PM8315

TEMUX

DATA SHEET ERRATA

PROPRIETARY AND CONFIDENTIAL

March, 2001

TABLE OF CONTENTS

1. Errata Specific to Revision C TEMUX Devices Only

1.1. Scope

This section of the document is a notification of additional information to Issue 5 of the TEMUX Datasheet, PMC-1981125, dated January 10, 2000. This section of the errata applies to issues specific to revision C TEMUX devices. The following information was full and complete as of March 8, 2001. Please feel free to contact PMC-Sierra's marketing department if any further information is required.

1.2. Data Corruption In Framer Slice #28

Within the revision C devices, it is possible for crosstalk to occur from the signals on the LREFCLK or CMV8MCLK pins into the transmit datapath of framer slice #28. This may happen at extremes of temperature (high), voltage (high) or process variations. The result of this crosstalk is corruption of data on tributary #28. In modes where LREFCLK and CMV8MCLK are not used (ie. applications not using VT/TU mappers or the H-MVIP system side interface), problems will not be encountered. This is not an issue in revision D, E , F and G TEMUX devices.

1.3. Diagnostic Loopback

When performing diagnostic loopback in the revision C TEMUX devices, the telecom drop bus must have valid signals from a SONET framer on the telecom bus. This can be done by sourcing data from a test set with the SONET framer presenting the SPE data to the telecom drop bus or it can be done by putting the SONET framer into diagnostic loopback. This issue has been fixed in revision D, E , F and G TEMUX devices.

1.4. Data Corruption with Clocks Operating Outside Specified Parameters

When operating revision C TEMUX devices with clocks well outside of their specified parameters, it has been noted that the jitter attenuator in the mapper can exhibit unstable characteristics. This can result in data corruption. If data corruption does occur, please check the integrity of all clocks and ensure they are operating within their specified parameters and then reset the mapper block. This has been fixed in revision D, E , F and G TEMUX devices.

1.5. Data Corruption in Mapping Mode due to VTPP

After a hardware or software reset of a revision C TEMUX device, while extremely rare, it is possible the device may enter a state where all tributaries in an STS-1 (AU-3) are corrupted. Symptoms of this issue include a high number of tributary BIP-2 errors, tributary loss-of-pointer and path signal label mismatch. This failure condition is automatically cleared as a result of normal AU pointer movements. This is specific for revision C TEMUX devices and NOT for revision D, E , F and G TEMUX devices.

If this rare failure condition does become an issue, it can be detected after a reset by monitoring a test mode register and cleared manually by toggling a register bit. To do this:

1. Put the VTPP into the appropriate test mode by writing 0Dh to *Register B*.
2. Read *Register C* 100 times. Count of the number of times that bits 0:2 are non-zero. i.e. AND the value in *Register C* with 07h, increment the count if the result is non-zero.
3. If the non-zero count is greater than 50, the VTPP is operating correctly, go to step 6. Otherwise, go to step 4.
4. VTPP is in error state. To clear, toggle the MSB of *Register D* to 1, and then back to 0. i.e. Write an 80h to *Register D*, then write 00h to *Register D*.
5. Repeat starting at step 2 to confirm that workaround was effective.
6. End

In the instructions above, the values for *Register B*, *C*, and *D* will change, depending on the VTPP of interest. Table 1 shows the corresponding values.

Table 1: VTPP Register Cross-Reference

Register	Ingress VTPP	Egress VTPP
B	3241h	3401h
C	3243h	3403h
D	124Eh	140Eh

1.6. Failure to Detect FIFO Over- and Underflow Conditions in VTPP

Under certain conditions the VTPP may fail to detect a tributary FIFO over- or underflow. During the failure, the device will corrupt the payload of affected tributaries, typically between one to ten tributaries being affected simultaneously. These failures have been observed after a sequence of service-affecting events like LOS, LOF and the like that cause non-contiguous jumps in the J1 position of LDC1J1V1 byte. A burst of AU or TU pointer justifications that exceeds the legal rate of one justification in every four frames/multiframe may also cause failures.

This failure is extremely rare and includes symptoms like high numbers of tributary BIP-2 errors, tributary loss-of-pointer and path signal label mismatch. The V5 signal may indicate several V5 bytes per tributary frame. This failure condition is automatically cleared as a result of normal AU and TU pointer movements.

Failures can be detected after every service-affecting event by looking at either detection of errors on the downstream framers or with the detection of high numbers of TU BIP errors (in RTOP) while upstream devices report low numbers of AU BIP errors.

If a failure is detected, it can cleared by toggling the corresponding Reserved (Bit 7) and TU11(Bit 6) bits to a new value and back again in the tributary Configuration and Status Register starting for TU#1 of TUG2#1 in register 1240H.

If further service-affecting events occur, some tributaries may enter a second failure event. Again, high numbers of tributary BIP-2 errors, tributary loss-of-pointer and path signal label mismatch may occur. In addition, the elastic store overflow/underflow error indicator (tributary ESEI bit) is always set. If this ESEI bit remains set after a service-affecting event, toggling the corresponding Reserved (Bit 7) and TU11 (Bit 6) bits to a new value and back again in the tributary Configuration and Status Register starting for TU#1 of TUG2#1 in register 1240H will clear it. This is not an issue in revision D, E , F and G TEMUX devices.

