[22]	rld-cor-_	**Packet Length Error Status Latch** "1" = one (or more) RXP packets were detected with a payload size larger than indicated by the header length fields (as indicated below). The RTP and Control Word fields are optional according to the B.BCDR4.RXRE and B.BCDR4.RXCWE settings. IPv4/IPv6 - IPv4 Total Length field > (actual payload + IP Header + Control Word + RTP) IPv6 Payload Length field > (actual payload + IP Header + Control Word + RTP) MPLS - Control Word Length field > (actual payload + Control Word + RTP)
SCJBEP DSL	[21]	rld-cor-_	**SAT/CES Jitter Buffer Early Packet Discard Status Latch.** SAT/CES Bundles "1" = one (or more) RXP packets were discarded due to a Sequence Number that was earlier than the Jitter Buffer Current Level. HDLC Bundles "1" = one (or more) RXP packets were discarded due to an HDLC buffer overflow.
JBCL	[20:6]	rod-_-_	**Jitter Buffer Current Level.** SAT Bundles Jitter Buffer Fill Level = JBCL * 32 TDM Port Bit Periods CES Bundles For Pn.PTCR1.BFD = 1: Jitter Buffer Fill Level = JBCL * 125 us For Pn.PTCR1.BFD = 2: Jitter Buffer Fill Level = JBCL * 250 us For Pn.PTCR1.BFD = 3: Jitter Buffer Fill Level = JBCL * 500 us
LBD	[5]	rod-_-_	**L Bit Data** = Control Word L-bit state in most recent RXP packet for this Bundle.
RBD	[4]	rod-_-_	**R Bit Data** = Control Word R-bit state in most recent RXP packet for this Bundle.
DMD	[3:2]	rod-_-_	**Defect Modifier Data** = the state of the Control Word M-bits in the most recent RXP packet for this Bundle (one for each M-bit).
FBD	[1:0]	rod-_-_	**Fragmentation Bit Data** = the state for the Control Word Frag-bits in the most recent RXP packet for this Bundle (one for each Frag bit).
BDSR8.	**A:00ECh**		**Bundle Decap Status Register 8. Default: 0x00.00.00.00**
RSVD	[31:20]		**Reserved.**
SCMPC	[19:0]	rld-cor-_	**SAT/CES Malformed Packet Count** = number received packets that fail to match the configured payload length, but excluding packets with L-bit = 1. This count is enabled and incremented by the Sanity Check function (B.BCDR1.SCSCFPD).
BDSR9.	**A:00F0h**		**Bundle Decap Status Register 9. Default: 0x00.00.00.00**
RSVD	[31:20]		**Reserved.**
SCRBPC	[19:0]	rld-cor-_	**SAT/CES R-Bit Packet Count** = # received packets with Control Word, R bit = 1.

10.3.2.4 Bundle Status Latch Registers (B.)

Table 10-9. Bundle Status Latch Registers (B.)

B. Field Name	Addr (A:) Bit [x:y]	Type	Description
G0SRL.	**A:0100h**		**Group 0 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDSL [7:0]	[7:0]	rls-cor-i3	**Control Word Change Detect Status Latch** = "1" indicates change detected in a Control Word for a Bundle. Bits [7:0] indicate Bundles [7:0] respectively.

B. Field Name	Addr (A:) Bit [x:y]	Type	Description
G1SRL.	**A:0104h**		**Group 1 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDSL [15:8]	[7:0]	rls-cor-i3	**Control Word Change Detect Status Latch** = "1" indicates change detected in a Control Word for a Bundle. Bits [7:0] indicate Bundles [15:8] respectively.
G2SRL.	**A:0108h**		**Group 2 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDSL [23:16]	[7:0]	rls-cor-i3	**Control Word Change Detect Status Latch** = "1" indicates change detected in a Control Word for a Bundle. Bits [7:0] indicate Bundles [23:16] respectively.
G3SRL.	**A:010Ch**		**Group 3 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDSL [31:24]	[7:0]	rls-cor-i3	**Control Word Change Detect Status Latch** = "1" indicates change detected in a Control Word for a Bundle. Bits [7:0] indicate Bundles [31:24] respectively.
G4SRL.	**A:0110h**		**Group 4 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDSL [39:32]	[7:0]	rls-cor-i3	**Control Word Change Detect Status Latch** = "1" indicates change detected in a Control Word for a Bundle. Bits [7:0] indicate Bundles [39:32] respectively.
G5SRL.	**A:0114h**		**Group 5 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDSL [47:40]	[7:0]	rls-cor-i3	**Control Word Change Detect Status Latch** = "1" indicates change detected in a Control Word for a Bundle. Bits [7:0] indicate Bundles [47:40] respectively.
G6SRL.	**A:0118h**		**Group 6 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDSL [55:48]	[7:0]	rls-cor-i3	**Control Word Change Detect Status Latch** = "1" indicates change detected in a Control Word for a Bundle. Bits [7:0] indicate Bundles [55:48] respectively.
G7SRL.	**A:011Ch**		**Group 7 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDSL [63:56]	[7:0]	rls-cor-i3	**Control Word Change Detect Status Latch** = "1" indicates change detected in a Control Word for a Bundle. Bits [7:0] indicate Bundles [63:56] respectively.
G8SRL.	**A:0120h**		**Group 8 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDSL [71:64]	[7:0]	rls-cor-i3	**Control Word Change Detect Status Latch** = "1" indicates change detected in a Control Word for a Bundle. Bits [7:0] indicate Bundles [71:64] respectively.
G9SRL.	**A:0124h**		**Group 9 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDSL [79:72]	[7:0]	rls-cor-i3	**Control Word Change Detect Status Latch** = "1" indicates change detected in a Control Word for a Bundle. Bits [7:0] indicate Bundles [79:72] respectively.
G10SRL.	**A:0128h**		**Group 10 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDSL [87:80]	[7:0]	rls-cor-i3	**Control Word Change Detect Status Latch** = "1" indicates change detected in a Control Word for a Bundle. Bits [7:0] indicate Bundles [87:80] respectively.
G11SRL.	**A:012Ch**		**Group 11 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDSL [95:88]	[7:0]	rls-cor-i3	**Control Word Change Detect Status Latch** = "1" indicates change detected in a Control Word for a Bundle. Bits [7:0] indicate Bundles [95:88] respectively.

B. Field Name	Addr (A:) Bit [x:y]	Type	Description
G12SRL.	**A:0130h**		**Group 12 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDSL [103:96]	[7:0]	rls-cor-i3	**Control Word Change Detect Status Latch** = "1" indicates change detected in a Control Word for a Bundle. Bits [7:0] indicate Bundles [103:96] respectively.
G13SRL.	**A:0134h**		**Group 13 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDSL [111:104]	[7:0]	rls-cor-i3	**Control Word Change Detect Status Latch** = "1" indicates change detected in a Control Word for a Bundle. Bits [7:0] indicate Bundles [111:104] respectively.
G14SRL.	**A:0138h**		**Group 14 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDSL [119:112]	[7:0]	rls-cor-i3	**Control Word Change Detect Status Latch** = "1" indicates change detected in a Control Word for a Bundle. Bits [7:0] indicate Bundles [119:112] respectively.
G15SRL.	**A:013Ch**		**Group 15 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDSL [127:120]	[7:0]	rls-cor-i3	**Control Word Change Detect Status Latch** = "1" indicates change detected in a Control Word for a Bundle. Bits [7:0] indicate Bundles [127:120] respectively.
G16SRL.	**A:0140h**		**Group 16 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDSL [135:128]	[7:0]	rls-cor-i3	**Control Word Change Detect Status Latch** = "1" indicates change detected in a Control Word for a Bundle. Bits [7:0] indicate Bundles [135:128] respectively.
G17SRL.	**A:0144h**		**Group 17 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDSL [143:136]	[7:0]	rls-cor-i3	**Control Word Change Detect Status Latch** = "1" indicates change detected in a Control Word for a Bundle. Bits [7:0] indicate Bundles [143:136] respectively.
G18SRL.	**A:0148h**		**Group 18 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDSL [151:144]	[7:0]	rls-cor-i3	**Control Word Change Detect Status Latch** = "1" indicates change detected in a Control Word for a Bundle. Bits [7:0] indicate Bundles [151:144] respectively.
G19SRL.	**A:014Ch**		**Group 19 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDSL [159:152]	[7:0]	rls-cor-i3	**Control Word Change Detect Status Latch** = "1" indicates change detected in a Control Word for a Bundle. Bits [7:0] indicate Bundles [159:152] respectively.
G20SRL.	**A:0150h**		**Group 20 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDSL [167:160]	[7:0]	rls-cor-i3	**Control Word Change Detect Status Latch** = "1" indicates change detected in a Control Word for a Bundle. Bits [7:0] indicate Bundles [167:160] respectively.
G21SRL.	**A:0154h**		**Group 21 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDSL [175:168]	[7:0]	rls-cor-i3	**Control Word Change Detect Status Latch** = "1" indicates change detected in a Control Word for a Bundle. Bits [7:0] indicate Bundles [175:168] respectively.
G22SRL.	**A:0158h**		**Group 22 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDSL [183:176]	[7:0]	rls-cor-i3	**Control Word Change Detect Status Latch** = "1" indicates change detected in a Control Word for a Bundle. Bits [7:0] indicate Bundles [183:176] respectively.

B. Field Name	Addr (A:) Bit [x:y]	Type	Description
G23SRL.	**A:015Ch**		**Group 23 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDSL [191:184]	[7:0]	rls-cor-i3	**Control Word Change Detect Status Latch** = "1" indicates change detected in a Control Word for a Bundle. Bits [7:0] indicate Bundles [191:184] respectively.
G24SRL.	A:0160h		**Group 24 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDSL [199:192]	[7:0]	rls-cor-i3	**Control Word Change Detect Status Latch** = "1" indicates change detected in a Control Word for a Bundle. Bits [7:0] indicate Bundles [199:192] respectively.
G25SRL.	**A:0164h**		**Group 25 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDSL [207:200]	[7:0]	rls-cor-i3	**Control Word Change Detect Status Latch** = "1" indicates change detected in a Control Word for a Bundle. Bits [7:0] indicate Bundles [207:200] respectively.
G26SRL.	**A:0168h**		**Group 26 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDSL [215:208]	[7:0]	rls-cor-i3	**Control Word Change Detect Status Latch** = "1" indicates change detected in a Control Word for a Bundle. Bits [7:0] indicate Bundles [215:208] respectively.
G27SRL.	**A:016Ch**		**Group 27 Status Register Latch.**
RSVD	[31:8]		**Reserved.**
CWCDSL [223:216]	[7:0]	rls-cor-i3	**Control Word Change Detect Status Latch** = "1" indicates change detected in a Control Word for a Bundle. Bits [7:0] indicate Bundles [223:216] respectively.
G28SRL.	**A:0170h**		**Group 28 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDSL [231:224]	[7:0]	rls-cor-i3	**Control Word Change Detect Status Latch** = "1" indicates change detected in a Control Word for a Bundle. Bits [7:0] indicate Bundles [231:224] respectively.
G29SRL.	**A:0174h**		**Group 29 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDSL [239:232]	[7:0]	rls-cor-i3	**Control Word Change Detect Status Latch** = "1" indicates change detected in a Control Word for a Bundle. Bits [7:0] indicate Bundles [239:232] respectively.
G30SRL.	**A:0178h**		**Group 30 Status Register Latch.**
RSVD	[31:8]		**Reserved.**
CWCDSL [247:240]	[7:0]	rls-cor-i3	**Control Word Change Detect Status Latch** = "1" indicates change detected in a Control Word for a Bundle. Bits [7:0] indicate Bundles [247:240] respectively.
G31SRL.	**A:017Ch**		**Group 31 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDSL [255:248]	[7:0]	rls-cor-i3	**Control Word Change Detect Status Latch** = "1" indicates change detected in a Control Word for a Bundle. Bits [7:0] indicate Bundles [255:248] respectively.

10.3.2.5 Bundle Status Register Interrupt Enables (B.)

Table 10-10. Bundle Status Register Interrupt Enables (B.)

B. Field Name	Addr (A:) Bit [x:y]	Type	Description
G0SRIE.	**A:0180h**		**Group 0 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**

B. Field Name	Addr (A:) Bit [x:y]	Type	Description
CWCDIE [7:0]	[7:0]	rwc-_-i3	**Control Word Change Detect Interrupt Enable.** For z = 0 to 7, the combination of B.G0SRL[z] = 1 and B.G0SRIE[z] = 1, forces G.GSR5[0] = 1.
G1SRIE.	**A:0184h**		**Group 1 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDIE [15:8]	[7:0]	rwc-_-i3	**Control Word Change Detect Interrupt Enable.** For z = 0 to 7, the combination of B.G1SRL[z] = 1 and B.G1SRIE[z] = 1, forces G.GSR5[1] = 1.
G2SRIE.	**A:0188h**		**Group 2 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDIE [23:16]	[7:0]	rwc-_-i3	**Control Word Change Detect Interrupt Enable.** For z = 0 to 7, the combination of B.G2SRL[z] = 1 and B.G2SRIE[z] = 1, forces G.GSR5[2] = 1.
G3SRIE.	**A:018Ch**		**Group 3 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDIE [31:24]	[7:0]	rwc-_-i3	**Control Word Change Detect Interrupt Enable.** For z = 0 to 7, the combination of B.G3SRL[z] = 1 and B.G3SRIE[z] = 1, forces G.GSR5[3] = 1.
G4SRIE.	**A:0190h**		**Group 4 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDIE [39:32]	[7:0]	rwc-_-i3	**Control Word Change Detect Interrupt Enable.** For z = 0 to 7, the combination of B.G4SRL[z] = 1 and B.G4SRIE[z] = 1, forces G.GSR5[4] = 1.
G5SRIE.	**A:0194h**		**Group 5 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDIE [47:40]	[7:0]	rwc-_-i3	**Control Word Change Detect Interrupt Enable.** For z = 0 to 7, the combination of B.G5SRL[z] = 1 and B.G5SRIE[z] = 1, forces G.GSR5[5] = 1.
G6SRIE.	**A:0198h**		**Group 6 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDIE [55:48]	[7:0]	rwc-_-i3	**Control Word Change Detect Interrupt Enable.** For z = 0 to 7, the combination of B.G6SRL[z] = 1 and B.G6SRIE[z] = 1, forces G.GSR5[6] = 1.
G7SRIE.	**A:019Ch**		**Group 7 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDIE [63:56]	[7:0]	rwc-_-i3	**Control Word Change Detect Interrupt Enable.** For z = 0 to 7, the combination of B.G7SRL[z] = 1 and B.G7SRIE[z] = 1, forces G.GSR5[7] = 1.
G8SRIE.	**A:01A0h**		**Group 8 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDIE [71:64]	[7:0]	rwc-_-i3	**Control Word Change Detect Interrupt Enable.** For z = 0 to 7, the combination of B.G8SRL[z] = 1 and B.G8SRIE[z] = 1, forces G.GSR5[8] = 1.
G9SRIE.	**A:01A4h**		**Group 9 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDIE [79:72]	[7:0]	rwc-_-i3	**Control Word Change Detect Interrupt Enable.** For z = 0 to 7, the combination of B.G9SRL[z] = 1 and B.G9SRIE[z] = 1, forces G.GSR5[9] = 1.
G10SRIE.	**A:01A8h**		**Group 10 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDIE [87:80]	[7:0]	rwc-_-i3	**Control Word Change Detect Interrupt Enable.** For z = 0 to 7, the combination of B.G10SRL[z] = 1 and B.G10SRIE[z] = 1, forces G.GSR5[10] = 1.
G11SRIE.	**A:01ACh**		**Group 11 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**

B. Field Name	Addr (A:) Bit [x:y]	Type	Description
CWCDIE [95:88]	[7:0]	rwc-_-i3	**Control Word Change Detect Interrupt Enable.** For z = 0 to 7, the combination of B.G11SRL[z] = 1 and B.G11SRIE[z] = 1, forces G.GSR5[11] = 1.
G12SRIE.	**A:01B0h**		**Group 12 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDIE [103:96]	[7:0]	rwc-_-i3	**Control Word Change Detect Interrupt Enable.** For z = 0 to 7, the combination of B.G12SRL[z] = 1 and B.G12SRIE[z] = 1, forces G.GSR5[12] = 1.
G13SRIE.	**A:01B4h**		**Group 13 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDIE [111:104]	[7:0]	rwc-_-i3	**Control Word Change Detect Interrupt Enable.** For z = 0 to 7, the combination of B.G13SRL[z] = 1 and B.G13SRIE[z] = 1, forces G.GSR5[13] = 1.
G14SRIE.	**A:01B8h**		**Group 14 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDIE [119:112]	[7:0]	rwc-_-i3	**Control Word Change Detect Interrupt Enable.** For z = 0 to 7, the combination of B.G14SRL[z] = 1 and B.G14SRIE[z] = 1, forces G.GSR5[14] = 1.
G15SRIE.	**A:01BCh**		**Group 15 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDIE [127:120]	[7:0]	rwc-_-i3	**Control Word Change Detect Interrupt Enable.** For z = 0 to 7, the combination of B.G15SRL[z] = 1 and B.G15SRIE[z] = 1, forces G.GSR5[15] = 1.
G16SRIE.	**A:01C0h**		**Group 16 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDIE [135:128]	[7:0]	rwc-_-i3	**Control Word Change Detect Interrupt Enable.** For z = 0 to 7, the combination of B.G16SRL[z] = 1 and B.G16SRIE[z] = 1, forces G.GSR5[16] = 1.
G17SRIE.	**A:01C4h**		**Group 17 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDIE [143:136]	[7:0]	rwc-_-i3	**Control Word Change Detect Interrupt Enable.** For z = 0 to 7, the combination of B.G17SRL[z] = 1 and B.G17SRIE[z] = 1, forces G.GSR5[17] = 1.
G18SRIE.	**A:01C8h**		**Group 18 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDIE [151:144]	[7:0]	rwc-_-i3	**Control Word Change Detect Interrupt Enable.** For z = 0 to 7, the combination of B.G18SRL[z] = 1 and B.G18SRIE[z] = 1, forces G.GSR5[18] = 1.
G19SRIE.	**A:01CCh**		**Group 19 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDIE [159:152]	[7:0]	rwc-_-i3	**Control Word Change Detect Interrupt Enable.** For z = 0 to 7, the combination of B.G19SRL[z] = 1 and B.G19SRIE[z] = 1, forces G.GSR5[19] = 1.
G20SRIE.	**A:01D0h**		**Group 20 Status Register Interrupt Enable.**
RSVD	[31:8]		**Reserved.**
CWCDIE [167:160]	[7:0]	rwc-_-i3	**Control Word Change Detect Interrupt Enable.** For z = 0 to 7, the combination of B.G20SRL[z] = 1 and B.G20SRIE[z] = 1, forces G.GSR5[20] = 1.
G21SRIE.	**A:01D4h**		**Group 21 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDIE [175:168]	[7:0]	rwc-_-i3	**Control Word Change Detect Interrupt Enable.** For z = 0 to 7, the combination of B.G21SRL[z] = 1 and B.G21SRIE[z] = 1, forces G.GSR5[21] = 1.
G22SRIE.	**A:01D8h**		**Group 22 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**

B. Field Name	Addr (A:) Bit [x:y]	Type	Description
CWCDIE [183:176]	[7:0]	rwc-_-i3	**Control Word Change Detect Interrupt Enable.** For z = 0 to 7, the combination of B.G22SRL[z] = 1 and B.G22SRIE[z] = 1, forces G.GSR5[22] = 1.
G23SRIE.	**A:01DCh**		**Group 23 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDIE [191:184]	[7:0]	rwc-_-i3	**Control Word Change Detect Interrupt Enable.** For z = 0 to 7, the combination of B.G23SRL[z] = 1 and B.G23SRIE[z] = 1, forces G.GSR5[23] = 1.
G24SRIE.	**A:01E0h**		**Group 24 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDIE [199:192]	[7:0]	rwc-_-i3	**Control Word Change Detect Interrupt Enable.** For z = 0 to 7, the combination of B.G24SRL[z] = 1 and B.G24SRIE[z] = 1, forces G.GSR5[24] = 1.
G25SRIE.	**A:01E4h**		**Group 25 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDIE [207:200]	[7:0]	rwc-_-i3	**Control Word Change Detect Interrupt Enable.** For z = 0 to 7, the combination of B.G25SRL[z] = 1 and B.G25SRIE[z] = 1, forces G.GSR5[25] = 1.
G26SRIE.	**A:01E8h**		**Group 26 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDIE [215:208]	[7:0]	rwc-_-i3	**Control Word Change Detect Interrupt Enable.** For z = 0 to 7, the combination of B.G26SRL[z] = 1 and B.G26SRIE[z] = 1, forces G.GSR5[26] = 1.
G27SRIE.	**A:01ECh**		**Group 27 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDIE [223:216]	[7:0]	rwc-_-i3	**Control Word Change Detect Interrupt Enable.** For z = 0 to 7, the combination of B.G27SRL[z] = 1 and B.G27SRIE[z] = 1, forces G.GSR5[27] = 1.
G28SRIE.	**A:01F0h**		**Group 28 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
CWCDIE [231:224]	[7:0]	rwc-_-i3	**Control Word Change Detect Interrupt Enable.** For z = 0 to 7, the combination of B.G28SRL[z] = 1 and B.G28SRIE[z] = 1, forces G.GSR5[28] = 1.
G29SRIE.	**A:01F4h**		**Group 29 Status Register Interrupt Enable.**
RSVD	[31:8]		**Reserved.**
CWCDIE [239:232]	[7:0]	rwc-_-i3	**Control Word Change Detect Interrupt Enable.** For z = 0 to 7, the combination of B.G29SRL[z] = 1 and B.G29SRIE[z] = 1, forces G.GSR5[29] = 1.
G30SRIE.	**A:01F8h**		**Group 30 Status Register Interrupt Enable.**
RSVD	[31:8]		**Reserved.**
CWCDIE [247:240]	[7:0]	rwc-_-i3	**Control Word Change Detect Interrupt Enable.** For z = 0 to 7, the combination of B.G30SRL[z] = 1 and B.G30SRIE[z] = 1, forces G.GSR5[30] = 1.
G31SRIE.	**A:01FCh**		**Group 31 Status Register Interrupt Enable.**
RSVD	[31:8]		**Reserved.**
CWCDIE [255:248]	[7:0]	rwc-_-i3	**Control Word Change Detect Interrupt Enable.** For z = 0 to 7, the combination of B.G31SRL[z] = 1 and B.G31SRIE[z] = 1, forces G.GSR5[31] = 1.

10.3.3 Jitter Buffer Registers (JB.)

10.3.3.1 Jitter Buffer Status Registers (JB.)

Table 10-11. Jitter Buffer Status Registers (JB.)

JB. Field Name	Addr (A:) Bit [x:y]	Type	Description
G0SRL.	**A:0200h**		**Group 0 Status Register Latch.**
RSVD	[31:8]		**Reserved.**
JBU [7:0]	[7:0]	rls-cor-i3	**Jitter Buffer Underrun** "1" = Jitter Buffer Underrun or "Start of Playout" according to the G.GCR.IPSE setting. Bits [7:0] indicate Bundles [7:0] respectively.
G1SRL.	**A:0204h**		**Group 1 Status Register Latch.**
RSVD	[31:8]		**Reserved.**
JBU [15:8]	[7:0]	rls-cor-i3	**Jitter Buffer Underrun** "1" = Jitter Buffer underrun. Bits [7:0] indicate Bundles [15:8] respectively. This can also optionally indicate the start of playout.
G2SRL.	**A:0208h**		**Group 2 Status Register Latch.**
RSVD	[31:8]		**Reserved.**
JBU [23:16]	[7:0]	rls-cor-i3	**Jitter Buffer Underrun** "1" = Jitter Buffer underrun. Bits [7:0] indicate Bundles [23:16] respectively. This can also optionally indicate the start of playout.
G3SRL.	**A:020Ch**		**Group 3 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBU [31:24]	[7:0]	rls-cor-i3	**Jitter Buffer Underrun** "1" = Jitter Buffer underrun. Bits [7:0] indicate Bundles [31:24] respectively. This can also optionally indicate the start of playout.
G4SRL.	**A:0210h**		**Group 4 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBU [39:32]	[7:0]	rls-cor-i3	**Jitter Buffer Underrun** "1" = Jitter Buffer underrun. Bits [7:0] indicate Bundles [39:32] respectively. This can also optionally indicate the start of playout.
G5SRL.	**A:0214h**		**Group 5 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBU [47:40]	[7:0]	rls-cor-i3	**Jitter Buffer Underrun** "1" = Jitter Buffer underrun. Bits [7:0] indicate Bundles [47:40] respectively. This can also optionally indicate the start of playout.
G6SRL.	**A:0218h**		**Group 6 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBU [55:48]	[7:0]	rls-cor-i3	**Jitter Buffer Underrun** "1" = Jitter Buffer underrun. Bits [7:0] indicate Bundles [55:48] respectively. This can also optionally indicate the start of playout.
G7SRL.	**A:021Ch**		**Group 7 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBU [63:56]	[7:0]	rls-cor-i3	**Jitter Buffer Underrun** "1" = Jitter Buffer underrun. Bits [7:0] indicate Bundles [63:56] respectively. This can also optionally indicate the start of playout.
G8SRL.	**A:0220h**		**Group 8 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBU [71:64]	[7:0]	rls-cor-i3	**Jitter Buffer Underrun** "1" = Jitter Buffer underrun. Bits [7:0] indicate Bundles [71:64] respectively. This can also optionally indicate the start of playout.
G9SRL.	**A:0224h**		**Group 9 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBU [79:72]	[7:0]	rls-cor-i3	**Jitter Buffer Underrun** "1" = Jitter Buffer underrun. Bits [7:0] indicate Bundles [79:72] respectively. This can also optionally indicate the start of playout.

JB. Field Name	Addr (A:) Bit [x:y]	Type	Description
G10SRL.	**A:0228h**		**Group 10 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBU [87:80]	[7:0]	rls-cor-i3	**Jitter Buffer Underrun** "1" = Jitter Buffer underrun. Bits [7:0] indicate Bundles [87:80] respectively. This can also optionally indicate the start of playout.
G11SRL.	**A:022Ch**		**Group 11 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBU [95:88]	[7:0]	rls-cor-i3	**Jitter Buffer Underrun** "1" = Jitter Buffer underrun. Bits [7:0] indicate Bundles [95:88] respectively. This can also optionally indicate the start of playout.
G12SRL.	**A:0230h**		**Group 12 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBU [103:96]	[7:0]	rls-cor-i3	**Jitter Buffer Underrun** "1" = Jitter Buffer underrun. Bits [7:0] indicate Bundles [103:96] respectively. This can also optionally indicate the start of playout.
G13SRL.	**A:0234h**		**Group 13 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBU [111:104]	[7:0]	rls-cor-i3	**Jitter Buffer Underrun** "1" = Jitter Buffer underrun. Bits [7:0] indicate Bundles [111:104] respectively. This can also optionally indicate the start of playout.
G14SRL.	**A:0238h**		**Group 14 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBU [119:112]	[7:0]	rls-cor-i3	**Jitter Buffer Underrun** "1" = Jitter Buffer underrun. Bits [7:0] indicate Bundles [119:112] respectively. This can also optionally indicate the start of playout.
G15SRL.	**A:023Ch**		**Group 15 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBU [127:120]	[7:0]	rls-cor-i3	**Jitter Buffer Underrun** "1" = Jitter Buffer underrun. Bits [7:0] indicate Bundles [127:120] respectively. This can also optionally indicate the start of playout.
G16SRL.	**A:0240h**		**Group 16 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBU [135:128]	[7:0]	rls-cor-i3	**Jitter Buffer Underrun** "1" = Jitter Buffer underrun. Bits [7:0] indicate Bundles [135:128] respectively. This can also optionally indicate the start of playout.
G17SRL.	**A:0244h**		**Group 17 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBU [143:136]	[7:0]	rls-cor-i3	**Jitter Buffer Underrun** "1" = Jitter Buffer underrun. Bits [7:0] indicate Bundles [143:136] respectively. This can also optionally indicate the start of playout.
G18SRL.	**A:0248h**		**Group 18 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBU [151:144]	[7:0]	rls-cor-i3	**Jitter Buffer Underrun** "1" = Jitter Buffer underrun. Bits [7:0] indicate Bundles [151:144] respectively. This can also optionally indicate the start of playout.
G19SRL.	**A:024Ch**		**Group 19 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBU [159:152]	[7:0]	rls-cor-i3	**Jitter Buffer Underrun** "1" = Jitter Buffer underrun. Bits [7:0] indicate Bundles [159:152] respectively. This can also optionally indicate the start of playout.
G20SRL.	**A:0250h**		**Group 20 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBU [167:160]	[7:0]	rls-cor-i3	**Jitter Buffer Underrun** "1" = Jitter Buffer underrun. Bits [7:0] indicate Bundles [167:160] respectively. This can also optionally indicate the start of playout.

JB. Field Name	Addr (A:) Bit [x:y]	Type	Description
G21SRL.	**A:0254h**		**Group 21 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBU [175:168]	[7:0]	rls-cor-i3	**Jitter Buffer Underrun** "1" = Jitter Buffer underrun. Bits [7:0] indicate Bundles [175:168] respectively. This can also optionally indicate the start of playout.
G22SRL.	**A:0258h**		**Group 22 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBU [183:176]	[7:0]	rls-cor-i3	**Jitter Buffer Underrun** "1" = Jitter Buffer underrun. Bits [7:0] indicate Bundles [183:176] respectively. This can also optionally indicate the start of playout.
G23SRL.	**A:025Ch**		**Group 23 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBU [191:184]	[7:0]	rls-cor-i3	**Jitter Buffer Underrun** "1" = Jitter Buffer underrun. Bits [7:0] indicate Bundles [191:184] respectively. This can also optionally indicate the start of playout.
G24SRL.	**A:0260h**		**Group 24 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBU [199:192]	[7:0]	rls-cor-i3	**Jitter Buffer Underrun** "1" = Jitter Buffer underrun. Bits [7:0] indicate Bundles [199:192] respectively. This can also optionally indicate the start of playout.
G25SRL.	**A:0264h**		**Group 25 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBU [207:200]	[7:0]	rls-cor-i3	**Jitter Buffer Underrun** "1" = Jitter Buffer underrun. Bits [7:0] indicate Bundles [207:200] respectively. This can also optionally indicate the start of playout.
G26SRL.	**A:0268h**		**Group 26 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBU [215:208]	[7:0]	rls-cor-i3	**Jitter Buffer Underrun** "1" = Jitter Buffer underrun. Bits [7:0] indicate Bundles [215:208] respectively. This can also optionally indicate the start of playout.
G27SRL.	**A:026Ch**		**Group 27 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBU [223:216]	[7:0]	rls-cor-i3	**Jitter Buffer Underrun** "1" = Jitter Buffer underrun. Bits [7:0] indicate Bundles [223:216] respectively. This can also optionally indicate the start of playout.
G28SRL.	**A:0270h**		**Group 28 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBU [231:224]	[7:0]	rls-cor-i3	**Jitter Buffer Underrun** "1" = Jitter Buffer underrun. Bits [7:0] indicate Bundles [231:224] respectively. This can also optionally indicate the start of playout.
G29SRL.	**A:0274h**		**Group 29 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBU [239:232]	[7:0]	rls-cor-i3	**Jitter Buffer Underrun** "1" = Jitter Buffer underrun. Bits [7:0] indicate Bundles [239:232] respectively. This can also optionally indicate the start of playout.
G30SRL.	**A:0278h**		**Group 30 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBU [247:240]	[7:0]	rls-cor-i3	**Jitter Buffer Underrun** "1" = Jitter Buffer underrun. Bits [7:0] indicate Bundles [247:240] respectively. This can also optionally indicate the start of playout.
G31SRL.	**A:027Ch**		**Group 31 Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBU [255:248]	[7:0]	rls-cor-i3	**Jitter Buffer Underrun** "1" = Jitter Buffer underrun. Bits [7:0] indicate Bundles [255:248] respectively. This can also optionally indicate the start of playout.

10.3.3.2 Jitter Buffer Status Register Interrupt Enables (JB.)

Table 10-12. Jitter Buffer Status Register Interrupt Enables (JB.)

JB. Field Name	Addr (A:) Bit [x:y]	Type	Description
G0SRIE.	**A:0280h**		**Group 0 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBUIE [7:0]	[7:0]	rwc-_-i3	**Jitter Buffer Underrun Interrupt Enable.** For z = 0 to 7, the combination of G0SRL[z] = 1 and G0SRIE[z] = 1, forces G.GSR6[0] = 1.
G1SRIE.	**A:0284h**		**Group 1 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBUIE [15:8]	[7:0]	rwc-_-i3	**Jitter Buffer Underrun Interrupt Enable.** For z = 0 to 7, the combination of JB.G1SRL[z] = 1 and JB.G1SRIE[z] = 1, forces G.GSR6[1] = 1.
G2SRIE.	**A:0288h**		**Group 2 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBUIE [23:16]	[7:0]	rwc-_-i3	**Jitter Buffer Underrun Interrupt Enable.** For z = 0 to 7, the combination of JB.G2SRL[z] = 1 and JB.G2SRIE[z] = 1, forces G.GSR6[2] = 1.
G3SRIE.	**A:028Ch**		**Group 3 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBUIE [31:24]	[7:0]	rwc-_-i3	**Jitter Buffer Underrun Interrupt Enable.** For z = 0 to 7, the combination of JB.G3SRL[z] = 1 and JB.G3SRIE[z] = 1, forces G.GSR6[3] = 1.
G4SRIE.	**A:0290h**		**Group 4 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBUIE [39:32]	[7:0]	rwc-_-i3	**Jitter Buffer Underrun Interrupt Enable.** For z = 0 to 7, the combination of JB.G4SRL[z] = 1 and JB.G4SRIE[z] = 1, forces G.GSR6[4] = 1.
G5SRIE.	**A:0294h**		**Group 5 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBUIE [47:40]	[7:0]	rwc-_-i3	**Jitter Buffer Underrun Interrupt Enable.** For z = 0 to 7, the combination of JB.G5SRL[z] = 1 and JB.G5SRIE[z] = 1, forces G.GSR6[5] = 1.
G6SRIE.	**A:0298h**		**Group 6 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBUIE [55:48]	[7:0]	rwc-_-i3	**Jitter Buffer Underrun Interrupt Enable.** For z = 0 to 7, the combination of JB.G6SRL[z] = 1 and JB.G6SRIE[z] = 1, forces G.GSR6[6] = 1.
G7SRIE.	**A:029Ch**		**Group 7 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBUIE [63:56]	[7:0]	rwc-_-i3	**Jitter Buffer Underrun Interrupt Enable.** For z = 0 to 7, the combination of JB.G7SRL[z] = 1 and JB.G7SRIE[z] = 1, forces G.GSR6[7] = 1.
G8SRIE.	**A:02A0h**		**Group 8 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBUIE [71:64]	[7:0]	rwc-_-i3	**Jitter Buffer Underrun Interrupt Enable.** For z = 0 to 7, the combination of JB.G8SRL[z] = 1 and JB.G8SRIE[z] = 1, forces G.GSR6[8] = 1.
G9SRIE.	**A:02A4h**		**Group 9 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBUIE [79:72]	[7:0]	rwc-_-i3	**Jitter Buffer Underrun Interrupt Enable.** For z = 0 to 7, the combination of JB.G9SRL[z] = 1 and JB.G9SRIE[z] = 1, forces G.GSR6[9] = 1.

JB. Field Name	Addr (A:) Bit [x:y]	Type	Description
G10SRIE.	**A:02A8h**		**Group 10 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBUIE [87:80]	[7:0]	rwc-_-i3	**Jitter Buffer Underrun Interrupt Enable.** For z = 0 to 7, the combination of JB.G10SRL[z] = 1 and JB.G10SRIE[z] = 1, forces G.GSR6[10] = 1.
G11SRIE.	**A:02ACh**		**Group 11 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBUIE [95:88]	[7:0]	rwc-_-i3	**Jitter Buffer Underrun Interrupt Enable.** For z = 0 to 7, the combination of JB.G11SRL[z] = 1 and JB.G11SRIE[z] = 1, forces G.GSR6[11] = 1.
G12SRIE.	**A:02B0h**		**Group 12 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBUIE [103:96]	[7:0]	rwc-_-i3	**Jitter Buffer Underrun Interrupt Enable.** For z = 0 to 7, the combination of JB.G12SRL[z] = 1 and JB.G12SRIE[z] = 1, forces G.GSR6[12] = 1.
G13SRIE.	**A:02B4h**		**Group 13 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBUIE [111:104]	[7:0]	rwc-_-i3	**Jitter Buffer Underrun Interrupt Enable.** For z = 0 to 7, the combination of JB.G13SRL[z] = 1 and JB.G13SRIE[z] = 1, forces G.GSR6[13] = 1.
G14SRIE.	**A:02B8h**		**Group 14 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBUIE [119:112]	[7:0]	rwc-_-i3	**Jitter Buffer Underrun Interrupt Enable.** For z = 0 to 7, the combination of JB.G14SRL[z] = 1 and JB.G14SRIE[z] = 1, forces G.GSR6[14] = 1.
G15SRIE.	**A:02BCh**		**Group 15 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBUIE [127:120]	[7:0]	rwc-_-i3	**Jitter Buffer Underrun Interrupt Enable.** For z = 0 to 7, the combination of JB.G15SRL[z] = 1 and JB.G15SRIE[z] = 1, forces G.GSR6[15] = 1.
G16SRIE.	**A:02C0h**		**Group 16 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBUIE [135:128]	[7:0]	rwc-_-i3	**Jitter Buffer Underrun Interrupt Enable.** For z = 0 to 7, the combination of JB.G16SRL[z] = 1 and JB.G16SRIE[z] = 1, forces G.GSR6[16] = 1.
G17SRIE.	**A:02C4h**		**Group 17 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBUIE [143:136]	[7:0]	rwc-_-i3	**Jitter Buffer Underrun Interrupt Enable.** For z = 0 to 7, the combination of JB.G17SRL[z] = 1 and JB.G17SRIE[z] = 1, forces G.GSR6[17] = 1.
G18SRIE.	**A:02C8h**		**Group 18 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBUIE [151:144]	[7:0]	rwc-_-i3	**Jitter Buffer Underrun Interrupt Enable.** For z = 0 to 7, the combination of JB.G18SRL[z] = 1 and JB.G18SRIE[z] = 1, forces G.GSR6[18] = 1.
G19SRIE.	**A:02CCh**		**Group 19 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBUIE [159:152]	[7:0]	rwc-_-i3	**Jitter Buffer Underrun Interrupt Enable.** For z = 0 to 7, the combination of JB.G19SRL[z] = 1 and JB.G19SRIE[z] = 1, forces G.GSR6[19] = 1.
G20SRIE.	**A:02D0h**		**Group 20 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBUIE [167:160]	[7:0]	rwc-_-i3	**Jitter Buffer Underrun Interrupt Enable.** For z = 0 to 7, the combination of JB.G20SRL[z] = 1 and JB.G20SRIE[z] = 1, forces G.GSR6[20] = 1.

JB. Field Name	Addr (A:) Bit [x:y]	Type	Description
G21SRIE.	**A:02D4h**		**Group 21 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBUIE [175:168]	[7:0]	rwc-_-i3	**Jitter Buffer Underrun Interrupt Enable.** For z = 0 to 7, the combination of JB.G21SRL[z] = 1 and JB.G21SRIE[z] = 1, forces G.GSR6[21] = 1.
G22SRIE.	**A:02D8h**		**Group 22 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBUIE [183:176]	[7:0]	rwc-_-i3	**Jitter Buffer Underrun Interrupt Enable.** For z = 0 to 7, the combination of JB.G22SRL[z] = 1 and JB.G22SRIE[z] = 1, forces G.GSR6[22] = 1.
G23SRIE.	**A:02DCh**		**Group 23 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBUIE [191:184]	[7:0]	rwc-_-i3	**Jitter Buffer Underrun Interrupt Enable.** For z = 0 to 7, the combination of JB.G23SRL[z] = 1 and JB.G23SRIE[z] = 1, forces G.GSR6[23] = 1.
G24SRIE.	**A:02E0h**		**Group 24 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBUIE [199:192]	[7:0]	rwc-_-i3	**Jitter Buffer Underrun Interrupt Enable.** For z = 0 to 7, the combination of JB.G24SRL[z] = 1 and JB.G24SRIE[z] = 1, forces G.GSR6[24] = 1.
G25SRIE.	**A:02E4h**		**Group 25 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBUIE [207:200]	[7:0]	rwc-_-i3	**Jitter Buffer Underrun Interrupt Enable.** For z = 0 to 7, the combination of JB.G25SRL[z] = 1 and JB.G25SRIE[z] = 1, forces G.GSR6[25] = 1.
G26SRIE.	**A:02E8h**		**Group 26 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBUIE [215:208]	[7:0]	rwc-_-i3	**Jitter Buffer Underrun Interrupt Enable.** For z = 0 to 7, the combination of JB.G26SRL[z] = 1 and JB.G26SRIE[z] = 1, forces G.GSR6[26] = 1.
G27SRIE.	**A:02ECh**		**Group 27 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBUIE [223:216]	[7:0]	rwc-_-i3	**Jitter Buffer Underrun Interrupt Enable.** For z = 0 to 7, the combination of JB.G27SRL[z] = 1 and JB.G27SRIE[z] = 1, forces G.GSR6[27] = 1.
G28SRIE.	**A:02F0h**		**Group 28 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBUIE [231:224]	[7:0]	rwc-_-i3	**Jitter Buffer Underrun Interrupt Enable.** For z = 0 to 7, the combination of JB.G28SRL[z] = 1 and JB.G28SRIE[z] = 1, forces G.GSR6[28] = 1.
G29SRIE.	**A:02F4h**		**Group 29 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBUIE [239:232]	[7:0]	rwc-_-i3	**Jitter Buffer Underrun Interrupt Enable.** For z = 0 to 7, the combination of JB.G29SRL[z] = 1 and JB.G29SRIE[z] = 1, forces G.GSR6[29] = 1.
G30SRIE.	**A:02F8h**		**Group 30 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBUIE [247:240]	[7:0]	rwc-_-i3	**Jitter Buffer Underrun Interrupt Enable.** For z = 0 to 7, the combination of JB.G30SRL[z] = 1 and JB.G30SRIE[z] = 1, forces G.GSR6[30] = 1.
G31SRIE.	**A:02FCh**		**Group 31 Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
JBUIE [255:248]	[7:0]	rwc-_-i3	**Jitter Buffer Underrun Interrupt Enable.** For z = 0 to 7, the combination of JB.G31SRL[z] = 1 and JB.G31SRIE[z] = 1, forces G.GSR6[31] = 1.

10.3.4 Packet Classifier Registers (PC.)

10.3.4.1 Packet Classifier Configuration Registers (PC.)

Table 10-13. Packet Classifier Configuration Registers (PC.)

PC. Field Name	Addr (A:) Bit [x:y]	Type	Description
CR1.	**A:0300h**		**Configuration Register 1. Default: 0x00.00.00.00**
DPDE	[31:30]	rwc-_-_	**Duplicate Packet Detect Enable.** 0 = Duplicate Packet Detect function is disabled 1 = Discard duplicate packet for Bundle used for clock recovery (BCDR4.PCRE) 2 = reserved 3 = Discard duplicate packets for all SAT/CES/HDLC Bundle types
RSVD	[29:26]		**Reserved.**
DPS10	[25]	rwc-_-_	**Discard Packet Switch 10.** Discard packets with too many MPLS Labels. 0 = Forward MPLS packets with > 3 MPLS labels to CPU (> 2 outer labels) 1 = Discard those packets.
DPS9	[24]	rwc-_-_	**Discard Packet Switch 9.** Discard packets with unknown Ethernet DA. 0 = Forward packets with unknown Ethernet DA to CPU (PC.CR17 – PC.CR19) 1 = Discard those packets.
DPS8	[23]	rwc-_-_	**Discard Packet Switch 8.** Discard packets with Ethernet type = CPU Destination. 0 = Forward packets with Ethernet type = PC.CR20.CDET to CPU (CPU Dest.) 1 = Discard those packets.
DPS7	[22]	rwc-_-_	**Discard Packet Switch 7.** Discard OAM packets. 0 = Forward MEF OAM, OAM BID and enabled In-band VCCV packets to CPU. 1= Discard those packets.
DPS6	[21]	rwc-_-_	**Discard Packet Switch 6.** Discard PW packets with Unknown PWID. 0 = Forward packets to CPU that have any PW header and includes a PWID that does not match any of the programmed BIDs or OAM BIDs 1 = Discard those packets.
DPS5	[20]	rwc-_-_	**Discard Packet Switch 5.** Discard UDP PW packets with wrong UDP Protocol. 0 = Forward UDP packets to CPU that have a recognized BID or OAM BID but have an unexpected UDP Protocol Type (Protocol ≠ PC.CR2.UPVC1 or PC.CR2.UPVC2, and PC.CR1.UPVCE = 1). The UDP Protocol Type may be in the UDP Destination or Source Port position (set using B.BCDR4.RXUBIDLS, PC.CR1.UBIDLS and PC.CR1.UBIDLCE). 1 = Discard those packets.
DPS4	[19]	rwc-_-_	**Discard Packet Switch 4.** Discard IP packets that do not have PW headers. 0 = Forward IP packets with unknown IP Protocol to CPU (not UDP or L2TPv3) 1 = Discard those packets.
DPS3	[18]	rwc-_-_	**Discard Packet Switch 3.** Discard ARP packets with a recognized IPv4 DA. 0 = Forward ARP packets with a recognized IPv4 DA to CPU (PC.CR6-PC.CR8) 1 = Discard those packets.
DPS2	[17]	rwc-_-_	**Discard Packet Switch 2.** Discard packets with unknown Ethernet Type. 0 = Forward packets to CPU that have an unknown Ethernet Type (not IPv4, IPv6, MPLS Unicast, MPLS Multi-cast, ARP, MEF = G.CR4.MOET, MEF OAM G.CR4.MET or CPU Destination Ethernet Type = G.CR20.CDET). 1 = Discard those packets.
DPS1	[16]	rwc-_-_	**Discard Packet Switch 1.** Discard IP packets with unknown IP DA. 0 = Forward IP packets with unknown IP DA to CPU (not PC.CR6 - PC.CR16) 1 = Discard those packets.
DPS0	[15]	rwc-_-_	**Discard Packet Switch 0.** Discard ARP packets with an unknown IPv4 DA. 0 = Forward ARP packets with unknown IP DA to CPU (not PC.CR6 – PC.CR8) 1 = Discard those packets

PC. Field Name	Addr (A:) Bit [x:y]	Type	Description
DICPE	[14]	rwc-_-_	**Discard IP Checksum Packet Error.** 0 = Do not discard packets due to IPv4 checksum errors. 1 = Discard those packets
DUCPE	[13]	rwc-_-_	**Discard UDP Checksum Packet Error.** 0 = Do not discard packets due to UDP checksum errors. 1 = Discard those packets For IPv4, a received zero UDP checksum (= checksum not calculated) is considered valid. For IPv6, a received zero UDP checksum is considered invalid and is discarded regardless of the DUCPE setting (see RFC1883). If the calculated UDP checksum = 0x0000 then the checksum is replaced with 0xFFFF.
DPLME	[12]	rwc-_-_	**Discard Packet Length Mismatch Error.** 0 = Do not discard packets due to a Control Word or IP Length field error. 1 = Discard packets with a received Control Word or IP Length field value that is greater than the number of bytes that are received (allows for Ethernet padding). This function does not test for an 802.3 or UDP Length field error.
DBTP	[11]	rwc-_-_	**Discard Broadcast TDM Packet.** 0 = Enable/accept the Broadcast DA as a valid Ethernet DA for PW packets. 1 = Discard PW packets that use the Broadcast Ethernet DA.
DBCP	[10]	rwc-_-_	**Discard Broadcast CPU Packet.** 0 = Enable Broadcast DA as a valid Ethernet DA for CPU (non-PW) packets. 1 = Discard CPU (non-PW) packets that use the Broadcast Ethernet DA.
RXPIVS	[9]	rwc-_-_	**RXP Packet IP Version Select.** (only valid when PC.CR1.RXPDSD = 1). 0 = Enable/accept the IPv4 protocol, discard all IPv6 packets. 1 = Enable/accept the IPv6 protocol, discard all IPv4 packets.
RXPDSD	[8]	rwc-_-_	**RXP Packet Dual Stack Disable.** 0 = Enable/accept both the IPv4 and IPv6 protocols. 1 = Enable/accept one IP version as selected by PC.CR1.RXPIVS.
UPVCE	[7]	rwc-_-_	**UDP Protocol Value Check Enable.** (only valid if PC.CR1.UBIDLS ≠ 3) 0 = Disable UDP Protocol Type test (accept any value) 1 = Discard packets with UDP Protocol Type ≠ PC.CR2.UPVC1 or UPVC2. The received UDP Protocol Type is tested in the UDP Port location (Source or Destination Port) not specified as the BID/PWID location (selected using PC.CR1.UBIDLS, PC.CR1.UBIDLCE, B.BCDR4.RXUBIDLS).
UBIDLCE	[6]	rwc-_-_	**UDP Bundle ID Location Check Enable.** (only valid if PC.CR1.UBIDLS ≠ 0) 0 = Auto-detect = Test for a BID/OAM BID match in both the UDP Source and Destination Port (a match in either port is accepted) 1 = Test for a BID/OAM BID match in only one UDP Port location as specified by B.BCDR4.RXUBIDLS
UBIDLS	[5:4]	rwc-_-_	**UDP Bundle ID Location Status Select.** 0 = Test for a 16-bit BID/OAM BID match in the UDP Source or Destination Port location specified by PC.CR1.UBIDLCE. 1 = Test for a 16-bit BID/OAM BID match in the UDP Destination Port location. 2 = Test for a 16-bit BID/OAM BID match in the UDP Source Port location. 3 = Test for a 32-bit BID/OAM BID match against the value of the combined Source and Destination Ports.
UICECS	[3:2]	rwc-_-_	**UDP IP Checksum Error Count Select.** 0 = PC.PCECR.UICPEC only counts IPv4 header checksum errors. 1 = PC.PCECR.UICPEC only counts UDP header checksum errors. 2 = PC.PCECR.UICPEC counts both IPv4 and UDP header checksum errors. 3 = Reserved.
RSVD	[1:0]		**Reserved.**

PC. Field Name	Addr (A:) Bit [x:y]	Type	Description
CR2.	**A:0304h**		**Configuration Register 2. Default: 0x00.00.00.00**
UPVC1	[31:16]	rwc-_-_	**UDP Protocol Value Constant 1** is used to compare against each received UDP Protocol Type field when UPVCE = 1 (1 of 2 recognized UDP Protocol Types).
UPVC2	[15:0]	rwc-_-_	**UDP Protocol Value Constant 2** is used to compare against each received UDP Protocol Type field when UPVCE = 1 (1 of 2 recognized UDP Protocol Types).
CR3.	**A:0308h**		**Configuration Register 3. Default: 81.00.81.00h**
VOTPID	[31:16]	rwc-_-_	**VLAN Outer Tag Protocol ID** specifies the Outer VLAN TPID value that is accepted when a packet header also includes an Inner VLAN Tag with TPID = PC.CR3.VITPID.
VITPID	[15:0]	rwc-_-_	**VLAN Inner Tag Protocol ID** specifies the Inner VLAN TPID value that is accepted when a packet includes 1 or 2 VLAN Tags. The common VITPID value that is used is 0x8100.
CR4.	**A:030Ch**		**Configuration Register 4. Default: 0x00.00.00.00**
MET	[31:16]	rwc-_-_	**MEF Ether Type** programs the Ethernet Type field value for the MEF-8 protocol. The IANA assigned Ethernet Type value for MEF is 0x88D8. Some systems may otherwise use Ethernet Type = 0x8847.
MOET	[15:0]	rwc-_-_	**MEF OAM Ether Type** programs the Ethernet Type field value for the MEF OAM protocol.
CR5.	**A:0310h**		**Configuration Register 5. Default: 0x00.00.00.00**
VOM	[31:16]	rwc-_-_	**VCCV OAM Mask** programs the mask of the 16 most significant bits of the Control Word that is used to identify In-band VCCV OAM packets (“1” = VOV bit is tested/enabled, one bit mask for each of the 16 VOV bits).
VOV	[15:0]	rwc-_-_	**VCCV OAM Value** programs the value of the 16 most significant bits of the Control Word that are used to identify an In-band VCCV OAM packet. The VOM bits can be used to ignore any of these 16 bits. To use the most common In-band VCCV identifier, program VOV = 0x1000 and VOM = 0xF000.
CR6.	**A:0314h**		**Configuration Register 6. Default: 0x00.00.00.00**
IV4A1	[31:0]	rwc-_-_	**IPv4 Address 1** programs the 32-bit value for the first IPv4 Destination Address.
CR7.	**A:0318h**		**Configuration Register 7. Default: 0x00.00.00.00**
IV4A2	[31:0]	rwc-_-_	**IPv4 Address 2** programs the 32-bit value for the 2nd IPv4 Destination Address.
CR8.	**A:031Ch**		**Configuration Register 8. Default: 0x00.00.00.00**
IV4A3	[31:0]	rwc-_-_	**IPv4 Address 3** programs the 32-bit value for the 3rd IPv4 Destination Address.
CR9.	**A:0320h**		**Configuration Register 9. Default: 0x00.00.00.00**
IV6A1A	[31:0]	rwc-_-_	**IPv6 Address 1 A-bits** programs bits 0 to 31 of the 1st 128-bit IPv6 Destination Address.
CR10.	**A:0324h**		**Configuration Register 10. Default: 0x00.00.00.00**
IV6A1B	[31:0]	rwc-_-_	**IPv6 Address 1 B-bits** programs bits 32 to 63 of the 1st 128-bit IPv6 Destination Address.
CR11.	**A:0328h**		**Configuration Register 11. Default: 0x00.00.00.00**
IV6A1C	[31:0]	rwc-_-_	**IPv6 Address 1 C-bits** programs bits 64 to 95 of the 1st 128-bit IPv6 Destination Address.
CR12.	**A:032Ch**		**Configuration Register 12. Default: 0x00.00.00.00**
IV6A1D	[31:0]	rwc-_-_	**IPv6 Address 1 D-bits** programs bits 96 to 127 of the 1st 128-bit IPv6 Destination Address.
CR13.	**A:0330h**		**Configuration Register 13. Default: 0x00.00.00.00**
IV6A2A	[31:0]	rwc-_-_	**IPv6 Address 2 A-bits** programs bits 0 to 31 of the 2nd 128-bit IPv6 Destination Address.

PC. Field Name	Addr (A:) Bit [x:y]	Type	Description
CR14.	**A:0334h**		**Configuration Register 14. Default: 0x00.00.00.00**
IV6A2B	[31:0]	rwc-_-_	**IPv6 Address 2 B-bits** programs bits 32 to 63 of the 2nd 128-bit IPv6 Destination Address.
CR15.	**A:0338h**		**Configuration Register 15. Default: 0x00.00.00.00**
IV6A2C	[31:0]	rwc-_-_	**IPv6 Address 2 C-bits** programs bits 64 to 95 of the 2nd 128-bit IPv6 Destination Address.
CR16.	**A:033Ch**		**Configuration Register 16. Default: 0x00.00.00.00**
IV6A2D	[31:0]	rwc-_-_	**IPv6 Address 2 D-bits** programs bits 96 to 127 of the 2nd 128-bit IPv6 Destination Address.
CR17.	**A:0340h**		**Configuration Register 17. Default: 0x00.00.00.00**
MA1B	[31:0]	rwc-_-_	**MAC Address 1 B-bits** programs bits 16 to 47 of the 1st 48-bit Ethernet Destination Address.
CR18.	**A:0344h**		**Configuration Register 18. Default: 0x00.00.00.00**
MA1A	[31:16]	rwc-_-_	**MAC Address 1 A-bits** programs bits 0 to 15 of the 1st 48-bit Ethernet Destination Address.
MA2A	[15:0]	rwc-_-_	**MAC Address 2 A-bits** programs bits 0 to 15 of the 2nd 48-bit Ethernet Destination Address.
CR19.	**A:0348h**		**Configuration Register 19. Default: 0x00.00.00.00**
MA2B	[31:0]	rwc-_-_	**MAC Address 2 B-bits** programs bits 16 to 47 of the 2nd 48-bit Ethernet Destination Address.
CR20.	**A:034Ch**		**Configuration Register 20. Default: 0x00.00.00.00**
CDET	[31:16]	rwc-_-_	**CPU Destination Ether Type** programs the Ethernet Type field value that is used to identify "CPU Destination Ethernet Type" packets.
UBIDM	[15:0]	rwc-_-_	**UDP Bundle ID Mask** selects which of the 16 LSB of a received UDP BID or OAM BID are tested for a match ("1" = test for match; "0" = ignore bit). For 32-bit UDP BIDs and 0AM BIDs the 16 MSB are always tested.
CR21.	**A:0350h**		**Configuration Register 21. Default: 0x00.00.00.03**
RSVD	[31:8]		**Reserved.**
PDCC	[7:4]	rwc-_-_	**Packet stream Defect Count Control** selects which Jitter Buffer fill defect conditions are counted by G.BDSR1.PDC (one bit per defect function; 1 = enable; any combination can be enabled): Too Early (bit 7), Too Late (bit 6), Overrun (bit 5), Underrun (bit 4). The Overrun level is programmed using B.BCDR5.MJBS.
JBECC	[3:0]	rwc-_-_	**Jitter Buffer Event Count Control** selects which Jitter Buffer fill defect conditions are counted by G.BDSR2.JBEC (one bit per defect function; 1 = enable; any combination can be enabled): Too Early (bit 7), Too Late (bit 6), Overrun (bit 5), Underrun (bit 4). The Overrun level is programmed using B.BCDR5.MJBS.

10.3.4.2 Packet Classifier Status Register Latches (PC.)

Table 10-14. Packet Classifier Register Latches (PC.)

PC. Field Name	Addr (A:) Bit [x:y]	Type	Description
SRL.	**A:0360h**		**Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
VMDSL	[7]	rls-crw-i3	**VLAN Mismatch Discard Status Latch** = "1" indicates 1 or more RXP packets have been received with an Outer VLAN TPID that matched PC.CR3.VOTPID, but the Inner VLAN TPID did not match PC.CR3.VITPID.

PC. Field Name	Addr (A:) Bit [x:y]	Type	Description
UPVCSL	[6]	rls-crw-i3	**UDP Port Value Check Status Latch** = "1" indicates 1 or more RXP UDP packets have been received with a BID or OAM BID match, but with a UDP Protocol Type mismatch (2 UDP Protocol values can be recognized: PC.CR2.UPVC1 and PC.CR2.UPVC2).
UBIDLCSL	[5]	rls-crw-i3	**UDP Bundle ID Location Check Status Latch** = "1" indicates 1 or more RXP UDP packets have been received with a BID match found, but not in the location specified by Bundle parameter B.BCDR4.RXUBIDLS .
BIDMSL	[4]	rls-crw-i3	**Bundle ID Mismatch Status Latch** = "1" indicates 1 or more RXP packets with any of the PW headers has been received with a PW-ID that did not match any of the active BIDs or OAM BIDs (indicates when an unknown PW-ID is received).
RXPFOSL	[3]	rls-crw-i3	**Reserved.**
RXPMESL	[2]	rls-crw-i3	**RXP Packet MPLS Error Status Latch** = "1" indicates 1 or more RXP MPLS packets have been received with a header that included more than 3 Labels.
ICPESL	[1]	rls-crw-i3	**IP Checksum Packet Error Status Latch** = "1" indicates 1 or more RXP IPv4 packets have been received with an IPv4 checksum error.
UCPESL	[0]	rls-crw-i3	**UDP Checksum Packet Error Status Latch** = "1" indicates 1 or more RXP UDP packets have been received with a UDP checksum error (IPv4 or IPv6).

10.3.4.3 Packet Classifier Status Register Interrupt Enables (PC.)

Table 10-15. Packet Classifier Status Register Interrupt Enables (PC.)

PC. Field Name	Addr (A:) Bit [x:y]	Type	Description
SRIE.	**A:0368h**		**Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
VMDIE	[7]	rwc-_-i3	**VLAN Mismatch Discard Interrupt Enable** = "1" enables an interrupt (INT_N) and forces G.GSR1.PCS = 1 when PC.SRL.VMDSL = "1".
UPVCIE	[6]	rwc-_-i3	**UDP Port Value Check Interrupt Enable** = "1" enables an interrupt (INT_N) and forces G.GSR1.PCS = 1 when PC.SRL.UPVCSL = "1".
UBIDLCIE	[5]	rwc-_-i3	**UDP Bundle ID Location Check Interrupt Enable** = "1" enables an interrupt (INT_N) and forces G.GSR1.PCS = 1 when PC.SRL.UBIDLCSL = "1".
RXPMIE	[4]	rwc-_-i3	**RXP Packet Mismatch Interrupt Enable** = "1" enables an interrupt (INT_N) and forces G.GSR1.PCS = 1 when PC.SRL.BIDMSL = "1".
RXPFOSIE	[3]	rwc-_-i3	**Reserved.**
RXPMEIE	[2]	rwc-_-i3	**RXP Packet MPLS Error Interrupt Enable** = "1" enables an interrupt (INT_N) and forces G.GSR1.PCS = 1 when PC.SRL.RXPMESL = "1".
ICPEIE	[1]	rwc-_-i3	**IP Checksum Packet Error Interrupt Enable** = "1" enables an interrupt (INT_N) and forces G.GSR1.PCS = 1 when PC.SRL.ICPESL = "1".
UCPEIE	[0]	rwc-_-i3	**UDP Checksum Packet Error Interrupt Enable** = "1" enables an interrupt (INT_N) and forces G.GSR1.PCS = 1 when PC.SRL.IUCPESL = "1".

10.3.4.4 Packet Classifier Counter Registers (PC.)

Table 10-16. Packet Classifier Counter Registers (PC.)

PC. Field Name	Addr (A:) Bit [x:y]	Type	Description
CPCR.	**A:0370h**		**Classified Packet Counter Register. Default: 0x00.00.00.00**
CPC	[31:0]	rcs-cor-nc	**Classified Packet Count** indicates # of "good" RXP packets that have been forwarded to a CES/SAT Engine, Clock Recovery Engine or the CPU Queue.
PCECR.	**A:0374h**		**IP/UDP Packet Checksum Error Counter Register. Default: 0x00.00.00.00**
RSVD	[31:16]		**Reserved.**
UICPEC	[15:0]	rcs-cor-nc	**UDP IP Checksum Packet Error Count** indicates the # of received IPv4 or UDP checksum errors (error type selected using PC.PCECR.UICPEC).
SPCR.	**A:0378h**		**Stray Packet Count Register. Default: 0x00.00.00.00**
SPC	[31:0]		**Stray Packet Count** indicates the # of received packets that include a PW Header but do not match any of the configured Bundle IDs or OAM Bundle IDs.
FOCR.	**A:037Ch**		**FIFO Overflow Counter Register. Default: 0x00.00.00.00**
RSVD	[31:16]		**Reserved.**
FOC	[15:0]		**Reserved.**

10.3.5 External Memory Interface Registers (EMI.)

10.3.5.1 External Memory Interface Configuration Registers (EMI.)

Table 10-17. External Memory Interface Configuration Registers (EMI.)

EMI. Field Name	Addr (A:) Bit [x:y]	Type	Description
BMCR1.	**A:0380h**		**Buffer Manager Configuration Register 1. Default: 0x00.00.00.00**
TXPSO	[31:16]	rwc-_-_	**TXP Packet Space Offset** specifies the starting address in the external SDRAM for storing TXP TDM payload (the location where Bundle 0 payload is stored). TXP TDM payload starting address = 2048 bytes * TXPSO
TXHSO	[15:0]	rwc-_-_	**TXP Header Space Offset** specifies the starting address in the external SDRAM for storing TXP TDM Headers (the location where the Bundle 0 Header is stored). TXP TDM Header starting address = 2048 bytes * TXHSO
BMCR2.	**A:0384h**		**Buffer Manager Configuration Register 2. Default: 0x00.00.00.00**
RSVD	[31:16]		**Reserved.**
JBSO	[15:0]	rwc-_-_	**Jitter Buffer Space Offset** specifies the starting address in the external SDRAM for storing RXP TDM packets (the location where Bundle 0 packets are stored). RXP TDM packet starting address = 2048 bytes * JBSO
BMCR3.	**A:0388h**		**Buffer Manager Configuration Register 3. Default: 0x00.00.00.00**
PTSO	[31:16]	rwc-_-_	**Packet Transmit Space Offset** specifies the starting address in the external SDRAM for storing TXP CPU packets. TXP CPU packet starting address = 2048 bytes * PTSO
PRSO	[15:0]	rwc-_-_	**Packet Receive Space Offset** specifies the starting address in the external SDRAM for storing RXP CPU packets. RXP CPU packet starting address = 2048 bytes * PRSO
DCR1.	**A:0390h**		**DDR SDRAM Configuration Register 1. Default: 0x00.00.00.00**
RSVD	[31:1]		**Reserved.**
DIR	[0]	rwc-_-_	**DDR SDRAM Initialization Reset** re-initializes the EMI.DCR3.DBMR and EMI.DCR3.DEMR register bits when DIR transitions from zero to one.

EMI. Field Name	Addr (A:) Bit [x:y]	Type	Description
DCR2.	**A:0394h**		**DDR SDRAM Configuration Register 2. Default: 00.02.90.10h**
RSVD	[31:19]		**Reserved.**
TRFC	[18:14]	rwc-_-_	**Time Refresh From Clock** selects the time the S132 allows for each SDRAM refresh cycle to complete. This can be set to any value between the minimum t_{RFC} allowed by the SDRAM and the max value (0x1F = 248 ns; 0 and 1 are invalid). Refresh Time = TRFC * 1/$freq_{DDRCLK}$ = TRFC * 8 ns
DCL	[13:11]	rwc-_-_	**DDR SDRAM CAS Latency** specifies the SDRAM CAS Latency. 2 = CAS Latency 2 (all other values are reserved).
DCW	[10:9]	rwc-_-_	**DDR SDRAM Column Width** specifies the external SDRAM Column Width. 0 = 2048 columns per row 1 = 1024 columns per row 2 = 512 columns per row 3 = reserved
DMS	[8:7]	rwc-_-_	**DDR SDRAM Memory Size** specifies the total external SDRAM memory size. 0 = 1 Gbit (two 32 Meg x 16-bit SDRAM devices) 1 = 512 Mbit (one 32 Meg x 16-bit SDRAM device) 2 = 256 Mbit (one 16 Meg x 16-bit SDRAM device) 3 = 128 Mbit (one 8 Meg x 16-bit SDRAM device)
DDW	[6:5]	rwc-_-_	**Reserved.**
DRRS	[4:0]	rwc-_-_	**DDR SDRAM Refresh Rate Select** = time period between each SDRAM Refresh (SDRAM t_{REFI} parameter) = DRRS * 512ns
DCR3.	**A:0398h**		**DDR SDRAM Configuration Register 3. Default: 00.22.40.00h**
DBMR	[31:16]	rwc-_-_	**Reserved.**
DEMR	[15:0]	rwc-_-_	**Reserved.**

10.3.5.2 External Memory Interface Status Registers (EMI.)

Table 10-18. External Memory Interface Status Registers (EMI.)

EMI. Field Name	Addr (A:) Bit [x:y]	Type	Description
BMSRL.	**A:03A0h**		**Buffer Manager Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:9]		**Reserved.**
CERCSL	[8]	rls-crw-i3	**CPU to Ethernet Read Check Status Latch** = “1” indicates one or more SDRAM Read operations were invalid due to EMI.BMCR3.PTSO. The TXP CPU Queue overlaps with another SDRAM queue due to an invalid EMI Start Address setting. The combination of CERCSL = 1 and CERCIE = 1 forces G.GSR1.EMIS = 1.
CEWCSL	[7]	rls-crw-i3	**CPU to Ethernet Write Check Status Latch** = “1” indicates one or more SDRAM Write operations were invalid due to EMI.BMCR3.PTSO. The TXP CPU Queue overlaps with another SDRAM queue due to an invalid EMI Start Address setting. The combination of CEWCSL = 1 and CEWCIE = 1 forces G.GSR1.EMIS = 1.
ECRCSL	[6]	rls-crw-i3	**Ethernet to CPU Read Check Status Latch** = “1” indicates one or more SDRAM Read operations were invalid due to EMI.BMCR3.PRSO. The RXP CPU Queue overlaps with another SDRAM queue due to an invalid EMI Start Address setting. The combination of ECRCSL = 1 and ECRCIE = 1 forces G.GSR1.EMIS = 1.
ECWCSL	[5]	rls-crw-i3	**Ethernet to CPU Write Check Status Latch** = “1” indicates one or more SDRAM Write operations were invalid due to EMI.BMCR3.PRSO. The RXP CPU Queue overlaps with another SDRAM queue due to an invalid EMI Start Address setting. The combination of ECWCSL = 1 and ECWCIE = 1 forces G.GSR1.EMIS = 1.

EMI. Field Name	Addr (A:) Bit [x:y]	Type	Description
ETRCSL	[4]	rls-crw-i3	**Ethernet to TDM Read Check Status Latch** = "1" indicates one or more SDRAM Read operations were invalid due to EMI.BMCR2.JBSO. The Jitter Buffer Queues overlap with another SDRAM queue due to an invalid EMI Start Address setting. The combination of ETRCSL = 1 and ETRCIE = 1 forces G.GSR1.EMIS = 1.
ETWCSL	[3]	rls-crw-i3	**Ethernet to TDM Write Check Status Latch** = "1" indicates 1 or more SDRAM Write operations were invalid due to EMI.BMCR2.JBSO. The Jitter Buffer Queues overlap with another SDRAM queue due to an invalid EMI Start Address setting. The combination of ETWCSL = 1 and ETWCIE = 1 forces G.GSR1.EMIS = 1.
TXPSRCSL	[2]	rls-crw-i3	**TXP Packet Space Read Check Status Latch** = "1" indicates 1 or more SDRAM Read operations were invalid due to EMI.BMCR1.TXPSO. The TXP TDM Packet Queues overlap with another queue due to an invalid EMI Start Address. The combination of TXPSRCSL = 1 and TXPSRCIE = 1 forces G.GSR1.EMIS = 1.
TXPSWCSL	[1]	rls-crw-i3	**TXP Packet Space Write Check Status Latch** = "1" indicates 1 or more SDRAM Write operations were invalid due to EMI.BMCR1.TXPSO. The TXP TDM Packet Queues overlap with another queue due to an invalid EMI Start Address. The combination of TXPSWCSL = 1 and TXPSWCIE = 1 forces G.GSR1.EMIS = 1.
TXHSRCSL	[0]	rls-crw-i3	**TXP Header Space Read Check Status Latch** = "1" indicates 1 or more SDRAM Read operations were invalid due to EMI.BMCR1.TXHSO. The TXP TDM Header space overlaps with another queue due to an invalid EMI Start Address. The combination of TXHSRCSL = 1 and TXHSRCIE = 1 forces G.GSR1.EMIS = 1.

10.3.5.3 External Memory Interface Status Register Interrupt Enables (EMI.)

Table 10-19. External Memory Interface Status Register Interrupt Enables (EMI.)

EMI. Field Name	Addr (A:) Bit [x:y]	Type	Description
BMSRIE.	**A:03B0h**		**Buffer Manager Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:9]		**Reserved.**
CERCIE	[8]	rwc-_-i3	**CPU to Ethernet Read Check Interrupt Enable.** (see EMI.BMSRL.CERCSL)
CEWCIE	[7]	rwc-_-i3	**CPU to Ethernet Write Check Interrupt Enable.** (see EMI.BMSRL.CEWCSL)
ECRCIE	[6]	rwc-_-i3	**Ethernet to CPU Read Check Interrupt Enable.** (see EMI.BMSRL.ECRCSL)
ECWCIE	[5]	rwc-_-i3	**Ethernet to CPU Write Check Interrupt Enable.** (see EMI.BMSRL.ECWCSL)
ETRCIE	[4]	rwc-_-i3	**Ethernet to TDM Read Check Interrupt Enable.** (see EMI.BMSRL.ETRCSL)
ETWCIE	[3]	rwc-_-i3	**Ethernet to TDM Write Check Interrupt Enable.** (see EMI.BMSRL.ETWCSL)
TXPSRCIE	[2]	rwc-_-i3	**TXP Packet Space Read Check Interrupt Enable.** (see EMI.BMSRL.TXPSRSL)
TXPSWCIE	[1]	rwc-_-i3	**TXP Packet Space Write Check Interrupt Enable.** (see EMI.BMSRL.TXPSWCSL)
TXHSRCIE	[0]	rwc-_-i3	**TXP Header Space Read Check Interrupt Enable.** (see EMI.BMSRL.TXHSRCSL)
TSRL	**A:03B4h**		**Test Status Register Latched**
RSVD	[31:11]		**Reserved.**
EMARERRSL	[10]	rls-crw-_	**Reserved.**
EMAWERRSL	[9]	rls-crw-_	**Reserved.**
RPIRERRSL	[8]	rls-crw-_	**Reserved.**

EMI. Field Name	Addr (A:) Bit [x:y]	Type	Description
RPI1WERRSL	[7]	rls-crw-_	**Reserved.**
RPI2WERRSL	[6]	rls-crw-_	**Reserved.**
TDI1ERRSL	[5]	rls-crw-_	**Reserved.**
TDI2ERRSL	[4]	rls-crw-_	**Reserved.**
TEI1ERRSL	[3]	rls-crw-_	**Reserved.**
TEI2ERRSL	[2]	rls-crw-_	**Reserved.**
TPI1ERRSL	[1]	rls-crw-_	**Reserved.**
TPI2ERRSL	[0]	rls-crw-_	**Reserved.**

10.3.5.4 External Memory DLL/PLL Test Registers (EMI.)

Table 10-20. External Memory DLL/PLL Test Registers (EMI.)

EMI. Field Name	Addr (A:) Bit [x:y]	Type	Description
TCR1.	**A:03B8h**		**Test Configuration Register 1. Default: 0x00.00.00.00**
PTR	[31:16]	rwc-_-_	**Reserved.**
DTR	[15:0]	rwc-_-_	**Reserved.**
TCR2.	**A:03BCh**		**Test Configuration Register 2. Default: 0x00.00.00.00**
RSVD	[31:9]		**Reserved.**
PPCR	[8:7]	rwc-_-_	**Reserved.**
DPCR	[6:0]	rwc-_-_	**Reserved.**

10.3.6 External Memory Access Registers (EMA.)

10.3.6.1 Write Registers (EMA.)

Table 10-21. Write Registers (EMA.)

EMA. Field Name	Addr (A:) Bit [x:y]	Type	Description
WCR.	**A:03C0h**		**Write Control Register. Default: 0x00.00.00.00**
RSVD	[31:17]		**Reserved.**
TLBE	[16:13]	rwc-_-_	**Transfer Last Byte Enable** is used to indicate to the S132 which bytes are valid in the last double-word stored in the TXP CPU FIFO (each bit enables 1 of 4 bytes). This function is used when TPCWC = 6 and a complete TXP CPU packet has already been stored at EMA.WDR.EMWD. TLBE = 0x1 = “1 byte in the least significant byte position”. TLBE = 0xF = “4 bytes”.

EMA. Field Name	Addr (A:) Bit [x:y]	Type	Description
TPCWC	[12:10]	rwc-_-_	**TXP Packet and Configuration Write Control** is used to control the transfer of packets from the internal TXP CPU FIFO to the TXP CPU SDRAM Queue. 0 = idle - no operations 2 = Flush/reset TXP CPU Queue (external SDRAM queue) 3 = Flush/reset TXP CPU FIFO (internal S132 FIFO Buffer) 6 = Transfer packet from TXP CPU FIFO to SDRAM TXP CPU Queue all other values are reserved
TL	[9:0]	rwc-_-_	**Transfer Length** is used to indicate how many double words are included in the packet that is to be transferred from the TXP CPU FIFO to the TXP CPU Queue. This function is used when TPCWC = 6 and a complete TXP CPU packet has already been stored at EMA.WDR.EMWD. The maximum TL value is 512. TL = 0 means "no data". To transfer a single byte, TL = 1, and TLBE = 0x1.
WAR.	**A:03C4h**		**WAR. Default: 0x00.00.00.00**
RSVD	[31:0]		**Reserved.**
WDR.	**A:03C8h**		**Write Data Register. Default: 0x00.00.00.00**
EMWD	[31:0]	woc-_-_	**External Memory Write Data.** Data written to EMWD is stored in the internal TXP CPU FIFO in preparation for transfer to the SDRAM TXP CPU Queue. Each EMWD write, auto increments the FIFO address (to be ready for the next write).
WSR1.	**A:03CCh**		**Write Status Register 1. Default: 0x00.00.00.00**
RSVD	[31:17]		**Reserved.**
WQNFS	[16]	ros-_-i3	**Write Queue Not Full Status** = "1" indicates the TXP CPU Queue is not full. Up to 512 packets can be stored in the SDRAM TXP CPU Queue (see WSR2.WQL)
RSVD	[15:7]		**Reserved.**
WFES	[6]	ros-_-i3	**Write FIFO Empty Status** = "1" = TXP CPU FIFO is empty, new data can be stored. The last packet was transferred or flushed, there is no data in the FIFO.
RSVD	[5:0]		**Reserved.**
WSR2.	**A:03D0h**		**Write Status Register 2. Default: 0x00.00.00.00**
RSVD	[31]		**Reserved.**
WQL	[30:21]	ros-_-_	**Write Queue Level** = # packets currently stored in SDRAM TXP CPU Queue.
RSVD	[20:0]		**Reserved.**
WSRL1.	**A:03D4h**		**Write Status Register Latch 1. Default: 0x00.00.00.00**
RSVD	[31:19]		**Reserved.**
WPNRSL	[18]	rls-crw-i3	**Write Preempted by New Request Status Latch** = "1" indicates one or more packet transfers from the TXP CPU FIFO to the TXP CPU Queue were preempted/corrupted by an invalid EMA.WDR.EMWD write (wait until WFES = 1 before beginning the write operation for each new packet). The combination of WPNRSL = 1 and WPNRIE = 1 forces G.GSR1.EMAWS = 1.
RSVD	[17]		**Reserved.**
WQNFSL	[16]	rls-crw-i3	**Write Queue Not Full Status Latch** is a latched "1" when EMA.WSR1.WQNFS transitions from 0 to 1. The combination of WQNFSL = 1 and WQNFIE = 1 forces G.GSR1.EMAWS = 1.
RSVD	[15:8]		**Reserved.**
WFOSL	[7]	rls-crw-i3	**Write FIFO Overflow Status Latch** = "1" = internal TXP CPU FIFO overflow (i.e. more than 512 EMA.WDR.EMWD writes before an EMA.WCR.TPCWC transfer). The combination of WFOSL = 1 and WFOIE = 1 forces G.GSR1.EMAWS = 1.
WFESL	[6]	rls-crw-i3	**Write FIFO Empty Status Latch** is a latched "1" when EMA.WSR1.WFES transitions from 0 to 1. The combination of WFESL = 1 and WFEIE = 1 forces G.GSR1.EMAWS = 1.
WTOSL	[5]	rls-crw-i3	**Reserved.**

EMA. Field Name	Addr (A:) Bit [x:y]	Type	Description
RSVD	[4:0]		**Reserved.**
WSRIE1.	**A:03D8h**		**Write Status Register Interrupt Enable 1. Default: 0x00.00.00.00**
RSVD	[31:19]		**Reserved.**
WPNRIE	[18]	rwc-_-i3	**Write Preempt New Request Interrupt Enable** (see EMA.WSRL1.WPNRSL)
RSVD	[17]		**Reserved.**
WQNFIE	[16]	rwc-_-i3	**Write Queue Not Full Interrupt Enable.** (see EMA.WSRL1.WQNFSL)
RSVD	[15:8]		**Reserved.**
WFOIE	[7]	rwc-_-i3	**Write FIFO Overflow Interrupt Enable.** (see EMA.WSRL1.WFOSL)
WFEIE	[6]	rwc-_-i3	**Write FIFO Empty Interrupt Enable.** (see EMA.WSRL1.WFESL)
WTOIE	[5]	rwc-_-i3	**Reserved.**
RSVD	[4:0]		**Reserved.**

10.3.6.2 Read Registers (EMA.)

Table 10-22. Read Registers (EMA.)

EMA. Field Name	Addr (A:) Bit [x:y]	Type	Description
RCR.	**A:03E0h**		**Read Control Register. Default: 0x00.00.00.00**
RSVD	[31:13]		**Reserved.**
RPCRC	[12:10]	rwc-_-_	**Receive Packet and Configuration Read Control** is used to control the transfer of packets from the RXP CPU SDRAM Queue to the internal RXP CPU FIFO. 0 = idle - no operations 2 = Flush/reset RXP CPU Queue (external SDRAM queue) 3 = Flush/reset RXP CPU FIFO (internal S132 FIFO Buffer) 6 = Transfer packet from SDRAM RXP CPU Queue to RXP CPU FIFO all other values are reserved
TL	[9:0]	rwc-_-_	**Transfer Length** indicates how many double words are to be transferred from the SDRAM RXP CPU Queue to the RXP CPU FIFO. This function is used when RPCRC = 6. The maximum TL value is 512. TL = 1 means “1 double word of data”. The CPU must read the first double word of each RXP CPU packet to learn how many bytes are included in each RXP CPU packet.
RAR.	**A:03E4h**		**Read Address Register. Default: 0x00.00.00.00**
RSVD	[31:0]		**Reserved.**
RDR.	**A:03E8h**		**Read Data Register. Default: 0x00.00.00.00**
EMRD	[31:0]	ros-_-_	**External Memory Read Data.** Each read from EMRD provides a double word of RXP CPU packet data from the internal RXP CPU FIFO and auto increments the FIFO address (to be ready for the next read). The data for each RXP CPU packet must first be transferred from the SDRAM RXP CPU Queue (using EMA.RCR.RPCRC) before the data is available at the RXP CPU FIFO.
RSR1.	**A:03ECh**		**Read Status Register 1. Default: 0x00.00.00.00**
RSVD	[31:17]		**Reserved.**
RQNES	[16]	ros-_-i3	**Read Queue Not Empty Status** = “1” indicates one or more packets are waiting in the SDRAM RXP CPU Queue (1 to 512 packets waiting; see RSR2.RQL).
RSVD	[15:7]		**Reserved.**
RFRS	[6]	ros-_-i3	**Read FIFO Ready Status** = “1” indicates the block of data for the RXP CPU packet (as requested by EMA.RCR.TL) has been transferred from the RXP CPU Queue to the RXP CPU FIFO and can now be read at EMA.RDR.EMRD.

EMA. Field Name	Addr (A:) Bit [x:y]	Type	Description
RSVD	[5:0]		**Reserved.**
RSR2.	**A:03F0h**		**Read Status Register 2. Default: 0x00.00.00.00**
RSVD	[31]		**Reserved.**
RQL	[30:21]	ros-_-_	**Read Queue Level** = # packets currently stored in SDRAM RXP CPU Queue.
RSVD	[20:19]		**Reserved.**
RQRP	[18:10]	ros-_-_	**Read Queue Read Pointer** indicates which SDRAM RXP CPU Queue packet is to be transferred next to the internal RXP CPU FIFO (0 to 512).
RFL	[9:0]	ros-_-_	**Read FIFO Level** = # double words currently in the RXP CPU FIFO.
RSRL1.	**A:03F4h**		**Read Status Register Latch 1. Default: 0x00.00.00.00**
RSVD	[31:19]		**Reserved.**
RPNRSL	[18]	rls-crw-i3	**Read Preempted by New Request Status Latch** = "1" indicates one or more data transfers from the RXP CPU Queue to the RXP CPU FIFO were preempted/corrupted by an invalid EMA.RCR.RPCRC transfer (wait until RFRS = 1 before beginning a new RPCRC = 6 transfer operation). The combination of RPNRSL = 1 and RPNRIE = 1 forces G.GSR1.EMARS = 1.
RQOSL	[17]	rls-crw-i3	**Read Queue Overflow Status Latch** = "1" = SDRAM RXP CPU Queue overflow. One or more packets were discarded from the tail of the queue. The combination of RQOSL = 1 and RQOIE = 1 forces G.GSR1.EMARS = 1.
RQNESL	[16]	rls-crw-i3	**Read Queue Not Empty Status Latch** = "1" indicates one or more packets are in the RXP CPU Queue waiting to be transferred to the RXP CPU FIFO. The combination of RQNESL = 1 and RQNEIE = 1 forces G.GSR1.EMARS = 1.
RSVD	[15:8]		**Reserved.**
RFUSL	[7]	rls-crw-i3	**Read FIFO Underflow Status Latch** = "1" indicates the RXP CPU FIFO was read (EMRD) when no data was present in the FIFO (read when empty). The combination of RFUSL = 1 and RFUIE = 1 forces G.GSR1.EMARS = 1.
RFRSL	[6]	rls-crw-i3	**Read FIFO Ready Status Latch** = "1" indicates the last request to transfer data from the SDRAM RXP CPU Queue to the RXP CPU FIFO (RPCRC = 6) is done. The data is can be read at EMRD. The combination of RFRSL = 1 and RFRIE = 1 forces G.GSR1.EMARS = 1.
RTOSL	[5]	rls-crw-i3	**Reserved.**
RSVD	[4:0]		**Reserved.**
RSRIE1.	**A:03F8h**		**Read Status Register Interrupt Enable 1. Default: 0x00.00.00.00**
RSVD	[31:19]		**Reserved.**
RPNRIE	[18]	rwc-_-i3	**Read Preempt by New Request Interrupt Enable.** (see EMA.RSRL1.RPNRSL)
RQOIE	[17]	rwc-_-i3	**Read Queue Overflow Interrupt Enable.** (see EMA.RSRL1.RQOSL)
RQNEIE	[16]	rwc-_-i3	**Read Queue Not Empty Interrupt Enable.** (see EMA.RSRL1.RQNESL)
RSVD	[15:8]		**Reserved.**
RFUIE	[7]	rwc-_-i3	**Read FIFO Underflow Interrupt Enable.** (see EMA.RSRL1.RFUSL)
RFRIE	[6]	rwc-_-i3	**Read FIFO Ready Interrupt Enable.** (see EMA.RSRL1.RFRSL)
RTOIE	[5]	rwc-_-i3	**Reserved.**
RSVD	[4:0]		**Reserved.**

10.3.7 Encap BERT Registers (EB.)

Table 10-23. Encap BERT Registers (EB.)

EB. Field Name	Addr (A:) Bit [x:y]	Type	Description
BCR.	**A:0400h**		**BERT Control Register. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
PMUM	[7]	rwc-_-_	**Reserved.**
LPMU	[6]	rwc-_-_	**Local Performance Monitoring Update.** A 0 to 1 transition of this bit updates the TXP TDM BERT Performance Monitoring registers (EB.RBECR.BEC and EB.RBECR.BC) with the latest counts and then resets the counters.
RNPL	[5]	rwc-_-_	**Receive New Pattern Load.** A 0 to 1 transition of this bit loads the test pattern into the receive TXP TDM BERT Monitor (QRSS, PTS, PLF[4:0], PTF[4:0], and BSP[31:0]). This forces the TXP TDM BERT Monitor to resynchronize to the incoming data pattern. Note: QRSS, PTS, PLF[4:0}, PTF[4:0], and BSP[31:0] must not change until 4 SYSCLK clock cycles after RNPL transitions from 0 to 1.
RPIC	[4]	rwc-_-_	**Receive Pattern Inversion Control.** (TXP TDM BERT Monitor) 0 = test normal (unaltered) incoming data pattern 1 = invert and then test the incoming data pattern
MPR	[3]	rwc-_-_	**Manual Pattern Resynchronization.** A 0 to 1 transition of this bit forces the TXP TDM BERT Monitor to resynchronize to the incoming pattern.
APRD	[2]	rwc-_-_	**Automatic Pattern Resynchronization Disable.** For APRD = 0, the TXP TDM BERT Monitor is forced to resynchronize to the incoming pattern when 6 received bits, within a 64-bit window, do not match the expected pattern. For APRD = 1, after the TXP TDM BERT Monitor finds synchronization lock, it does not attempt to resynchronize regardless of how many bit errors are detected.
TNPL	[1]	rwc-_-_	**Transmit New Pattern Load.** A 0 to 1 transition of this bit loads the test pattern into the transmit TXP Packet BERT Generator (QRSS, PTS, PLF[4:0], PTF[4:0], and BSP[31:0]). Note: QRSS, PTS, PLF[4:0}, PTF[4:0], and BSP[31:0] must not change until 4 SYSCLK clock cycles after TNPL transitions from 0 to 1.
TPIC	[0]	rwc-_-_	**Transmit Pattern Inversion Control.** (TXP Packet BERT Generator) 0 = transmit normal (unaltered) outgoing data pattern 1 = transmit inverted outgoing data pattern
BPCR.	**A:0404h**		**BERT Pattern Configuration Register. Default: 0x00.00.00.00**
RSVD	[31:13]		**Reserved.**
PTF	[12:8]	rwc-_-_	**Test Pattern "y" Coefficient** is used by the TXP TDM BERT Monitor and TXP Packet BERT Generator to specify the "y" coefficient in the PRBS pattern: $x^n + x^y + 1$, where y = (PTF[4:0] + 1). PTF is ignored when a QRSS or Repetitive Pattern is enabled.
RSVD	[7]		**Reserved.**
QRSS	[6]	rwc-_-_	**QRSS Sequence Select** is used with PTS to select the transmit TXP Packet BERT Generator and the receive TXP TDM BERT Monitor Test Pattern: QRSS/PTS 0 / 0b = $x^z + x^y + 1$ PRBS Pattern (using PLF, PTF and BSP) 0 / 1b = Repetitive Pattern (using PLF and BSP) 1 / 0b = $x^{20} + x^{17} + 1$ QRSS Pattern with a forced "1" if the next 14 bits are "0" 1 / 1b = invalid setting
PTS	[5]	rwc-_-_	**Pattern Type Select.** Used with QRSS to select the TXP BERT Test Pattern
PLF	[4:0]	rwc-_-_	**Test Pattern "z" Coefficient or Length** is used by the TXP TDM BERT Monitor and TXP Packet BERT Generator to specify the "z" coefficient in the PRBS pattern: $x^z + x^y + 1$, where z = (PLF[4:0] + 1); or to specify the length for a Repetitive Pattern. PLF is ignored when the QRSS Pattern is enabled.

EB. Field Name	Addr (A:) Bit [x:y]	Type	Description
BSPR.	**A:0408h**		**BERT Seed / Pattern Register. Default: 0x00.00.00.00**
BSP	[31:0]	rwc-_-_	**BERT Seed/Pattern** specifies the seed value for the transmit PRBS pattern, or the transmit and receive Repetitive Pattern. BSP[31] is the 1st transmitted bit and the expected 1st receive bit. BSP is ignored when the QRSS Pattern is enabled.
TEICR.	**A:0410h**		**Transmit Error Insertion Control Register. Default: 0x00.00.00.00**
RSVD	[31:6]		**Reserved.**
TEIR	[5:3]	rwc-_-_	**Transmit Error Insertion Rate** specifies the rate at which errors are inserted in the TXP Packet BERT Generator output data stream (TSEI = 0). One out of every 10^k bits is inverted where k = TEIR and k > 0. TEIR = 0 disables the Transmit Error Insertion Rate function. TEIR = 1 results in every 10th bit being inverted. If this register is written to during the middle of an error insertion process, the TEIR insertion rate is updated after the next error is inserted.
BEI	[2]	rwc-_-_	**Bit Error Insertion Enable** = "0" disables error insertion (disables TEIR & TSEI)
TSEI	[1]	rwc-_-_	**Transmit Single Error Insert** A 0 to 1 transition forces a single bit error in the TXP Packet BERT Generator output stream (TEIR = 0). If this bit transitions more than once between error insertion opportunities, only one error will be inserted.
MEIMS	[0]	rwc-_-_	**Reserved.**
BSR.	**A:0414h**		**BERT Status Register. Default: 0x00.00.00.00**
RSVD	[31:2]		**Reserved.**
BEC	[1]	ros-_-i3	**Performance Monitoring Update Status** = "1" indicates the TXP TDM BERT Monitor bit error count > 0 (EB.RBECR.BEC).
OOS	[0]	ros-_-i3	**Out Of Synchronization** = "1" indicates the TXP TDM BERT Monitor is not synchronized to the incoming pattern.
BSRL.	**A:0418h**		**BERT Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:3]		**Reserved.**
BEL	[2]	rls-crw-i3	**Bit Error Latched** = "1" when one or more bit errors are detected.
BECL	[1]	rls-crw- i3	**Bit Error Count Latched** = "1" when EB.BSR.BEC transitions from 0 to 1.
OOSL	[0]	rls-crw- i3	**Out Of Synchronization Latched** = "1" when EB.BSR.OOS changes state.
BSRIE.	**A:041Ch**		**BERT Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:3]		**Reserved.**
BEIE	[2]	rwc-_-i3	**Bit Error Interrupt Enable.** The combination of BEIE = 1 and EB.BSRL.BEL = 1 forces G.GSR1.EBS = 1.
BECIE	[1]	rwc-_-i3	**Bit Error Count Interrupt Enable.** The combination of BECIE = 1 and EB.BSRL.BECL = 1 forces G.GSR1.EBS = 1.
OOSIE	[0]	rwc-_-i3	**Out Of Synchronization Interrupt Enable.** The combination of OOSIE = 1 and EB.BSRL.OOSL = 1 forces G.GSR1.EBS = 1.
RBECR.	**A:0420h**		**Receive Bit Error Count Register. Default: 0x00.00.00.00**
RSVD	[31:24]		**Reserved.**
BEC	[23:0]	rcs-cor-sc	**Bit Error Count** = # bit errors during the previous update period (EB.BCR.LPMU) but not including errors during an Out of Sync condition (EB.BSR.OOS = 1).
RBCR.	**A:0424h**		**Receive Bit Count Register. Default: 0x00.00.00.00**
BC	[31:0]	rcs-cor-sc	**Bit Count** = # received bits during the previous update period (EB.BCR.LPMU) but not including errors during an Out of Sync condition (EB.BSR.OOS = 1).
TSTCR.	**A:0430h**		**Test Control Register. Default: 0x00.00.00.00**
RSVD	[31:0]		**Reserved.**

10.3.8 Decap BERT Registers (DB.)

Table 10-24. Decap BERT Registers (DB.)

DB. Field Name	Addr (A:) Bit [x:y]	Type	Description
BCR.	**A:0400h**		**BERT Control Register. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
PMUM	[7]	rwc-_-_	**Reserved.**
LPMU	[6]	rwc-_-_	**Local Performance Monitoring Update.** A 0 to 1 transition of this bit updates the RXP Packet BERT Performance Monitoring registers (DB.RBECR.BEC and DB.RBECR.BC) with the latest counts and then resets the counters.
RNPL	[5]	rwc-_-_	**Receive New Pattern Load.** A 0 to 1 transition of this bit loads the test pattern into the receive RXP Packet BERT Monitor (QRSS, PTS, PLF[4:0], PTF[4:0], and BSP[31:0]). This forces the RXP Packet BERT Monitor to resynchronize to the incoming data pattern. Note: QRSS, PTS, PLF[4:0}, PTF[4:0], and BSP[31:0] must not change until 4 SYSCLK clock cycles after RNPL transitions from 0 to 1.
RPIC	[4]	rwc-_-_	**Receive Pattern Inversion Control.** (RXP Packet BERT Monitor) 0 = test normal (unaltered) incoming data pattern 1 = invert and then test the incoming data pattern
MPR	[3]	rwc-_-_	**Manual Pattern Resynchronization.** A 0 to 1 transition of this bit forces the RXP Packet BERT Monitor to resynchronize to the incoming pattern.
APRD	[2]	rwc-_-_	**Automatic Pattern Resynchronization Disable.** For APRD = 0, the RXP Packet BERT Monitor is forced to resynchronize to the incoming pattern when 6 received bits, within a 64-bit window, do not match the expected pattern. For APRD = 1, after the RXP Packet BERT Monitor finds synchronization lock, it does not attempt to resynchronize regardless of how many bit errors are detected.
TNPL	[1]	rwc-_-_	**Transmit New Pattern Load.** A 0 to 1 transition of this bit loads the test pattern into the transmit RXP TDM BERT Generator (QRSS, PTS, PLF[4:0], PTF[4:0], and BSP[31:0]). Note: QRSS, PTS, PLF[4:0}, PTF[4:0], and BSP[31:0] must not change until 4 SYSCLK clock cycles after TNPL transitions from 0 to 1.
TPIC	[0]	rwc-_-_	**Transmit Pattern Inversion Control.** (RXP TDM BERT Generator) 0 = transmit normal (unaltered) outgoing data pattern 1 = transmit inverted outgoing data pattern
BPCR.	**A:0404h**		**BERT Pattern Configuration Register. Default: 0x00.00.00.00**
RSVD	[31:13]		**Reserved.**
PTF	[12:8]	rwc-_-_	**Test Pattern "y" Coefficient** is used by the RXP Packet BERT Monitor and RXP TDM BERT Generator to specify the "y" coefficient in the PRBS pattern: $x^n + x^y + 1$, where y = (PTF[4:0] + 1). PTF is ignored when a QRSS or Repetitive Pattern is enabled.
RSVD	[7]		**Reserved.**
QRSS	[6]	rwc-_-_	**QRSS Sequence Select** is used with PTS to select the transmit RXP TDM BERT Generator and the receive RXP Packet BERT Monitor Test Pattern: <u>QRSS/PTS</u> 0 / 0b = $x^z + x^y + 1$ PRBS Pattern (using PLF, PTF and BSP) 0 / 1b = Repetitive Pattern (using PLF and BSP) 1 / 0b = $x^{20} + x^{17} + 1$ QRSS Pattern with a forced "1" if the next 14 bits are "0" 1 / 1b = invalid setting
PTS	[5]	rwc-_-_	**Pattern Type Select.** Used with QRSS to select the RXP BERT Test Pattern
PLF	[4:0]	rwc-_-_	**Test Pattern "z" Coefficient or Length** is used by the RXP Packet BERT Monitor and RXP TDM BERT Generator to specify the "z" coefficient in the PRBS pattern: $x^z + x^y + 1$, where z = (PLF[4:0] + 1); or to specify the length for a Repetitive Pattern. PLF is ignored when the QRSS Pattern is enabled.

DB. Field Name	Addr (A:) Bit [x:y]	Type	Description
BSPR1.	**A:0408h**		**BERT Seed / Pattern Register. Default: 0x00.00.00.00**
BSP	[31:0]	rwc-_-_	**BERT Seed/Pattern** specifies the seed value for the transmit PRBS pattern, or the transmit and receive Repetitive Pattern. BSP[31] is the 1st transmitted bit and the expected 1st receive bit. BSP is ignored when the QRSS Pattern is enabled.
TEICR.	**A:0410h**		**Transmit Error Insertion Control Register. Default: 0x00.00.00.00**
RSVD	[31:6]		**Reserved.**
TEIR	[5:3]	rwc-_-_	**Transmit Error Insertion Rate** specifies the rate at which errors are inserted in the RXP TDM BERT Generator output data stream (TSEI = 0). One out of every 10^k bits is inverted where k = TEIR and k > 0. TEIR = 0 disables the Transmit Error Insertion Rate function. TEIR = 1 results in every 10th bit being inverted. If this register is written to during the middle of an error insertion process, the TEIR insertion rate is updated after the next error is inserted.
BEI	[2]	rwc-_-_	**Bit Error Insertion Enable** = "0" disables error insertion (disables TEIR & TSEI)
TSEI	[1]	rwc-_-_	**Transmit Single Error Insert** A 0 to 1 transition forces a single bit error in the RXP TDM BERT Generator output stream (TEIR = 0). If this bit transitions more than once between error insertion opportunities, only one error will be inserted.
MEIMS	[0]	rwc-_-_	**Reserved.**
BSR.	**A:0414h**		**BERT Status Register. Default: 0x00.00.00.00**
RSVD	[31:2]		**Reserved.**
BEC	[1]	ros-_-i3	**Performance Monitoring Update Status** = "1" indicates the RXP Packet BERT Monitor bit error count > 0 (DB.RBECR.BEC).
OOS	[0]	ros-_-i3	**Out Of Synchronization** = "1" indicates the RXP Packet BERT Monitor is not synchronized to the incoming pattern.
BSRL.	**A:0418h**		**BERT Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:3]		**Reserved.**
BEL	[2]	rls-crw-i3	**Bit Error Latched** = "1" when one or more bit errors are detected.
BECL	[1]	rls-crw-i3	**Bit Error Count Latched** = "1" when DB.BSR.BEC transitions from 0 to 1.
OOSL	[0]	rls-crw-i3	**Out Of Synchronization Latched** = "1" when DB.BSR.OOS changes state.
BSRIE.	**A:041Ch**		**BERT Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:3]		**Reserved.**
BEIE	[2]	rwc-_-i3	**Bit Error Interrupt Enable.** The combination of BEIE = 1 and DB.BSRL.BEL = 1 forces G.GSR1.EBS = 1.
BECIE	[1]	rwc-_-i3	**Bit Error Count Interrupt Enable.** The combination of BECIE = 1 and DB.BSRL.BECL = 1 forces G.GSR1.EBS = 1.
OOSIE	[0]	rwc-_-i3	**Out Of Synchronization Interrupt Enable.** The combination of OOSIE = 1 and DB.BSRL.OOSL = 1 forces G.GSR1.EBS = 1.
RBECR.	**A:0420h**		**Receive Bit Error Count Register. Default: 0x00.00.00.00**
RSVD	[31:24]		**Reserved.**
BEC	[23:0]	rcs-cor-sc	**Bit Error Count** = # bit errors during the previous update period (DB.BCR.LPMU) but not including errors during an Out of Sync condition (DB.BSR.OOS = 1).
RBCR.	**A:0424h**		**Receive Bit Count Register. Default: 0x00.00.00.00**
BC	[31:0]	rcs-cor-sc	**Bit Count** = # received bits during the previous update period (DB.BCR.LPMU) but not including errors during an Out of Sync condition (DB.BSR.OOS = 1).
TSTCR.	**A:0430h**		**Test Control Register. Default: 0x00.00.00.00**
RSVD	[31:0]		**Reserved.**

10.3.9 Miscellaneous Diagnostic Registers (MD.)

Table 10-25. Miscellaneous Diagnostic Registers (MD.)

MD. Field Name	Addr (A:) Bit [x:y]	Type	Description
DCR.	**A:0480h**		**Diagnostic Control Register. Default: 0x00.00.00.00**
RSVD	[31:1]		**Reserved.**
MBE	[0]	rwc-_-_	**Memory BIST Enable** enables the memory BIST test. This test runs until complete. The result is visible in the diagnostic register
EBCR.	**A:0484h**		**Encap BERT Control Register. Default: 0x00.00.00.00**
RSVD	[31:24]		**Reserved.**
ETBE	[24]	rwc-_-_	**Encap Transmit BERT Enable** enables TXP Packet BERT Generator to insert a test pattern into the packet payload section of a TXP Bundle (see ETBBS).
ETBBS	[23:16]	rwc-_-_	**Encap Transmit BERT Bundle Select** selects the TXP Bundle # that carries the output data stream of the TXP Packet BERT Generator (ETBE = 1).
RSVD	[15:9]		**Reserved.**
ERBE	[8]	rwc-_-_	**Encap Receive BERT Enable** enables the TXP TDM BERT Monitor to test the receive TDM Port Timeslot data for a TXP Bundle (see ERBBS).
ERBBS	[7:0]	rwc-_-_	**Encap Receive BERT Bundle Select** selects the TXP Bundle # that receives the TDM Port Timeslot data that is tested by the TXP TDM BERT Monitor (ERBE = 1).
DBCR.	**A:0488h**		**Decap BERT Control Register. Default: 0x00.00.00.00**
RSVD	[31:24]		**Reserved.**
DTBE	[24]	rwc-_-_	**Decap Transmit BERT Enable** enables RXP TDM BERT Generator to insert a test pattern into the transmit TDM Port Timeslot of an RXP Bundle (see DTBBS).
DTBBS	[23:16]	rwc-_-_	**Decap Transmit BERT Bundle Select** selects RXP Bundle # for the TDM Port Timeslots that transmit the RXP TDM BERT Generator output data (DTBE = 1).
RSVD	[15:9]		**Reserved.**
DRBE	[8]	rwc-_-_	**Decap Receive BERT Enable** enables the RXP Packet BERT Monitor to test the RXP packet payload data for an RXP Bundle (see DRBBS).
DRBBS	[7:0]	rwc-_-_	**Decap Receive BERT Bundle Select** selects the RXP Bundle # for the RXP packet payload data that is tested by the RXP Packet BERT Monitor (DRBE = 1).
MBSR1.	**A:04A0h**		**Memory BIST Status Register 1. Default: 0x00.00.00.00**
MBD	[31:0]	ros-_-_	**Memory BIST Done.** Memory BIST Done Status Bits (only valid if DCR.MBE = 1).
MBSR2.	**A:04A4h**		**Memory BIST Status Register 2. Default: 0x00.00.00.00**
MBD	[31:0]	ros-_-_	**Memory BIST Done.** Memory BIST Done Status Bits (only valid if DCR.MBE = 1).
MBSR3.	**A:04A8h**		**Memory BIST Status Register 3. Default: 0x00.00.00.00**
MBF	[31:0]	ros-_-_	**Memory BIST Fail.** Memory BIST Fail Status Bits. M.MBSR3.MBF[x] is only valid when M.MBRS1.MBD[x] = 1 and M.DCR.MBE = 1 (x = 0 to 31).
MBSR4.	**A:04ACh**		**Memory BIST Status Register 4. Default: 0x00.00.00.00**
MBF	[31:0]	ros-_-_	**Memory BIST Fail.** Memory BIST Fail Status Bits. M.MBSR4.MBF[x] is only valid when M.MBRS2.MBD[x] = 1 and M.DCR.MBE = 1 (x = 0 to 31).
MBSR5.	**A:04B0h**		**Memory BIST Status Register 5. Default: 0x00.00.00.00**
RBS	[31:0]	ros-_-_	**ROM BIST Signature. (**only valid if DCR.MBE = 1 and the BIST has completed).

10.3.10 Test Registers (TST.)

Table 10-26. Test Registers (TST.)

TST. Field Name	Addr (A:) Bit [x:y]	Type	Description
GTR1.	**A:0600h**		**Global Test Control Register 1. Default: 0x00.00.00.00**
RSVD	[31:7]		**Reserved.**
CTCE	[6]	rwc-_-_	**Reserved.**
CWLUPM	[5]	rwc-_-_	**Reserved.**
SOEE	[4]	rwc-_-_	**Reserved.**
COEE	[3]	rwc-_-_	**Reserved.**
MTPOE	[2]	rwc-_-_	**Reserved.**
MCRS	[1]	rwc-_-_	**Reserved.**
INTE	[0]	rwc-_-_	**Reserved.**
BTCR1.	**A:0604h**		**Block Test Control Register 1. Default: 0x00.00.00.00**
RSVD	[31:16]		**Reserved.**
TPIBTC	[15:0]	rwc-_-_	**Reserved.**
BTCR2.	**A:0608h**		**Block Test Control Register 2. Default: 0x00.00.00.00**
RSVD	[31:16]		**Reserved.**
RPIBTC	[15:0]	rwc-_-_	**Reserved.**
BTCR3.	**A:060Ch**		**Block Test Control Register 3. Default: 0x00.00.00.00**
RSVD	[31:16]		**Reserved.**
TDIBTC	[15:0]	rwc-_-_	**Reserved.**
BTCR4.	**A:0610h**		**Block Test Control Register 4. Default: 0x00.00.00.00**
RSVD	[31:16]		**Reserved.**
TEIBTC	[15:0]	rwc-_-_	**Reserved.**
BTCR5.	**A:0614h**		**Block Test Control Register 5. Default: 0x00.00.00.00**
RSVD	[31:16]		**Reserved.**
EMIBTC	[15:0]	rwc-_-_	**Reserved.**
BTCR6.	**A:0618h**		**Block Test Control Register 6. Default: 0x00.00.00.00**
RSVD	[31:16]		**Reserved.**
SBIBTC	[15:8]	rwc-_-_	**Reserved.**
SBIBTC	[7:4]	ros-_-_	**Silicon Revision ID**
SBIBTC	[3:0]	rwc-_-_	**Reserved.**
CRJBT	**A:061Ch**		**Clock Recovery Jitter Buffer Test. Default: 0x00.00.00.00**
RSVD	[31:16		**Reserved.**
JBBS	[15:8]	rwc-_-_	**Reserved.**
RSVD	[7:5]		**Reserved.**
CRCS	[4:0]	rwc-_-_	**Reserved.**
BTSR1.	**A:0624h**		**Block Test Status Register 1. Default: 0x00.00.00.00**
TPIBTS	[31:0]		**Reserved.**
BTSR2.	**A:0628h**		**Block Test Status Register 2. Default: 0x00.00.00.00**
RPIBTS	[31:0]		**Reserved.**
BTSR3.	**A:062Ch**		**Block Test Status Register 3. Default: 0x00.00.00.00**
TDIBTS	[31:0]		**Reserved.**

TST. Field Name	Addr (A:) Bit [x:y]	Type	Description
BTSR4.	**A:0630h**		**Block Test Status Register 4. Default: 0x00.00.00.00**
TEIBTS	[31:0]		**Reserved.**
BTSR5.	**A:0634h**		**Block Test Status Register 5. Default: 0x00.00.00.00**
EMIBTS	[31:0]		**Reserved.**
BTSR6.	**A:0638h**		**Block Test Status Register 6. Default: 0x00.00.00.00**
SBIBTS	[31:0]		**Reserved.**
CTCR1.	**A:0640h**		**CLAD Test Control Register 1. Default: 0x00.00.00.00**
PD	[31:28]	rwc-_-_	**Reserved.**
RST	[27:24]	rwc-_-_	**Reserved.**
TCS	[23:22]	rwc-_-_	**Reserved.**
IRA	[21:19]	rwc-_-_	**Reserved.**
VRA	[18:17]	rwc-_-_	**Reserved.**
RSVD	[16:7]		**Reserved.**
PMIA	[6:0]	rwc-_-_	**Reserved.**
CTCR2.	**A:0644h**		**CLAD Test Control Register 2. Default: 0x00.00.00.00**
RSVD	[31:9]		**Reserved.**
PPCS	[8:7]	rwc-_-_	**Reserved.**
PPIA	[6:0]	rwc-_-_	**Reserved.**
CTCR3.	**A:0648h**		**CLAD Test Control Register 3. Default: 0x00.00.00.00**
RSVD	[31:9]		**Reserved.**
PTECS	[8:7]	rwc-_-_	**Reserved.**
PTEIA	[6:0]	rwc-_-_	**Reserved.**
CTCR4.	**A:064Ch**		**CLAD Test Control Register 4. Default: 0x00.00.00.00**
RSVD	[31:25]		**Reserved.**
PTSTVA	[24:9]	rwc-_-_	**Reserved.**
PTSTCS	[8:7]	rwc-_-_	**Reserved.**
PTSTIA	[6:0]	rwc-_-_	**Reserved.**
EDTCR	**A:0660h**		**Encap/Decap Test Control Register. Default: 0x00.00.00.00**
RSVD	[31:10]		**Reserved.**
EDMEM	[9:8]	rwc-_-_	**Reserved.**
EDBDL	[7:0]	rwc-_-_	**Reserved.**
EDTSR1	**A:0664h**		**Encap/Decap Test Status Register 1**
RSVD	[31:1]		**Reserved.**
EDVLD	[0]	ros-_-_	**Reserved.**
EDTSR2	**A:0668h**		**Encap/Decap Test Status Register 2**
EDRDT	[31:0]	ros-_-_	**Reserved.**
EDTSR3	**A:066Ch**		**Encap/Decap Test Status Register 3**
EDRDT	[31:0]	ros-_-_	**Reserved.**
EDTSR4	**A:0670h**		**Encap/Decap Test Status Register 4**
EDRDT	[31:0]	ros-_-_	**Reserved.**
EDTSR5	**A:0674h**		**Encap/Decap Test Status Register 5**
EDRDT	[31:0]	ros-_-_	**Reserved.**

TST. Field Name	Addr (A:) Bit [x:y]	Type	Description
FID.	**A:06FCh**		**Block Test Control Register 6. Default: 0x00.00.00.1A**
FRI	[31:0]	ros-_-_	**Reserved.**

10.3.11 Clock Recovery Registers (CR.)

These registers are defined by the S132 Clock Recovery firmware load (according to the firmware revision).

10.3.12 MAC Registers (M.)

Table 10-27. MAC Registers (M.)

M. Field Name	Addr (A:) Bit [x:y]	Type	Description
NET_CONTROL.	**A:0C00h**		**Network Control Register. Default: 0x00.00.00.00**
RSVD	[31:9]		**Reserved.**
RD_SNAP	[14]	rwc-_-_	**Read Snapshot** = "1" enables the Ethernet RMON statistics registers to provide latched values. When "0" they provide real-time/raw values.
TAKE_SNAP	[13]	woc-_-_	**Take Snapshot** A 0 to 1 transition latches the current Ethernet statistics into the statistics registers and then resets the counters (RD_SNAP = 1).
TX_0Q_PAUSE	[12]	woc-_-_	**Reserved.**
TX_PAUSE	[11]	woc-_-_	**Reserved.**
TX_HALT	[10]	woc-_-_	**Transmit Halt** = "1" disables MAC transmission. If a packet is already partially transmitted, the complete packet is transmitted before stopping.
START_TX	[9]	woc-_-_	**Start Transmission** = "1" starts transmission.
RSVD	[8]		**Reserved.**
STATS_WR_EN	[7]	rwc-_-_	**Reserved.**
STATS_INC	[6]	woc-_-_	**Reserved.**
STATS_CLR	[5]	woc-_-_	**Statistics Clear = "1" clears the statistics registers.**
MAN_PORT_EN	[4]	rwc-_-_	**Management Port Enable** = "1" to enable the MDIO management port. When "0" forces MDIO to high impedance state and MDC low.
TX_EN	[3]	rwc-_-_	**Transmit Enable** = "1" enables MAC transmission. "0" immediately stops transmission (partially transmitted packets are aborted).
RX_EN	[2]	rwc-_-_	**Receive Enable** = "1" enables the MAC to receive data. When "0", frame reception will stop immediately (partially received packets are aborted).
LB_LOCAL	[1]	rwc-_-_	**Loop Back Local** = "1" enables the Ethernet Loopback (TXP to RXP)
LB	[0]	rwc-_-_	**Reserved.**
NET_CONFIG.	**A:0C04h**		**Network Configuration Register. Default: 00.0C.00.00h**
RSVD	[31:30]		**Reserved.**
BAD_PREAMB	[29]	rwc-_-_	**Reserved.**
IPG	[28]	rwc-_-_	**IPG Stretch Enable** = "1" enables the MAC to increase the Inter Packet Gap to > 96 bit times (see M.IPG_STRETCH).
RSVD	[27]	rwc-_-_	**Reserved.**
IGN_RX_FCS	[26]	rwc-_-_	**Reserved.**
EN_FRMS_HDUP	[25]	rwc-_-_	**Reserved.**
RX_CHK_EN	[24]	rwc-_-_	**Reserved.**
DIS_CP_PAUSE	[23]	rwc-_-_	**Reserved.** This must be programmed to "1".
RSVD	[22:21]		**Reserved.**

M. Field Name	Addr (A:) Bit [x:y]	Type	Description
MDC_CLK_DIV	[20:18]	rwc-_-_	**MDC Clock Division** selects the MDC frequency where MDC_{freq} = SYSCLK ÷ MDC_CLK_DIV. To comply with IEEE 802.3, MDC_{freq} must not exceed 2.5 MHz. 1 = divide by 32 (for SYSCLK ≤ 80 MHz) 2 = divide by 48 (for SYSCLK ≤ 120MHz) 4 = divide by 64 (for SYSCLK ≤ 160 MHz) 5 = divide by 96 (for SYSCLK ≤ 240 MHz) 6 = divide by 128 (for SYSCLK ≤ 320 MHz) 7 = divide by 224 (for SYSCLK ≤ 540 MHz)
FCS_REMOVE	[17]	rwc-_-_	**Reserved.** This must be programmed to “1”.
LGTH_FRM_DIS	[16]	rwc-_-_	**Reserved.** This must be programmed to “1”.
RX_BUF_OFFSET	[15:14]	rwc-_-_	**Reserved.**
PAUSE_EN	[13]	rwc-_-_	**Reserved.**
RETRY_TST	[12]	rwc-_-_	**Reserved.**
RSVD	[11]		**Reserved.**
GIG_MODE_EN	[10]	rwc-_-_	**Gigabit Mode Enable.** 0 = 100 Mb/s operation using an MII interface 1 = 1000 Mb/s operation using a GMII interface
EXT_AMATCHEN	[9]	rwc-_-_	**Reserved.**
RX_1536FRMS	[8]	rwc-_-_	**Receive 1536 Byte Frames.** 0 = maximum receive Ethernet packet length is 1518 bytes 1 = maximum receive Ethernet packet length is 1536 bytes
UNI_HSH_EN	[7]	rwc-_-_	**Reserved.**
MULT_HSH_EN	[6]	rwc-_-_	**Reserved.**
NO_BROADCAST	[5]	rwc-_-_	**No Broadcast.** 0 = This function is disabled. 1 = Packets with the Ethernet Broadcast DA are discarded.
COPY_FRMS	[4]	rwc-_-_	**Reserved.** This must be programmed to “1”.
JUMBO_FRMS	[3]	rwc-_-_	**Reserved.**
DISC_NONVLAN	[2]	rwc-_-_	**Discard Non-VLAN** = 1 = discard with no VLAN tags.
FULL_DUPLEX	[1]	rwc-_-_	**Reserved.** This must be programmed to “1”.
SPEED	[0]	rwc-_-_	**Reserved.** This must be programmed to “1”.
NET_STATUS.	**A:0C08h**		**Network Status Register. Default: 00.00.00.04h**
RSVD	[31:3]		**Reserved.**
PHY_MAN_IDLE	[2]	ros-_-_	**PHY Management Idle** = 1 = MDIO (Phy) management is idle (i.e. has completed).
MDIOS	[1]	ros-_-_	**MDIO Status** indicates the status/value of the MDIO signal.
SYNC_STAT	[0]	ros-_-_	**Reserved.**
RSVD.	**A:0C0Ch**		**Reserved.**
USER_IO.	**A:0C10h**		**User Input/Output Register. Default: 0x00.00.00.00**
USER_PRG_IN	[31:16]	ros-_-_	**Reserved.**
USER_PRG_OUT	[15:0]	rwc-_-_	**Reserved.**
TX_STATUS.	**A:0C14h**		**Transmit Status Register. Default: 0x00.00.00.00**
RSVD	[31:9]		**Reserved.**
TX_HRESP	[8]	rls-cow-_	**Reserved.**
LATE_COL	[7]	rls-cow-_	**Reserved.**
TX_URUN	[6]	rls-cow-_	**Reserved.**

M. Field Name	Addr (A:) Bit [x:y]	Type	Description
TX_COMPLETE	[5]	rls-cow-_	**Reserved.**
TX_BUF_EXH	[4]	rls-cow-_	**Reserved.**
TX_GO	[3]	rls-cow-_	**Reserved.**
TX_RETRY_EXC	[2]	rls-cow-_	**Reserved.**
TX_COL	[1]	rls-cow-_	**Reserved.**
TX_USED	[0]	rls-cow-_	**Reserved.**
RX_QPTR.	**A:0C18h**		**Receive Buffer Queue Base Address. Default: 0x00.00.00.00**
RX_BUF_QBA	[31:2]	rwc-_-_	**Reserved.**
RSVD	[1:0]		**Reserved.**
TX_QPTR.	**A:0C1Ch**		**Transmit Queue Base Address. Default: 0x00.00.00.00**
TX_B UF_QBA	[31:2]	rwc-_-_	**Reserved.**
RSVD	[1:0]		**Reserved.**
RX_STATUS.	**A:0C20h**		**Receive Status Register. Default: 0x00.00.00.00**
RSVD	[31:4]		**Reserved.**
RX_HRESP	[3]	rls-cow-_	**Reserved.**
RX_ORUN	[2]	rls-cow-_	**Reserved.**
RX_DONE	[1]	rls-cow-_	**Reserved.**
RX_BUF_USED	[0]	rls-cow-_	**Reserved.**
IRQ_STATUS.	**A:0C24h**		**Interrupt Status Register. Default: 0x00.00.00.00**
RSVD	[31:16]		**Reserved.**
IRQ_EXT_INT	[15]	rls-cor-_	**Reserved.**
IRQ_PAUSE_TX	[14]	rls-cor-_	**Reserved.**
IRQ_PAUSE_0	[13]	rls-cor-_	**Reserved.**
IRQ_PAUSE_RX	[12]	rls-cor-_	**Reserved.**
IRQ_HRESP	[11]	rls-cor-_	**Reserved.**
IRQ_RX_ORUN	[10]	rls-cor-_	**Reserved.**
RSVD	[9:8]		**Reserved.**
IRQ_TX_DONE	[7]	rls-cor-_	**Reserved.**
IRQ_TX_ERROR	[6]	rls-cor-_	**Reserved.**
IRQ_RETRY_EXC	[5]	rls-cor-_	**Reserved.**
IRQ_TX_URUN	[4]	rls-cor-_	**Reserved.**
IRQ_TX_USED	[3]	rls-cor-_	**Reserved.**
IRQ_RX_USED	[2]	rls-cor-_	**Reserved.**
IRQ_RX_DONE	[1]	rls-cor-_	**Reserved.**
IRQ_MAN_DONE	[0]	rls-cor-i3	**PHY Management Operation Complete** = "1" = MDIO operation done.
IRQ_ENABLE.	**A:0C28h**		**Interrupt Enable Register. Default: 0x00.00.00.00**
RSVD	[31:16]		**Reserved.**
EN_IRQ_EXT_INT	[15]	woc-_-_	**Reserved.**
EN_IRQ_PAUSE_TX	[14]	woc-_-_	**Reserved.**
EN_IRQ_PAUSE_0	[13]	woc-_-_	**Reserved.**
EN_IRQ_PAUSE_RX	[12]	woc-_-_	**Reserved.**
EN_IRQ_HRESP	[11]	woc-_-_	**Reserved.**
EN_IRQ_RX_ORUN	[10]	woc-_-_	**Reserved.**

M. Field Name	Addr (A:) Bit [x:y]	Type	Description
RSVD	[9:8]		**Reserved.**
EN_IRQ_TX_DONE	[7]	woc-_-_	**Reserved.**
EN_IRQ_TX_ERROR	[6]	woc-_-_	**Reserved.**
EN_IRQ_RETRY_EXC	[5]	woc-_-_	**Reserved.**
EN_IRQ_TX_URUN	[4]	woc-_-_	**Reserved.**
EN_IRQ_TX_USED	[3]	woc-_-_	**Reserved.**
EN_IRQ_RX_USED	[2]	woc-_-_	**Reserved.**
EN_IRQ_RX_DONE	[1]	woc-_-_	**Reserved.**
EN_IRQ_MAN_DONE	[0]	woc-_-i3	**Enable PHY Management Operation Complete.** The combination of EN_IRQ_MAN_DONE = 1, DIS_IRQ_MAN_DONE = 0 and IRQ_MAN_DONE = 1, forces MIRS = 1.
IRQ_DISABLE.	**A:0C2Ch**		**Interrupt Disable Register. Default: 0x00.00.00.00**
RSVD	[31:18]		**Reserved.**
RSVD	[17:16]		**Reserved.**
DIS_IRQ_EXT_INT	[15]	woc-_-_	**Reserved.**
DIS_IRQ_PAUSE_TX	[14]	woc-_-_	**Reserved.**
DIS_IRQ_PAUSE_0	[13]	woc-_-_	**Reserved.**
DIS_IRQ_PAUSE_RX	[12]	woc-_-_	**Reserved.**
DIS_IRQ_HRESP	[11]	woc-_-_	**Reserved.**
DIS_IRQ_RX_ORUN	[10]	woc-_-_	**Reserved.**
RSVD	[9:8]		**Reserved.**
DIS_IRQ_TX_DONE	[7]	woc-_-_	**Reserved.**
DIS_IRQ_TX_ERROR	[6]	woc-_-_	**Reserved.**
DIS_IRQ_RETRY_EXC	[5]	woc-_-_	**Reserved.**
DIS_IRQ_TX_URUN	[4]	woc-_-_	**Reserved.**
DIS_IRQ_TX_USED	[3]	woc-_-_	**Reserved.**
DIS_IRQ_RX_USED	[2]	woc-_-_	**Reserved.**
DIS_IRQ_RX_DONE	[1]	woc-_-_	**Reserved.**
DIS_IRQ_MAN_DONE	[0]	woc-_-i3	**Disable PHY Management Operation Complete.** (see EN_IRQ_MAN_DONE)
IRQ_MASK.	**A:0C30h**		**Interrupt Mask Register. Default: 0x00.00.00.00**
RSVD	[31:16]		**Reserved.**
MSK_IRQ_EXT_INT	[15]	ros-_-_	**Reserved.**
MSK_IRQ_PAUSE_TX	[14]	ros-_-_	**Reserved.**
MSK_IRQ_PAUSE_0	[13]	ros-_-_	**Reserved.**
MSK_IRQ_PAUSE_RX	[12]	ros-_-_	**Reserved.**
MSK_IRQ_HRESP	[11]	ros-_-_	**Reserved.**
MSK_IRQ_RX_ORUN	[10]	ros-_-_	**Reserved.**
RSVD	[9:8]		**Reserved.**
MSK_IRQ_TX_DONE	[7]	ros-_-_	**Reserved.**
MSK_IRQ_TX_ERROR	[6]	ros-_-_	**Reserved.**
MSK_IRQ_RETRY_EXC	[5]	ros-_-_	**Reserved.**
MSK_IRQ_TX_URUN	[4]	ros-_-_	**Reserved.**
MSK_IRQ_TX_USED	[3]	ros-_-_	**Reserved.**

M. Field Name	Addr (A:) Bit [x:y]	Type	Description
MSK_IRQ_RX_USED	[2]	ros-_-_	**Reserved.**
MSK_IRQ_RX_DONE	[1]	ros-_-_	**Reserved.**
MSK_IRQ_MAN_DONE	[0]	ros-_-_	**Mask PHY Management Operation Complete.** A read of this register returns the value of the management done interrupt mask. 0: Interrupt is enabled 1: Interrupt is disabled A write to this register directly affects the state of the corresponding bit in the interrupt status register, causing an interrupt to be generated if a 1 is written.
PHY_MAN.	**A:0C34h**		**Phy Maintenance Register. Default: 0x00.00.00.00**
PHY_SET3	[31]	rwc-_-_	**Reserved.**
PHY_CL22	[30]	rwc-_-_	**Reserved.** This must be programmed to “1”.
PHY_SET2	[29:28]	rwc-_-_	**Phy Set 2** selects the MDIO Operation: 2 = Read; 1 = Write.
PHY_ADDR	[27:23]	rwc-_-_	**Phy Address** selects the MDIO Phy address.
PHY_REG_ADDR	[22:18]	rwc-_-_	**Phy Register Address** selects the MDIO Register address.
PHY_SET1	[17:16]	rwc-_-_	**Reserved.** This must be programmed to “2”.
PHY_DATA_WR	[15:0]	rwc-_-_	**Phy Data to be Written** provides the Write data sent to the Phy or the Read data received from the Phy according to the PHY_SET2 operation.
RX_PAUSE_TIME.	**A:0C38h**		**Received Pause Quantum Register. Default: 0x00.00.00.00**
RSVD	[31:16]		**Reserved.**
RX_PAUSE_Q	[15:0]	ros-_-_	**Reserved.**
TX_PAUSE_QUANT.	**A:0C3Ch**		**Transmit Pause Quantum Register. Default: 00.00.FF.FFh**
RSVD	[31:16]		**Reserved.**
TX_PAUSE_Q	[15:0]	rwc-_-_	**Reserved.**
HASH_BOT.	**A:0C80h**		**Hash Register Bottom. Default: 0x00.00.00.00**
HASH_BOT	[31:0]	rwc-_-_	**Reserved.**
HASH_TOP.	**A:0C84h**		**Hash Register Top. Default: 0x00.00.00.00**
HASH_TOP	[31:0]	rwc-_-_	**Reserved.**
LADDR1_BOT.	**A:0C88h**		**Specific Address 1 Bottom. Default: 0x00.00.00.00**
SPEC_ADD1_BOT	[31:0]	rwc-_-_	**Reserved.**
LADDR1_TOP.	**A:0C8Ch**		**Specific Address 1 Top. Default: 0x00.00.00.00**
RSVD	[31:16]		**Reserved.**
SPEC_ADD1_TOP	[15:0]	rwc-_-_	**Reserved.**
LADDR2_BOT.	**A:0C90h**		**Specific Address 2 Bottom. Default: 0x00.00.00.00**
SPEC_ADD2_BOT	[31:0]	rwc-_-_	**Reserved.**
LADDR2_TOP.	**A:0C94h**		**Specific Address 2 Top. Default: 0x00.00.00.00**
RSVD	[31:16]		**Reserved.**
SPEC_ADD2_TOP	[15:0]	rwc-_-_	**Reserved.**
LADDR3_BOT.	**A:0C98h**		**Specific Address 3 Bottom. Default: 0x00.00.00.00**
SPEC_ADD3_BOT	[31:0]	rwc-_-_	**Reserved.**
LADDR3_TOP.	**A:0C9Ch**		**Specific Address 3 Top. Default: 0x00.00.00.00**
RSVD	[31:16]		**Reserved.**
SPEC_ADD3_TOP	[15:0]	rwc-_-_	**Reserved.**
LADDR4_BOT.	**A:0CA0h**		**Specific Address 4 Bottom. Default: 0x00.00.00.00**
SPEC_ADD4_BOT	[31:0]	rwc-_-_	**Reserved.**

M. Field Name	Addr (A:) Bit [x:y]	Type	Description
LADDR4_TOP.	**A:0CA4h**		**Specific Address 4 Top. Default: 0x00.00.00.00**
RSVD	[31:16]		**Reserved.**
SPEC_ADD4_TOP	[15:0]	rwc-_-_	**Reserved.**
ID_CHECK1.	**A:0CA8h**		**Type ID Match 1. Default: 0x00.00.00.00**
EN_TYPE_ID_M1	[31]	rwc-_-_	**Reserved.**
RSVD	[30:16]		**Reserved.**
TYPE_ID_M1	[15:0]	rwc-_-_	**Reserved.**
ID_CHECK2.	**A:0CACh**		**Type ID Match 2. Default: 0x00.00.00.00**
EN_TYPE_ID_M2	[31]	rwc-_-_	**Reserved.**
RSVD	[30:16]		**Reserved.**
TYPE_ID_M2	[15:0]	rwc-_-_	**Reserved.**
ID_CHECK3.	**A:0CB0h**		**Type ID Match 3. Default: 0x00.00.00.00**
EN_TYPE_ID_M3	[31]	rwc-_-_	**Reserved.**
RSVD	[30:16]		**Reserved.**
TYPE_ID_M3	[15:0]	rwc-_-_	**Reserved.**
ID_CHECK4.	**A:0CB4h**		**Type ID Match 4. Default: 0x00.00.00.00**
EN_TYPE_ID_M4	[31]	rwc-_-_	**Reserved.**
RSVD	[30:16]		**Reserved.**
TYPE_ID_M4	[15:0]	rwc-_-_	**Reserved.**
RSVD.	**A:0CB8h**		**Reserved.**
IPG_STRETCH.	**A:0CBCh**		**IPG Stretch Register. Default: 0x00.00.00.00**
RSVD	[31:16]		**Reserved.**
IPG	[15:0]	rwc-_-_	**Inter-Packet Gap** can be used to modify the Inter Packet Gap between transmitted packets. Bits 7:0 are multiplied with the previously transmitted frame length (including preamble) bits 15:8 +1 divide the frame length. If the resulting number is greater than 96 and bit 28 is set in the M.NET_CONFIG.IPG = 1 network configuration register then the resulting number is used for the transmit inter-packet-gap. 1 is added to bits 15:8 to prevent a divide by zero.
MOD_ID.	**A:0CFCh**		**Module Revision ID Register. Default: 00.02.00.00h**
RSVD	[31:16]	ros-_-_	**Reserved.**
MOD_REV	[15:0]	ros-_-_	**Reserved.**
OCT_TX_BOT.	**A:0D00h**		**Octet Transmitted Bottom. Default: 0x00.00.00.00**
TX_OCTETS_FRM	[31:0]	rcs-cor-sc	**Transmitted Octets in Frame [31:0]** = # octets in transmitted frames (48-bit count using OCT_TX_BOT and OCT_TX_TOP).
OCT_TX_TOP.	**A:0D04h**		**Octet Transmitted Top. Default: 0x00.00.00.00**
RSVD	[31:16]		**Reserved.**
TX_OCTETS_FRM	[15:0]	rcs-cor-sc	**Transmitted Octets in Frame [47:32].** (see OCT_TX_BOT)
STATS_FRAMES_ TX.	**A:0D08h**		**Frames Transmitted Top. Default: 0x00.00.00.00**
FRMS_TX	[31:0]	rcs-cor-sc	**Frames Transmitted** = # transmitted frames.
BROADCAST_TX.	**A:0D0Ch**		**Broadcast Frames Transmitted. Default: 0x00.00.00.00**
BRDCST_TX	[31:0]	rcs-cor-sc	**Broadcast Frames Transmitted** = # transmitted Ethernet Broadcast frames.

M. Field Name	Addr (A:) Bit [x:y]	Type	Description
MULTICAST_TX.	**A:0D10h**		**Multicast Frames Transmitted. Default: 0x00.00.00.00**
MLTCST_TX	[31:0]	rcs-cor-sc	**Multicast Frames Transmitted** = # transmitted Ethernet Multicast frames.
STATS_PAUSE_TX.	**A:0D14h**		**Pause Frames Transmitted. Default: 0x00.00.00.00**
RSVD	[31:16]		**Reserved.**
PAUSE_TX	[15:0]	rcs-cor-sc	**Reserved.**
FRAME64_TX.	**A:0D18h**		**64 Byte Frames Transmitted. Default: 0x00.00.00.00**
64B_TX	[31:0]	rcs-cor-sc	**64 Byte Frames Transmitted** = # transmitted frames with 64 bytes.
FRAME65_TX.	**A:0D1Ch**		**65 to 127 Byte Frames Transmitted. Default: 0x00.00.00.00**
65TO127B_TX	[31:0]	rcs-cor-sc	**65 to 127 Byte Frames Transmitted** = # transmitted frames with 65 to 127 bytes.
FRAME128_TX.	**A:0D20h**		**128 to 255 Byte Frames Transmitted. Default: 0x00.00.00.00**
128TO255B_TX	[31:0]	rcs-cor-sc	**128 to 255 Byte Frames Transmitted** = # transmitted frames with 128 to 255 bytes.
FRAME256_TX.	**A:0D24h**		**256 to 511 Byte Frames Transmitted. Default: 0x00.00.00.00**
256TO511B_TX	[31:0]	rcs-cor-sc	**256 to 511 Byte Frames Transmitted** = # transmitted frames with 256 to 511 bytes.
FRAME512_TX.	**A:0D28h**		**512 to 1023 Byte Frames Transmitted. Default: 0x00.00.00.00**
512TO1023B_TX	[31:0]	rcs-cor-sc	**512 to 1023 Byte Frames Transmitted** = # transmitted frames with 512 to 1023 bytes.
FRAME1024_TX.	**A:0D2Ch**		**1024 to 1518 Byte Frames Transmitted. Default: 0x00.00.00.00**
1024TO1518B_TX	[31:0]	rcs-cor-sc	**1024 to 1518 Byte Frames Transmitted** = # transmitted frames with 1024 to 1518 bytes.
FRAME1519_TX.	**A:0D30h**		**Greater Than 1518 Byte Frames Transmitted. Default: 0x00.00.00.00**
1519B_OR_MORE	[31:0]	rcs-cor-sc	**1519 Bytes or More Frames Transmitted** = # transmitted frames with > 1519-bytes**.**
STATS_TX_URUN.	**A:0D34h**		**Transmit Under Runs. Default: 0x00.00.00.00**
RSVD	[31:10]		**Reserved.**
TX_URUNS	[9:0]	rcs-cor-sc	**Reserved.**
STATS_SINGLE_COL.	**A:0D38h**		**Single Collision Frames. Default: 0x00.00.00.00**
RSVD	[32:18]		**Reserved.**
SINGLE_COL	[17:0]	rcs-cor-sc	**Reserved.**
STATS_MULTI_COL.	**A:0D3Ch**		**Multiple Collision Frames. Default: 0x00.00.00.00**
RSVD	[32:18]		**Reserved.**
MLT_COL	[17:0]	rcs-cor-sc	**Reserved.**
STATS_EXCESS_COL.	**A:0D40h**		**Excessive Collisions. Default: 0x00.00.00.00**
RSVD	[31:10]		**Reserved.**
EXC_COL	[9:0]	rcs-cor-sc	**Reserved.**
STATS_LATE_COL.	**A:0D44h**		**Late Collisions. Default: 0x00.00.00.00**
RSVD	[31:10]		**Reserved.**

M. Field Name	Addr (A:) Bit [x:y]	Type	Description
LATE_COL	[9:0]	rcs-cor-sc	**Reserved.**
STATS_DEF_TX.	**A:0D48h**		**Deferred Transmission Frames. Default: 0x00.00.00.00**
RSVD	[32:18]		**Reserved.**
DEF_TX_FRMS	[17:0]	rcs-cor-sc	**Reserved.**
STATS_CRS_ERRORS.	**A:0D4Ch**		**Carrier Sense Errors. Default: 0x00.00.00.00**
RSVD	[31:10]		**Reserved.**
CRS_ERRORS	[9:0]	rcs-cor-sc	**Reserved.**
OCT_RX_BOT.	**A:0D50h**		**Octets Received Bottom. Default: 0x00.00.00.00**
RX_OCTETS_FRM	[31:0]	rcs-cor-sc	**Received Octets in Frame [31:0]** = # octets in received frames (48-bit count using OCT_RX_BOT and OCT_RX_TOP). This count does not include octets for frames discarded by enabled MAC discard functions (e.g. packet length > 1536 bytes). OCT_RX_BOT should be read before OCT_RX_TOP.
OCT_RX_TOP.	**A:0D54h**		**Octets Received Top. Default: 0x00.00.00.00**
RSVD	[31:16]		**Reserved.**
RX_OCTETS_FRM	[15:0]	rcs-cor-sc	**Received Octets in Frame [47:32].** (see OCT_RX_BOT)
STATS_FRAMES_RX.	**A:0D58h**		**Frames Received. Default: 0x00.00.00.00**
FRMS_RX	[31:0]	rcs-cor-sc	**Frames Received** = # received frames, not including frames discarded by enabled MAC discard functions.
BROADCAST_RX.	**A:0D5Ch**		**Broadcast Frames Received. Default: 0x00.00.00.00**
BRDCST_RX	[31:0]	rcs-cor-sc	**Broadcast Frames Received** = # received Ethernet Broadcast frames, not including frames discarded by enabled MAC discard functions.
MULTICAST_RX.	**A:0D60h**		**Multicast Frames Received. Default: 0x00.00.00.00**
MLTCST_RX	[31:0]	rcs-cor-sc	**Multicast Frames Received** = # received Ethernet Multicast frames, not including frames discarded by enabled MAC discard functions.
STATS_PAUSE_RX.	**A:0D64h**		**Pause Frames Received. Default: 0x00.00.00.00**
RSVD	[31:16]		**Reserved.**
PAUSE_RX	[15:0]	rcs-cor-sc	**Reserved.**
FRAME64_RX.	**A:0D68h**		**64 Byte Frames Received. Default: 0x00.00.00.00**
64B_RX	[31:0]	rcs-cor-sc	**64 Byte Frames Received** = # received frames with 64 bytes, not including frames discarded by enabled MAC discard functions.
FRAME65_RX.	**A:0D6Ch**		**65 to 127 Byte Frames Received. Default: 0x00.00.00.00**
65TO127B_RX	[31:0]	rcs-cor-sc	**65 to 127 Byte Frames Received** = # received frames with 65 to 127 bytes, not including frames discarded by enabled MAC discard functions.
FRAME128_RX.	**A:0D70h**		**128 to 255 Byte Frames Received. Default: 0x00.00.00.00**
128TO255B_RX	[31:0]	rcs-cor-sc	**128 to 255 Byte Frames Received** = # received frames with 128 to 255 bytes, not including frames discarded by enabled MAC discard functions.
FRAME256_RX.	**A:0D74h**		**256 to 511 Byte Frames Received. Default: 0x00.00.00.00**
256TO511B_RX	[31:0]	rcs-cor-sc	**256 to 511 Byte Frames Received** = # received frames with 256 to 511 bytes, not including frames discarded by enabled MAC discard functions.

M. Field Name	Addr (A:) Bit [x:y]	Type	Description
FRAME512_RX.	**A:0D78h**		**512 to 1023 Byte Frames Received. Default: 0x00.00.00.00**
512TO1023B_RX	[31:0]	rcs-cor-sc	**512 to 1023 Byte Frames Received** = # received frames with 512 to 1023 bytes, not including frames discarded by enabled MAC discard functions.
FRAME1024_RX.	**A:0D7Ch**		**1024 to 1518 Byte Frames Received. Default: 0x00.00.00.00**
1024TO1518B_RX	[31:0]	rcs-cor-sc	**1024 to 1518 Byte Frames Received** = # received frames with 1024 to 1518 bytes, not including frames discarded by enabled MAC discard functions.
FRAME1519_RX.	**A:0D80h**		**1519 to Maximum Byte Frames Received. Default: 0x00.00.00.00**
1519B_OR_MORE_RX	[31:0]	rcs-cor-sc	**1519 Bytes or More Frames Received** = # received frames with > 1518 bytes, not including frames discarded by enabled MAC discard functions.
STATS_USIZE_FRAMES.	**A:0D84h**		**Undersized Frames Received. Default: 0x00.00.00.00**
RSVD	[31:10]		**Reserved.**
USIZE_RX	[9:0]	rcs-cor-sc	**Undersized Frames Received** = # received frames with < 64 bytes, not including frames with an Ethernet FCS error or an alignment error.
STATS_EXCESS_LEN.	**A:0D88h**		**Oversized Frames Received. Default: 0x00.00.00.00**
RSVD	[31:10]		**Reserved.**
OSIZE_RX	[9:0]	rcs-cor-sc	**Oversized Frames Received** = # received frames with more than 1518 or 1536 bytes as specified by M.NET_CONFIG.RX_1536FRMS. This count does not include frames that have either a CRC error, an alignment error or a receive symbol error.
STATS_JABBERS.	**A:0D8Ch**		**Jabbers Received. Default: 0x00.00.00.00**
RSVD	[31:10]		**Reserved.**
JAB_RX	[9:0]	rcs-cor-sc	**Jabbers Received** = # received frames with more than 1518 or 1536 bytes, as specified by M.NET_CONFIG.RX_1536FRMS, and that also include a CRC error, an alignment error or a receive symbol error.
STATS_FCS_ERRORS.	**A:0D90h**		**Frame Check Sequence Errors. Default: 0x00.00.00.00**
RSVD	[31:10]		**Reserved.**
FCS_ERR	[9:0]	rcs-cor-sc	**Frame Check Sequence Errors** = # received frames with FCS errors and a length between 64 and 1518 bytes (1536 if RX_1536FRMS = 1).
STATS_LENGTH_ERRORS.	**A:0D94h**		**Length Field Frame Errors. Default: 0x00.00.00.00**
RSVD	[31:10]		**Reserved.**
LGTH_FRM_ERR	[9:0]	rcs-cor-sc	**Length Field Frame Errors** = # received frames with an Ethernet Length field error and a measured length between 64 and 1518 bytes (1536 bytes if RX_1536FRMS = 1).
STATS_RX_SYM_ERR.	**A:0D98h**		**Receive Symbol Errors. Default: 0x00.00.00.00**
RSVD	[31:10]		**Reserved.**

M. Field Name	Addr (A:) Bit [x:y]	Type	Description
RX_SYM_ERR	[9:0]	rcs-cor-sc	**Received Symbol Errors** = # received frames with input pin RX_ER = 1 during reception. For the 100 Mb/s mode Symbol Errors are counted regardless of the frame length. For the 1000 Gb/s mode the frame must satisfy the Ethernet Slot Time requirements to be counted as a Symbol Error. Receive Symbol Errors are also counted as an FCS Error or an Alignment Error if the frame is between 64 and 1518 bytes (1536 bytes if RX_1536FRMS = 1). If the frame is larger it is also counted as a jabber error. If the frame is too small it is also counted as an Undersized Error.
STATS_ALIGN_ERRORS.	**A:0D9Ch**		**Alignment Errors. Default: 0x00.00.00.00**
RSVD	[31:10]		**Reserved.**
ALIGN_ERR	[9:0]	rcs-cor-sc	**Alignment Errors** = # received frames with a length that is not an integral number of bytes, has a bad FCS when the length is truncated to the nearest integral number of bytes and the integral number of bytes is between 64 and 1518 bytes (1536 bytes if RX_1536FRMS = 1).
STATS_RX_RES_ERR.	**A:0DA0h**		**Receive Resource Errors. Default: 0x00.00.00.00**
RSVD	[32:18]		**Reserved.**
RX_RES_ERR	[17:0]	rcs-cor-sc	**Reserved.**
STATS_RX_ORUN.	**A:0DA4h**		**Receive Overruns. Default: 0x00.00.00.00**
RSVD	[31:10]		**Reserved.**
RX_ORUNS	[9:0]	rcs-cor-sc	**Reserved.**
IP_HDR_CHK.	**A:0DA8h**		**IP Header Checksum Errors. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
IP_HDR_CHK	[7:0]	rcs-cor-sc	**Reserved.**
TCP_CHK.	**A:0DACh**		**TCP Checksum Errors. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
TCP_CHK	[7:0]	rcs-cor-sc	**Reserved.**
UDP_CHK.	**A:0DB0h**		**UDP Checksum Errors. Default: 0x00.00.00.00**
RSVD	[31:8]		**Reserved.**
UDP_CHK	[7:0]	rcs-cor-sc	**Reserved.**
RSVD.	**A:0E00h**		**Reserved.**
RSVD	[31:0]		**Reserved.**
REG_TOP.	**A:0E3Ch**		**Reserved.**
RSVD	[31:0]		**Reserved.**

10.3.13 TXP SW CAS Registers (TXSCn.)

Table 10-28. TXP SW CAS Registers (TXSCn.)

TXSCn Field Name	Addr (A:) Bit [x:y]	Type	Description
CR1.	**A:1000h**		**Configuration Register 1. Default: na**
CTS0	[31:28]	rwd-_-_	**CAS Time Slot 0** = Timeslot 0 TXP Bundle Conditioning SW CAS code.
CTS1	[27:24]	rwd-_-_	**CAS Time Slot 1** = Timeslot 1 TXP Bundle Conditioning SW CAS code.
CTS2	[23:20]	rwd-_-_	**CAS Time Slot 2** = Timeslot 2 TXP Bundle Conditioning SW CAS code.
CTS3	[19:16]	rwd-_-_	**CAS Time Slot 3** = Timeslot 3 TXP Bundle Conditioning SW CAS code.

TXSCn Field Name	Addr (A:) Bit [x:y]	Type	Description
CTS4	[15:12]	rwd-_-_	**CAS Time Slot 4** = Timeslot 4 TXP Bundle Conditioning SW CAS code.
CTS5	[11:8]	rwd-_-_	**CAS Time Slot 5** = Timeslot 5 TXP Bundle Conditioning SW CAS code.
CTS6	[7:4]	rwd-_-_	**CAS Time Slot 6** = Timeslot 6 TXP Bundle Conditioning SW CAS code.
CTS7	[3:0]	rwd-_-_	**CAS Time Slot 7** = Timeslot 7 TXP Bundle Conditioning SW CAS code.
CR2.	**A:1004h**		**Configuration Register 2. Default: na**
CTS8	[31:28]	rwd-_-_	**CAS Time Slot 8** = Timeslot 8 TXP Bundle Conditioning SW CAS code.
CTS9	[27:24]	rwd-_-_	**CAS Time Slot 9** = Timeslot 9 TXP Bundle Conditioning SW CAS code.
CTS10	[23:20]	rwd-_-_	**CAS Time Slot 10** = Timeslot 10 TXP Bundle Conditioning SW CAS code.
CTS11	[19:16]	rwd-_-_	**CAS Time Slot 11** = Timeslot 11 TXP Bundle Conditioning SW CAS code.
CTS12	[15:12]	rwd-_-_	**CAS Time Slot 12** = Timeslot 12 TXP Bundle Conditioning SW CAS code.
CTS13	[11:8]	rwd-_-_	**CAS Time Slot 13** = Timeslot 13 TXP Bundle Conditioning SW CAS code.
CTS14	[7:4]	rwd-_-_	**CAS Time Slot 14** = Timeslot 14 TXP Bundle Conditioning SW CAS code.
CTS15	[3:0]	rwd-_-_	**CAS Time Slot 15** = Timeslot 15 TXP Bundle Conditioning SW CAS code.
CR3.	**A:1008h**		**Configuration Register 3. Default: na**
CTS16	[31:28]	rwd-_-_	**CAS Time Slot 16** = Timeslot 16 TXP Bundle Conditioning SW CAS code.
CTS17	[27:24]	rwd-_-_	**CAS Time Slot 17** = Timeslot 17 TXP Bundle Conditioning SW CAS code.
CTS18	[23:20]	rwd-_-_	**CAS Time Slot 18** = Timeslot 18 TXP Bundle Conditioning SW CAS code.
CTS19	[19:16]	rwd-_-_	**CAS Time Slot 19** = Timeslot 19 TXP Bundle Conditioning SW CAS code.
CTS20	[15:12]	rwd-_-_	**CAS Time Slot 20** = Timeslot 20 TXP Bundle Conditioning SW CAS code.
CTS21	[11:8]	rwd-_-_	**CAS Time Slot 21** = Timeslot 21 TXP Bundle Conditioning SW CAS code.
CTS22	[7:4]	rwd-_-_	**CAS Time Slot 22** = Timeslot 22 TXP Bundle Conditioning SW CAS code.
CTS23	[3:0]	rwd-_-_	**CAS Time Slot 23** = Timeslot 23 TXP Bundle Conditioning SW CAS code.
CR4.	**A:100Ch**		**Configuration Register 4. Default: na**
CTS24	[31:28]	rwd-_-_	**CAS Time Slot 24** = Timeslot 24 TXP Bundle Conditioning SW CAS code.
CTS25	[27:24]	rwd-_-_	**CAS Time Slot 25** = Timeslot 25 TXP Bundle Conditioning SW CAS code.
CTS26	[23:20]	rwd-_-_	**CAS Time Slot 26** = Timeslot 26 TXP Bundle Conditioning SW CAS code.
CTS27	[19:16]	rwd-_-_	**CAS Time Slot 27** = Timeslot 27 TXP Bundle Conditioning SW CAS code.
CTS28	[15:12]	rwd-_-_	**CAS Time Slot 28** = Timeslot 28 TXP Bundle Conditioning SW CAS code.
CTS29	[11:8]	rwd-_-_	**CAS Time Slot 29** = Timeslot 29 TXP Bundle Conditioning SW CAS code.
CTS30	[7:4]	rwd-_-_	**CAS Time Slot 30** = Timeslot 30 TXP Bundle Conditioning SW CAS code.
CTS31	[3:0]	rwd-_-_	**CAS Time Slot 31** = Timeslot 31 TXP Bundle Conditioning SW CAS code.

10.3.14 Xmt (RXP) SW CAS Registers (RXSCn.)

Table 10-29. Xmt (RXP) SW CAS Registers (RXSCn.)

RXSCn. Field Name	Addr (A:) Bit [x:y]	Type	Description
CR1.	**A:1200h**		**Configuration Register 1. Default: na**
CTS0	[31:28]	rwd-_-_	**CAS Time Slot 0** = Timeslot 0 RXP Bundle Conditioning SW CAS code.
CTS1	[27:24]	rwd-_-_	**CAS Time Slot 1** = Timeslot 1 RXP Bundle Conditioning SW CAS code.
CTS2	[23:20]	rwd-_-_	**CAS Time Slot 2** = Timeslot 2 RXP Bundle Conditioning SW CAS code.
CTS3	[19:16]	rwd-_-_	**CAS Time Slot 3** = Timeslot 3 RXP Bundle Conditioning SW CAS code.
CTS4	[15:12]	rwd-_-_	**CAS Time Slot 4** = Timeslot 4 RXP Bundle Conditioning SW CAS code.
CTS5	[11:8]	rwd-_-_	**CAS Time Slot 5** = Timeslot 5 RXP Bundle Conditioning SW CAS code.

RXSCn. Field Name	Addr (A:) Bit [x:y]	Type	Description
CTS6	[7:4]	rwd-_-_	**CAS Time Slot 6** = Timeslot 6 RXP Bundle Conditioning SW CAS code.
CTS7	[3:0]	rwd-_-_	**CAS Time Slot 7** = Timeslot 7 RXP Bundle Conditioning SW CAS code.
CR2.	**A:1204h**		**Configuration Register 2. Default: na**
CTS8	[31:28]	rwd-_-_	**CAS Time Slot 8** = Timeslot 8 RXP Bundle Conditioning SW CAS code.
CTS9	[27:24]	rwd-_-_	**CAS Time Slot 9** = Timeslot 9 RXP Bundle Conditioning SW CAS code.
CTS10	[23:20]	rwd-_-_	**CAS Time Slot 10** = Timeslot 10 RXP Bundle Conditioning SW CAS code.
CTS11	[19:16]	rwd-_-_	**CAS Time Slot 11** = Timeslot 11 RXP Bundle Conditioning SW CAS code.
CTS12	[15:12]	rwd-_-_	**CAS Time Slot 12** = Timeslot 12 RXP Bundle Conditioning SW CAS code.
CTS13	[11:8]	rwd-_-_	**CAS Time Slot 13** = Timeslot 13 RXP Bundle Conditioning SW CAS code.
CTS14	[7:4]	rwd-_-_	**CAS Time Slot 14** = Timeslot 14 RXP Bundle Conditioning SW CAS code.
CTS15	[3:0]	rwd-_-_	**CAS Time Slot 15** = Timeslot 15 RXP Bundle Conditioning SW CAS code.
CR3.	**A:1208h**		**Configuration Register 3. Default: na**
CTS16	[31:28]	rwd-_-_	**CAS Time Slot 16** = Timeslot 16 RXP Bundle Conditioning SW CAS code.
CTS17	[27:24]	rwd-_-_	**CAS Time Slot 17** = Timeslot 17 RXP Bundle Conditioning SW CAS code.
CTS18	[23:20]	rwd-_-_	**CAS Time Slot 18** = Timeslot 18 RXP Bundle Conditioning SW CAS code.
CTS19	[19:16]	rwd-_-_	**CAS Time Slot 19** = Timeslot 19 RXP Bundle Conditioning SW CAS code.
CTS20	[15:12]	rwd-_-_	**CAS Time Slot 20** = Timeslot 20 RXP Bundle Conditioning SW CAS code.
CTS21	[11:8]	rwd-_-_	**CAS Time Slot 21** = Timeslot 21 RXP Bundle Conditioning SW CAS code.
CTS22	[7:4]	rwd-_-_	**CAS Time Slot 22** = Timeslot 22 RXP Bundle Conditioning SW CAS code.
CTS23	[3:0]	rwd-_-_	**CAS Time Slot 23** = Timeslot 23 RXP Bundle Conditioning SW CAS code.
CR4.	**A:120Ch**		**Configuration Register 4. Default: na**
CTS24	[31:28]	rwd-_-_	**CAS Time Slot 24** = Timeslot 24 RXP Bundle Conditioning SW CAS code.
CTS25	[27:24]	rwd-_-_	**CAS Time Slot 25** = Timeslot 25 RXP Bundle Conditioning SW CAS code.
CTS26	[23:20]	rwd-_-_	**CAS Time Slot 26** = Timeslot 26 RXP Bundle Conditioning SW CAS code.
CTS27	[19:16]	rwd-_-_	**CAS Time Slot 27** = Timeslot 27 RXP Bundle Conditioning SW CAS code.
CTS28	[15:12]	rwd-_-_	**CAS Time Slot 28** = Timeslot 28 RXP Bundle Conditioning SW CAS code.
CTS29	[11:8]	rwd-_-_	**CAS Time Slot 29** = Timeslot 29 RXP Bundle Conditioning SW CAS code.
CTS30	[7:4]	rwd-_-_	**CAS Time Slot 30** = Timeslot 30 RXP Bundle Conditioning SW CAS code.
CTS31	[3:0]	rwd-_-_	**CAS Time Slot 31** = Timeslot 31 RXP Bundle Conditioning SW CAS code.

10.3.15 TDM Port n Registers (Pn.; n = 0 to 31)

10.3.15.1 Port n Transmit Configuration Registers (Pn.)

Table 10-30. Port n Transmit Configuration Registers (Pn.)

Pn. Field Name	Addr (A:) Bit [x:y]	Type	Description
PTCR1.	**A:2000h**		**Port Transmit Configuration Register 1. Default: 81.FC.00.00h**
DR	[31]	rwc-_-_	**Reserved.**
RSVD	[30:29]		**Reserved.**
SFS	[28]	rwc-_-_	**Structured Format Select** selects the transmit TDM Port Structure type. 0 = unstructured format (no framing; for SAT or HDLC applications) 1 = structured format (with T1 or E1 framing; for CES or HDLC applications)

Pn. Field Name	Addr (A:) Bit [x:y]	Type	Description
FFS	[27]	rwc-_-_	**Frame Format Select** selects the transmit TDM Port Frame type (only valid if SFS = 1). 0 = E1 frame structure 1 = T1 frame structure
MFS	[26:25]	rwc-_-_	**Multiframe Format Select** selects the transmit TDM Port CAS multi-frame type (only valid if SFS = 1). 0 = No multiframe structure 1 = E1 multiframe structure (16 frame multiframe structure) 2 = T1 - SF multiframe structure (12 frame multiframe structure) 3 = T1 – ESF multiframe structure (24 frame multiframe structure)
BFD	[24:23]	rwc-_-_	**Buffer Frame/Fragment Depth** selects the number of 125 us periods of TDM data internally buffered by the S132 for the CES/SAT engines (see PTCR1.BPF). The number of bytes specified by BFD * BPF must be ≤ B.BCDR1.PMS for all RXP Bundles assigned to this TDM Port. 0 = Disable SAT/CES data path for transmit TDM Port 1 = 1 Frame/Fragment per staging buffer (125 us staging buffer) 2 = 2 Frame/Fragments per staging buffer (250 us staging buffer) 3 = 4 Frame/Fragments per staging buffer (500 us staging buffer)
BPF	[22:18]	rwc-_-_	**Bytes Per Frame/Fragment** = # bytes transmitted from TDM Port during a 125 us period (125 us = time period for 1 CES Frame or 1 SAT Fragment). For T1, BPF = 23 (0x17; SAT/CES). For E1, BPF = 31 (0x1F; SAT/CES). 0 = 1 byte per 125 us period 31 = 32 bytes per 125 us period
DP	[17]	rwc-_-_	**Decap Priority** selects the RXP (Decap) SAT/CES/HDLC processing priority 0 = low priority (processed after all high priority data has been completed) 1 = high priority
DOSOT	[16]	rwc-_-_	**Disable Overwrite Signaling On TDAT** = "1" disables the S132 from over-writing CAS codes in the transmit TDM Port TDAT pin, T1/E1 CAS code positions (T1 robbed-bit signaling and E1 Timeslot 16). When DOSOT = 0, the S132 over-writes those TDAT time positions. DOSOT does not affect the transmit TDM Port TSIG data (when a TDM Port is programmed to transmit CAS the CAS codes are transmitted on TSIG regardless of the DOSOT setting).
RSVD	[15:0]		**Reserved.**
PTCR2.	**A:2004h**		**Port Transmit Configuration Register 2. Default: 00.00.00.08h**
RSVD	[31:8]		**Reserved.**
PRPTLL	[9]	rwc-_-_	**Port Receive to Port Transmit Line Loopback** = "1" enables the TDM Port Line Loopback from RDAT, RSYNC, RSIG to TDAT, TSYNC, TSIG respectively. The transmit timing is not automatically changed when PRPTLL = 1. Pn.PTCR2.TSS must be programmed so the incoming RDAT data stream timing is used to time TCLKO and RCLK. PRPTLL = 1 over-rides Pn.PTCR2.TDS forcing TSYNC to be an output. 0 = TDAT/TSYNC/TSIG pass data from RXP packets (normal) 1 = TDAT/TSYNC/TSIG loopback data from receive inputs RDAT/RSYNC/RSIG
TIOE	[8]	rwc-_-_	**Transmit Input Output Enable.** 0 = TDAT/TSIG/TSYNC disabled (high-impedance) 1 = TDAT/TSIG/TSYNC enabled (Pn.PTCR2.TDS selects TSYNC direction)
TCE	[7]	rwc-_-_	**Transmit Clock Enable.** 0 = TCLKO disabled (high-impedance) 1 = TCLKO enabled with timing source selected by Pn.PTCR2.TSS
TSRS	[6]	rwc-_-_	**Transmit Frame Timing Synchronized to RSYNC.** 0 = Transmit Frame Timing synchronized to RSYNC input 1 = Transmit Frame Timing synchronized to TSYNC input (when TDS = 0)

Pn. Field Name	Addr (A:) Bit [x:y]	Type	Description
TDS	[5]	rwc-_-_	**TSYNC Direction Select.** (only valid when TIOE = 1). 0 = TSYNC is an input 1 = TSYNC is an output
TOES	[4]	rwc-_-_	**Transmit Output Edge Select.** 0 = TDM Port TDAT/TSIG/TSYNC outputs timed using TCLKO positive edge 1 = TDM Port TDAT/TSIG/TSYNC outputs timed using TCLKO negative edge
TIES	[3]	rwc-_-_	**Transmit Input Edge Select.** (only valid when TDS = 1) 0 = TDM Port TSYNC input timed using TCLKO positive edge 1 = TDM Port TSYNC input timed using TCLKO negative edge
TSS	[2:0]	rwc-_-_	**TCLKO Source Select** selects the timing source for TCLKO. 0 = RCLKn pin input signal 1 = internal aclk_n signal (port "n" recovered clock) 2 = internal grclk signal (globally selected recovered clock) 4 = EXTCLK[0] pin input signal 5 = EXTCLK[1] pin input signal all other values are reserved
PTCR3.	**A:2008h**		**Port Transmit Configuration Register 3. Default: 0x00.00.00.00**
PRPTTSL	[31:0]	rwc-_-_	**PR to PT Time Slot Loop** = "1" enables the TDM Port Timeslot Loopback from RDAT, RSYNC, RSIG to TDAT, TSYNC, TSIG respectively (1 bit for each timeslot; for T1, bits 31:24 are not used). The transmit timing is not automatically changed when PRPTLL = 1. Pn.PTCR2.TSS must be programmed so the incoming RDAT data stream timing is used to time TCLKO and RCLK. PRPTLL = 1 over-rides Pn.PTCR2.TDS forcing TSYNC to be an output. Any number of timeslots can be in loopback while others Timeslot are not in loopback, but the receive TDM Port timing and all RXP packet data streams must be frequency synchronized to work error free. This loopback is only valid if Pn.PTCR1.SFS = 1 (CES). 0 = pass Timeslot data from RXP packets to TDAT/TSYNC/TSIG (normal) 1 = loopback Timeslot data from RDAT/RSYNC/RSIG to TDAT/TSYNC/TSIG

10.3.15.2 Port n Transmit Status Registers (Pn.)

Table 10-31. Port n Transmit Status Registers (Pn.)

Pn. Field Name	Addr (A:) Bit [x:y]	Type	Description
PTSR1.	**A:2020h**		**Port Transmit Status Register 1. Default: 0x00.00.00.00**
CTS0	[31:28]	ros-_-_	**CAS Time Slot 0 =** value of CAS code transmitted at TDM Port for Timeslot 0.
CTS1	[27:24]	ros-_-_	**CAS Time Slot 1 =** value of CAS code transmitted at TDM Port for Timeslot 1.
CTS2	[23:20]	ros-_-_	**CAS Time Slot 2 =** value of CAS code transmitted at TDM Port for Timeslot 2.
CTS3	[19:16]	ros-_-_	**CAS Time Slot 3 =** value of CAS code transmitted at TDM Port for Timeslot 3.
CTS4	[15:12]	ros-_-_	**CAS Time Slot 4 =** value of CAS code transmitted at TDM Port for Timeslot 4.
CTS5	[11:8]	ros-_-_	**CAS Time Slot 5 =** value of CAS code transmitted at TDM Port for Timeslot 5.
CTS6	[7:4]	ros-_-_	**CAS Time Slot 6 =** value of CAS code transmitted at TDM Port for Timeslot 6.
CTS7	[3:0]	ros-_-_	**CAS Time Slot 7 =** value of CAS code transmitted at TDM Port for Timeslot 7.
PTSR2.	**A:2024h**		**Port Transmit Status Register 2. Default: 0x00.00.00.00**
CTS8	[31:28]	ros-_-_	**CAS Time Slot 8 =** value of CAS code transmitted at TDM Port for Timeslot 8
CTS9	[27:24]	ros-_-_	**CAS Time Slot 9 =** value of CAS code transmitted at TDM Port for Timeslot 9
CTS10	[23:20]	ros-_-_	**CAS Time Slot 10 =** value of CAS code transmitted at TDM Port for Timeslot 10.
CTS11	[19:16]	ros-_-_	**CAS Time Slot 11 =** value of CAS code transmitted at TDM Port for Timeslot 11.
CTS12	[15:12]	ros-_-_	**CAS Time Slot 12 =** value of CAS code transmitted at TDM Port for Timeslot 12.

Pn. Field Name	Addr (A:) Bit [x:y]	Type	Description
CTS13	[11:8]	ros-_-_	**CAS Time Slot 13 =** value of CAS code transmitted at TDM Port for Timeslot 13.
CTS14	[7:4]	ros-_-_	**CAS Time Slot 14 =** value of CAS code transmitted at TDM Port for Timeslot 14.
CTS15	[3:0]	ros-_-_	**CAS Time Slot 15 =** value of CAS code transmitted at TDM Port for Timeslot 15.
PTSR3.	**A:2028h**		**Port Transmit Status Register 3. Default: 0x00.00.00.00**
CTS16	[31:28]	ros-_-_	**CAS Time Slot 16 =** value of CAS code transmitted at TDM Port for Timeslot 16.
CTS17	[27:24]	ros-_-_	**CAS Time Slot 17 =** value of CAS code transmitted at TDM Port for Timeslot 17.
CTS18	[23:20]	ros-_-_	**CAS Time Slot 18 =** value of CAS code transmitted at TDM Port for Timeslot 18.
CTS19	[19:16]	ros-_-_	**CAS Time Slot 19 =** value of CAS code transmitted at TDM Port for Timeslot 19.
CTS20	[15:12]	ros-_-_	**CAS Time Slot 20 =** value of CAS code transmitted at TDM Port for Timeslot 20.
CTS21	[11:8]	ros-_-_	**CAS Time Slot 21** = value of CAS code transmitted at TDM Port for Timeslot 21
CTS22	[7:4]	ros-_-_	**CAS Time Slot 22** = value of CAS code transmitted at TDM Port for Timeslot 22.
CTS23	[3:0]	ros-_-_	**CAS Time Slot 23** = value of CAS code transmitted at TDM Port for Timeslot 23.
PTSR4.	**A:202Ch**		**Port Transmit Status Register 4. Default: 0x00.00.00.00**
CTS24	[31:28]	ros-_-_	**CAS Time Slot 24** = value of CAS code transmitted at TDM Port for Timeslot 24.
CTS25	[27:24]	ros-_-_	**CAS Time Slot 25** = value of CAS code transmitted at TDM Port for Timeslot 25.
CTS26	[23:20]	ros-_-_	**CAS Time Slot 26** = value of CAS code transmitted at TDM Port for Timeslot 26.
CTS27	[19:16]	ros-_-_	**CAS Time Slot 27** = value of CAS code transmitted at TDM Port for Timeslot 27.
CTS28	[15:12]	ros-_-_	**CAS Time Slot 28** = value of CAS code transmitted at TDM Port for Timeslot 28.
CTS29	[11:8]	ros-_-_	**CAS Time Slot 29** = value of CAS code transmitted at TDM Port for Timeslot 29.
CTS30	[7:4]	ros-_-_	**CAS Time Slot 30** = value of CAS code transmitted at TDM Port for Timeslot 30.
CTS31	[3:0]	ros-_-_	**CAS Time Slot 31** = value of CAS code transmitted at TDM Port for Timeslot 31.

10.3.15.3 Port n Transmit Status Register Latches (Pn.)

Table 10-32. Port n Transmit Status Register Latches (Pn.)

Pn. Field Name	Addr (A:) Bit [x:y]	Type	Description
PTSRL.	**A:2030h**		**Port Transmit Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:2]		**Reserved.**
BUSL	[1]	rls-crw-i3	**Buffer Underrun Status Latch** = "1" indicates the transmit TDM Port ran out of data. The S132 internal transmit processes were not able to keep up with the transmit rate for this TDM Port.
COFASL	[0]	rls-crw-i3	**Change Of Frame Alignment Status Latch** = "1" indicates a change of frame or multi-frame timing error was detected (only valid for SFS = 1).

10.3.15.4 Port n Transmit Status Register Interrupt Enables (Pn.)

Table 10-33. Port n Transmit Status Register Interrupt Enables (Pn.)

Pn. Field Name	Addr (A:) Bit [x:y]	Type	Description
PTSRIE.	**A:2038h**		**Port Transmit Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:2]		**Reserved.**
BUIE	[1]	rwc-_-i3	**Buffer Underrun Interrupt Enable.** The combination of PTSRL.BUSL = 1 and BUIE = 1 forces G.GSR1.PS = 1.
COFAIE	[0]	rwc-_-i3	**Change Of Frame Alignment Interrupt Enable.** The combination of PTSRL.COFASL = 1 and COFAIE = 1 forces G.GSR1.PS = 1.

10.3.15.5 Port n Receive Configuration Registers (Pn.)

Table 10-34. Port n Receive Configuration Registers (Pn.)

Pn. Field Name	Addr (A:) Bit [x:y]	Type	Description
PRCR1.	**A:2040h**		**Port Receive Configuration Register 1. Default: 81.FC.00.00h**
DR	[31]	rwc-_-_	**Datapath Reset.** When this bit is set, it will force the internal data path registers in the corresponding port receive interface to their default state. This bit must be set high for a minimum of 100ns. See section 10.3 Reset And Power Down. 0 = Normal operation 1 = Force all data path registers to their default values
RSVD	[30:29]		**Reserved.**
SFS	[28]	rwc-_-_	**Structured Format Select.** This bit selects structured or unstructured formatting. Unstructured format is used for SAT and unstructured HDLC. Structured format is used for CES and structured HDLC. 0 = unstructured format 1 = structured format
FFS	[27]	rwc-_-_	**Frame Format Select.** This bit selects the frame format for the port receive. 0 = E1 frame select 1 = T1 frame select
MFS	[26:25]	rwc-_-_	**Multiframe Format Select.** Used to determine the type of multiframe format being used. The CAS machine uses this to determine when data may be captured and passed to the packet interface. Additionally, the RSYNC uses this to know the multiframe frame size for aligning the frame and the multiframe counters. Note that this register has no affect in unstructured modes. 0 = none No multiframes. 1 = E1 MF 16 2 = T1 SF 12 3 = T1 ESF 24
BFD	[24:23]	rwc-_-_	**Buffer Frame Depth.** Used to indicate the number of frames per segment. It is also used to indicate the port has been disabled. When the frames of the segment are filled, the PRDME is toggled. 0 = Disable Request to Encap 1 = 1 frame per segment 2 = 2 frame per segment 3 = 4 frame per segment
BPF	[22:18]	rwc-_-_	**Bytes Per Frame.** This is used to select the number of bytes to capture for unstructured modes. For T1 this must be set to 17h and for E1 should be 1Fh. 00h = 1 byte captured 01h = 2 bytes captured ... 17h = 24 bytes captured ... 1Fh = 32 bytes captured
EP	[17]	rwc-_-_	**Encap Priority.** This is used to prioritize processing of this port by the encap engine. If more than one port has this bit set, then all of the high priority ports are processed first, followed by the low priority ports. 0 = low priority for encap processing 1 = high priority for encap processing
CS	[16]	rwc-_-_	**CAS Source** selects the receive T1/E1 Port CAS Signaling Source. 0 = RDAT pin 1 = RSIG pin
CBVSE	[15]	rwc-_-_	**C Bit Value for SF to ESF.** Sets the value of the C bit when mapping SF locally to ESF at the destination.

Pn. Field Name	Addr (A:) Bit [x:y]	Type	Description
DBVSE	[14]	rwc-_-_	**D Bit Value for SF to ESF.** Sets the value of the D bit when mapping SF locally to ESF at the destination.
LB	[13]	rwc-_-_	**L Bit.** This sets the L bit value for all Bundles sourced by this port.
LBSS	[12]	rwc-_-_	**L Bit Source Select** selects the L-bit source for all TXP Bundles for this T1/E1 port: 0 = PRCR1.LB value or 1 = L-bit programmed in each TXP Bundle header.
SPL	[11:1]	rwc-_-_	**SAT Payload Length.** Set to the # bytes per packet payload (SAT mode only). SPL must = PMS and must be ≥ BPF. For example for T1 SAT, SPL = PMS = 0x17 (for 24 timeslots) and BPF must be set to 0x17 or less.
RSVD	[0]		**Reserved.**
PRCR2.	**A:2044h**		**Port Receive Configuration Register 2. Default: 00.00.00.08h**
RSVD	[31:7]		**Reserved.**
RSTS	[6]	rwc-_-_	**Receive Frame Synchronization .** 0 = synchronized to RSYNC signal input 1 = synchronized to internal TDM Port Transmit frame timing (system timing)
RDS	[5]	rwc-_-_	**RSYNC Direction Select.** This bit selects the direction of the RSYNC signal. 0 = input 1 = output
RSVD	[4]		**Reserved.**
RIES	[3]	rwc-_-_	**Receive Input Edge Select.** This bit selects the edge to be used for port receive data capture on inputs relative to RCLK. 0 = positive edge 1 = negative edge
RSVD	[2:1]		**Reserved.**
RSS	[0]	rwc-_-_	**RCLK Source Select.** This bit is used to select the source of the clock used to time the port receive interface. This selects the source clock for capture of RDAT, RSIG, and RSYNC. 0 = RCLK Signal input 1 = TCLKO Signal output
PRCR3.	**A:2048h**		**Port Receive Configuration Register 3. Default: 0x00.00.00.00**
PTPRTSL	[31:0]	rwc-_-_	**PT to PR Time Slot Loopback.** Each bit selects the TDM loopback for the corresponding time slot from the port transmit to the port receive; bit 0 enables port loop back for time slot 0, bit 1 enables port loop back for time slot 1, etc. You may use either the loopback for PR to PT or PT to PR, but not both at the same time; i.e. the control for the unused direction must not have any timeslots selected for loopback. Note that for T1, bits 31:24 are not used.
PRCR4.	**A:204Ch**		**Port Receive Configuration Register 4. Default: 0x00.00.00.00**
RSVD	[31:17]		**Reserved.**
TSGMS	[16]	rwc-_-_	**RTP Time Stamp Generator Mode Select.** 0 = derived from CMNCLK (Differential Timestamp) 1 = derived from RSS selected receive TDM Port timing (Absolute Timestamp)
TSGMC	[15:0]	rwc-_-_	**Timestamp Generator M Coefficient** is defined by the following equation where TSPCLK = "remote PW Timestamp clock rate" (TSPCLK and CMNCLK are specified in bits/sec; only valid for TSGMS = 0). In most applications TSPCLK = CMNCLK and TSGMC = 4096 decimal = 0x1000. TSGMC = Integer [4096 * (TSPCLK ÷ CMNCLK)]
PRCR5.	**A:2050h**		**Port Receive Configuration Register 5. Default: 0x00.00.00.00**
RSVD	[31:29]		**Reserved.**
TSGN1C	[28:16]	rwc-_-_	**Timestamp Generator N1 Coefficient** is defined by the following equation (see TSGMC). In most applications TSPCLK = CMNCLK and TSGN1C = 0x0000. TSGN1C = (CMNCLK ÷ 8000) * [TSGMC – 4096 * (TSPCLK ÷ CMNCLK)]

Pn. Field Name	Addr (A:) Bit [x:y]	Type	Description
RSVD	[15:13]		**Reserved.**
TSGN0C	[12:0]	rwc-_-_	**Timestamp Generator N0 Coefficient** is defined by the following equation (see TSGN1C). In most applications TSPCLK = CMNCLK and TSGN0C = CMNCLK ÷ 8000 (e.g. if TSGN0C = CMNCLK = 2.048 Mb/s then TSGN0C = 256 = 0x0100). TSGN0C = TSGN1C + (CMNCLK ÷ 8000)

10.3.15.6 Port n Receive Status Registers (Pn.)

Table 10-35. Port n Receive Status Registers (Pn.)

Pn. Field Name	Addr (A:) Bit [x:y]	Type	Description
PRSR1.	**A:2060h**		**Port Receive Status Register 1. Default: 0x00.00.00.00**
CTS0	[31:28]	ros-_-_	**CAS Time Slot 0 =** CAS code received at TDM Port (Pn.PRCR2.CS) for Timeslot 0.
CTS1	[27:24]	ros-_-_	**CAS Time Slot 1 =** CAS code received at TDM Port (Pn.PRCR2.CS) for Timeslot 1
CTS2	[23:20]	ros-_-_	**CAS Time Slot 2 =** CAS code received at TDM Port (Pn.PRCR2.CS) for Timeslot 2
CTS3	[19:16]	ros-_-_	**CAS Time Slot 3 =** CAS code received at TDM Port (Pn.PRCR2.CS) for Timeslot 3
CTS4	[15:12]	ros-_-_	**CAS Time Slot 4 =** CAS code received at TDM Port (Pn.PRCR2.CS) for Timeslot 4
CTS5	[11:8]	ros-_-_	**CAS Time Slot 5 =** CAS code received at TDM Port (Pn.PRCR2.CS) for Timeslot 5
CTS6	[7:4]	ros-_-_	**CAS Time Slot 6 =** CAS code received at TDM Port (Pn.PRCR2.CS) for Timeslot 6
CTS7	[3:0]	ros-_-_	**CAS Time Slot 7 =** CAS code received at TDM Port (Pn.PRCR2.CS) for Timeslot 7
PRSR2.	**A:2064h**		**Port Receive Status Register 2. Default: 0x00.00.00.00**
CTS8	[31:28]	ros-_-_	**CAS Time Slot 8 =** CAS code received at TDM Port (Pn.PRCR2.CS) for Timeslot 8
CTS9	[27:24]	ros-_-_	**CAS Time Slot 9 =** CAS code received at TDM Port (Pn.PRCR2.CS) for Timeslot 9
CTS10	[23:20]	ros-_-_	**CAS Time Slot 10 =** CAS code received at TDM Port (Pn.PRCR2.CS) for TS 10
CTS11	[19:16]	ros-_-_	**CAS Time Slot 11 =** CAS code received at TDM Port (Pn.PRCR2.CS) for TS 11
CTS12	[15:12]	ros-_-_	**CAS Time Slot 12 =** CAS code received at TDM Port (Pn.PRCR2.CS) for TS 12
CTS13	[11:8]	ros-_-_	**CAS Time Slot 13 =** CAS code received at TDM Port (Pn.PRCR2.CS) for TS 13
CTS14	[7:4]	ros-_-_	**CAS Time Slot 14 =** CAS code received at TDM Port (Pn.PRCR2.CS) for TS 14
CTS15	[3:0]	ros-_-_	**CAS Time Slot 15 =** CAS code received at TDM Port (Pn.PRCR2.CS) for TS 15
PRSR3.	**A:2068h**		**Port Receive Status Register 3. Default: 0x00.00.00.00**
CTS16	[31:28]	ros-_-_	**CAS Time Slot 16 =** CAS code received at TDM Port (Pn.PRCR2.CS) for TS 16
CTS17	[27:24]	ros-_-_	**CAS Time Slot 17 =** CAS code received at TDM Port (Pn.PRCR2.CS) for TS 17
CTS18	[23:20]	ros-_-_	**CAS Time Slot 18 =** CAS code received at TDM Port (Pn.PRCR2.CS) for TS 18
CTS19	[19:16]	ros-_-_	**CAS Time Slot 19 =** CAS code received at TDM Port (Pn.PRCR2.CS) for TS 19
CTS20	[15:12]	ros-_-_	**CAS Time Slot 20 =** CAS code received at TDM Port (Pn.PRCR2.CS) for TS 20
CTS21	[11:8]	ros-_-_	**CAS Time Slot 21** = CAS code received at TDM Port (Pn.PRCR2.CS) for TS 21
CTS22	[7:4]	ros-_-_	**CAS Time Slot 22** = CAS code received at TDM Port (Pn.PRCR2.CS) for TS 22
CTS23	[3:0]	ros-_-_	**CAS Time Slot 23** = CAS code received at TDM Port (Pn.PRCR2.CS) for TS 23
PRSR4.	**A:206Ch**		**Port Receive Status Register 4. Default: 0x00.00.00.00**
CTS24	[31:28]	ros-_-_	**CAS Time Slot 24** = CAS code received at TDM Port (Pn.PRCR2.CS) for TS 24
CTS25	[27:24]	ros-_-_	**CAS Time Slot 25** = CAS code received at TDM Port (Pn.PRCR2.CS) for TS 25
CTS26	[23:20]	ros-_-_	**CAS Time Slot 26** = CAS code received at TDM Port (Pn.PRCR2.CS) for TS 26
CTS27	[19:16]	ros-_-_	**CAS Time Slot 27** = CAS code received at TDM Port (Pn.PRCR2.CS) for TS 27
CTS28	[15:12]	ros-_-_	**CAS Time Slot 28** = CAS code received at TDM Port (Pn.PRCR2.CS) for TS 28
CTS29	[11:8]	ros-_-_	**CAS Time Slot 29** = CAS code received at TDM Port (Pn.PRCR2.CS) for TS 29

Pn. Field Name	Addr (A:) Bit [x:y]	Type	Description
CTS30	[7:4]	ros-_-_	**CAS Time Slot 30** = CAS code received at TDM Port (Pn.PRCR2.CS) for TS 30
CTS31	[3:0]	ros-_-_	**CAS Time Slot 31** = CAS code received at TDM Port (Pn.PRCR2.CS) for TS 31

10.3.15.7 Port n Receive Status Register Latches (Pn.)

Table 10-36. Port n Receive Status Register Latches (Pn.)

Pn. Field Name	Addr (A:) Bit [x:y]	Type	Description
PRSRL.	**A:2070h**		**Port Receive Status Register Latch. Default: 0x00.00.00.00**
RSVD	[31:2]		**Reserved.**
BOSL	[1]	rls-crw-i3	**Buffer Overrun Status Latch** = "1" indicates the receive TDM Port received data faster than it could be processed (TXP payload data was lost).
COFASL	[0]	rls-crw-i3	**Change Of Frame Alignment Status Latch** = "1" indicates a change of frame or multi-frame timing error was detected (only valid for SFS = 1).

10.3.15.8 Port n Receive Status Register Interrupt Enables (Pn.)

Table 10-37. Port n Receive Status Register Interrupt Enables (Pn.)

Pn. Field Name	Addr (A:) Bit [x:y]	Type	Description
PRSRIE.	**A:2078h**		**Port Receive Status Register Interrupt Enable. Default: 0x00.00.00.00**
RSVD	[31:2]		**Reserved.**
BOIE	[1]	rwc-_-i3	**Buffer Overrun Interrupt Enable.** The combination of BOIE = 1 and PRSRL.BOSL = 1 forces G.GSR1.PS = 1.
COFAIE	[0]	rwc-_-i3	**Change Of Frame Alignment Interrupt Enable.** The combination of and COFAIE = 1 and PRSRL.COFASL = 1 forces G.GSR1.PS = 1.

10.3.16 Timeslot Assignment Registers (TSAn.m.; "n" = TDM Port n; "m" = Timeslot m)

Table 10-38. Timeslot Assignment Registers (TSAn.m.; "n" = TDM Port n; "m" = Timeslot m)

TSAn.m. Field Name	Addr (A:) Bit [x:y]	Type	Description
CR.	**A[1]:3000h +n*0020h +m*0004h**		**Configuration Register. Default: na (SRAM unknown values after reset)**
RSVD	[31:17]		**Reserved.**
TSAS	[16]	rwd-_-_	**Timeslot Assigned Select** = "1" = TDM Port "n" Timeslot "m" is assigned to the Bundle # specified by BNS ("0" = unassigned/unused).
RSVD	[15:8]		**Reserved.**
BNS	[7:0]	rwd-_-_	**Bundle Number Select** = Bundle # for TDM Port "n", Timeslot "m" (for TSAS = 1).

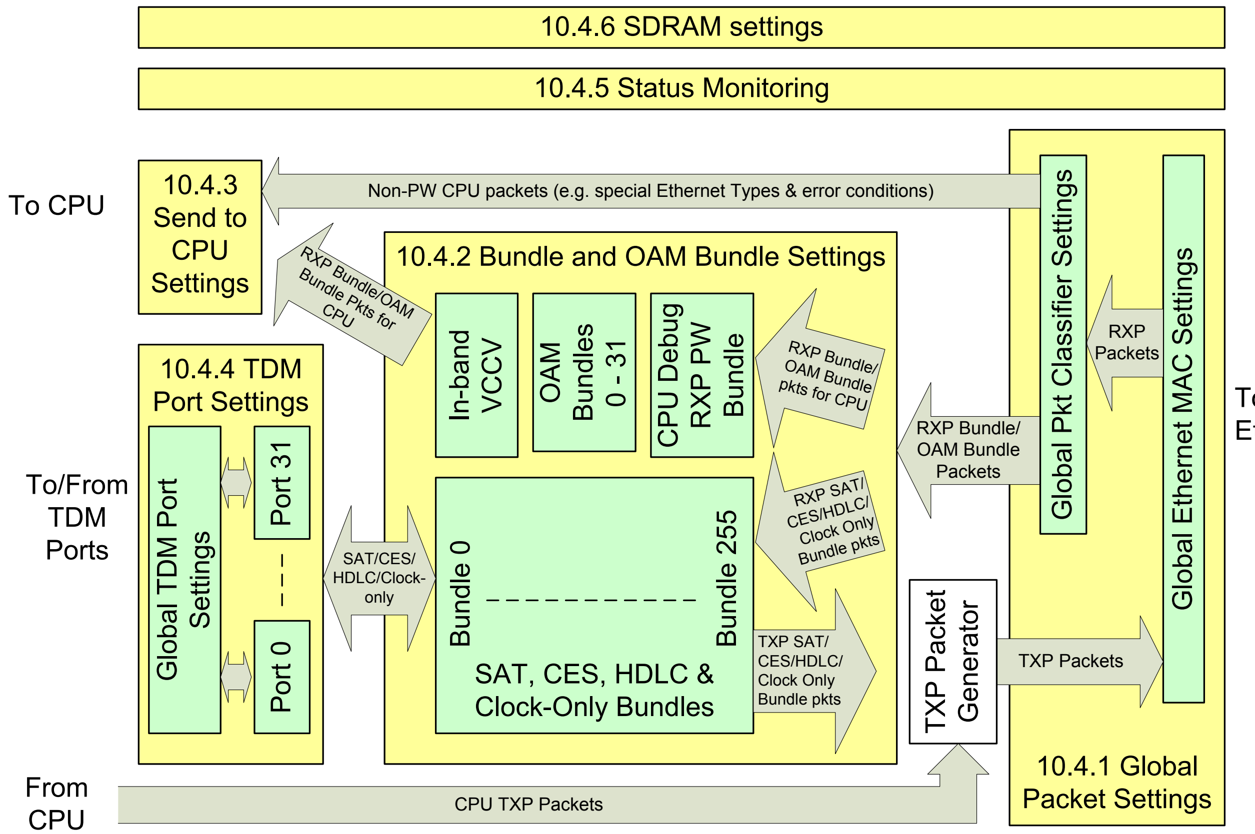
Note: [1] There are 1024 TSAn.m. registers (32 TDM Ports * 32 Timeslots = 1024). The TSAn.m. address = 3000h+ (n*0020h + m*0004h) where the TDM Port "n" varies from 0 to 0x1F and the TS "m" varies from 0 to 0x1F. In binary this can viewed as 11.00P_4P_3.$P_2P_1P_0T_4$.$T_3T_2T_1T_0$ where $P_4P_3P_2P_1P_0$ = 5-bit TDM Port # (0 – 31 decimal) and $T_4T_3T_2T_1T_0$ = 5-bit TS # (0 – 31 decimal; T1 does not use the values 24 – 31). For an Unstructured TDM Port (SAT or HDLC) TS 0 must be assigned.

10.4 Register Guide

The Register Guide Section provides example settings for some of the more common applications, especially for applications in which one register setting determines which settings are valid for other related registers. The S132 registers and their functional operation cannot be fully understood without also reading the Functional Description and Register Definition sections. When those two sections are understood this section enables an S132 user to quickly identify interactions and settings that must be made for particular applications.

Figure 10-1 provides a high level view of how the Register Guide sub-sections relate to each other. The arrows depict the flow of RXP and TXP packet data. The boxes each represent one of the Register Guide sub-sections and give a high level view of how the sub-sections relate to each other.

Figure 10-1. Register Guide High Level Diagram



Throughout this section example register values are presented as decimal values except when the "0x" notation is used to identify a hex value (e.g. 0x17) or when the letter "b" follows a "1" or "0" to indicate a binary value (e.g. "10b"). This means that a "5" when indicated for a 3-bit register field equates to "101" binary (register bits are always programmed using binary equivalent values). Register bit numbers, paragraph text and equations are always indicated using decimal values (e.g. for "bit 10", the value "10" is a decimal value).

An "x" value (by itself) is used in the tables that follow to indicate "any valid value". In many cases the only "valid values" are listed in the "Comment" column of the table. If the "Comment column" does not provide specific values, then "any" value is legal. Dark shading and/or "NA" are used to identify rows or cells within each table that are "Not Applicable" to the identified application. When a Write register bit is identified in this way, the "0" value should be written to that register unless specified otherwise. When a Read register bit is identified in this way, the returned register bit value should be ignored.

10.4.1 Global Packet Settings

Table 10-39. Global Ethernet MAC (M.) Control Register Settings (Values are in hex)

Bit #	Register Bit Name	r/w	Val	Comments
M.NET_CONTROL		**- Network Control Register**		
10	TX_HALT	wo	x	TXP Transmit Halt (wait to finish if packet already started)
9	START_TX	wo	x	Start TXP Transmission
4	MAN_PORT_EN	rw	x	MDIO Management Port Enable
3	TX_EN	rw	x	TXP Transmit Enable (immediate)
2	RX_EN	rw	x	Receive Enable (immediate)
NET_CONFIG		**- Network Configuration Register**		
25	EN_FRMS_HDUP	rw	0	Reserved
22:21	DATBUS_WIDTH	rw	0	Reserved
20:18	MDC_CLK_DIV	rw	4	Reserved
17	FCS_REMOVE	rw	1	Reserved
16	LGTH_FRM_DIS	rw	1	Discard Frames with Length Field Errors
15:14	RX_BUF_OFFSET	rw	0	Reserved
13	PAUSE_EN	rw	0	Receive Pause Enable
10	GIG_MODE_EN	rw	x	Select MAC Interface type: 0 = MII I/F (10/100 Mbps); 1 = GMII I/F (1 Gbps)
9	EXT_AMATCHEN	rw	0	Reserved
8	RX_1536FRMS	rw	x	Maximum Receive Frame Size: 0 = 1518 bytes; 1 = 1536 bytes
5	NO_BROADCAST	rw	x	Discard Ethernet Broadcast Frames
4	COPY_FRMS	rw	?	Forward all valid Ethernet Frames (disregard filter settings)
3	JUMBO_FRMS	rw	0	Reserved
2	DISC_NONVLAN	rw	x	Discard frames that do not include VLAN tags
1	FULL_DUPLEX	rw	1	Enable Full Duplex
0	SPEED	rw	1	Reserved
PHY_MAN		**- Phy Maintenance Register**		
31	PHY_SET3	rw	0	Reserved
30	PHY_CL22	rw	1	Reserved
29:28	PHY_SET2	rw	x	MDIO Operation: 2 = Read; 1 = Write.
27:23	PHY_ADDR	rw	x	MDIO Phy Address
22:18	PHY_REG_ADDR	rw	x	MDIO Register Address (register address in Phy that is selected by PHY_ADDR)
17:16	PHY_SET1	rw	2	Reserved
15:0	PHY_DATA_WR	rw	x	“Write data sent to PHY” or “Read data from PHY” according to the selected PHY_SET2 operation

Notes: “s” = Status; “x” = any valid value; r = Read; w = Write; “wo” = “Write Only”; “Val” = “Value”.

Table 10-40. Global Ethernet Packet Classification (PC.) Settings

Register	Functional Description	Comments
Ethernet		
CR17 - CR19	Ethernet DA1 and DA2	The S132 can recognize up to 2 programmed Ethernet DAs.
CR1.DBTP	Broadcast TDMoP Pkt Discard	TDMoP packets with the Broadcast DA: continue processing (0) or discard(1)
CR1.DBCP	Broadcast CPU Pkt Discard	Non-TDMoP packets with the Broadcast DA: continue processing (0) or discard(1)
CR1.DPS9	Unknown Ethernet DA Discard	Packets with DA ≠ DA1, DA2 or Broadcast DA: send to CPU (0) or discard (1)
CR3.VITPID	VLAN Inner Tag Protocol ID	Packets with 1 or 2 VLAN Tags must use this TPID in the Inner Tag position.
CR3.VOTPID	VLAN Outer Tag Protocol ID	Packets with 2 VLAN Tags must use this TPID in the Outer Tag position.
CR1.DPS2	Unknown Ethernet Type Discard	Packets with unknown Ethernet Type: send to CPU (0) or discard (1)
IPv4 & IPv6		
CR1.RXPIVS & CR1.RXPDSD	IP Version	Select “Only IPv4” (0x1), “Only IPv6” (0x3) or “both IPv4 and IPv6” (0x0)
CR6 - CR8	IPv4 Destination Address 1 - 3	The S132 can recognize up to 3 CPU configured IPv4 DAs.
CR9 – CR16	IPv6 Destination Address 1 - 2	The S132 can recognize up to 2 CPU configured IPv6 DAs.
CR1.DPS1	Unknown IP DA Discard	Send to CPU (0) or Discard (1).
CR1.DPS4	Unknown IP Protocol Discard	Packets with unknown IP Protocol: send to CPU (0) or discard (1)
CR1.DICPE	Bad IPv4 Checksum	Ignore Bad Checksum and Forward pkt (0) or Discard pkt (1).
All PW Protocols (MEF-8, MPLS, IP/UDP and IP/L2TPv3)		
CR1.DPS6	Unknown PW-ID Discard	PW packets with unknown PW-ID: send to CPU (0) or discard (1)
CR1.DPS7	OAM Discard	Send to CPU (0) or Discard (1) MEF OAM, In-band VCCV and OAM BIDs
MEF-8		
CR4.MET	MEF Ether Type	Identify pkt with this Ether Type as a MEF-8 TDMoP pkt. Default = 0x88D8.
CR4.MOET	MEF OAM Ether Type	Identify pkt with this Ether Type as a MEF-8 OAM (CPU) pkt.
MPLS		
CR1.DPS10	>2 MPLS Outer Label Discard	Send to CPU (0) or Discard (1).
UDP[1]		
CR1.UBIDLS	UDP BID Global Location Select	0 = Test UDP pkt for 16-bit BID based on UBIDLCE setting and test UDP pkt for 16-bit OAM BID in either UDP Source or Destination Port location 1 = Test UDP pkt for 16-bit BID/OAM BID match in UDP Destination Port 2 = Test UDP pkt for 16-bit BID/OAM BID match in UDP Source Port 3 = Test UDP pkt for 32-bit BID/OAM BID match in Source and Dest. Port
CR1.UBIDLCE	UDP BID per-Bundle Location Select (only for UBIDLS=0)	0 = Auto detect = Test UDP pkt for 16-bit BID match in UDP Source or Dest. Port 1 = Test UDP pkt for 16-bit BID in UDP Port selected by B.BCDR4.RXUBIDLS
CR20.UBIDM	UDP BID Mask	UDP BID/OAM BID bit Mask (0 = ignore bit; 1 = test bit; 0xFFFF = test all bits)
CR1.UPVCE	UDP Protocol Check En (valid for UBIDLS ≠ 3)	0 = Ignore UDP Protocol Type 1 = Process UDP pkt with BID but UDP Protocol ≠ UPVC1/2 according to DPS5
CR1.DPS5	Unknown UDP Protocol Type Discard (valid for UPVCE = 1)	0 = Send to CPU, UDP pkt with BID but UDP Protocol ≠ UPVC1/2 1 = Discard UDP pkt with BID match but UDP Protocol ≠ UPVC1/2
CR1.DUCPE	UDP Checksum Error	For UDP pkt with a checksum error: Ignore checksum (0) or Discard packet (1).
CR2.UPVC1 & CR2.UPVC2	UDP Protocol Type 1 - 2	UDP Protocol Type values for when UPVCE is enabled (default 0x085E).
L2TPv3 (there are no specialized Global L2TPv3 settings)		
ARP		
CR1.DPS3	ARP with Known IP DA	Send to CPU (0) or Discard (1) ARP packets with known IPv4 DA.
CR1.DPS0	ARP with Unknown IP DA	Send to CPU (0) or Discard (1) ARP packets with unknown IPv4 DA.
CPU Destination Ethernet Type		
CR20.CDET	MEF OAM Ether Type	Identify pkt with this Ether Type as CPU Destination Ethernet Type.
CR1.DPS8	CPU Dest. Ether Type Discard	Send to CPU (0) or Discard (1) packets with CPU Destination Ethernet Type

Note: [1] The interactions between the various UDP settings are further described in Table 10-41.

Table 10-41. Valid UDP BID Location and UDP Protocol Type Settings

UDP BID Location Test Mode		BID Test Settings			Protocol Test Settings		UDP Protocol Type Test Location
		PC.CR1 UBIDLS	PC.CR1 UBIDLCE	B.BCDR4. RXUBIDLS	PC.CR1. UPVCE	PC.CR1. DPS5	
1 A	All Bundles: "16-bit auto discover"	0	0	0	0	0	UDP Protocol is Ignored
					1	0/1	For BID & OAM BID: 16-bit auto-discover[2]
B	Per-Bundle setting: "16-bit Source Port" [1]	0	1	0	0	0	UDP Protocol is Ignored
					1	0/1	For BID: 16-bit Destination Port For OAM BID: auto-discover[2]
	Per-Bundle setting: "16-bit Destination Port" [1]	0	1	1	0	0	UDP Protocol is Ignored
					1	0/1	For BID: 16-bit Source Port For OAM BID: auto-discover[2]
C	All Bundles: "16-bit Destination Port"	1	0	0	0	0	UDP Protocol is Ignored
					1	0/1	For BID & OAM BID: 16-bit Source Port
D	All Bundles: "16-bit Source Port"	2	0	0	0	0	UDP Protocol is Ignored
					1	0/1	For BID & OAM BID: 16-bit Destination Port
E	All Bundles: "32-bit"	3	0	0	0	0	UDP Protocol is Ignored

Notes: [1] The BID test location for the Per-Bundle tests are programmed per Bundle using B.BCDR4.RXUBIDLS.
[2] The BID is auto discovered and the UPVC1/UPVC2 test is performed on the "other" UDP Port position.

10.4.2 Bundle and OAM Bundle Settings

Table 10-42. Bundle and OAM Bundle Control Registers (B.)

Register	Bits	Functional Description	Comments
Bundle Reset Control			
BRCR1	SNS	Sequence Number Seed	**RESET Bundle:** To Reset an RXP Bundle payload data path, first select the Bundle reset direction ("RXP only", "TXP only", "RXP and TXP" or "none") using RXBRE and TXBRE (1 = reset; 0 = release = no reset). When the Bundle number (0-255) is written to RXTXBS, the Bundle will be reset. The Reset Status can be monitored using RXBRS and TXBRS. This function is not used with OAM Bundles. **Release Bundle:** To Release a TXP Bundle payload data path from Reset, first select the direction using RXBRE and TXBRE (0 = release; 1 = reset = do not release). When the Bundle # is written to RXTXBS and Sequence Seed to SNS, the Bundle is ready with a new Sequence Seed value waiting to be activated. A Bundle's Status Registers are not enabled until the Bundle is released from Reset.
	RXTXBS	Bundle # to be Reset	
BRCR2	RXBRE	RXP Bundle Reset Enable	
	TXBRE	TXP Bundle Reset Enable	
BRSR	RXBRS	RXP Bundle Reset Status	
	TXBRS	TXP Bundle Reset Status	
Bundle Activation Control			
BACR	OBS	OAM Bundle Select	**Assign Bundle ID (PWID):** To Assign a Bundle ID to a Bundle, first program the Bundle ID using BIDV. Then use OBS = 0 and BS to select the Bundle Number (0 – 255). The BIDV value will be written to that Bundle Number when the WE transitions from "0 to 1". **Bundle Activate State:** To Activate or De-activate a Bundle ID, first program the Activate state using ABE. Then use OBS = 0 and BIDV to select the Bundle Number (0 – 255). The Activate state will be written to that Bundle Number when WE transitions from "0 to 1". All Bundles must be released from Reset after a Power up/Reset before they can be Activated.
	WE	Write Enable	
	RE	Read Enable	
	BS	Bundle Number	
BADR1	ABE	Activate Bundle	
BADR2	BIDV	Bundle ID	
Bundle Configuration Control			
BCCR	WE	Write Enable	**Configure Bundle Attributes:** To configure the attributes of a Bundle, first program all of the attributes in B.BCDR1 through B.BCDR5. Then use BS to specify the Bundle Number (0-255). The new set of attributes will be written to that Bundle when WE transitions from "0 to 1". The BCDR1 – BCDR5 settings are described in Bundle Configuration tables (that follow) according to the application.
	RE	Read Enable	
	BS	Bundle Number	
BCDR1-5		Misc Bundle Functions	

In the Bundle Configuration tables that follow (for B.BCDR1 through B.BCDR5): "x" = "any valid value"; for the column titles, "M" = "MPLS", "U" = "UDP", "L" = "L2TPv3", "E" = "MEF". Values not identified in the Comment column are invalid. Values included in "[ ]" brackets means "recommended value", but other values in the comment column are possible. The "RT" column indicates whether the configuration register is used in the "RXP only", "TXP only" or "RXP and TXP" directions.

10.4.2.1 SAT Bundle Settings

Table 10-43. SAT Bundle Settings

Reg-bit	Bit Name	RT	M	U	L	E	Bit Name Description	Comments
BCDR1								
23	LBCAI	R	x	x	x	x	L Bit Conditioning Auto Insert	1 = Discard payload if "L-bit = 1"; 0 = disable
22:21	PMT	RT	3	3	3	3	Payload Machine Type	3 = SAT/CES Payload Machine Type
20:10	PMS	RT	x	x	x	x	RXP & TXP Payload Monitored Size in bytes	For T1, PCT = 1 ms: PMS = 0x0C1 (193 bytes decimal) For E1, PCT = 1 ms: PMS = 0x100 (256 bytes decimal) For 256 Kb/s, PCT = 1 ms: PMS = 0x020 (32 bytes decimal)
9	SCSCFPD	R	[0]	[0]	[0]	[0]	SAT/CES Sanity Check	1 = Discard if rcvd pkt ≠ PMS; 0 = do not test against PMS
8	SCSNRE	R	[1]	[1]	[1]	[1]	SAT/CES Seq # Reorder En	0 = Disable Reordering; 1 = Enable Reordering
7	SCRXBCSS	NA	0	0	0	0	CES RXP CAS Source Select	NA
6	SCTXBCSS	NA	0	0	0	0	CES TXP CAS Source Select	NA
5	RSNS	R	[0]	[0]	[0]	[0]	Reorder Seq Number Select	0 = Control Word Sequence #; 1 = RTP Sequence #
4	SCTXCE	T	x	x	x	x	SAT/CES TXP Condition En	0 = normal; 1 = use TXP Conditioning data
3	SCTXDFSE	NA	0	0	0	0	CES T1 TXP Framing	NA
2:0	SCTXCOS	T	x	x	x	x	SAT/CES TXP Cond. Octet	Select 1 of 8 TXP Conditioning Octets
BCDR2								
31:0	ATSS[1]	RT	1	1	1	1	Active Timeslot Select	0x0000.0001
BCDR3								
4:3	TXPMS	T	x	x	x	x	TXP Packet Mode Select	0 = Disable; 1 = Xmt with payload; 2 = Xmt without payload
2:1	TXBTS	T	0	0	0	0	TXP Bundle Type	0 = SAT for Unstructured TDM Port
0	TXBPS	T	x	x	x	x	TXP Bundle Priority	0=low priority (normal); 1=high (for PW Timing Connections)
BCDR4								
21	RXRE	R	x	x	x	x	RXP RTP Enable	0 = RTP is not included; 1 = RTP is required
20	RXCWE	R	1	1	1	1	RXP Control Word Enable	1 = Control Word is required
19:18	RXHTS	R	0	1	2	3	RXP Header Type Select	0 = MPLS; 1 = UDP; 2 = L2TPv3; 3 = MEF
17:16	RXBTS	R	0	0	0	0	RXP Bundle Type	0 = SAT for Unstructured TDM Port
15:14	RXLCS	R	x	0	x	0	RXP Label/Cookie Select	MPLS: 0x1 = 1 Label; 0x2 = 2 Labels; 0x3 = 3 Labels L2TPV3: 0x0 = 0 Cookies; 0x1 = 1 Cookie; 0x2 = 2 Cookies
13	RXUBIDLS	R	0	[1]	0	0	RXP UDP BID Location	0 = UDP Source Port; 1 = UDP Destination Port
12	SCLVI	R	[0]	[0]	[0]	[0]	SAT/CES Last Value Insert	0 = disable last value insert; 1 = insert last value if pkt lost
11:9	RXCOS	R	x	x	x	x	Xmt (RXP) Conditioning Octet	Selects 1 of 8 Conditioning Octets for the transmit TDM Port
8	RXOICWE	R	[1]	[0]	[1]	[1]	RXP OAM in CW Enable	0 = ignore CW OAM indication; 1 = look for OAM indication
7:6	RXBDS	R	x	x	x	x	RXP Bundle Data Destination	0 = TDM Port; 3 = Discard (timing still available for ck recov)
5:1	PNS[1]	RT	x	x	x	x	TDM Port Number Select	Select TDM Port #0 - #31
0	PCRE	R	x	x	x	x	TDM Port Ck Recov. Enable	0 = do not use for Ck Recovery; 1 = use for Ck Recovery
BCDR5								
24:10	PDVT	R	x	x	x	x	PDV Tolerance	(see Table 10-44)
9:0	MJBS	R	x	x	x	x	Max Jitter Buffer Size	(see Table 10-44)

Note: [1] TSAn.m must be programmed to enable the port and timeslots selected by PNS (n = PNS) and ATSS (m = Timeslot).

Table 10-44. PMS/PDVT/MJBS for SAT with various PCT, PDV and BFD values

Example Applications					PMS		PDVT			MJBS		
Line Rate	Jitter Buffer Discard Method	Given Parameters			Settings		JB Fill	Settings		JB Fill	Settings	
		PCT	Tot PDV	BFD	Decimal	Hex	Level	Decimal	Hex	Level	Decimal	Hex
T1	"No Discard"	1 ms	5 ms	125 us	193	C1	10 ms	483	1E3	11 ms	17	11
	Limited Overrun	6 ms	10 ms	125 us	1,158	486	10 ms	483	1E3	16 ms	25	19
	Limited Underrun	20 ms	20 ms	125 us	3,860	F14	NA	1	1	40 ms	61	3D
E1	"No Discard"	1 ms	5 ms	125 us	256	100	10 ms	640	280	11 ms	22	16
	Limited Overrun	6 ms	10 ms	125 us	1,536	600	10 ms	640	280	16 ms	32	20
	Limited Underrun	20 ms	20 ms	125 us	5,120	1400	NA	1	1	40 ms	80	50
64 Kb/s	"No Discard"	1 ms	5 ms	125 us	8	8	10 ms	20	14	11 ms	1	1
	"Limited Overrun"	6 ms	10 ms	125 us	48	30	10 ms	20	14	16 ms	1	1
	"Limited Underrun"	20 ms	20 ms	125 us	160	A0	NA	1	1	40 ms	3	3

10.4.2.2 CES without CAS Bundle Settings

Table 10-45. CES without CAS Bundle Settings

Reg-bit	Bit Abbrev	RT	M	U	L	E	Bit Name Description	Comments
BCDR1								
23	LBCAI	R	x	x	x	x	L Bit Conditioning Auto Insert	1 = Discard payload if "L-bit = 1"; 0 = disable
22:21	PMT	RT	3	3	3	3	Payload Machine Type	3 = SAT/CES Payload Machine Type
20:10	PMS	RT	x	x	x	x	Payload Monitored Size	# of Frames of data in TXP & RXP pkt payload. For PCT = 1 ms: PMS = 0x008 (8 frames) For PCT = 8 ms: PMS = 0x040 (64 frames)
9	SCSCFPD	R	[0]	[0]	[0]	[0]	SAT/CES Sanity Check	1 = Discard if rcvd pkt ≠ PMS; 0 = do not test against PMS
8	SCSNRE	R	[1]	[1]	[1]	[1]	SAT/CES Seq # Reorder En	0 = Disable Reordering; 1 = Enable Reordering
7	SCRXBCSS	NA	0	0	0	0	CES RXP CAS Source Select	NA
6	SCTXBCSS	NA	0	0	0	0	CES TXP CAS Source Select	NA
5	RSNS	R	[0]	[0]	[0]	[0]	Reorder Seq Number Select	0 = Control Word Sequence #; 1 = RTP Sequence #
4	SCTXCE	T	x	x	x	x	SAT/CES TXP Conditioning	0 = normal; 1 = use TXP Conditioning data
3	SCTXDFSE	NA	0	0	0	0	CES T1 TXP Framing	NA
2:0	SCTXCOS	T	x	x	x	x	SAT/CES TXP Cond. Octet	Select 1 of 8 TXP Conditioning Octets
BCDR2								
31:0	ATSS[1]	RT	x	x	x	x	Active Timeslot Select	1b = included in Bundle (T1: TS #0 - 23; E1: TS #1 - 31)
BCDR3								
4:3	TXPMS	T	x	x	x	x	TXP Packet Mode Select	0 = Disable; 1 = Transmit with payload; 2 = Transmit without payload (for TDM Port faults)
2:1	TXBTS	T	1	1	1	1	TXP Bundle Type	1 = CES without CAS for Structured T1/E1 Port
0	TXBPS	T	x	x	x	x	TXP Bundle Priority	0=low priority (normal); 1=high (for PW Timing Connections)
BCDR4								
21	RXRE	R	x	x	x	x	RXP RTP Enable	0 = RTP is not included; 1 = RTP is required
20	RXCWE	R	1	1	1	1	RXP Control Word Enable	1 = Control Word is required
19:18	RXHTS	R	0	1	2	3	RXP Header Type Select	0 = MPLS; 1 = UDP; 2 = L2TPv3; 3 = MEF
17:16	RXBTS	R	1	1	1	1	RXP Bundle Type	1 = CES without CAS for Structured T1/E1 Port
15:14	RXLCS	R	x	0	x	0	RXP Label/Cookie Select	MPLS: 0x1 = 1 Label; 0x2 = 2 Labels; 0x3 = 3 Labels L2TPV3: 0x0 = 0 Cookies; 0x1 = 1 Cookie; 0x2 = 2 Cookies
13	RXUBIDLS	R	0	[1]	0	0	RXP UDP BID Location	0 = UDP Source Port; 1 = UDP Destination Port
12	SCLVI	R	[0]	[0]	[0]	[0]	SAT/CES Last Value Insert	0 = insert last value if pkt lost; 1 = disable last value insert
11:9	RXCOS	R	x	x	x	x	Xmt (RXP) Conditioning Octet	Selects 1 of 8 Conditioning Octets for the transmit TDM Port
8	RXOICWE	R	[1]	[0]	[1]	[1]	RXP OAM in CW Enable	0 = ignore CW OAM indication; 1 = look for OAM indication
7:6	RXBDS	R	x	x	x	x	RXP Bundle Data Destination	0 = TDM Port; 3 = Discard (timing still available for ck recov)
5:1	PNS[1]	RT	x	x	x	x	TDM Port Number Select	Select TDM Port #0 - #31
0	PCRE	R	x	x	x	x	TDM Port Ck Recov. Enable	0 = do not use for Ck Recovery; 1 = use for Ck Recovery
BCDR5								
24:10	PDVT	R	x	x	x	x	Packet Delay Variation Time	(see Table 10-46 for examples)
9:0	MJBS	R	x	x	x	x	Max Jitter Buffer Size	(see Table 10-46 for examples)

Note: [1] TSAn.m must be programmed to enable the port and timeslots selected by PNS (n = PNS) and ATSS (m = Timeslot).

Table 10-46. PMS/PDVT/MJBS for T1/E1 CES without CAS for various PCT, PDV and BFD values

Example Applications					PMS		PDVT			MJBS		
Line Rate	Jitter Buffer Discard Method	Given Timing Parameters			Settings		JB Fill	Settings		JB Fill	Settings	
		PCT	Tot PDV	BFD	Decimal	Hex	Level	Decimal	Hex	Level	Decimal	Hex
T1 or E1	"No Discard"	1 ms	5 ms	125 us	8	8	10 ms	80	14	11 ms	22	16
	"Limited Overrun"	6 ms	10 ms	125 us	48	30	10 ms	80	14	16 ms	32	20
	"Limited Underrun"	20 ms	20 ms	125 us	160	A0	NA	1	1	40 ms	80	50

10.4.2.3 CES with CAS Bundle Settings

Table 10-47. CES with CAS Bundle Settings

Reg-bit	Bit Abbrev	RT	M	U	L	E	Bit Name Description	Comments
BCDR1								
23	LBCAI	R	x	x	x	x	L Bit Conditioning Auto Insert	1 = Discard payload if "L-bit = 1"; 0 = disable
22:21	PMT	RT	3	3	3	3	Payload Machine Type	3 = SAT/CES Payload Machine Type
20:10	PMS	RT	x	x	x	x	Payload Monitored Size	# of Frames of data in TXP & RXP Packet Payload. For PCT = 1 ms: PMS = 0x008 (8 frames) For PCT = 8 ms: PMS = 0x040 (64 frames)
9	SCSCFPD	R	[0]	[0]	[0]	[0]	SAT/CES Sanity Check	1 = Discard if rcvd pkt ≠ PMS; 0 = do not test against PMS
8	SCSNRE	R	[1]	[1]	[1]	[1]	SAT/CES Seq # Reorder En	0 = Disable Reordering; 1 = Enable Reordering
7	SCRXBCSS	R	x	x	x	x	CES RXP CAS Source Select	0 = Use CAS from RXP Pkt; 1 = use Xmt SW CAS
6	SCTXBCSS	T	x	x	x	x	CES TXP CAS Source Select	0 = Use CAS from Rcv T1/E1 Port; 1 = use TXP SW CAS
5	RSNS	R	[0]	[0]	[0]	[0]	Reorder Seq Number Select	0 = Control Word Sequence #; 1 = RTP Sequence #
4	SCTXCE	T	x	x	x	x	SAT/CES TXP Conditioning	0 = normal; 1 = use TXP Conditioning data
3	SCTXDFSE	T	x	x	x	x	CES T1 TXP Framing	0 = SF Framing; 1 = ESF Framing (for E1 this is NA)
2:0	SCTXCOS	T	x	x	x	x	SAT/CES TXP Cond. Octet	Select 1 of 8 TXP Conditioning Octets
BCDR2								
31:0	ATSS[1]	RT	x	x	x	x	Active Timeslot Select	1b= included in Bundle (T1:TS #0-23; E1:TS #1-15 & 17-31)
BCDR3								
4:3	TXPMS	T	x	x	x	x	TXP Packet Mode Select	0 = Disable; 1 = Transmit with payload; 2 = Transmit without payload (for TDM Port faults)
2:1	TXBTS	T	2	2	2	2	TXP Bundle Type	2 = CES with CAS for Structured T1/E1 Port
0	TXBPS	T	x	x	x	x	TXP Bundle Priority	0=low priority (normal); 1=high (for PW Timing Connections)
BCDR4								
21	RXRE	R	x	x	x	x	RXP RTP Enable	0 = RTP is not included; 1 = RTP is required
20	RXCWE	R	1	1	1	1	RXP Control Word Enable	1 = Control Word is required
19:18	RXHTS	R	0	1	2	3	RXP Header Type Select	0 = MPLS; 1 = UDP; 2 = L2TPv3; 3 = MEF
17:16	RXBTS	R	2	2	2	2	RXP Bundle Type	2 = CES with CAS for Structured T1/E1 Port
15:14	RXLCS	R	x	0	x	0	RXP Label/Cookie Select	MPLS: 0x1 = 1 Label; 0x2 = 2 Labels; 0x3 = 3 Labels L2TPV3: 0x0 = 0 Cookies; 0x1 = 1 Cookie; 0x2 = 2 Cookies
13	RXUBIDLS	R	0	[1]	0	0	RXP UDP BID Location	0 = UDP Source Port; 1 = UDP Destination Port
12	SCLVI	R	[0]	[0]	[0]	[0]	SAT/CES Last Value Insert	0 = insert last value if pkt lost; 1 = disable last value insert
11:9	RXCOS	R	x	x	x	x	Xmt (RXP) Conditioning Octet	Selects 1 of 8 Conditioning Octets for the transmit TDM Port
8	RXOICWE	R	[1]	[0]	[1]	[1]	RXP OAM in CW Enable	0 = ignore CW OAM indication; 1 = look for OAM indication
7:6	RXBDS	R	x	x	x	x	RXP Bundle Data Destination	0 = TDM Port; 3 = Discard (timing still available for ck recov)
5:1	PNS[1]	RT	x	x	x	x	TDM Port Number Select	Select TDM Port #0 - #31
0	PCRE	R	x	x	x	x	TDM Port Ck Recov. Enable	0 = do not use for Ck Recovery; 1 = use for Ck Recovery
BCDR5								
24:10	PDVT	R	x	x	x	x	Packet Delay Variation Time	(see Table 10-48 for examples)
9:0	MJBS	R	x	x	x	x	Max Jitter Buffer Size	(see Table 10-48 for examples)

Note: [1] TSAn.m must be programmed to enable the port and timeslots selected by PNS (n = PNS) and ATSS (m = Timeslot).

Table 10-48. PMS/PDVT/MJBS for T1/E1 CES with CAS for various PCT, PDV and BFD values

Example Applications					PMS		PDVT			MJBS		
Line Rate	Jitter Buffer Discard Method	Given Timing Parameters			Settings		JB Fill	Settings		JB Fill	Settings	
		PCT	Tot PDV	BFD	Decimal	Hex	Level	Decimal	Hex	Level	Decimal	Hex
T1 or E1	"No Discard"	1 ms	5 ms	125 us	8	8	10 ms	80	14	11 ms	22	16
	"Limited Overrun"	6 ms	10 ms	125 us	48	30	10 ms	80	14	16 ms	32	20
	"Limited Underrun"	20 ms	20 ms	125 us	160	A0	NA	1	1	40 ms	80	50

10.4.2.4 Unstructured HDLC Bundle (any Line Rate) Settings

Table 10-49. Unstructured HDLC Bundle (any Line Rate) Settings

Reg-bit	Bit Abbrev	RT	M	U	L	E	Bit Name Description	Comments
BCDR1								
23	LBCAI	NA	0	0	0	0	L Bit Conditioning Auto Insert	NA
22:21	PMT	RT	0	0	0	0	Payload Machine Type	0 = HDLC Payload Machine Type
20:10	PMS	R	x	x	x	x	Payload Max Size	Maximum # of bytes in RXP Packet Payload (not incl. FCS)
9	SCSCFPD	NA	0	0	0	0	SAT/CES Sanity Check	NA
8	SCSNRE	RT	[0]	[0]	[0]	[0]	HDLC Bit Reorder Enable	0 = transmit MS bit first; 1 = transmit LS bit first
7	SCRXBCSS	RT	[1]	[1]	[1]	[1]	HDLC FCS Disable	0 = FCS enabled; 1 = FCS disabled
6	SCTXBCSS	RT	[1]	[1]	[1]	[1]	HDLC RXP FCS bit Width	0 = 16-bit; 1 = 32-bit
5	RSNS	NA	0	0	0	0	Reorder Seq Number Select	NA
4:3	SCTXCE/ SCTXDFSE	T	x	x	x	x	HDLC frame Seq # Mode	0 = Seq Num always 0; 1 = Wrap around using “0” 3 = Wrap around skipping “0”
2:0	SCTXCOS	RT	0	0	0	0	HDLC Channel Width Select	0 = Nx8-bit (Nx64 Kb/s)
BCDR2								
31:0	ATSS[1]	RT	1	1	1	1	Active Timeslot Select	0x0000.0001
BCDR3								
4:3	TXPMS	T	x	x	x	x	TXP Packet Mode Select	0 = Disable TXP Bundle; 1 = Enable TXP HDLC Bundle
2:1	TXBTS	T	0	0	0	0	TXP Bundle TDM Port Mode	0 = HDLC for Unstructured TDM Port
0	TXBPS	NA	0	0	0	0	TXP Bundle Priority	NA
BCDR4								
21	RXRE	R	x	x	x	x	RXP RTP Enable	0 = RTP is not included; 1 = RTP is required
20	RXCWE	R	[1]	[1]	[1]	[1]	RXP Control Word Enable	0 = Control Word is not included; 1 = CW is required
19:18	RXHTS	R	0	1	2	3	RXP Header Type Select	0 = MPLS; 1 = UDP; 2 = L2TPv3; 3 = MEF
17:16	RXBTS	R	0	0	0	0	RXP Bundle TDM Port Mode	0 = HDLC for Unstructured TDM Port
15:14	RXLCS	R	x	0	x	0	RXP Label/Cookie Select	MPLS: 0x1 = 1 Label; 0x2 = 2 Labels; 0x3 = 3 Labels L2TPV3: 0x0 = 0 Cookies; 0x1 = 1 Cookie; 0x2 = 2 Cookies
13	RXUBIDLS	R	0	[1]	0	0	RXP UDP BID Location	0 = UDP Source Port; 1 = UDP Destination Port
12	SCLVI	R	[0]	[0]	[0]	[0]	HDLC Inter-frame Fill	0 = 0x7E Inter-frame Fill; 1 = 0xFF Inter-frame Fill
11:9	RXCOS	NA	0	0	0	0	Xmt (RXP) Conditioning Octet	NA
8	RXOICWE	R	[1]	[0]	[1]	[1]	RXP OAM in CW Enable	0 = ignore CW OAM indication; 1 = look for OAM indication
7:6	RXBDS	R	x	x	x	x	RXP Bundle Data Destination	0 = TDM Port; 3 = Discard packet
5:1	PNS[1]	R	x	x	x	x	TDM Port Number Select	Select TDM Port #0 - #31
0	PCRE	R	0	0	0	0	TDM Port Ck Recov. Enable	0 = do not use for Ck Recov
BCDR5								
24:10	PDVT	NA	0	0	0	0	Packet Delay Variation Time	NA
9:0	MJBS	R	NA	NA	NA	NA	Max Jitter Buffer Size	NA

Note: [1] TSAn.m must be programmed to enable the port and timeslots selected by PNS (n = PNS) and ATSS (m = Timeslot).

10.4.2.5 Structured Nx64 Kb/s HDLC Bundle Settings

Table 10-50. Structured Nx64 Kb/s HDLC Bundle Settings

Reg-bit	Bit Abbrev	RT	M	U	L	E	Bit Name Description	Comments
BCDR1								
23	LBCAI	NA	0	0	0	0	L Bit Conditioning Auto Insert	NA
22:21	PMT	RT	0	0	0	0	Payload Machine Type	0 = HDLC Payload Machine Type
20:10	PMS	R	x	x	x	x	Payload Max Size	Maximum # of bytes in RXP Packet Payload (not incl. FCS)
9	SCSCFPD	NA	0	0	0	0	SAT/CES Sanity Check	NA
8	SCSNRE	RT	[0]	[0]	[0]	[0]	HDLC Bit Reorder Enable	0 = transmit MS bit first; 1 = transmit LS bit first
7	SCRXBCSS	RT	[1]	[1]	[1]	[1]	HDLC FCS Disable	0 = FCS enabled; 1 = FCS disabled
6	SCTXBCSS	RT	[1]	[1]	[1]	[1]	HDLC RXP FCS bit Width	0 = 16-bit; 1 = 32-bit
5	RSNS	NA	0	0	0	0	Reorder Seq Number Select	NA
4:3	SCTXCE/ SCTXDFSE	T	x	x	x	x	HDLC frame Seq # Mode	0 = Seq Num always 0; 1 = Wrap around using “0” 3 = Wrap around skipping “0”
2:0	SCTXCOS	RT	0	0	0	0	HDLC Channel Width Select	0 = Nx8-bit (Nx64 Kb/s)
BCDR2								
31:0	ATSS[1]	RT	x	x	x	x	Active Timeslot Select	1b = included in Bundle (T1: TS #0 - 23; E1: TS #1 - 31)
BCDR3								
4:3	TXPMS	T	x	x	x	x	TXP Packet Mode Select	0 = Disable TXP Bundle; 1 = Enable TXP HDLC Bundle
2:1	TXBTS	T	1	1	1	1	TXP Bundle TDM Port Mode	1 = HDLC for Structured T1/E1 Port
0	TXBPS	NA	0	0	0	0	TXP Bundle Priority	NA
BCDR4								
21	RXRE	R	x	x	x	x	RXP RTP Enable	0 = RTP is not included; 1 = RTP is required
20	RXCWE	R	[1]	[1]	[1]	[1]	RXP Control Word Enable	0 = Control Word is not included; 1 = CW is required
19:18	RXHTS	R	0	1	2	3	RXP Header Type Select	0 = MPLS; 1 = UDP; 2 = L2TPv3; 3 = MEF
17:16	RXBTS	R	1	1	1	1	RXP Bundle TDM Port Mode	1 = HDLC for Structured T1/E1 Port
15:14	RXLCS	R	x	0	x	0	RXP Label/Cookie Select	MPLS: 0x1 = 1 Label; 0x2 = 2 Labels; 0x3 = 3 Labels L2TPV3: 0x0 = 0 Cookies; 0x1 = 1 Cookie; 0x2 = 2 Cookies
13	RXUBIDLS	R	0	[1]	0	0	RXP UDP BID Location	0 = UDP Source Port; 1 = UDP Destination Port
12	SCLVI	R	[0]	[0]	[0]	[0]	HDLC Inter-frame Fill	0 = 0x7E Inter-frame Fill; 1 = 0xFF Inter-frame Fill
11:9	RXCOS	NA	0	0	0	0	Xmt (RXP) Conditioning Octet	NA
8	RXOICWE	R	[1]	[0]	[1]	[1]	RXP OAM in CW Enable	0 = ignore CW OAM indication; 1 = look for OAM indication
7:6	RXBDS	R	x	x	x	x	RXP Bundle Data Destination	0 = TDM Port; 3 = Discard packet
5:1	PNS[1]	R	x	x	x	x	TDM Port Number Select	Select TDM Port #0 - #31
0	PCRE	R	0	0	0	0	TDM Port Ck Recov. Enable	0 = do not use for Ck Recov
BCDR5								
24:10	PDVT	NA	0	0	0	0	Packet Delay Variation Time	NA
9:0	MJBS	R	NA	NA	NA	NA	Max Jitter Buffer Size	NA

Note: [1] TSAn.m must be programmed to enable the port and timeslots selected by PNS (n = PNS) and ATSS (m = Timeslot).

10.4.2.6 Structured 16 Kb/s or 56 Kb/s HDLC Bundle Settings

Table 10-51. Structured 16 Kb/s or 56 Kb/s HDLC Bundle Settings

Reg-bit	Bit Abbrev	RT	M	U	L	E	Bit Name Description	Comments
BCDR1								
23	LBCAI	NA	0	0	0	0	L Bit Conditioning Auto Insert	NA
22:21	PMT	RT	0	0	0	0	Payload Machine Type	0 = HDLC Payload Machine Type
20:10	PMS	R	x	x	x	x	Payload Max Size	Maximum # of bytes in RXP Packet Payload (not incl. FCS)
9	SCSCFPD	NA	0	0	0	0	SAT/CES Sanity Check	NA
8	SCSNRE	RT	[0]	[0]	[0]	[0]	HDLC Bit Reorder Enable	0 = transmit MS bit first; 1 = transmit LS bit first
7	SCRXBCSS	RT	[1]	[1]	[1]	[1]	HDLC FCS Disable	0 = FCS enabled; 1 = FCS disabled
6	SCTXBCSS	RT	[1]	[1]	[1]	[1]	HDLC RXP FCS bit Width	0 = 16-bit; 1 = 32-bit
5	RSNS	NA	0	0	0	0	Reorder Seq Number Select	NA
4:3	SCTXCE/ SCTXDFSE	T	x	x	x	x	HDLC frame Seq # Mode	0 = Seq Num always 0; 1 = Wrap around using "0" 3 = Wrap around skipping "0"
2:0	SCTXCOS	RT	x	x	x	x	HDLC Channel Width Select	1 = 7-bit + 1 unused bit (56 Kb/s); 2 = 2-bit coding in 2 LSbit position + 6 unused bits (16 Kb/s) 3 = 2-bit coding in 2 MSbit position + 6 unused bits (16 Kb/s)
BCDR2								
31:0	ATSS[1]	RT	x	x	x	x	Active Timeslot Select	1b = included in Bundle (T1: TS #0 - 23; E1: TS #1 - 31)
BCDR3								
4:3	TXPMS	T	x	x	x	x	TXP Packet Mode Select	0 = Disable TXP Bundle; 1 = Enable TXP HDLC Bundle
2:1	TXBTS	T	1	1	1	1	TXP Bundle TDM Port Mode	1 = HDLC for Structured T1/E1 Port
0	TXBPS	NA	0	0	0	0	TXP Bundle Priority	NA
BCDR4								
21	RXRE	R	x	x	x	x	RXP RTP Enable	0 = RTP is not included; 1 = RTP is required
20	RXCWE	R	[1]	[1]	[1]	[1]	RXP Control Word Enable	0 = Control Word is not included; 1 = CW is required
19:18	RXHTS	R	0	1	2	3	RXP Header Type Select	0 = MPLS; 1 = UDP; 2 = L2TPv3; 3 = MEF
17:16	RXBTS	R	1	1	1	1	RXP Bundle TDM Port Mode	1 = HDLC for Structured T1/E1 Port
15:14	RXLCS	R	x	0	x	0	RXP Label/Cookie Select	MPLS: 0x1 = 1 Label; 0x2 = 2 Labels; 0x3 = 3 Labels L2TPV3: 0x0 = 0 Cookies; 0x1 = 1 Cookie; 0x2 = 2 Cookies
13	RXUBIDLS	R	0	[1]	0	0	RXP UDP BID Location	0 = UDP Source Port; 1 = UDP Destination Port
12	SCLVI	R	[0]	[0]	[0]	[0]	HDLC Inter-frame Fill	0 = 0x7E Inter-frame Fill; 1 = 0xFF Inter-frame Fill
11:9	RXCOS	NA	0	0	0	0	Xmt (RXP) Conditioning Octet	NA
8	RXOICWE	R	[1]	[0]	[1]	[1]	RXP OAM in CW Enable	0 = ignore CW OAM indication; 1 = look for OAM indication
7:6	RXBDS	R	x	x	x	x	RXP Bundle Data Destination	0 = TDM Port; 3 = Discard packet
5:1	PNS[1]	R	x	x	x	x	TDM Port Number Select	Select TDM Port #0 - #31
0	PCRE	R	0	0	0	0	TDM Port Ck Recov. Enable	0 = do not use for Ck Recov
BCDR5								
24:10	PDVT	NA	0	0	0	0	Packet Delay Variation Time	NA
9:0	MJBS	R	NA	NA	NA	NA	Max Jitter Buffer Size	NA

Note: [1] TSAn.m must be programmed to enable the port and timeslots selected by PNS (n = PNS) and ATSS (m = Timeslot).

10.4.2.7 Clock Only Bundle Settings

10.4.2.7.1 Combined RXP and TXP (Bidirectional) Clock Only Bundle Settings

Table 10-52. Combined RXP and TXP (Bidirectional) Clock Only Bundle Settings

Reg-bit	Bit Abbrev	RT	M	U	L	E	Bit Name Description	Comments
BCDR1								
23	LBCAI	NA	0	0	0	0	L Bit Conditioning Auto Insert	NA
22:21	PMT	RT	3	3	3	3	Payload Machine Type	3 = SAT/CES Payload Machine Type
20:10	PMS	RT	x	x	x	x	Ck Only Packet Rate	SAT Packet rate (# TDM Port bytes per pkt) For T1, PCT = 1 ms: PMS = 0x0C1 (193 bytes decimal) For E1, PCT = 1 ms: PMS = 0x100 (256 bytes decimal) For 256 Kb/s, PCT = 1 ms: PMS = 0x020 (32 bytes dec.) CES Packet rate (# TDM Port frames per pkt) For PCT = 1 ms: PMS = 0x008 (8 frames) For PCT = 8 ms: PMS = 0x040 (64 frames)
9	SCSCFPD	R	0	0	0	0	Sanity Check	0 = do not discard based on PMS setting
8	SCSNRE	R	[1]	[1]	[1]	[1]	Seq # Reorder En	0 = Disable Reordering; 1 = Enable Reordering
7	SCRXBCSS	NA	0	0	0	0	CES RXP CAS Source Select	NA
6	SCTXBCSS	NA	0	0	0	0	CES TXP CAS Source Select	NA
5	RSNS	R	[0]	[0]	[0]	[0]	Reorder Seq Number Select	0 = Control Word Sequence #; 1 = RTP Sequence #
4	SCTXCE	NA	0	0	0	0	SAT/CES TXP Condition En	NA
3	SCTXDFSE	NA	0	0	0	0	CES T1 TXP Framing	NA
2:0	SCTXCOS	NA	0	0	0	0	SAT/CES TXP Cond. Octet	NA
BCDR2								
31:0	ATSS[1]	RT	x	x	x	x	Active Timeslot Select	SAT: 0x0000.0001 CES with out CAS: 1 = enable TS (T1: 0-23; E1: 1-31) CES with CAS: 1 = enable TS (T1: 0-23; E1: 1-15 & 17-31)
BCDR3								1 = HDLC for Structured T1/E1 Port
4:3	TXPMS	T	x	x	x	x	TXP Packet Mode Select	0 = Disable; 2 = Transmit without payload (Clock Only)
2:1	TXBTS	T	x	x	x	x	RXP Bundle Type	0 = SAT for Unstructured TDM Port 1 = CES without CAS for Structured T1/E1 Port
0	TXBPS	T	[1]	[1]	[1]	[1]	TXP Bundle Priority	0 = low priority; 1 = high (for PW Timing Connections)
BCDR4								
21	RXRE	R	x	x	x	x	RXP RTP Enable	0 = RTP is not included; 1 = RTP is required
20	RXCWE	R	1	1	1	1	RXP Control Word Enable	1 = Control Word is required
19:18	RXHTS	R	0	1	2	3	RXP Header Type Select	0 = MPLS; 1 = UDP; 2 = L2TPv3; 3 = MEF
17:16	RXBTS	R	x	x	x	x	RXP Bundle Type	0 = SAT for Unstructured TDM Port 1 = CES without CAS for Structured T1/E1 Port
15:14	RXLCS	R	x	0	x	0	RXP Label/Cookie Select	MPLS: 0x1 = 1 Label; 0x2 = 2 Labels; 0x3 = 3 Labels L2TPV3: 0x0 = 0 Cookies; 0x1 = 1 Cookie; 0x2 = 2 Cookies
13	RXUBIDLS	R	0	[1]	0	0	RXP UDP BID Location	0 = UDP Source Port; 1 = UDP Destination Port
12	SCLVI	NA	0	0	0	0	SAT/CES Last Value Insert	NA
11:9	RXCOS	NA	0	0	0	0	Xmt (RXP) Conditioning Octet	NA
8	RXOICWE	R	[1]	[0]	[1]	[1]	RXP OAM in CW Enable	0 = ignore CW OAM indication; 1 = look for OAM indication
7:6	RXBDS	R	3	3	3	3	RXP Bundle Data Destination	3 = Discard (timing information is still available for ck recov)
5:1	PNS[1]	RT	x	x	x	x	TDM Port Number Select	Select TDM Port #0 - #31
0	PCRE	R	1	1	1	1	TDM Port Ck Recov. Enable	1 = use for Ck Recovery
BCDR5								
24:10	PDVT	NA	0	0	0	0	Packet Delay Variation Time	NA
9:0	MJBS	NA	0	0	0	0	Max Jitter Buffer Size	NA

Note: [1] TSAn.m must be programmed to enable the port and timeslots selected by PNS (n = PNS) and ATSS (m = Timeslot).

The Datapath for an RXP Clock Only Bundle should not be released from Reset (B.BRCR2.RXBRE = 1). The Clock Only Bundle does not include payload data. Holding the Bundle's Datapath in Reset prevents the S132 from attempting to process the packet after the packet header has been fully interpreted.

10.4.2.7.2 RXP (Unidirectional) Clock Only Bundle Settings

Table 10-53. RXP (Unidirectional) Clock Only Bundle Settings

Reg-bit	Bit Abbrev	RT	M	U	L	E	Bit Name Description	Comments
BCDR1								
23	LBCAI	NA	0	0	0	0	L Bit Conditioning Auto Insert	NA
22:21	PMT	R	3	3	3	3	Payload Machine Type	3 = SAT/CES Payload Machine Type
20:10	PMS	R	x	x	x	x	Ck Only Packet Rate	SAT Packet rate (# TDM Port bytes per pkt) For T1, PCT = 1 ms: PMS = 0x0C1 (193 bytes decimal) For E1, PCT = 1 ms: PMS = 0x100 (256 bytes decimal) For 256 Kb/s, PCT = 1 ms: PMS = 0x020 (32 bytes dec.) CES Packet rate (# TDM Port frames per pkt) For PCT = 1 ms: PMS = 0x008 (8 frames) For PCT = 8 ms: PMS = 0x040 (64 frames)
9	SCSCFPD	R	0	0	0	0	Sanity Check	0 = do not discard based on PMS setting
8	SCSNRE	R	[1]	[1]	[1]	[1]	Seq # Reorder En	0 = Disable Reordering; 1 = Enable Reordering
7	SCRXBCSS	NA	0	0	0	0	CES RXP CAS Source Select	NA
6	SCTXBCSS	NA	0	0	0	0	CES TXP CAS Source Select	NA
5	RSNS	R	[0]	[0]	[0]	[0]	Reorder Seq Number Select	0 = Control Word Sequence #; 1 = RTP Sequence #
4	SCTXCE	NA	0	0	0	0	SAT/CES TXP Condition En	NA
3	SCTXDFSE	NA	0	0	0	0	CES T1 TXP Framing	NA
2:0	SCTXCOS	NA	0	0	0	0	SAT/CES TXP Cond. Octet	NA
BCDR2								
31:0	ATSS[1]	R	x	x	x	x	Active Timeslot Select	SAT: 0x0000.0001 CES with out CAS: 1 = enable TS (T1: 0-23; E1: 1-31) CES with CAS: 1 = enable TS (T1: 0-23; E1: 1-15 & 17-31)
BCDR3								
4:3	TXPMS	NA	0	0	0	0	TXP Packet Mode Select	NA
2:1	TXBTS	NA	0	0	0	0	RXP Bundle Type	NA
0	TXBPS	NA	0	0	0	0	TXP Bundle Priority	NA
BCDR4								
21	RXRE	R	x	x	x	x	RXP RTP Enable	0 = RTP is not included; 1 = RTP is required
20	RXCWE	R	1	1	1	1	RXP Control Word Enable	1 = Control Word is required
19:18	RXHTS	R	0	1	2	3	RXP Header Type Select	0 = MPLS; 1 = UDP; 2 = L2TPv3; 3 = MEF
17:16	RXBTS	R	x	x	x	x	RXP Bundle Type	0 = SAT for Unstructured TDM Port 1 = CES without CAS for Structured T1/E1 Port
15:14	RXLCS	R	x	0	x	0	RXP Label/Cookie Select	MPLS: 0x1 = 1 Label; 0x2 = 2 Labels; 0x3 = 3 Labels L2TPV3: 0x0 = 0 Cookies; 0x1 = 1 Cookie; 0x2 = 2 Cookies
13	RXUBIDLS	R	0	[1]	0	0	RXP UDP BID Location	0 = UDP Source Port; 1 = UDP Destination Port
12	SCLVI	NA	0	0	0	0	SAT/CES Last Value Insert	NA
11:9	RXCOS	NA	0	0	0	0	Xmt (RXP) Conditioning Octet	NA
8	RXOICWE	R	[1]	[0]	[1]	[1]	RXP OAM in CW Enable	0 = ignore CW OAM indication; 1 = look for OAM indication
7:6	RXBDS	R	3	3	3	3	RXP Bundle Data Destination	3 = Discard (timing still available for ck recov)
5:1	PNS[1]	R	x	x	x	x	TDM Port Number Select	Select TDM Port #0 - #31
0	PCRE	R	1	1	1	1	TDM Port Ck Recov. Enable	1 = use for Ck Recovery
BCDR5								
24:10	PDVT	NA	0	0	0	0	Packet Delay Variation Time	NA
9:0	MJBS	NA	0	0	0	0	Max Jitter Buffer Size	NA

Note: [1] TSAn.m must be programmed to enable the port and timeslots selected by PNS (n = PNS) and ATSS (m = Timeslot).

The Datapath for an RXP Clock Only Bundle should not be released from Reset (B.BRCR2.RXBRE = 1). The Clock Only Bundle does not include payload data. Holding the Bundle's Datapath in Reset prevents the S132 from attempting to process the packet after the packet header has been fully interpreted.

10.4.2.7.3 TXP (Unidirectional) Clock Only Bundle Settings

Table 10-54. TXP (Unidirectional) Clock Only Bundle Settings

Reg-bit	Bit Abbrev	RT	M	U	L	E	Bit Name Description	Comments
BCDR1								
23	LBCAI	NA	0	0	0	0	L Bit Conditioning Auto Insert	NA
22:21	PMT	T	3	3	3	3	Payload Machine Type	3 = SAT/CES Payload Machine Type
20:10	PMS	T	x	x	x	x	Ck Only Packet Rate	SAT Packet rate (# TDM Port bytes per pkt) For T1, PCT = 1 ms: PMS = 0x0C1 (193 bytes decimal) For E1, PCT = 1 ms: PMS = 0x100 (256 bytes decimal) For 256 Kb/s, PCT = 1 ms: PMS = 0x020 (32 bytes dec.) CES Packet rate (# TDM Port frames per pkt) For PCT = 1 ms: PMS = 0x008 (8 frames) For PCT = 8 ms: PMS = 0x040 (64 frames)
9	SCSCFPD	NA	0	0	0	0	Sanity Check	NA
8	SCSNRE	NA	0	0	0	0	Seq # Reorder En	NA
7	SCRXBCSS	NA	0	0	0	0	CES RXP CAS Source Select	NA
6	SCTXBCSS	NA	0	0	0	0	CES TXP CAS Source Select	NA
5	RSNS	NA	0	0	0	0	Reorder Seq Number Select	NA
4	SCTXCE	NA	0	0	0	0	SAT/CES TXP Condition En	NA
3	SCTXDFSE	NA	0	0	0	0	CES T1 TXP Framing	NA
2:0	SCTXCOS	NA	0	0	0	0	SAT/CES TXP Cond. Octet	NA
BCDR2								
31:0	ATSS[1]	T	x	x	x	x	Active Timeslot Select	SAT: 0x0000.0001 CES with out CAS: 1 = enable TS (T1: 0-23; E1: 1-31) CES with CAS: 1 = enable TS (T1: 0-23; E1: 1-15 & 17-31)
BCDR3								
4:3	TXPMS	T	x	x	x	x	TXP Packet Mode Select	0 = Disable; 2 = Transmit without payload (Clock Only)
2:1	TXBTS	T	x	x	x	x	RXP Bundle Type	0 = SAT for Unstructured TDM Port 1 = CES without CAS for Structured T1/E1 Port
0	TXBPS	T	[1]	[1]	[1]	[1]	TXP Bundle Priority	0 = low priority; 1 = high (for PW Timing Connections)
BCDR4								
21	RXRE	NA	0	0	0	0	RXP RTP Enable	NA
20	RXCWE	NA	0	0	0	0	RXP Control Word Enable	NA
19:18	RXHTS	NA	0	0	0	0	RXP Header Type Select	NA
17:16	RXBTS	NA	0	0	0	0	RXP Bundle Type	NA
15:14	RXLCS	NA	0	0	0	0	RXP Label/Cookie Select	NA
13	RXUBIDLS	NA	0	0	0	0	RXP UDP BID Location	NA
12	SCLVI	NA	0	0	0	0	SAT/CES Last Value Insert	NA
11:9	RXCOS	NA	0	0	0	0	Xmt (RXP) Conditioning Octet	NA
8	RXOICWE	NA	0	0	0	0	RXP OAM in CW Enable	NA
7:6	RXBDS	NA	0	0	0	0	RXP Bundle Data Destination	NA
5:1	PNS[1]	T	x	x	x	x	TDM Port Number Select	Select TDM Port #0 - #31
0	PCRE	NA	0	0	0	0	TDM Port Ck Recov. Enable	NA
BCDR5								
24:10	PDVT	NA	0	0	0	0	Packet Delay Variation Time	NA
9:0	MJBS	NA	0	0	0	0	Max Jitter Buffer Size	NA

Note: [1] TSAn.m must be programmed to enable the port and timeslots selected by PNS (n = PNS) and ATSS (m = Timeslot).

10.4.2.8 “CPU RXP PW Debug” Bundle Settings

The minimum Bundle settings that must be configured to properly detect packets for CPU Debug RXP PW Bundles are provided in Table 10-55. In the table, “NR” indicates “Not Required”. An “NR” value can be set according to the “normal” CES, SAT, HDLC or Clock Only Bundle setting but is not required by a CPU Debug RXP PW Bundle.

Table 10-55. “CPU RXP PW Debug” Bundle Settings

Reg-bit	Bit Abbrev	RT	M	U	L	E	Bit Name Description	Comments
BCDR1								
31:0		-	NR	NR	NR	NR		NR
BCDR2								
31:0		-	NR	NR	NR	NR		NR
BCDR3								
4:3	TXPMS	T	[0]	[0]	[0]	[0]	TXP Packet Mode Select	0 = Disable; 1 = Transmit with payload; 2 = Transmit without payload (for TDM Port faults)
2:1	TXBTS	-	NR	NR	NR	NR	RXP Bundle Type	NR
0	TXBPS	-	NR	NR	NR	NR	TXP Bundle Priority	NR
BCDR4								
21	RXRE	-	NR	NR	NR	NR	RXP RTP Enable	NR
20	RXCWE	-	NR	NR	NR	NR	RXP Control Word Enable	NR
19:18	RXHTS	R	0	1	2	3	RXP Header Type Select	0 = MPLS; 1 = UDP; 2 = L2TPv3; 3 = MEF
17:16	RXBTS	-	NR	NR	NR	NR	RXP Bundle Type	NR
15:14	RXLCS	-	NR	NR	NR	NR	RXP Label/Cookie Select	NR
13	RXUBIDLS	R	NR	[1]	NR	NR	RXP UDP BID Location	0 = UDP Source Port; 1 = UDP Destination Port
12	SCLVI	-	NR	NR	NR	NR	SAT/CES Last Value Insert	NR
11:9	RXCOS	-	NR	NR	NR	NR	Xmt (RXP) Condition Octet	NR
8	RXOICWE	-	NR	NR	NR	NR	RXP OAM in CW Enable	NR
7:6	RXBDS	R	x	x	x	x	RXP Bundle Data Destination	1 = forward to CPU; 3 = discard
5:1	PNS	-	NR	NR	NR	NR	TDM Port Number Select	NR
0	PCRE	-	NR	NR	NR	NR	TDM Port Ck Recov. Enable	NR
BCDR5								
31:0		-	NR	NR	NR	NR		NR

10.4.2.9 In-band VCCV OAM Connection Settings

When In-band VCCV OAM is used it is always part of a CES, SAT, HDLC or Clock Only Bundle. The In-band VCCV connection can be to be enabled before all of the Bundle function/settings are known/programmed for the CES, SAT, HDLC, Clock Only Bundle. The minimum Bundle settings that must be configured to properly detect In-band VCCV are provided in Table 10-56. In the table, "NR" indicates "Not Required". An "NR" value can be set according to the "normal" CES, SAT, HDLC or Clock Only Bundle setting but is not required by an In-band VCCV Connection.

Table 10-56. In-band VCCV OAM Connection Settings

Reg-bit	Bit Abbrev	RT	M	U	L	E	Bit Name Description	Comments
BCDR1								
31:0		-	NR	NR	NR	NR		NR
BCDR2								
31:0		-	NR	NR	NR	NR		NR
BCDR3								
31:0		-	NR	NR	NR	NR		NR
BCDR4								
21	RXRE	-	NR	NR	NR	NR	RXP RTP Enable	NR
20	RXCWE	R	1	1	1	1	RXP Control Word Enable	1 = Control Word is required
19:18	RXHTS	R	0	1	2	3	RXP Header Type Select	0 = MPLS; 1 = UDP; 2 = L2TPv3; 3 = MEF
17:16	RXBTS	-	NR	NR	NR	NR	RXP Bundle Type	NR
15:14	RXLCS	-	NR	NR	NR	NR	RXP Label/Cookie Select	NR
13	RXUBIDLS	R	0	[1]	0	0	RXP UDP BID Location	0 = UDP Source Port; 1 = UDP Destination Port
12	SCLVI	-	NR	NR	NR	NR	SAT/CES Last Value Insert	NR
11:9	RXCOS	-	NR	NR	NR	NR	Xmt (RXP) Conditioning Octet	NR
8	RXOICWE	R	1	[0]	1	1	RXP OAM in CW Enable	1 = look for OAM indication
7:6	RXBDS	R	x	x	x	x	RXP Bundle Data Destination	0 = TDM Port; 3 = Discard packet
5:1	PNS	-	NR	NR	NR	NR	TDM Port Number Select	NR
0	PCRE	-	NR	NR	NR	NR	TDM Port Ck Recov. Enable	NR
BCDR5								
31:0		-	NR	NR	NR	NR		NR

The B.BCDR4.RXCWE setting is ignored if PC.CR1.DPS7 = 1 (discard all In-band VCCV packets).

10.4.2.10 OAM Bundle (Out-band VCCV OAM) Settings

OAM Bundles only include programmable settings for the OAM BID and for the Activate state of the OAM Bundle. OAM Bundles do not included the other register/functions that are provided for the "normal" Bundles (described in the previous sections).

Table 10-57. OAM Bundle PWID and Activation Control Registers (B.)

Register	Bits	Functional Description	Comments
BACR	OBS	OAM Bundle Select	**Assign Bundle ID (PWID):** To assign an OAM Bundle ID to an OAM Bundle, first program the OAM Bundle ID using BIDV. Then use OBS = 1 and BS to select the OAM Bundle Number (0 to 31). The BIDV value for that OAM Bundle will be Written when the WE transitions from "0 to 1".
	WE	Write Enable	
	RE	Read Enable	
	BS	Bundle Number	**Bundle Activate State:** To Activate or De-activate an OAM Bundle, first program the Activate state using ABE. Then use OBS = 1 and BS to select the OAM Bundle Number (0 to 31). The Activate state for that OAM Bundle will be Written when WE transitions from "0 to 1".
BADR1	ABE	Activate Bundle	
BADR2	BIDV	Bundle ID	

10.4.3 Send to CPU Settings

There are several RXP packet conditions that can be used to forward packets to the CPU that have been described in previous Register Guide sections. Table 10-58 provides a Quick Reference list for each of these "send to CPU" conditions using an abbreviated detected condition description.

Table 10-58. "Send to CPU" Quick Reference Settings

Send to CPU Type	Detected RX Packet Condition	"Send to CPU" Program Settings
CPU Debug RXP PW Bundle	PW-ID = Activated BID	B.BCDR4.RXBDS = 1
In-band VCCV OAM	PW-ID = Activated BID	B.BCDR4.RXIOCWE=1 & PC.CR1.DPS7 = 0
MEF OAM	Ethernet Type = PC.CR4.MOET	PC.CR4.MOET & PC.CR1.DPS7 = 0
Too many MPLS Labels	# MPLS outer labels > 2	PC.CR1.DPS10 = 0
Unknown Ethernet DA	Ethernet DA ≠ PC.CR17 – PC.CR19	PC.CR1.DPS9 = 0
CPU Destination Ethernet Type	Ethernet Type = PC.CR20.CDET	PC.CR1.DPS8=0
OAM Bundle (Out-band VCCV)	PW-ID = Activated OAM BID	PC.CR1.DPS7 = 0
Unknown PW-ID	PW-ID ≠ PC.CR6 – PC.CR16	PC.CR1.DPS6 = 0
Unknown UDP Protocol	Unknown UDP Protocol Type	PC.CR1.DPS5 = 0
Unknown IP Protocol	IP Protocol ≠ UDP or L2TPv3	PC.CR1.DPS4 = 0
ARP with known IP DA	ARP IP DA = PC.CR6 – PC.CR8	PC.CR1.DPS3 = 0
Unknown Ether Type	Ethernet Type ≠ ARP, IPv4, IPv6, Multicast MPLS, Unicast MPLS, PC.CR20.CDET, PC.CR4.MET or PC.CR4.MOET	PC.CR1.DPS2 = 0
Unknown IP DA	IP DA ≠ PC.CR6 – PC.CR16	PC.CR1.DPS1 = 0
ARP w/ unknown IP DA	ARP IP DA ≠ PC.CR6 – PC.CR8	PC.CR1.DPS0 = 0

10.4.4 TDM Port Settings

Table 10-59. Global TDM Port Settings

Register	Functional Description	Comments
G.ECCR1 - G.ECCR2	TXP TDM Conditioning Octets A – H	Data transmitted in TXP TDM Bundle (in place of received TDM port data).
G.TCCR1 - G.TCCR2	RXP TDM Conditioning Octets A –H	Data transmitted at TDM Port (in place of RXP TDM Bundle data).

The tables that follow provide most of the settings for T1/E1 and slower TDM Port applications. When a TDM Port uses a Clock Recovery Engine there are some Clock Recovery Engine Registers that must be set that are not identified in this section (e.g. selection between Adaptive Clock Recovery and Differential Clock Recovery). These are defined by the S132 DSP Firmware load.

Table 10-60. TDM Port "n" Register Settings for T1 Applications (Pn.; n = 0 to 31)

Reg-bit	Bit Name	RT	SAT	CES no CAS	CES w/ CAS	Bit Name Description	Comments
TXSCn.CR1 - CR4							
31:0	CTS0-CTS23	T	0	0	x	TXP SW CAS (TS 0 – 23)	SW CAS transmitted in TXP TDM Bundle.
RXSCn.CR1 - CR4							
31:0	CTS0-CTS23	R	0	0	x	Xmt (RXP) SW CAS (TS0-23)	SW CAS transmitted at TDM Port.
PTCR1.							
31	DR	R	x	x	x	Data path Reset	0 = Normal; 1 = Hold all data path registers in reset value
28	SFS	R	0	1	1	Structure Format Select	Unstructured (0) or Structured (1)
27	FFS	R	0	1	1	Frame Format	1 = T1
26:25	MFS	R	0	0	x	CAS Multi-frame Format	0 = no CAS or multi-frame; 2 = T1 SF; 3 = T1 ESF
24:23	BFD	R	[3]	[3]	[3]	# Frame(Block) per Buffer	0 = disable port rcv; 1 = 1 block; 2 = 2 blocks; 3 = 4 blocks
22:18	BPF	R	[0x17]	[0x17]	[0x17]	Bytes per Frame(Block)	0x17 (24 bytes; 0x00 = 1 byte/frame; BPF ≤ PMS)
17	DP	R	x	x	x	Decap Priority	Low priority (0) or High priority (1)
16	DOSOT	R	0	0	x	Disable CAS on TDAT	Overwrite CAS on TDAT (0) or do not overwrite TDAT (1)
PTCR2.							
9	PRPTLL	RT	x	x	x	Port Rcv to Xmt Line Loopbk	0 = loopback disabled (normal); 1 = loopback enabled
8	TIOE	R	x	x	x	Transmit Input/Output Enable	0 = TDAT/TSYNC/TSIG disabled (high Z); 1 = enabled
7	TCE	R	x	x	x	Transmit Clock Enable	0 = TCLKO disabled (high Z); 1 = enabled
6	TSRS	R	0	x	x	Transmit Framing Source	Synchronize transmit timing to RSYNC (0) or TSYNC (1)
5	TDS	R	0	x	x	TSYNC Direction Select	Input (0) or output (1).
4	TOES	R	x	x	x	Transmit Output Edge Select	0 = positive edge; 1 = negative edge
3	TIES	R	x	x	x	Transmit Input Edge Select	0 = positive edge; 1 = negative edge
2:0	TSS	R	x	x	x	TCLKO Source Select	0 = RCLK; 1 = aclk; 2 = grclk; 4 = EXTCLK0; 5 = EXTCLK1
PTCR3.							
31:0	PRPTTSL	RT	0	x	x	Port Rcv to Xmt TS Loopback	0 = loopback disabled (normal); 1 = enabled (1 bit per TS)
PRCR1.							
31	DR	T	x	x	x	Data path Reset	0 = Normal; 1 = Hold all data path registers in reset value
28	SFS	T	0	1	1	Structure Format Select	Unstructured (0) or Structured (1)
27	FFS	T	0	1	1	Frame Format	1 = T1
26:25	MFS	T	0	0	x	CAS Multi-frame Format	0 = no CAS multi-frame; 2 = T1 SF; 3 = T1 ESF
24:23	BFD	T	[3]	[3]	[3]	# Frame(Block) per Buffer	0 = disable port rcv; 1 = 1 block; 2 = 2 blocks; 3 = 4 blocks
22:18	BPF	T	[0x17]	[0x17]	[0x17]	Bytes per Frame(Block)	0x17 (24 bytes; 0x00 = 1 byte/frame; BPF ≤ PMS)
17	EP	T	x	x	x	Encap Priority	Low priority (0) or High priority (1)
16	CS	T	0	0	x	CAS Source	RDAT (0) or RSIG (1)
15	CBVSE	T	0	0	x	C-bit Value	CAS C-bit value (T1 ESF only)
14	DBVSE	T	0	0	x	D-bit Value	CAS D-bit value (T1 ESF only)
13	LB	T	x	x	x	L-bit Value	L-bit Value for all TXP Bundles associated with Port "n"
12	LBSS	T	x	x	x	L-bit Source (all Pn Bundles)	TXP Bundle L-bit source: LB (0); TXP Bundle Descriptor (1)
11:1	SPL	T	[0x17]	0	0	SAT Packet Payload Length	SPL = B.BCDR1.PMS ≥ BPF (T1: 0x17)
PRCR2.							
6	RSTS	T	0	x	x	Receive Framing Source	0 = RSYNC input; 1 = synchronize to Transmit Port timing
5	RDS	T	0	x	x	RSYNC Direction Select	Input (0) or output (1).
3	RIES	T	x	x	x	RCLK Edge Select	Positive edge (0) or negative edge (1)
0	RSS	T	x	x	x	Receive clock Source Select	RCLK (0) or TCLKO (1)
PRCR3.		T					
31:0	PTPRTSL	RT	0	x	x	Port Xmt to Rcv TS Loopback	0 = loopback disabled (normal); 1 = enabled (1 bit per TS)
PRCR4.							
16	TSGMS	T	x	x	x	TXP Timestamp Gen Mode	0=Differential (CMNCLK); 1=Absolute (RSS selects Rcv ck)
15:0	TSGMC	T	x	x	x	TXP Timestamp Gen M Coeff	M = INT(637009920000 / CMNCLK)
PRCR5.							
28:16	TSGN1C	T	x	x	x	TXP T-stamp Gen N1 Coeff	N1 = 79626240 + (M * CMNCLK / 8000)
12:0	TSGN0C	T	x	x	x	TXP T-stamp Gen N0 Coeff	N0 = N1 + (CMNCLK / 8000)

Table 10-61. TDM Port "n" Register Settings for E1 Applications (Pn.; n = 0 to 31)

Reg-bit	Bit Name	RT	SAT	CES no CAS	CES w/ CAS	Bit Name Description	Comments
TXSCn.CR1 - CR4							
31:0	CTS0-CTS23	T	0	0	x	TXP SW CAS (TS 0 – 31)	SW CAS transmitted in TXP TDM Bundle.
RXSCn.CR1 - CR4							
31:0	CTS0-CTS23	R	0	0	x	Xmt (RXP) SW CAS (TS0-23)	SW CAS transmitted at TDM Port.
PTCR1.							
31	DR	R	x	x	x	Data path Reset	0 = Normal; 1 = Hold all data path registers in reset value
28	SFS	R	0	1	1	Structure Format Select	Unstructured (0) or Structured (1)
27	FFS	R	0	0	0	Frame Format	0 = E1
26:25	MFS	R	0	0	x	CAS Multi-frame Format	0 = no CAS multi-frame; 1 = E1
24:23	BFD	R	[3]	[3]	[3]	# Frame(Block) per Buffer	0 = disable port rcv; 1 = 1 block; 2 = 2 blocks; 3 = 4 blocks
22:18	BPF	R	[0x1F]	[0x1F]	[0x1F]	Bytes per Frame(Block)	0x1F (32 bytes; 0x00 = 1 byte/frame; BPF ≤ PMS)
17	DP	R	x	x	x	Decap Priority	Low priority (0) or High priority (1)
16	DOSOT	R	0	0	x	Disable CAS on TDAT	Overwrite CAS on TDAT (0) or do not overwrite TDAT (1)
PTCR2.							
9	PRPTLL	RT	x	x	x	Port Rcv to Xmt Line Loopbk	0 = loopback disabled (normal); 1 = loopback enabled
8	TIOE	R	x	x	x	Transmit Input/Output Enable	0 = TDAT/TSYNC/TSIG disabled (high Z); 1 = enabled
7	TCE	R	x	x	x	Transmit Clock Enable	0 = TCLKO disabled (high Z); 1 = enabled
6	TSRS	R	0	x	x	Transmit Framing Source	Synchronize transmit timing to RSYNC (0) or TSYNC (1)
5	TDS	R	0	x	x	TSYNC Direction Select	Input (0) or output (1).
4	TOES	R	x	x	x	Transmit Output Edge Select	0 = positive edge; 1 = negative edge
3	TIES	R	x	x	x	Transmit Input Edge Select	0 = positive edge; 1 = negative edge
2:0	TSS	R	x	x	x	TCLKO Source Select	0 = RCLK; 1 = aclk; 2 = grclk; 4 = EXTCLK0; 5 = EXTCLK1
PTCR3.							
31:0	PRPTTSL	RT	0	x	x	Port Rcv to Xmt TS Loopback	0 = loopback disabled (normal); 1 = enabled (1 bit per TS)
PRCR1.							
31	DR	T	x	x	x	Data path Reset	0 = Normal; 1 = Hold all data path registers in reset value
28	SFS	T	0	1	1	Structure Format Select	Unstructured (0) or Structured (1)
27	FFS	T	0	0	0	Frame Format	0 = E1
26:25	MFS	T	0	0	x	CAS Multi-frame Format	0 = no CAS multi-frame; 1 = E1
24:23	BFD	T	[3]	[3]	[3]	# Frame(Block) per Buffer	0 = disable port rcv; 1 = 1 block; 2 = 2 blocks; 3 = 4 blocks
22:18	BPF	T	[0x1F]	[0x1F]	[0x1F]	Bytes per Frame(Block)	0x1F (32 bytes; 0x00 = 1 byte/frame; BPF ≤ PMS)
17	EP	T	x	x	x	Encap Priority	Low priority (0) or High priority (1)
16	CS	T	0	0	x	CAS Source	RDAT (0) or RSIG (1)
15	CBVSE	NA	0	0	0	C-bit Value	NA
14	DBVSE	NA	0	0	0	D-bit Value	NA
13	LB	T	x	x	x	L-bit Value	L-bit Value for all TXP Bundles associated with Port "n"
12	LBSS	T	x	x	x	L-bit Source (all Pn Bundles)	TXP Bundle L-bit source: LB (0); TXP Bundle Descriptor (1)
11:1	SPL	T	[0x1F]	0	0	SAT Packet Payload Length	SPL = B.BCDR1.PMS ≥ BPF (E1: 0x1F)
PRCR2.							
6	RSTS	T	0	x	x	Receive Framing Source	0 = RSYNC input; 1 = synchronize to Transmit Port timing
5	RDS	T	0	x	x	RSYNC Direction Select	Input (0) or output (1).
3	RIES	T	x	x	x	RCLK Edge Select	Positive edge (0) or negative edge (1)
0	RSS	T	x	x	x	Receive clock Source Select	RCLK (0) or TCLKO (1)
PRCR3.		T					
31:0	PTPRTSL	RT	0	x	x	Port Xmt to Rcv TS Loopback	0 = loopback disabled (normal); 1 = enabled (1 bit per TS)
PRCR4.							
16	TSGMS	T	x	x	x	TXP Timestamp Gen Mode	0=Differential (CMNCLK); 1=Absolute (RSS select Rcv ck)
15:0	TSGMC	T	x	x	x	TXP Timestamp Gen M Coeff	M = INT(637009920000 / CMNCLK)
PRCR5.							
28:16	TSGN1C	T	x	x	x	TXP T-stamp Gen N1 Coeff	N1 = 79626240 + (M * CMNCLK / 8000)
12:0	TSGN0C	T	x	x	x	TXP T-stamp Gen N0 Coeff	N0 = N1 + (CMNCLK / 8000)

Table 10-62. TDM Port "n" Register Settings for non-T1/E1 Applications (Pn.; n = 0 to 31)

Reg-bit	Bit Name	RT	SAT	Bit Name Description	Comments
PTCR1.					
31	DR	R	x	Data path Reset	0 = Normal; 1 = Hold all data path registers in reset value
28	SFS	R	0	Structure Format Select	Unstructured = 0
27	FFS	R	0	Frame Format	NA
26:25	MFS	R	0	CAS Multi-frame Format	NA
24:23	BFD	R	[3]	# Frame(Block) per Buffer	0 = disable port rcv; 1 = 1 block; 2 = 2 blocks; 3 = 4 blocks
22:18	BPF	R	x	Bytes per Frame(Block)	# bytes per Frame(Block) "pseudo frame period" (0x00 = 1 byte; BPF ≤ PMS)
17	DP	R	x	Decap Priority	Low priority (0) or High priority (1)
16	DOSOT	R	0	Disable CAS on TDAT	NA
PTCR2.					
9	PRPTLL	RT	x	Port Rcv to Xmt Line Loopbk	0 = loopback disabled (normal); 1 = loopback enabled
8	TIOE	R	x	Transmit Input/Output Enable	0 = TDAT/TSYNC/TSIG disabled (high Z); 1 = enabled
7	TCE	R	x	Transmit Clock Enable	0 = TCLKO disabled (high Z); 1 = enabled
6	TSRS	R	0	Transmit Framing Source	NA
5	TDS	R	0	TSYNC Direction Select	NA
4	TOES	R	x	Transmit Output Edge Select	0 = positive edge; 1 = negative edge
3	TIES	R	x	Transmit Input Edge Select	0 = positive edge; 1 = negative edge
2:0	TSS	R	x	TCLKO Source Select	0 = RCLK; 1 = aclk; 2 = grclk; 4 = EXTCLK0; 5 = EXTCLK1
PTCR3.					
31:0	PRPTTSL	RT	0	Port Rcv to Xmt TS Loopback	NA
PRCR1.					
31	DR	T	x	Data path Reset	0 = Normal; 1 = Hold all data path registers in reset value
28	SFS	T	0	Structure Format Select	Unstructured (0) or Structured (1)
27	FFS	T	0	Frame Format	NA
26:25	MFS	T	0	CAS Multi-frame Format	NA
24:23	BFD	T	[3]	# Frame(Block) per Buffer	0 = disable port rcv; 1 = 1 block; 2 = 2 blocks; 3 = 4 blocks
22:18	BPF	T	x	Bytes per Frame(Block)	# bytes per Frame(Block) "pseudo frame period" (0x00 = 1 byte; BPF ≤ PMS)
17	EP	T	x	Encap Priority	Low priority (0) or High priority (1)
16	CS	T	0	CAS Source	NA
15	CBVSE	T	0	C-bit Value	NA
14	DBVSE	T	0	D-bit Value	NA
13	LB	T	x	L-bit Value	L-bit Value for TXP Bundle
12	LBSS	T	x	L-bit Source	TXP Bundle L-bit source: LB (0); TXP Bundle Descriptor (1)
11:1	SPL	T	x	SAT Packet Payload Length	Payload length in bytes where SPL = B.BCDR1.PMS ≥ BPF.
PRCR2.					
6	RSTS	T	0	Receive Framing Source	NA
5	RDS	T	0	RSYNC Direction Select	NA
3	RIES	T	x	RCLK Edge Select	Positive edge (0) or negative edge (1)
0	RSS	T	x	RCLK Source Select	RCLK (0) or TCLKO (1)
PRCR3.					
31:0	PTPRTSL	RT	0	Port Xmt to Rcv TS Loopback	NA
PRCR4.					
16	TSGMS	T	x	TXP Timestamp Gen Mode	0= Differential (CMNCLK); 1= Absolute (RSS selected RCLK)
15:0	TSGMC	T	x	TXP Timestamp Gen M Coeff	M = INT(637009920000 / CMNCLK)
PRCR5.					
28:16	TSGN1C	T	x	TXP T-stamp Gen N1 Coeff	N1 = - FCMN x ((2^12 x FOUT/FCMN) - M)
12:0	TSGN0C	T	x	TXP T-stamp Gen N0 Coeff	N0 = FCMN + N1

10.4.5 Status Monitoring

10.4.5.1 Ethernet Port Monitoring

Table 10-63. Ethernet MAC Status Registers (M.)

Bit #	Register Bit Name	r/w	Default	Comments
NET_CONTROL		**- Network Control Register**		
14	RD_SNAP	rw	0	Read Snapshot – 1 = Read latched register; 0 = Read current (real-time/raw) statistics
13	TAKE_SNAP	wo	0	Take Snapshot – "0 to 1" = store current statistics values in latched registers
5	STATS_CLR	wo	0	Clear Statistics Register - When set, clears the statistics registers.
NET_STATUS		**- Network Status Register**		
2	PHY_MAN_IDLE	ro	-	PHY Management Idle - The PHY management logic is idle (i.e. has completed).
1	MDIOS	ro	-	MDIO Status - Returns status of the MDIO signal
IRQ_STATUS		**- Interrupt Status Register**		
0	IRQ_MAN_DONE	ro	-	PHY Management Operation Done status. Cleared on read.
IRQ_ENABLE		**- Interrupt Enable Register**		
0	EN_IRQ_MAN_DONE	wo	0	Enable PHY Management Operation Done Interrupt
IRQ_DISABLE		**- Interrupt Disable Register**		
0	DIS_IRQ_MAN_DONE	wo	0	Disable PHY Management Done interrupt
IRQ_MASK		**- Interrupt Mask Register**		
0	MSK_IRQ_MAN_DONE	ro	0	Mask PHY Management Complete - 0 = Interrupt is enabled (1 = disabled).

Table 10-64. Ethernet RMON Count Registers (M.; all are Read Only)

Register Name	Bits	Bit Name	Description
OCT_TX_BOT	31:0	TX_OCTETS_FRM	Transmitted Octets in Frame without errors [31:0].
OCT_TX_TOP	15:0	TX_OCTETS_FRM	Transmitted Octets in Frame without errors [47:32].
STATS_FRAMES_TX	31:0	FRMS_TX	Frames transmitted without error.
BROADCAST_TX	31:0	BRDCST_TX	Broadcast Frames Transmitted without error.
MULTICAST_TX	31:0	MLTCST_TX	Multicast Frames Transmitted without error.
STATS_PAUSE_TX	15:0	PAUSE_TX	Pause Frames Transmitted
FRAME64_TX	31:0	64B_TX	64 Byte Frames Transmitted without error.
FRAME65_TX	31:0	65TO127B_TX	65 to 127 Byte Frames Transmitted without error.
FRAME128_TX	31:0	128TO255B_TX	128 to 255 Byte Frames Transmitted without error.
FRAME256_TX	31:0	256TO511B_TX	256 to 511 Byte Frames Transmitted without error.
FRAME512_TX	31:0	512TO1023B_TX	512 to 1023 Byte Frames Transmitted without error.
FRAME1024_TX	31:0	1024TO1518B_TX	1024 to 1518 Byte Frames Transmitted without error.
FRAME1519_TX	31:0	1519B_OR_MORE	1519 Bytes or More Frames Transmitted without error.
OCT_RX_BOT	31:0	OCT_RX_BOT	Octets (bottom) received without errors and passed filter [31:0]
OCT_RX_TOP	15:0	OCT_RX_TOP	Octets (top) received without errors and passed filter [47:32]
STATS_FRAMES_RX	31:0	FRMS_RX	Frames Received without error and passed filter.
BROADCAST_RX	31:0	BRDCST_RX	Broadcast Frames Received without errors and passed filter.
MULTICAST_RX	31:0	MLTCST_RX	Multicast Frames Received without errors and passed filter.
STATS_PAUSE_RX	15:0	PAUSE_RX	Pause Frames Received
FRAME64_RX	31:0	64B_RX	64 Byte Frames Received without errors and passed filter.
FRAME65_RX	31:0	65TO127B_RX	65 to 127 Byte Frames Rcvd without errors and passed filter.
FRAME128_RX	31:0	128TO255B_RX	128 to 255 Byte Frames Rcvd without errors and passed filter.
FRAME256_RX	31:0	256TO511B_RX	256 to 511 Byte Frames Rcvd without errors and passed filter.
FRAME512_RX	31:0	512TO1023B_RX	512 to 1023 Byte Frames Rcvd without errors and passed filter.
FRAME1024_RX	31:0	1024TO1518B_RX	1024 to 1518 Byte Frames Rcvd without errors and passed filter.
FRAME1519_RX	31:0	1519B_OR_MORE_RX	1519 Bytes or More Frames Rcvd without errors and passed filter.
STATS_USIZE_FRAMES	9:0	USIZE_RX	Frames received with < 64 bytes in length
STATS_EXCESS_LEN	9:0	OSIZE_RX	Oversized Frames Received
STATS_JABBERS	9:0	JAB_RX	Jabbers Received
STATS_FCS_ERRORS	9:0	FCS_ERR	10-bit count of frames discarded with Ether FCS errors.
STATS_LENGTH_ERRORS	9:0	LGTH_FRM_ERR	10-bit count of frames with Length field not equal to measured length
STATS_RX_SYM_ERR	9:0	RX_SYM_ERR	10-bit count of frames with RX_ER = 1 during reception.
STATS_ALIGN_ERRORS	9:0	ALIGN_ERR	10 bit count of frames discarded with non-integral byte count.

10.4.5.2 Global Packet Classifier Monitoring Control

Table 10-65. Global Packet Classifier Monitoring Settings (PC.)

Register	Packet Classifier Function	Description
CPCR.CPC	Good Packet Count	# received packets forwarded toward a TDM Port or CPU
PCECR.UICPEC	UDP & IP Pkt FCS Error Count	# received packets with UDP & IP checksum errors (see UICECS)
CR1.UICECS	UDP & IP FCS Error Select	Selects whether UICPEC counts UDP, IP or “UDP and IP” checksum errors
SPCR.SPC	Stray Packet Count	# received packets with PW header, but unknown PWID (no BID or OAM BID match)

10.4.5.3 Global RXP Bundle Monitoring Control

Table 10-66. Global RXP Bundle Control Word Change Monitor Settings(G.)

Register	Control Word Function	SAT/CES Bundle	SAT Bundle	HDLC Bundle	Clock-only Bundle	CPU Debug Bundle[1]
GCR.LBCDE	L-bit Change Detect Enable	Yes	Yes	NA	NA	Yes
GCR.RBCDE	R-bit Change Detect Enable	Yes	Yes	NA	Yes	Yes
GCR.MBCDE	M-bit Change Detect Enable	Yes	Yes	NA	Yes	Yes
GCR.FBCDE	Frag-bit Change Detect Enable	Yes	NA	NA	NA	Yes

Notes: [1] When an intended SAT/CES Bundle is programmed to be sent to the CPU the Control Word Change Detect bits can be monitored for debug purposes (this is not a normal CPU Bundle function).

10.4.5.4 Global TXP Packet Queue Monitoring

Table 10-67. Global TXP Output Queue Status Registers (G.)

Status Register	Functional Description	SAT/CES Bundle	HDLC Bundle	Clock-only Bundle	All CPU connection types
TPISR1.TXHPQML	TXP High Priority Queue Max Level[1]	Yes	Yes	Yes	NA
TPISR2.TXLPQML	TXP Low Priority Queue Max Level	Yes	Yes	Yes	NA
TPISR3.TXCQML	TXP CPU Queue Max Level	NA	NA	NA	Yes

Notes: [1] High priority normally is only assigned to SAT/CES/Clock Only Bundles used for Clock Recovery at PW far end.

10.4.5.5 PW Bundle Monitoring

Table 10-68. TXP Bundle Status/Statistics Registers

Bundle/Port Select	Status Register	Status Bits	Functional Description	SAT/CES Bundle	HDLC Bundles	Clock-only Bundles	CPU Debug Bundles
B.BESCR	BESR1	PRHEFC	Bad Rcv HDLC Frame Count	NA	Yes	NA	NA
	BESR2	GPTXC	Good TXP Packet (Ethernet) Count	Yes	Yes	Yes	NA
	BESR3	TXPSFSL	TXP Queue Overflow	Yes	Yes	Yes	NA
Pn.	PTSR1-4	CTSx	TXP CAS in Time Slot x	Yes	NA	NA	NA

Table 10-69. RXP Bundle Status/Statistics Registers[3]

Bundle Select	Register	Status Bits	Function	SAT/CES Bundles	HDLC Bundles	Clock-only Bundles	CPU Debug Bundles
B.BDSCR	B.BDSR1	JBLPDSL	Jitter Buffer Late Packet Discard	Yes	NA	NA	NA
		PDC	RXP Pkt Discard Count[1]	Yes	NA	NA	NA
	B.BDSR2	JBEC	Jitter Buffer Event Count[1]	Yes	NA	NA	NA
	B.BDSR3	JBLL	Jitter Buffer Low Level	Yes	NA	NA	NA
		JBHL	Jitter Buffer High Level	Yes	NA	NA	NA
	B.BDSR4	GPRXC	Ethernet Good RXP Packet Count	Yes	Yes	Yes	NA
	B.BDSR5	SCJPC	Jumped/Lost Packet Count	Yes	Yes	NA	NA
	B.BDSR6	SCRPC	Reorder/Out-of-Window Count	Yes	Yes	NA	NA
	B.BDSR7	SCPSESL	Payload Size/Sequence Error	Yes	Yes	NA	NA
		PLESL	Packet Length Error	Yes	Yes	NA	NA
		SCJBEPDSL	Early Pkt/Buffer overflow Discard	Yes	Yes	NA	NA
		JBCL	Jitter Buffer Current Level	Yes	NA	NA	NA
		LBD	Control Word L-bit	Yes	NA	Yes	NA
		RBD	Control Word R-bit	Yes	NA	Yes	NA
		DMD	Control Word M-bits	Yes	NA	Yes	NA
		FBD	Control Word Frag-bits	Yes	NA	NA	NA
	B.BDSR8	SCMPC	Malformed Packet Count	Yes	NA	Yes	NA
	B.BDSR9	SCRBPC	R-bit Packet Count	Yes	NA	Yes	NA
-	B.GxSRL	CWCDSL	Control Word Change	Yes	NA	Yes	NA
-	JB.GxSRL	JBU	Group Jitter Buffer Underrun/Playout	Yes	NA	NA	NA
-	Pn.PRSR1-4	CTSx	RXP CAS in Time Slot x	Yes	NA	NA	NA

Notes: [1] PC.CR1.PDCC and PC.CR1.JBECC select what conditions are counted by PDC and JBEC (respectively).
[2] G.GCR.IPSE selects whether JB.GxSRL.JBU indicates Underrun or "Underrun and Start of Playout".
[3] The Bundle Status Registers do not function until a Bundle is released from Reset.

10.4.6 SDRAM Settings

Table 10-70. SDRAM Settings (EMI.)

Register bits	Functional Description	Total SDRAM Memory Size				Comments
		128 Mbit	256 Mbit	512 Mbit	1024 Mbit	
DCR2.TRFC	Refresh Pulse Period	0x1F	0x1F	0x1F	0x1F	TRFC * 8ns (0x1F = 248 ns)
DCR2.DCL	CAS Latency	2	2	2	2	CAS Latency = 2
DCR2.DCW	Column Width	2	2	1	0	512 = 2; 1024 = 1; 2048 = 0
DCR2.DMS	Total External Memory	3	2	1	0	128= 3; 256= 2; 512= 1; 1024= 0
DCR2.DRRS	Repeat Refresh Time	0x10	0x10	0x10	0x10	DRRS * 512ns (0x10 = 8.192 us)

Table 10-71. SDRAM Starting Address Assignments (EMI.; all SDRAM sizes)

EMI Register	Description	Contents	Block size	Start Addr (Hex)
BMCR3.PRSO	RXP CPU Queue	512 RXP CPU Packets	8 Mbit	000.0000
BMCR3.PTSO	TXP CPU Queue	512 TXP CPU Packets	8 Mbit	080.0000
BMCR1.TXPSO	TXP Payload Queue	256 TXP Bundle Packet Payloads	16 Mbit	100.0000
BMCR1.TXHSO	TXP Header Descriptors	256 TXP Bundle Header Descriptors	1 Mbit	200.0000
BMCR2.JBSO	Jitter Buffer	256 RXP Bundle Jitter Buffers	to end of SDRAM	210.0000

Table 10-72. Example Max PDV (ms) for various PCT, JBMD and # of TS Combinations

PCT	JBMD Setting (Kbytes)	Number of Timeslots per Bundle						
		32	16	12	8	4	2	1
0.125 ms	256	451	812	1016	1354	2032	2709	3251
	128	224	403	504	672	1008	1344	1612
	64	110	198	248	330	496	661	793
	32	53	96	120	160	240	320	384

1 ms	**256**	500	985	1300	1912	3612	6502	10837
	128	248	489	645	949	1792	3226	5376
	64	122	240	317	467	882	1587	2645
	32	59	116	154	226	427	768	1280
5 ms	**256**	507	1010	1345	2007	3965	7742	14780
	128	252	502	667	997	1967	3842	7332
	64	125	247	330	490	970	1890	3607
	32	60	120	160	237	470	915	1747
10 ms	**256**	INVALID	1015	1350	2020	4015	7930	15485
	128	INVALID	505	670	1005	1995	3935	7685
	64	INVALID	250	330	495	980	1940	3780
	32	INVALID	120	160	240	475	940	1830

Note: “INVALID” means that the packet size would exceed the 2 Kbyte maximum packet size.

It is expected that a 256 Mbit DDR SDRAM will support most applications. With a 256 Mbit device and a JBMD setting of 64 KByte, the system can support up to 110 ms of PDV on any combination of Bundle sizes (1 Timeslot to 32 Timeslots per Bundle). The Packet Creation Time (PCT) and BFD can be set to any valid values. To minimize the S132 process latency the BFD can be set to a 1 frame period. Larger SDRAM devices can be used to support this same application description (e.g. if pricing or availability makes a larger device more desirable). If the maximum PDV can be decreased, for example to 53 ms then the smaller 128 Mbit could be used.

The SDRAM size selection can be complicated because there are so many variables. One approach is to begin by knowing the maximum PDV and the maximum number of Timeslots in a Bundle. With this information Table 10-72 indicates the minimum JBMD. The SDRAM size can then be calculated from:

DDR SDRAM size = (JBMD in Kbytes) * # Bundles + total memory for other queues (e.g. TXP CPU queue)

11 JTAG INFORMATION

This device supports the standard instruction codes SAMPLE/PRELOAD, BYPASS, and EXTEST. Optional public instructions included are HIGHZ, CLAMP, and IDCODE. The device contains the following items, which meet the requirements set by the IEEE 1149.1 Standard Test Access Port (TAP) and Boundary Scan Architecture:

Test Access Port (TAP)	Instruction Register	Boundary Scan Register
TAP Controller	Bypass Register	Device Identification Register

The Test Access Port has the necessary interface pins, namely JTCLK, JTDI, JTDO, JTMS and JTRST_N. Details about the boundary scan architecture and the TAP can be found in IEEE 1149.1-1990, IEEE 1149.1a-1993, and IEEE 1149.1b-1994.

IEEE 1149.1 requires a minimum of two test registers—the bypass register and the boundary scan register. The bypass register is a 1-bit shift register used with the BYPASS, CLAMP, and HIGHZ instructions to provide a short path between JTDI and JTDO. The boundary scan register contains a shift register path and a latched parallel output for control cells and digital I/O cells. DS34S132 BSDL files are available at http://www.maxim-ic.com/tools/bsdl/. An optional test register, the identification register, has also been included in the device design. The identification register contains a 32-bit shift register and a 32-bit latched parallel output. Table 11-1 shows the identification register contents for the DS34S132 device.

Table 11-1. JTAG ID Code

DEVICE	REVISION ID[31:28]	DEVICE CODE ID[27:12]	MANUFACTURER'S CODE ID[11:0]	STD Bit[0]
DS34S132	4'b0001	009Fh	0A1h	1

12 DC ELECTRICAL CHARACTERISTICS

ABSOLUTE MAXIMUM RATINGS

Voltage Range on Any Input, Bi-directional or Open Drain Output Lead with Respect to VSS (except VDD)	-0.5V to +5.5V
Supply Voltage (VDD33) with Respect to VSS	-0.5V to +3.6V
Supply Voltage (VDD18) with Respect to VSS	-0.5V to +2.0V
Ambient Operating Temperature Range	-40°C to +85°C
Junction Operating Temperature Range	-40°C to +125°C
Storage Temperature Range	-55°C to +125°C
Soldering Temperature Range	See IPC/JEDEC J-STD-020A

These are stress ratings only and functional operation of the device at these or any other conditions beyond those indicated in the operation sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods can affect reliability. Ambient Operating Temperature Range is assuming the device is mounted on a JEDEC standard test board in a convection cooled JEDEC test enclosure.

Note 1: The "typ" (typical) values listed below are not production tested.

Note 2: Production tests are done at room temperature and at TA=+85°C. All functionality and parametric values through temperature range are guaranteed by design.

Table 12-1. Recommended DC Operating Conditions (T_j = -40°C to +85°C.)

Parameter	Symbol	Notes	Min	Typ	Max	Units
Input Logic 1	V_{IH}		2.40		5.50	V
Input Logic 0	V_{IL}		-0.30		+0.80	V
SDRAM Input Reference +/- 5%	V_{RF}	1	1.188	1.25	1.313	V
Input Voltage DDR SDRAM Logic 0	V_{IL}		-0.30		V_{RF} - 0.15	V
Input Voltage DDR SDRAM Logic 1	V_{IH}		V_{RF} + 0.15		V_{DDQ} + 0.3	V
Core Digital 3.3V Supply +/- 5%	V_{DD33}		3.135	3.300	3.465	V
SDRAM Core 2.5V Supply +/- 5%	V_{DDP}		2.375	2.500	2.625	V
SDRAM Output 2.5V Supply +/- 5%	V_{DDQ}		2.375	2.500	2.625	V
Core Digital 1.8 V Supply +/- 5%	V_{DD18}		1.710	1.800	1.890	V
SDRAM 1.8 V PLL Supply +/- 5%	A_{VDD}		1.710	1.800	1.890	V
CLAD 1.8 V PLL Supply +/- 5%	C_{VDD}		1.710	1.800	1.890	V

Notes: [1] The value of VRF can be selected by the user to provide optimum noise margins in the system. Typically, the value of VRF is expected to be about 0.5 x VDDQ of the transmitting device and VRF is expected to track variations in VDDQ.

Table 12-2. DC Electrical Characteristics (T_j = -40°C to +85°C.)

Parameter	Symbol	Note	Min	Typ	Max	Units
VDD33 I/O Supply Current (VDD33 = 3.465V)	I_{dd33}	1		60	100	mA
VDDQ I/O + VDDP I/O Supply Current (VDD = 2.625)	I_{ddq}			100	120	mA
VDD18 Supply Current (VDD18 = 1.89)	I_{dd18}			250	350	mA
AVDD Supply Current (AVDD = 1.89)	I_{add}			5	10	mA
CVDD Supply Current (CVDD = 1.89)	I_{cdd}			1	5	mA
Power-Down Current (All DISABLE and power down bits set)	I_{DDD}	1		1		mA
Lead Capacitance	C_{IO}			7		pF
Input Leakage	I_{IL}		-10		+10	μA
Input Leakage	I_{ILP}		-100		+10	μA
Output Leakage (when Hi-Z)	I_{LO}		-10		+10	μA
Output Voltage (I_{OH} = -8.0mA)	V_{OH}		2.4			V
Output Voltage (I_{OL} = +8.0mA)	V_{OL}				0.4	V
Output Voltage (I_{OH} = -16.0mA)	V_{OH}		2.4			V
Output Voltage (I_{OL} = +16.0mA)	V_{OL}				0.4	V
Output Voltage DDR SDRAM (I_{OH} = -8.0mA)	V_{OH}		1.9			V
Output Voltage DDR SDRAM (I_{OL} = +8.0mA)	V_{OL}			0.2	0.4	V

Notes: [1] All outputs loaded with rated capacitance; all inputs between DVDD33 and DVSS; inputs with pull-ups connected to VDD33.

13 AC TIMING CHARACTERISTICS

13.1 CPU Interface

Table 13-1. CPU Interface Timing (VDD = 3.3V ±5%, Tj = -40°C to 125°C.)

SIGNAL	SYMBOL	DESCRIPTION	MIN	TYP	MAX	UNITS	NOTES
SYSCLK		System Clock Frequency	50		85	MHz	
PCS_N, PA, PWR	t_1	Setup Time to SYSCLK active edge	3.5			ns	1,2,4
PCS_N, PA, PWR	t_2	Hold Time from SYSCLK active edge	1			ns	1,2,4
PD[31:0]	t_3	Input Setup Time to SYSCLK active edge	3.5			ns	1,2,4
PD[31:0]	t_4	Input Hold Time from SYSCLK active edge	1			ns	1,2,4
PD[31:0]	t_7	Output Delay from SYSCLK active edge			6	ns	1,2,4
PD[31:0]	t_8	Output Hold from SYSCLK active edge	1			ns	1,2,4
PTA_N	t_9	Output Delay from SYSCLK active edge			6	ns	1,2,3,4
PTA_N	t_{10}	Output Tristate from SYSCLK active edge			6	ns	1,2,3,4
PCS_N	t_{11}	Delay between Consecutive Accesses	1			clock period	
PA	t_{12}	Setup Time to PALE falling edge	11			ns	
PA	t_{13}	Hold Time from PALE falling edge	1.5			ns	
PALE	t_{15}	Width	4			ns	

Notes: [1] The input/output timing reference level for all signals is VDD/2.

Figure 13-1. MPC870-like processor CPU Interface Write Cycle

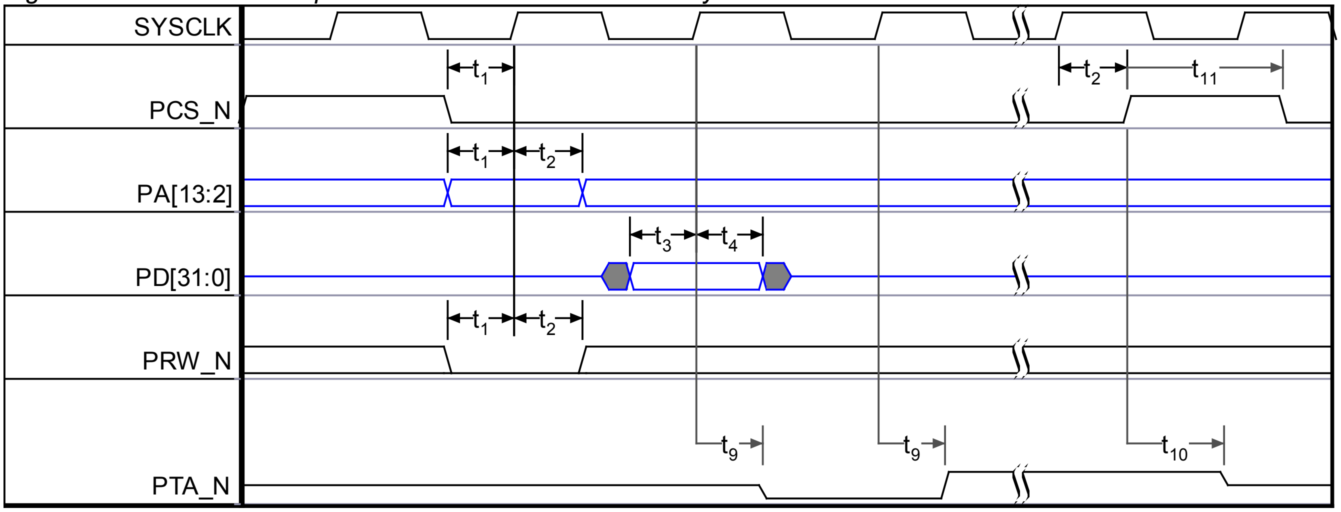


Figure 13-2. MPC870-like processor CPU Interface Read Cycle

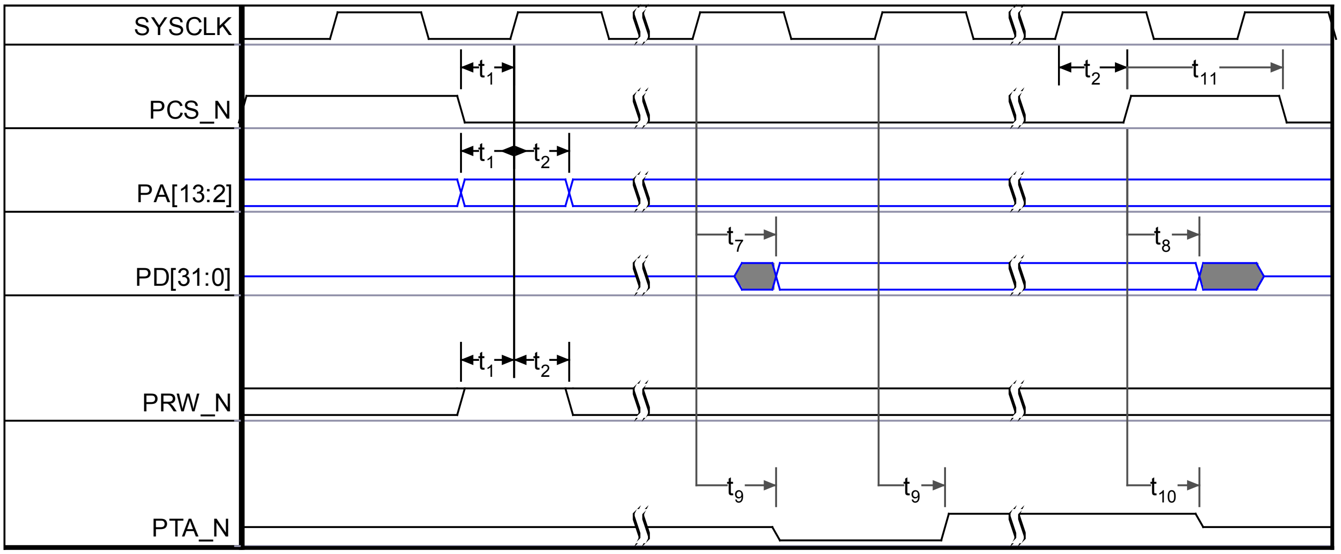


Figure 13-3. MPC8313-like processor CPU Interface Write Cycle

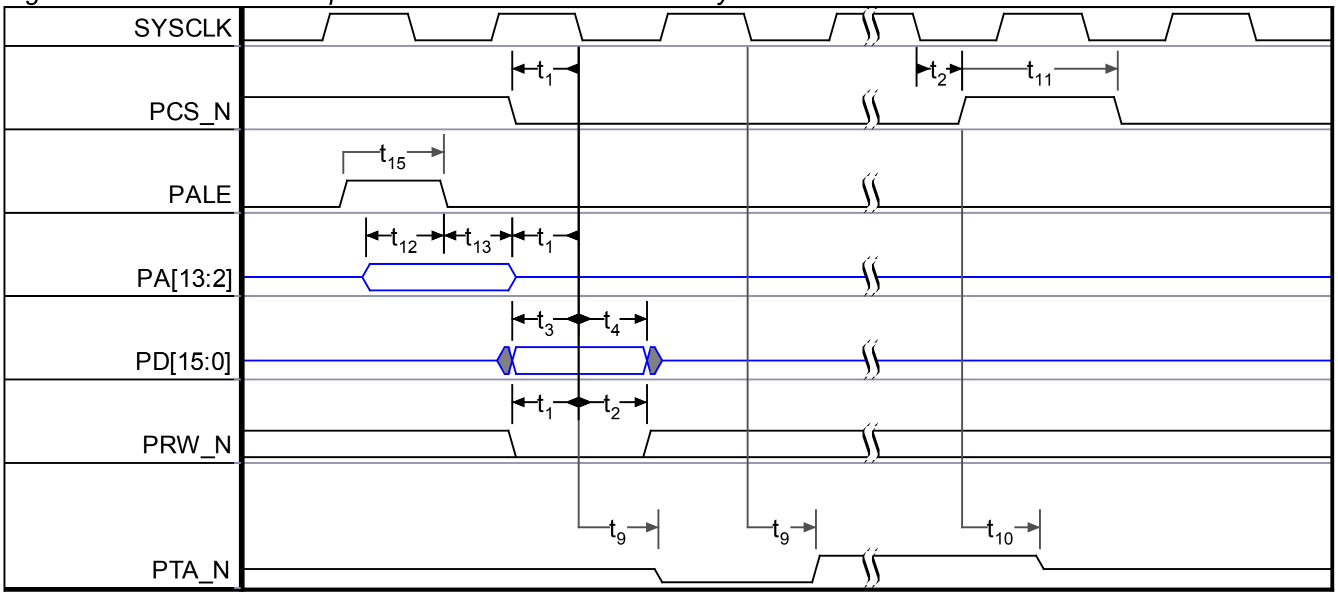
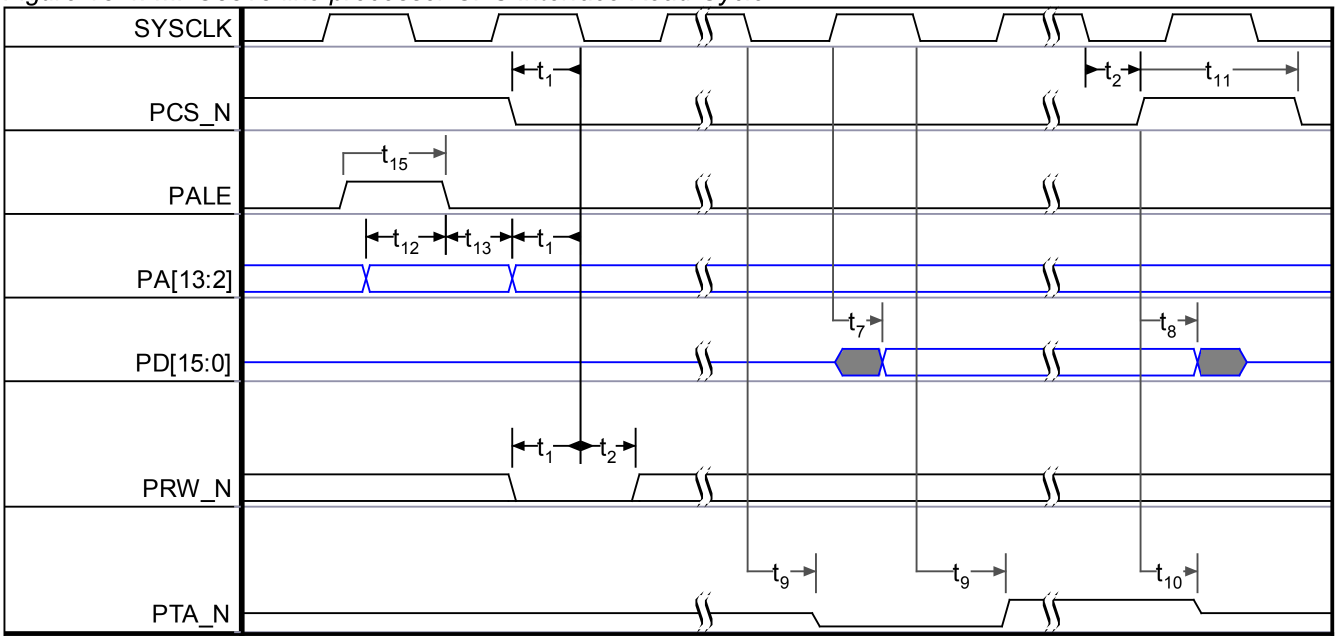


Figure 13-4. MPC8313-like processor CPU Interface Read Cycle



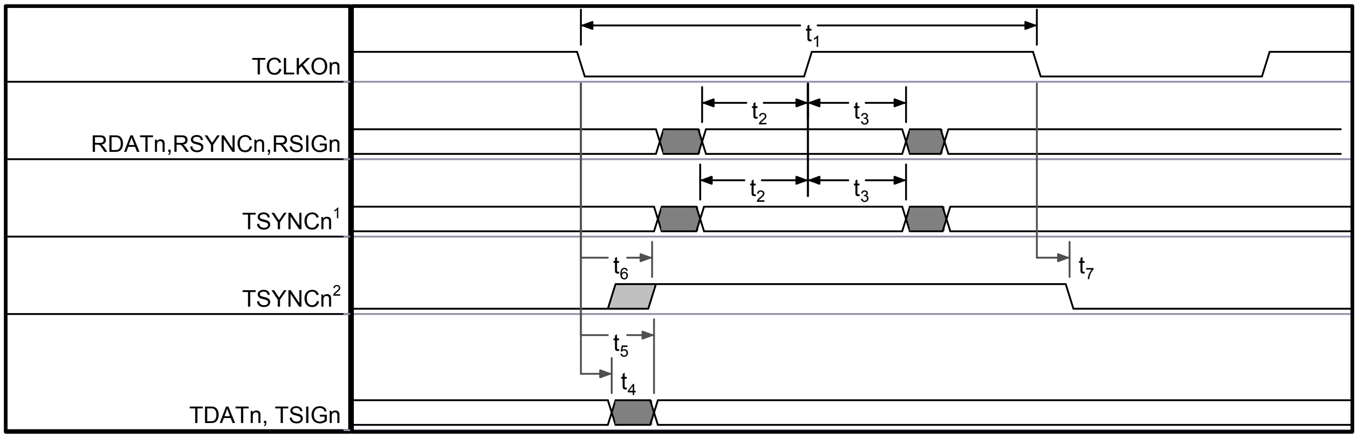
13.2 TDM Interface

Table 13-2. TDM Ports

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
TCLKO Output Period	t_1 t_1		648 488		ns ns	1 2
TSYNC, RSYNC, RDAT, RSIG input setup to TCLKO	t_2	8			ns	3 4
TSYNC, RSYNC, RDAT, RSIG input hold from TCLKO	t_3	2			ns	3 4
TCLKO to TDAT, TSIG output hold	t_4	0			ns	
TCLKO to TDAT, TSIG output valid	t_5			10	ns	
TCLKO to TSYNC output valid	t_6			10	ns	6
TCLKO to TSYNC output hold	t_7	0			ns	6
RCLK Input Period	t_8 t_8		648 488		ns ns	1 2
RSYNC, RDAT, RSIG input setup to RCLK	t_9	8			ns	5
RSYNC, RDAT, RSIG input hold from RCLK	t_{10}	2			ns	5

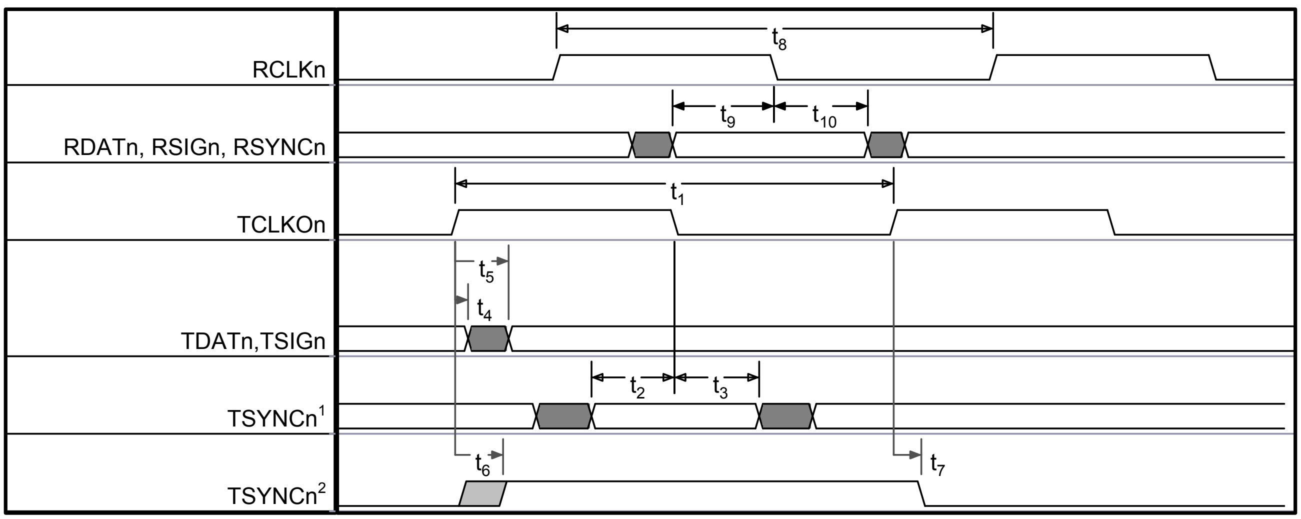
Notes:
1 T1 Mode
2 E1 Mode
3 TSYNC is in input mode
4 RSYNC, RDAT and RSIG are timed relative to TCLKO when using a single clock for the port.
5 RSYNC, RDAT and RSIG are timed relative to RCLK when using two clocks for the port.
6 TSYNC is in output mode
7 The output timing specification for each port signal is with a 30pF load.

Figure 13-5. TDM Port using Single Clock (TCLKO), positive edge timing (RSS = 1, TIES = RIES = 0)



Notes: [1] TSYNC programmed to be an Output
[2] TSYNC programmed to be an Input

Figure 13-6. TDM Port using Two Clock, negative edge timing (RSS = 0, TIES = RIES = 1)



Notes: [1] TSYNC programmed to be an Output
[2] TSYNC programmed to be an Input

13.3 MAC Interface

13.3.1 GMII Interface

Table 13-3. GMII Transmit Timing

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
GTXCLK Output Period GTXCLK Stability	t_1 t_1	 -100	8	 +100	ns ppm	1
GTXCLK Duty Cycle	t_4	40		60	%	
TXD,TXEN, & TXER valid after rising edge GTXCLK	t_2			5.5	ns	
TXD,TXEN, & TXER hold after rising edge GTXCLK	t_3	0.5			ns	

Notes: [1] The rise time and the fall time shall be 1ns measured from $V_{IL\_AC(MAX)}$ = 0.7V to $V_{HI\_AC(MIN)}$ = 1.9V.
The output timing specification for each signal is with a 5 pF load.

Figure 13-7. GMII Transmit Timing

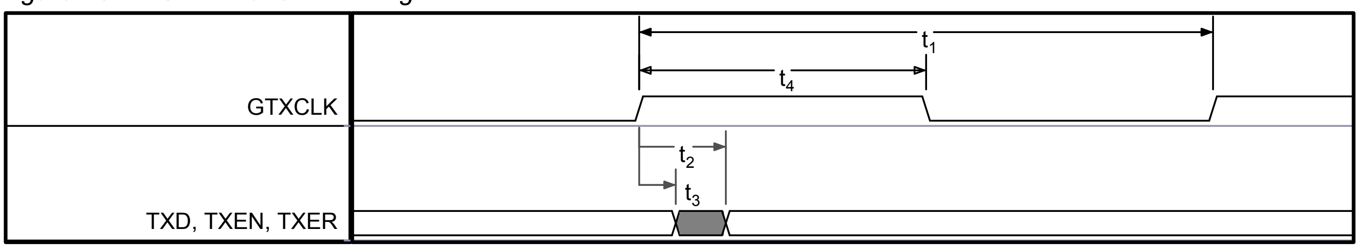
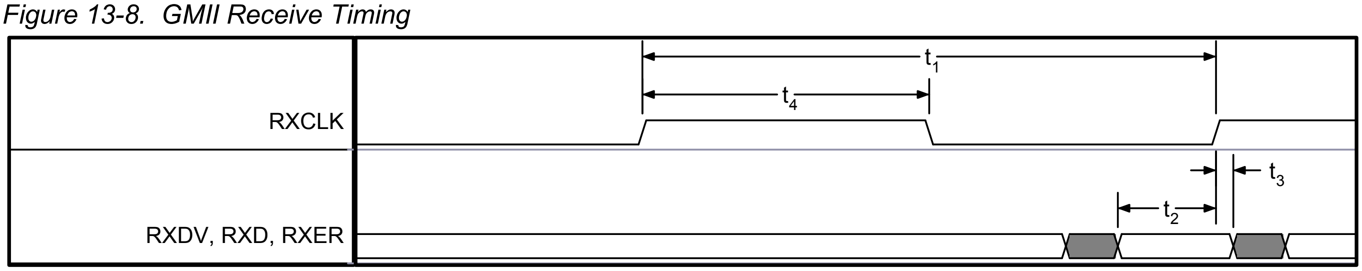


Table 13-4. GMII Receive Timing

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
RXCLK input Period	t_1		8		ns	1
RXCLK Duty Cycle	t_4	40		60	%	
RXDV,& RXD setup prior to RXCLK	t_2	2.0			ns	
RXDV, & RXD hold after RXCLK	t_3			0	ns	

Notes: [1] The rise time and the fall time shall be 1ns measured from $V_{IL\_AC(MAX)}$ = 0.7V to $V_{HI\_AC(MIN)}$ = 1.9V.
[2] The output timing specification for each signal is with a 5pF load.

Figure 13-8. GMII Receive Timing



13.3.2 MII Interface

Table 13-5. MII Transmit Timing

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
TXCLK input Period	t_1		40		ns	2
TXCLK Duty Cycle	t_4	40		60	%	
TXD,TXEN, & TXER valid after rising edge TXCLK	t_2			25	ns	1
TXD,TXEN, & TXER hold after rising edge TXCLK	t_3	0			ns	1

Notes: [1] The output timing specification for each signal is with a 20pF load.
[2] Input low and input high are from VIL_AC(MAX) = 0.8V to VHI_AC(MIN) = 2.0V.

Figure 13-9. MII Transmit Timing

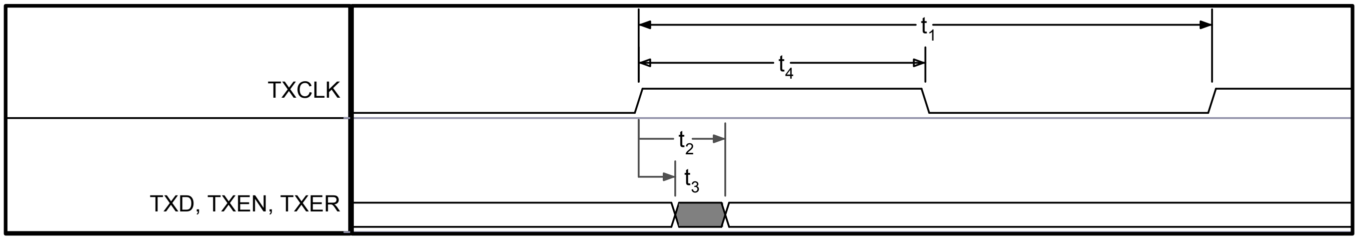
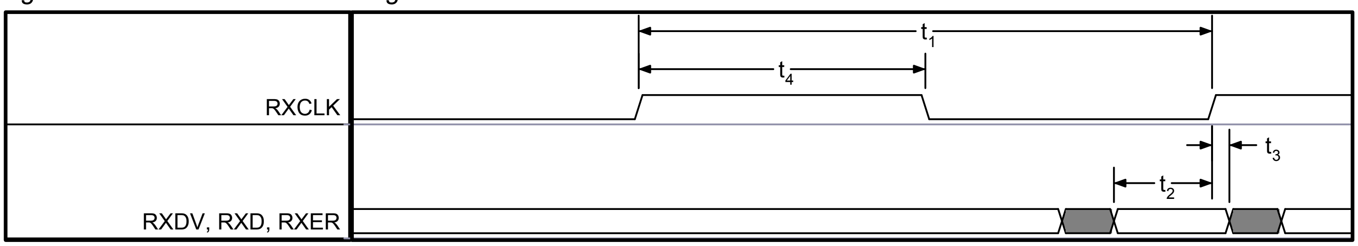


Table 13-6. MII Receive Timing

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
RXCLK input Period	t_1		8		ns	1
TXCLK Duty Cycle	t_4	40		60	%	
RXDV,RXD, & RXER setup prior to RXCLK	t_2	10			ns	1
RXDV, RXD, RXER hold after RXCLK	t_3			0	ns	1

Notes: [1] Input low and input high are from $V_{IL\_AC(MAX)}$ = 0.8V to $V_{HI\_AC(MIN)}$ = 2.0V.

Figure 13-10. MII Receive Timing

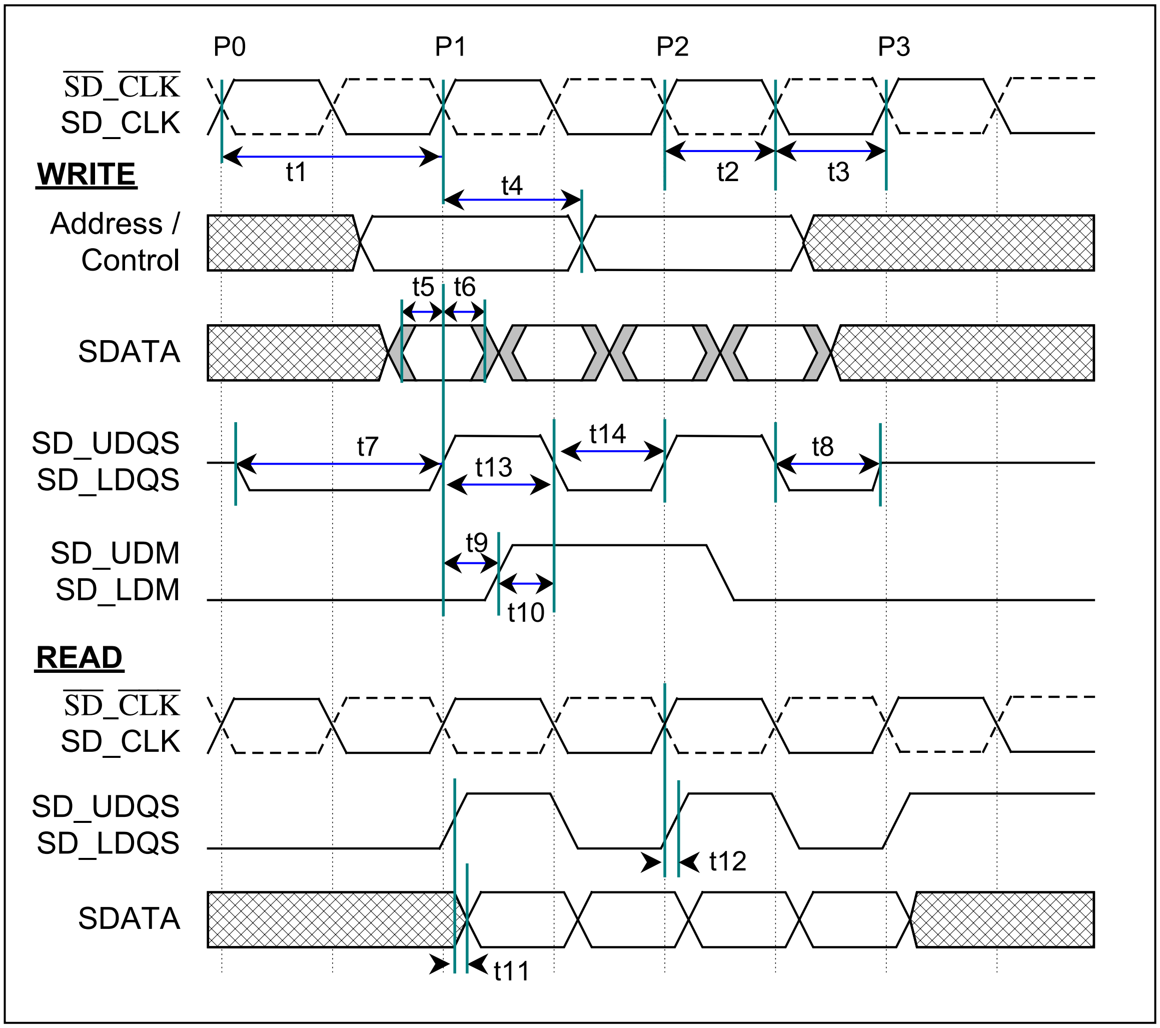


13.4 DDR SDRAM Timing

Table 13-7. DDR SDRAM Interface Timing

Parameter	Symbol	Min	Typ	Max	Units
SD_CLK Output Period	t1	7.5		8.5	ns
SD_CLK Output High Period	t2	3.6		4.4	ns
SD_CLK Output Low Period	t3	3.6		4.4	ns
Address and Control Output Hold Time	t4	3		5	ns
SDDQ Setup to SDUDQS, SDLDQS	t5	0.8			ns
SDDQ Output hold to SDUDQS, SDLDQS	t6	0.8			ns
SDUDQS, SDLDQS Write Preamble	t7	6		10	ns
SDUDQS, SDLDQS Write Postamble	t8	3.2		5.0	ns
SDUDQS, SDLDQS to SDUDM, SDLDM Hold Time	t9	1			ns
SDUDM, SDLDM to SDUDQS, SDLDQS Setup Time	t10	1			ns
SDUDQS, SDLDQS to SDDQ (Read)	t11	-1		+1	ns
SD_CLK to SDLDQS, SDUDQS (Read)	t12	-1		+1	ns
SDLDQS, SDUDQS High Pulse Width (Read)	t13	3.4		4.5	ns
SDLDQS, SDUDQS Low Pulse Width (Read)	t14	3.4		4.5	ns

Figure 13-11. DDR SDRAM Timing



14 PIN ASSIGNMENT

Table 14-1. Pins Sorted by Signal Name

Signal	Ball#	Signal	Ball#	Signal	Ball#	Signal	Ball#
AVDD	A6	PA[13]	AB24	PINT_N	N25	RDAT14	T6
AVSS	A7	PA[2]	Y25	PRW	W23	RDAT15	U6
CMNCLK	AC10	PA[3]	Y24	PRWCTRL	W26	RDAT16	V6
COL	H24	PA[4]	Y23	PTA_N	W24	RDAT17	Y7
CRS	H25	PA[5]	Y22	PWIDTH	W22	RDAT18	AB6
CVDD	AF9	PA[6]	AA26	RCLK0	D6	RDAT19	AA8
CVSS	AF8	PA[7]	AA25	RCLK1	D5	RDAT2	G7
DDRCLK	B7	PA[8]	AA24	RCLK10	R2	RDAT20	AA9
EPHYRST_N	H26	PA[9]	AA23	RCLK11	U3	RDAT21	AB9
ETHCLK	B26	PALE	AB23	RCLK12	R5	RDAT22	AA11
EXTCLK[0]	AA10	PCS_N	W25	RCLK13	U4	RDAT23	AA13
EXTCLK[1]	Y11	PD[0]	N23	RCLK14	V4	RDAT24	AA14
EXTINT	H23	PD[1]	N24	RCLK15	W5	RDAT25	AA15
GTXCLK	K26	PD[10]	R23	RCLK16	Y5	RDAT26	AA16
HIZ_N	D24	PD[11]	R24	RCLK17	AA5	RDAT27	AA17
JTCLK	A22	PD[12]	R25	RCLK18	AD4	RDAT28	Y18
JTDI	C22	PD[13]	R26	RCLK19	AC6	RDAT29	AA20
JTDO	D22	PD[14]	T21	RCLK2	F5	RDAT3	J6
JTMS	B22	PD[15]	T22	RCLK20	AC7	RDAT30	AA21
JTRST_N	B23	PD[16]	T23	RCLK21	AC9	RDAT31	AD22
LIUCLK	AF10	PD[17]	T24	RCLK22	AB11	RDAT4	K6
MDC	D26	PD[18]	T25	RCLK23	AB12	RDAT5	L6
MDIO	D25	PD[19]	T26	RCLK24	AB13	RDAT6	M6
MT[0]	AF23	PD[2]	P21	RCLK25	AC15	RDAT7	N6
MT[1]	AE23	PD[20]	U21	RCLK26	AC16	RDAT8	M4
MT[10]	V20	PD[21]	U22	RCLK27	AC17	RDAT9	M3
MT[11]	U20	PD[22]	U23	RCLK28	AB18	REFCLK	AE9
MT[12]	T20	PD[23]	U24	RCLK29	AB19	RSIG0	F8
MT[13]	R20	PD[24]	U25	RCLK3	G5	RSIG1	F7
MT[14]	N22	PD[25]	U26	RCLK30	AB20	RSIG10	P3
MT[15]	N21	PD[26]	V21	RCLK31	AF22	RSIG11	P4
MT[2]	AD23	PD[27]	V22	RCLK4	H5	RSIG12	P7
MT[3]	AF24	PD[28]	V23	RCLK5	J4	RSIG13	R7
MT[4]	AE24	PD[29]	V24	RCLK6	K4	RSIG14	T7
MT[5]	AF25	PD[3]	P22	RCLK7	M5	RSIG15	U7
MT[6]	AD26	PD[30]	V25	RCLK8	K3	RSIG16	V7
MT[7]	AD25	PD[31]	V26	RCLK9	M2	RSIG17	W7
MT[8]	W21	PD[4]	P23	RDAT0	E8	RSIG18	AA7
MT[9]	W20	PD[5]	P24	RDAT1	E7	RSIG19	Y8
PA[1]	Y26	PD[6]	P25	RDAT10	R3	RSIG2	H7
PA[10]	AA22	PD[7]	P26	RDAT11	R4	RSIG20	Y9
PA[11]	AB26	PD[8]	R21	RDAT12	P6	RSIG21	Y10
PA[12]	AB25	PD[9]	R22	RDAT13	R6	RSIG22	Y12

Signal	Ball#
RSIG23	Y13
RSIG24	Y14
RSIG25	Y15
RSIG26	Y16
RSIG27	Y17
RSIG28	Y19
RSIG29	Y20
RSIG3	J7
RSIG30	Y21
RSIG31	AC22
RSIG4	K7
RSIG5	L7
RSIG6	M7
RSIG7	N7
RSIG8	N4
RSIG9	N3
RST_N	A23
RSYNC0	D7
RSYNC1	E6
RSYNC10	P2
RSYNC11	T3
RSYNC12	P5
RSYNC13	T5
RSYNC14	U5
RSYNC15	V5
RSYNC16	W6
RSYNC17	Y6
RSYNC18	AC5
RSYNC19	AB7
RSYNC2	G6
RSYNC20	AB8
RSYNC21	AD8
RSYNC22	AB10
RSYNC23	AA12
RSYNC24	AB14
RSYNC25	AB15
RSYNC26	AB16
RSYNC27	AB17
RSYNC28	AA18
RSYNC29	AA19
RSYNC3	H6
RSYNC30	AB21
RSYNC31	AE22
RSYNC4	J5
RSYNC5	K5
RSYNC6	L5

Signal	Ball#
RSYNC7	N5
RSYNC8	L3
RSYNC9	N2
RXCLK	G26
RXD[0]	F26
RXD[1]	F25
RXD[2]	F24
RXD[3]	F23
RXD[4]	E26
RXD[5]	E25
RXD[6]	E24
RXD[7]	E23
RXDV	G25
RXERR	G24
SDA[0]	D19
SDA[1]	C19
SDA[10]	C18
SDA[11]	B17
SDA[12]	A17
SDA[13]	B16
SDA[2]	C20
SDA[3]	B20
SDA[4]	A20
SDA[5]	B19
SDA[6]	A19
SDA[7]	B18
SDA[8]	A18
SDA[9]	D18
SDBA[0]	C17
SDBA[1]	D17
SDCAS_N	D15
SDCLK	A16
SDCLK_N	A15
SDCLKEN	B15
SDCS_N	D16
SDDQ[0]	C9
SDDQ[1]	C10
SDDQ[10]	A12
SDDQ[11]	A11
SDDQ[12]	B11
SDDQ[13]	B10
SDDQ[14]	A10
SDDQ[15]	A9
SDDQ[2]	D10
SDDQ[3]	C11
SDDQ[4]	D11

Signal	Ball#
SDDQ[5]	C12
SDDQ[6]	D12
SDDQ[7]	D13
SDDQ[8]	A13
SDDQ[9]	B12
SDLDM	B14
SDLDQS	C13
SDRAS_N	C16
SDUDM	A14
SDUDQS	B13
SDWE_N	C15
SMTI	B8
SMTO	C7
SYSCLK	N26
TCLKO0	A4
TCLKO1	A2
TCLKO10	T1
TCLKO11	V1
TCLKO12	W1
TCLKO13	AA1
TCLKO14	AB1
TCLKO15	AC1
TCLKO16	AD1
TCLKO17	AE1
TCLKO18	AF2
TCLKO19	AF4
TCLKO2	B1
TCLKO20	AF6
TCLKO21	AF7
TCLKO22	AF12
TCLKO23	AF13
TCLKO24	AF14
TCLKO25	AF15
TCLKO26	AF16
TCLKO27	AF17
TCLKO28	AF18
TCLKO29	AF19
TCLKO3	C1
TCLKO30	AF20
TCLKO31	AF21
TCLKO4	D1
TCLKO5	E1
TCLKO6	F1
TCLKO7	H1
TCLKO8	J1
TCLKO9	L1

Signal	Ball#
TDAT0	B5
TDAT1	B3
TDAT10	P1
TDAT11	V2
TDAT12	V3
TDAT13	Y2
TDAT14	Y3
TDAT15	AA3
TDAT16	AB3
TDAT17	AC3
TDAT18	AE3
TDAT19	AE5
TDAT2	D3
TDAT20	AD6
TDAT21	AE7
TDAT22	AE11
TDAT23	AD12
TDAT24	AD13
TDAT25	AD14
TDAT26	AD15
TDAT27	AE17
TDAT28	AD18
TDAT29	AD19
TDAT3	E3
TDAT30	AD20
TDAT31	AD21
TDAT4	F3
TDAT5	G3
TDAT6	G2
TDAT7	J3
TDAT8	J2
TDAT9	N1
TEST_N	G23
TSIG0	C5
TSIG1	C4
TSIG10	T2
TSIG11	U2
TSIG12	T4
TSIG13	W3
TSIG14	W4
TSIG15	Y4
TSIG16	AA4
TSIG17	AB4
TSIG18	AE4
TSIG19	AD5
TSIG2	E4

Signal	Ball#
TSIG20	AD7
TSIG21	AC8
TSIG22	AD11
TSIG23	AC12
TSIG24	AC13
TSIG25	AC14
TSIG26	AD16
TSIG27	AD17
TSIG28	AC18
TSIG29	AC19
TSIG3	F4
TSIG30	AC20
TSIG31	AC21
TSIG4	G4
TSIG5	H4
TSIG6	H3
TSIG7	L4
TSIG8	K2
TSIG9	L2
TSYNC0	B4
TSYNC1	A3
TSYNC10	R1
TSYNC11	U1
TSYNC12	W2
TSYNC13	Y1
TSYNC14	AA2
TSYNC15	AB2
TSYNC16	AC2
TSYNC17	AD2
TSYNC18	AF3
TSYNC19	AF5
TSYNC2	C2
TSYNC20	AE6
TSYNC21	AE8
TSYNC22	AC11
TSYNC23	AE12
TSYNC24	AE13
TSYNC25	AE14
TSYNC26	AE15
TSYNC27	AE16
TSYNC28	AE18
TSYNC29	AE19
TSYNC3	D2
TSYNC30	AE20
TSYNC31	AE21
TSYNC4	E2

Signal	Ball#
TSYNC5	F2
TSYNC6	G1
TSYNC7	H2
TSYNC8	K1
TSYNC9	M1
TXCLK	J26
TXD[0]	L26
TXD[1]	L25
TXD[2]	L24
TXD[3]	L23
TXD[4]	K25
TXD[5]	K24
TXD[6]	J25
TXD[7]	J24
TXEN	J23
TXERR	K23
VDD18	H10
VDD18	H11
VDD18	H12
VDD18	H13
VDD18	H14
VDD18	H15
VDD18	H16
VDD18	H17
VDD18	H18
VDD18	H19
VDD18	H8
VDD18	H9
VDD18	J19
VDD18	J8
VDD18	K19
VDD18	K8
VDD18	L19
VDD18	L8
VDD18	M19
VDD18	M8
VDD18	N19
VDD18	N8
VDD18	P19
VDD18	P8
VDD18	R19
VDD18	R8
VDD18	T19
VDD18	T8
VDD18	U19
VDD18	U8

Signal	Ball#
VDD18	V19
VDD18	V8
VDD18	W10
VDD18	W11
VDD18	W12
VDD18	W13
VDD18	W14
VDD18	W15
VDD18	W16
VDD18	W17
VDD18	W18
VDD18	W19
VDD18	W8
VDD18	W9
VDD33	A1
VDD33	A26
VDD33	AA6
VDD33	AB22
VDD33	AB5
VDD33	AC23
VDD33	AC4
VDD33	AD24
VDD33	AD3
VDD33	AE2
VDD33	AE25
VDD33	AF1
VDD33	AF26
VDD33	B2
VDD33	B25
VDD33	C24
VDD33	C3
VDD33	D23
VDD33	D4
VDD33	E22
VDD33	E5
VDD33	F6
VDD33	M25
VDD33	M26
VDDP	E11
VDDP	E15
VDDP	E18
VDDQ	A21
VDDQ	B9
VDDQ	C21
VDDQ	D14
VDDQ	D20

Signal	Ball#
VDDQ	D9
VDDQ	E12
VDDQ	E16
VDDQ	E19
VDDQ	F14
VREF	E14
VSS	A25
VSS	A5
VSS	AD10
VSS	AD9
VSS	AE10
VSS	AE26
VSS	AF11
VSS	B6
VSS	C25
VSS	C26
VSS	C6
VSS	D8
VSS	E21
VSS	F12
VSS	F18
VSS	F22
VSS	J10
VSS	J11
VSS	J12
VSS	J13
VSS	J14
VSS	J15
VSS	J16
VSS	J17
VSS	J18
VSS	J22
VSS	J9
VSS	K10
VSS	K11
VSS	K12
VSS	K13
VSS	K14
VSS	K15
VSS	K16
VSS	K17
VSS	K18
VSS	K22
VSS	K9
VSS	L10
VSS	L11

Signal	Ball#
VSS	L12
VSS	L13
VSS	L14
VSS	L15
VSS	L16
VSS	L17
VSS	L18
VSS	L22
VSS	L9
VSS	M10
VSS	M11
VSS	M12
VSS	M13
VSS	M14
VSS	M15
VSS	M16
VSS	M17
VSS	M18
VSS	M22
VSS	M23
VSS	M24
VSS	M9
VSS	N10
VSS	N11
VSS	N12
VSS	N13
VSS	N14
VSS	N15
VSS	N16
VSS	N17
VSS	N18
VSS	N9
VSS	P10
VSS	P11
VSS	P12
VSS	P13
VSS	P14
VSS	P15
VSS	P16
VSS	P17
VSS	P18
VSS	P9
VSS	R10
VSS	R11
VSS	R12
VSS	R13
VSS	R14
VSS	R15
VSS	R16
VSS	R17
VSS	R18
VSS	R9
VSS	T10
VSS	T11
VSS	T12
VSS	T13
VSS	T14
VSS	T15
VSS	T16
VSS	T17
VSS	T18
VSS	T9
VSS	U10
VSS	U11
VSS	U12
VSS	U13
VSS	U14
VSS	U15
VSS	U16
VSS	U17
VSS	U18
VSS	U9
VSS	V10
VSS	V11
VSS	V12
VSS	V13
VSS	V14
VSS	V15
VSS	V16
VSS	V17
VSS	V18
VSS	V9
VSSQ	A8
VSSQ	B21
VSSQ	C14
VSSQ	C8
VSSQ	D21
VSSQ	E10
VSSQ	E13
VSSQ	E17
VSSQ	E20
VSSQ	F15

Table 14-2. Pins Sorted by Ball Grid Array - Ball Number

Ball#	Signal
A1	VDD33
A10	SDDQ[14]
A11	SDDQ[11]
A12	SDDQ[10]
A13	SDDQ[8]
A14	SDUDM
A15	SDCLK_N
A16	SDCLK
A17	SDA[12]
A18	SDA[8]
A19	SDA[6]
A2	TCLKO1
A20	SDA[4]
A21	VDDQ
A22	JTCLK
A23	RST_N
A24	
A25	VSS
A26	VDD33
A3	TSYNC1
A4	TCLKO0
A5	VSS
A6	AVDD
A7	AVSS
A8	VSSQ
A9	SDDQ[15]
AA1	TCLKO13
AA10	EXTCLK[0]
AA11	RDAT22
AA12	RSYNC23
AA13	RDAT23
AA14	RDAT24
AA15	RDAT25
AA16	RDAT26
AA17	RDAT27
AA18	RSYNC28
AA19	RSYNC29
AA2	TSYNC14
AA20	RDAT29
AA21	RDAT30
AA22	PA[10]
AA23	PA[9]
AA24	PA[8]
AA25	PA[7]
AA26	PA[6]
AA3	TDAT15
AA4	TSIG16
AA5	RCLK17
AA6	VDD33

Ball#	Signal
AA7	RSIG18
AA8	RDAT19
AA9	RDAT20
AB1	TCLKO14
AB10	RSYNC22
AB11	RCLK22
AB12	RCLK23
AB13	RCLK24
AB14	RSYNC24
AB15	RSYNC25
AB16	RSYNC26
AB17	RSYNC27
AB18	RCLK28
AB19	RCLK29
AB2	TSYNC15
AB20	RCLK30
AB21	RSYNC30
AB22	VDD33
AB23	PALE
AB24	PA[13]
AB25	PA[12]
AB26	PA[11]
AB3	TDAT16
AB4	TSIG17
AB5	VDD33
AB6	RDAT18
AB7	RSYNC19
AB8	RSYNC20
AB9	RDAT21
AC1	TCLKO15
AC10	CMNCLK
AC11	TSYNC22
AC12	TSIG23
AC13	TSIG24
AC14	TSIG25
AC15	RCLK25
AC16	RCLK26
AC17	RCLK27
AC18	TSIG28
AC19	TSIG29
AC2	TSYNC16
AC20	TSIG30
AC21	TSIG31
AC22	RSIG31
AC23	VDD33
AC24	
AC25	
AC26	
AC3	TDAT17

Ball#	Signal
AC4	VDD33
AC5	RSYNC18
AC6	RCLK19
AC7	RCLK20
AC8	TSIG21
AC9	RCLK21
AD1	TCLKO16
AD10	VSS
AD11	TSIG22
AD12	TDAT23
AD13	TDAT24
AD14	TDAT25
AD15	TDAT26
AD16	TSIG26
AD17	TSIG27
AD18	TDAT28
AD19	TDAT29
AD2	TSYNC17
AD20	TDAT30
AD21	TDAT31
AD22	RDAT31
AD23	MT[2]
AD24	VDD33
AD25	MT[7]
AD26	MT[6]
AD3	VDD33
AD4	RCLK18
AD5	TSIG19
AD6	TDAT20
AD7	TSIG20
AD8	RSYNC21
AD9	VSS
AE1	TCLKO17
AE10	VSS
AE11	TDAT22
AE12	TSYNC23
AE13	TSYNC24
AE14	TSYNC25
AE15	TSYNC26
AE16	TSYNC27
AE17	TDAT27
AE18	TSYNC28
AE19	TSYNC29
AE2	VDD33
AE20	TSYNC30
AE21	TSYNC31
AE22	RSYNC31
AE23	MT[1]
AE24	MT[4]

Ball#	Signal
AE25	VDD33
AE26	VSS
AE3	TDAT18
AE4	TSIG18
AE5	TDAT19
AE6	TSYNC20
AE7	TDAT21
AE8	TSYNC21
AE9	REFCLK
AF1	VDD33
AF10	LIUCLK
AF11	VSS
AF12	TCLKO22
AF13	TCLKO23
AF14	TCLKO24
AF15	TCLKO25
AF16	TCLKO26
AF17	TCLKO27
AF18	TCLKO28
AF19	TCLKO29
AF2	TCLKO18
AF20	TCLKO30
AF21	TCLKO31
AF22	RCLK31
AF23	MT[0]
AF24	MT[3]
AF25	MT[5]
AF26	VDD33
AF3	TSYNC18
AF4	TCLKO19
AF5	TSYNC19
AF6	TCLKO20
AF7	TCLKO21
AF8	CVSS
AF9	CVDD
B1	TCLKO2
B10	SDDQ[13]
B11	SDDQ[12]
B12	SDDQ[9]
B13	SDUDQS
B14	SDLDM
B15	SDCLKEN
B16	SDA[13]
B17	SDA[11]
B18	SDA[7]
B19	SDA[5]
B2	VDD33
B20	SDA[3]
B21	VSSQ

Ball#	Signal	Ball#	Signal	Ball#	Signal	Ball#	Signal
B22	JTMS	D21	VSSQ	F20		H2	TSYNC7
B23	JTRST_N	D22	JTDO	F21		H20	
B24		D23	VDD33	F22	VSS	H21	
B25	VDD33	D24	HIZ_N	F23	RXD[3]	H22	
B26	ETHCLK	D25	MDIO	F24	RXD[2]	H23	EXTINT
B3	TDAT1	D26	MDC	F25	RXD[1]	H24	COL
B4	TSYNC0	D3	TDAT2	F26	RXD[0]	H25	CRS
B5	TDAT0	D4	VDD33	F3	TDAT4	H26	EPHYRST_N
B6	VSS	D5	RCLK1	F4	TSIG3	H3	TSIG6
B7	DDRCLK	D6	RCLK0	F5	RCLK2	H4	TSIG5
B8	SMTI	D7	RSYNC0	F6	VDD33	H5	RCLK4
B9	VDDQ	D8	VSS	F7	RSIG1	H6	RSYNC3
C1	TCLKO3	D9	VDDQ	F8	RSIG0	H7	RSIG2
C10	SDDQ[1]	E1	TCLKO5	F9		H8	VDD18
C11	SDDQ[3]	E10	VSSQ	G1	TSYNC6	H9	VDD18
C12	SDDQ[5]	E11	VDDP	G10		J1	TCLKO8
C13	SDLDQS	E12	VDDQ	G11		J10	VSS
C14	VSSQ	E13	VSSQ	G12		J11	VSS
C15	SDWE_N	E14	VREF	G13		J12	VSS
C16	SDRAS_N	E15	VDDP	G14		J13	VSS
C17	SDBA[0]	E16	VDDQ	G15		J14	VSS
C18	SDA[10]	E17	VSSQ	G16		J15	VSS
C19	SDA[1]	E18	VDDP	G17		J16	VSS
C2	TSYNC2	E19	VDDQ	G18		J17	VSS
C20	SDA[2]	E2	TSYNC4	G19		J18	VSS
C21	VDDQ	E20	VSSQ	G2	TDAT6	J19	VDD18
C22	JTDI	E21	VSS	G20		J2	TDAT8
C23		E22	VDD33	G21		J20	
C24	VDD33	E23	RXD[7]	G22		J21	
C25	VSS	E24	RXD[6]	G23	TEST_N	J22	VSS
C26	VSS	E25	RXD[5]	G24	RXERR	J23	TXEN
C3	VDD33	E26	RXD[4]	G25	RXDV	J24	TXD[7]
C4	TSIG1	E3	TDAT3	G26	RXCLK	J25	TXD[6]
C5	TSIG0	E4	TSIG2	G3	TDAT5	J26	TXCLK
C6	VSS	E5	VDD33	G4	TSIG4	J3	TDAT7
C7	SMTO	E6	RSYNC1	G5	RCLK3	J4	RCLK5
C8	VSSQ	E7	RDAT1	G6	RSYNC2	J5	RSYNC4
C9	SDDQ[0]	E8	RDAT0	G7	RDAT2	J6	RDAT3
D1	TCLKO4	E9		G8		J7	RSIG3
D10	SDDQ[2]	F1	TCLKO6	G9		J8	VDD18
D11	SDDQ[4]	F10		H1	TCLKO7	J9	VSS
D12	SDDQ[6]	F11		H10	VDD18	K1	TSYNC8
D13	SDDQ[7]	F12	VSS	H11	VDD18	K10	VSS
D14	VDDQ	F13		H12	VDD18	K11	VSS
D15	SDCAS_N	F14	VDDQ	H13	VDD18	K12	VSS
D16	SDCS_N	F15	VSSQ	H14	VDD18	K13	VSS
D17	SDBA[1]	F16		H15	VDD18	K14	VSS
D18	SDA[9]	F17		H16	VDD18	K15	VSS
D19	SDA[0]	F18	VSS	H17	VDD18	K16	VSS
D2	TSYNC3	F19		H18	VDD18	K17	VSS
D20	VDDQ	F2	TSYNC5	H19	VDD18	K18	VSS

Ball#	Signal
K19	VDD18
K2	TSIG8
K20	
K21	
K22	VSS
K23	TXERR
K24	TXD[5]
K25	TXD[4]
K26	GTXCLK
K3	RCLK8
K4	RCLK6
K5	RSYNC5
K6	RDAT4
K7	RSIG4
K8	VDD18
K9	VSS
L1	TCLKO9
L10	VSS
L11	VSS
L12	VSS
L13	VSS
L14	VSS
L15	VSS
L16	VSS
L17	VSS
L18	VSS
L19	VDD18
L2	TSIG9
L20	
L21	
L22	VSS
L23	TXD[3]
L24	TXD[2]
L25	TXD[1]
L26	TXD[0]
L3	RSYNC8
L4	TSIG7
L5	RSYNC6
L6	RDAT5
L7	RSIG5
L8	VDD18
L9	VSS
M1	TSYNC9
M10	VSS
M11	VSS
M12	VSS
M13	VSS
M14	VSS
M15	VSS
M16	VSS
M17	VSS
M18	VSS
M19	VDD18
M2	RCLK9
M20	
M21	
M22	VSS
M23	VSS
M24	VSS
M25	VDD33
M26	VDD33
M3	RDAT9
M4	RDAT8
M5	RCLK7
M6	RDAT6
M7	RSIG6
M8	VDD18
M9	VSS
N1	TDAT9
N10	VSS
N11	VSS
N12	VSS
N13	VSS
N14	VSS
N15	VSS
N16	VSS
N17	VSS
N18	VSS
N19	VDD18
N2	RSYNC9
N20	
N21	MT[15]
N22	MT[14]
N23	PD[0]
N24	PD[1]
N25	PINT_N
N26	SYSCLK
N3	RSIG9
N4	RSIG8
N5	RSYNC7
N6	RDAT7
N7	RSIG7
N8	VDD18
N9	VSS
P1	TDAT10
P10	VSS
P11	VSS
P12	VSS
P13	VSS
P14	VSS
P15	VSS
P16	VSS
P17	VSS
P18	VSS
P19	VDD18
P2	RSYNC10
P20	
P21	PD[2]
P22	PD[3]
P23	PD[4]
P24	PD[5]
P25	PD[6]
P26	PD[7]
P3	RSIG10
P4	RSIG11
P5	RSYNC12
P6	RDAT12
P7	RSIG12
P8	VDD18
P9	VSS
R1	TSYNC10
R10	VSS
R11	VSS
R12	VSS
R13	VSS
R14	VSS
R15	VSS
R16	VSS
R17	VSS
R18	VSS
R19	VDD18
R2	RCLK10
R20	MT[13]
R21	PD[8]
R22	PD[9]
R23	PD[10]
R24	PD[11]
R25	PD[12]
R26	PD[13]
R3	RDAT10
R4	RDAT11
R5	RCLK12
R6	RDAT13
R7	RSIG13
R8	VDD18
R9	VSS
T1	TCLKO10
T10	VSS
T11	VSS
T12	VSS
T13	VSS
T14	VSS
T15	VSS
T16	VSS
T17	VSS
T18	VSS
T19	VDD18
T2	TSIG10
T20	MT[12]
T21	PD[14]
T22	PD[15]
T23	PD[16]
T24	PD[17]
T25	PD[18]
T26	PD[19]
T3	RSYNC11
T4	TSIG12
T5	RSYNC13
T6	RDAT14
T7	RSIG14
T8	VDD18
T9	VSS
U1	TSYNC11
U10	VSS
U11	VSS
U12	VSS
U13	VSS
U14	VSS
U15	VSS
U16	VSS
U17	VSS
U18	VSS
U19	VDD18
U2	TSIG11
U20	MT[11]
U21	PD[20]
U22	PD[21]
U23	PD[22]
U24	PD[23]
U25	PD[24]
U26	PD[25]
U3	RCLK11
U4	RCLK13
U5	RSYNC14
U6	RDAT15
U7	RSIG15
U8	VDD18
U9	VSS
V1	TCLKO11
V10	VSS
V11	VSS
V12	VSS
V13	VSS
V14	VSS

Ball#	Signal
V15	VSS
V16	VSS
V17	VSS
V18	VSS
V19	VDD18
V2	TDAT11
V20	MT[10]
V21	PD[26]
V22	PD[27]
V23	PD[28]
V24	PD[29]
V25	PD[30]
V26	PD[31]
V3	TDAT12
V4	RCLK14
V5	RSYNC15
V6	RDAT16
V7	RSIG16

Ball#	Signal
V8	VDD18
V9	VSS
W1	TCLKO12
W10	VDD18
W11	VDD18
W12	VDD18
W13	VDD18
W14	VDD18
W15	VDD18
W16	VDD18
W17	VDD18
W18	VDD18
W19	VDD18
W2	TSYNC12
W20	MT[9]
W21	MT[8]
W22	PWIDTH
W23	PRW

Ball#	Signal
W24	PTA_N
W25	PCS_N
W26	PRWCTRL
W3	TSIG13
W4	TSIG14
W5	RCLK15
W6	RSYNC16
W7	RSIG17
W8	VDD18
W9	VDD18
Y1	TSYNC13
Y10	RSIG21
Y11	EXTCLK[1]
Y12	RSIG22
Y13	RSIG23
Y14	RSIG24
Y15	RSIG25
Y16	RSIG26

Ball#	Signal
Y17	RSIG27
Y18	RDAT28
Y19	RSIG28
Y2	TDAT13
Y20	RSIG29
Y21	RSIG30
Y22	PA[5]
Y23	PA[4]
Y24	PA[3]
Y25	PA[2]
Y26	PA[1]
Y3	TDAT14
Y4	TSIG15
Y5	RCLK16
Y6	RSYNC17
Y7	RDAT17
Y8	RSIG19
Y9	RSIG20

Table 14-3. Pin Assignments according to Device Outline

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	
A	VDD33	TCLKO1	TSYNC1	TCLKO0	VSS	AVDD	AVSS	VSSQ	SDDQ[15]	SDDQ[14]	SDDQ[11]	SDDQ[10]	SDDQ[8]	SDUDM	SDCLK_N	SDCLK	SDA[12]	SDA[8]	SDA[6]	SDA[4]	VDDQ	JTCLK	RST_N		VSS	VDD33	**A**
B	TCLKO2	VDD33	TDAT1	TSYNC0	TDAT0	VSS	DDRCLK	SMTI	VDDQ	SDDQ[13]	SDDQ[12]	SDDQ[9]	SDUDQS	SDLDM	SDCLKEN	SDA[13]	SDA[11]	SDA[7]	SDA[5]	SDA[3]	VSSQ	JTMS	JTRST_N		VDD33	ETHCLK	**B**
C	TCLKO3	TSYNC2	VDD33	TSIG1	TSIG0	VSS	SMTO	VSSQ	SDDQ[0]	SDDQ[1]	SDDQ[3]	SDDQ[5]	SDLDQS	VSSQ	SDWE_N	SDRAS_N	SDBA[0]	SDA[10]	SDA[1]	SDA[2]	VDDQ	JTDI		VDD33	VSS	VSS	**C**
D	TCLKO4	TSYNC3	TDAT2	VDD33	RCLK1	RCLK0	RSYNC0	VSS	VDDQ	SDDQ[2]	SDDQ[4]	SDDQ[6]	SDDQ[7]	VDDQ	SDCAS_N	SDCS_N	SDBA[1]	SDA[9]	SDA[0]	VDDQ	VSSQ	JTDO	VDD33	HIZ_N	MDIO	MDC	**D**
E	TCLKO5	TSYNC4	TDAT3	TSIG2	VDD33	RSYNC1	RDAT1	RDAT0		VSSQ	VDDP	VDDQ	VSSQ	VREF	VDDP	VDDQ	VSSQ	VDDP	VDDQ	VSSQ	VSS	VDD33	RXD[7]	RXD[6]	RXD[5]	RXD[4]	**E**
F	TCLKO6	TSYNC5	TDAT4	TSIG3	RCLK2	VDD33	RSIG1	RSIG0				VSS		VDDQ	VSSQ			VSS				VSS	RXD[3]	RXD[2]	RXD[1]	RXD[0]	**F**
G	TSYNC6	TDAT6	TDAT5	TSIG4	RCLK3	RSYNC2	RDAT2																TEST_N	RXERR	RXDV	RXCLK	**G**
H	TCLKO7	TSYNC7	TSIG6	TSIG5	RCLK4	RSYNC3	RSIG2	VDD18	VDD18	VDD18	VDD18	VDD18	VDD18	VDD18	VDD18	VDD18	VDD18	VDD18	VDD18				EXTINT	COL	CRS	EPHYRST_N	**H**

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
J	TCLKO8	TDAT8	TDAT7	RCLK5	RSYNC4	RDAT3	RSIG3	VDD18	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VDD18			VSS	TXEN	TXD[7]	TXD[6]	TXCLK
K	TSYNC8	TSIG8	RCLK8	RCLK6	RSYNC5	RDAT4	RSIG4	VDD18	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VDD18			VSS	TXERR	TXD[5]	TXD[4]	GTXCLK
L	TCLKO9	TSIG9	RSYNC8	TSIG7	RSYNC6	RDAT5	RSIG5	VDD18	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VDD18			VSS	TXD[3]	TXD[2]	TXD[1]	TXD[0]
M	TSYNC9	RCLK9	RDAT9	RDAT8	RCLK7	RDAT6	RSIG6	VDD18	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VDD18			VSS	VSS	VSS	VDD33	VDD33
N	TDAT9	RSYNC9	RSIG9	RSIG8	RSYNC7	RDAT7	RSIG7	VDD18	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VDD18		MT[15]	MT[14]	PD[0]	PD[1]	PINT_N	SYSCLK
P	TDAT10	RSYNC10	RSIG10	RSIG11	RSYNC12	RDAT12	RSIG12	VDD18	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VDD18		PD[2]	PD[3]	PD[4]	PD[5]	PD[6]	PD[7]
R	TSYNC10	RCLK10	RDAT10	RDAT11	RCLK12	RDAT13	RSIG13	VDD18	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VDD18	MT[13]	PD[8]	PD[9]	PD[10]	PD[11]	PD[12]	PD[13]
T	TCLKO10	TSIG10	RSYNC11	TSIG12	RSYNC13	RDAT14	RSIG14	VDD18	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VDD18	MT[12]	PD[14]	PD[15]	PD[16]	PD[17]	PD[18]	PD[19]

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	
U	TSYNC11	TSIG11	RCLK11	RCLK13	RSYNC14	RDAT15	RSIG15	VDD18	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VDD18	MT[11]	PD[20]	PD[21]	PD[22]	PD[23]	PD[24]	PD[25]	U
V	TCLKO11	TDAT11	TDAT12	RCLK14	RSYNC15	RDAT16	RSIG16	VDD18	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VSS	VDD18	MT[10]	PD[26]	PD[27]	PD[28]	PD[29]	PD[30]	PD[31]	V
W	TCLKO12	TSYNC12	TSIG13	TSIG14	RCLK15	RSYNC16	RSIG17	VDD18	VDD18	VDD18	VDD18	VDD18	VDD18	VDD18	VDD18	VDD18	VDD18	VDD18	VDD18	MT[9]	MT[8]	PWIDTH	PRW	PTA_N	PCS_N	PRWCTRL	W
Y	TSYNC13	TDAT13	TDAT14	TSIG15	RCLK16	RSYNC17	RDAT17	RSIG19	RSIG20	RSIG21	EXTCLK[1]	RSIG22	RSIG23	RSIG24	RSIG25	RSIG26	RSIG27	RDAT28	RSIG28	RSIG29	RSIG30	PA[5]	PA[4]	PA[3]	PA[2]	PA[1]	Y
AA	TCLKO13	TSYNC14	TDAT15	TSIG16	RCLK17	VDD33	RSIG18	RDAT19	RDAT20	EXTCLK[0]	RDAT22	RSYNC23	RDAT23	RDAT24	RDAT25	RDAT26	RDAT27	RSYNC28	RSYNC29	RDAT29	RDAT30	PA[10]	PA[9]	PA[8]	PA[7]	PA[6]	AA
AB	TCLKO14	TSYNC15	TDAT16	TSIG17	VDD33	RDAT18	RSYNC19	RSYNC20	RDAT21	RSYNC22	RCLK22	RCLK23	RCLK24	RSYNC24	RSYNC25	RSYNC26	RSYNC27	RCLK28	RCLK29	RCLK30	RSYNC30	VDD33	PALE	PA[13]	PA[12]	PA[11]	AB
AC	TCLKO15	TSYNC16	TDAT17	VDD33	RSYNC18	RCLK19	RCLK20	TSIG21	RCLK21	CMNCLK	TSYNC22	TSIG23	TSIG24	TSIG25	RCLK25	RCLK26	RCLK27	TSIG28	TSIG29	TSIG30	TSIG31	RSIG31	VDD33				AC
AD	TCLKO16	TSYNC17	VDD33	RCLK18	TSIG19	TDAT20	TSIG20	RSYNC21	VSS	VSS	TSIG22	TDAT23	TDAT24	TDAT25	TDAT26	TSIG26	TSIG27	TDAT28	TDAT29	TDAT30	TDAT31	RDAT31	MT[2]	VDD33	MT[7]	MT[6]	AD

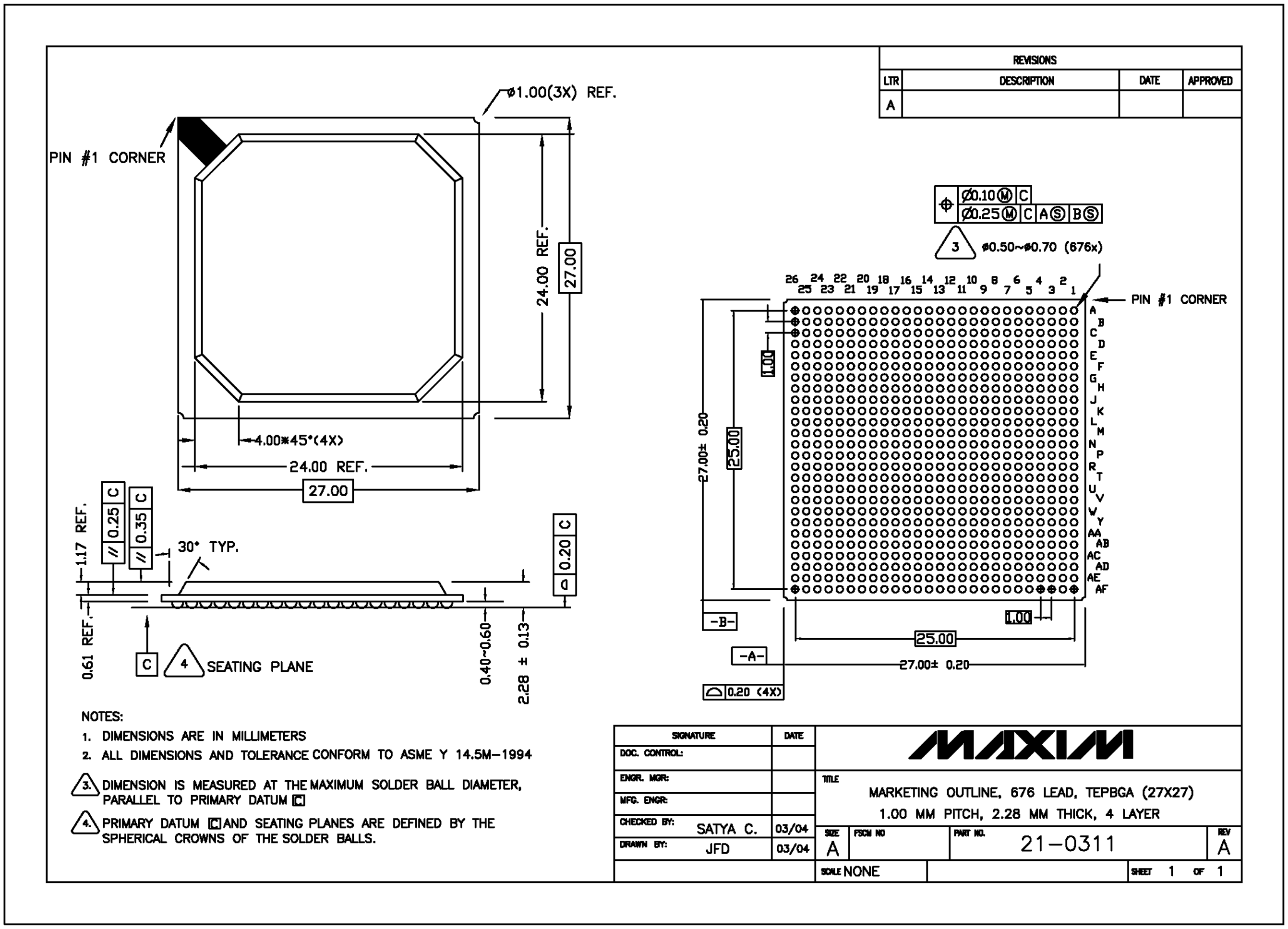
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	
AE	TCLKO17	VDD33	TDAT18	TSIG18	TDAT19	TSYNC20	TDAT21	TSYNC21	REFCLK	VSS	TDAT22	TSYNC23	TSYNC24	TSYNC25	TSYNC26	TSYNC27	TDAT27	TSYNC28	TSYNC29	TSYNC30	TSYNC31	RSYNC31	MT[1]	MT[4]	VDD33	VSS	**AE**
AF	VDD33	TCLKO18	TSYNC18	TCLKO19	TSYNC19	TCLKO20	TCLKO21	CVSS	CVDD	LIUCLK	VSS	TCLKO22	TCLKO23	TCLKO24	TCLKO25	TCLKO26	TCLKO27	TCLKO28	TCLKO29	TCLKO30	TCLKO31	RCLK31	MT[0]	MT[3]	MT[5]	VDD33	**AF**

15 PACKAGE INFORMATION

The package drawing(s) in this data sheet may not reflect the most current specifications. For the latest package outline information and land patterns, go to **www.maxim-ic.com/packages**.

PACKAGE TYPE	PACKAGE CODE	OUTLINE NO.	LAND PATTERN NO.
676 TEPBGA (27mm x 27mm)	V676H+1	21-0311	90-0269

Figure 15-1. 676-Ball TEPBGA



16 THERMAL INFORMATION

Table 16-1. Thermal Package Information

Parameter	Value
Target Ambient Temperature Range	-40°C to +85°C
Die Junction Temperature Range	-40°C to +125°C
Theta-JA, Still Air	14.5°C/W (Note 1)
Theta-JC, Still Air	3.9°C/W
Psi Jt (Junction to Top of Case)	0.23°C/W

Note 1: Theta-JA is based on the package mounted on a four-layer JEDEC board and measured in a JEDEC test chamber.

17 DATA SHEET REVISION HISTORY

REVISION NUMBER	REVISION DATE	DESCRIPTION	PAGES CHANGED
0	7/09	Initial release.	—
1	7/11	Rev A2 – DCR bug fixed.	1,83,127,171

Maxim Integrated Products, 120 San Gabriel Drive, Sunnyvale, CA 94086 408-737-7600