1.7. TFPO Pulses

When configured for DS3 operation, TFPO pulses every 680 TICLK cycles instead of every 85 TICLK cycles as stated in the datasheet. This is specific for revision C TEMUX devices and NOT for revision D, E , F and G TEMUX devices. Revision D, E , F and G TEMUX devices pulse once every 85 TICLK pulses.

1.8. DS3 Framer-Only Mode over Serial Clock and Data Pins

In a revision C TEMUX device, when in DS3 framer-only with the serial clock and data pins on the system side interface, the SBI bus is not tri-stated. As a result Revision C TEMUX cannot simultaneously support a system where DS3 serial clock and data is being used in parallel with other TEMUXs configured for DS3/T1/E1 over SBI. This is an issue with revision C TEMUX devices only. When connected to an SBI bus, the revision D, E , F and G TEMUX devices can drive data in DS3 framer-only mode via the serial clock and data interface without corrupting the data on the SBI bus.

1.9. RFALL in DS3 Line Loopback Mode

When the user sets the Revision C TEMUX in DS3 line loopback mode, RPOS/RDAT and RNEG/RLCV are only sampled on the rising edge of RCLK. The RFALL bit (bit 0 in register 1004) has no effect. When not in line loopback mode, however, RFALL does indicate whether RPOS/RDAT and RNEG/RLCV are sampled on the rising or falling edge of RCLK. This is an issue with revision C devices only. In the revision D, E , F and G TEMUX devices, the RFALL bit does provide proper polarity control for the input RCLK.

1.10. Tri-State Enable

To tri-state the revision C TEMUX devices, the boundary scan tri-state sequence must be used. Holding the three inputs CSB, WRB and RDB low in revision C devices does not guarantee that the outputs are tri-stated. This is an issue with revision C devices only. Holding the three inputs CSB, WRB and RDB low in revision D, E , F and G TEMUX devices will tri-state the outputs.

1.11. RDI Indication

When ERDI bit in Register 1500H: TTOP TU #1 in TUG2 #1 of TUG3 #1, Control is set high, extended RDI is selected. The sampled RDI indication is inserted into bit 5 of the Z7 byte and bit 8 of the V5 byte. The RFI indication is inserted into bit 6 of the Z7 byte and its complement should be inserted into bit 7. When ERDI is set however, bit 7 in Z7 is the inverse of bit 5 of Z7 rather than bit 6 of Z7. In Revision D, E , F and G TEMUX devices, bit 7 of Z7 is the complement of bit 6 of Z7.

Please note that when both RDI and RFI alarms are active on serial remote alarm ports, bit 7 operates properly (since in this case both bits 5 and 6 are the same).

1.12. Incorrect Egress Alarm Indications

It is possible that incorrect egress alarms can be generated in the Revision C TEMUX device. In AU3 mode the device can indicate an intermittent RDI/RFI/REI alarm when it should indicate a constant RDI/RFI/REI alarm. Revision D, E , F and G TEMUX devices indicate constant RDI/RFI/REI alarm correctly.

1.13. AIS Generation in the Receive Direction

The revision C TEMUX does not have the ability to generate an AIS signal in the receive stream upon a received alarm condition such as an LOS, OOF and RED. This feature is desirable in unchannelized DS3 CES design using TEMUX with the AAL1gator-32 and can be accomplished with external logic. Revision D, E , F and G TEMUX devices are able to generate AIS in the receive direction upon received alarm conditions without the use of external logic.

2. Errata Common to Both Revision C and Revision D TEMUX Devices

2.1. Scope

This section of the document is a notification of additional information to Issue 5 of the TEMUX Datasheet, PMC-1981125, dated January 10, 2000. This section of the errata applies issues common in both revision C and revision D TEMUX devices. The following information was full and complete as of March 8, 2001. Please feel free to contact PMC-Sierra's marketing department if any further information is required.

2.2. Transmux Mode

Due to jitter on the derived clocks from the DS3 line, Transmux mode does not work reliably in the revision C and revision D TEMUX devices. This jitter causes repeated reports of DS1 Loss of Pattern / Loss of Frame condition. This was fixed in revisions E , F and G of the TEMUX device by running the derived clocks through the internal jitter attenuator.

2.3. CAS over SBI Mode

When TEMUX is configured for SBI ASYNC mode, the Channel Associated Signaling (CAS) bits can become corrupted. As CAS is only required in SBI SYNC mode, this is not seen to be a problem and will be taken out of the data sheet.

A corruption to CAS in SBI SYNC mode had been observed on very rare occasions (only twice) and cannot even be reproduced on the test bench. Even though the event was not reproducible, the SBI circuitry was modified in revisions E , F and G of the TEMUX to eliminate any chance of corruption to CAS over SBI in SYNC mode.

2.4. Using RGAPCLK Clock in DS Framer Only Mode

When in DS3 framer only mode with a gapped serial clock and data interface, and using the RGAPEN bit of register 1002H is set to create RGAPCLK, same edge (rising update, falling sampling) clocking should not be used with the serial interfaces of different link layer devices. As well, opposite edge (rising update, falling sampling) clocking with the serial interfaces of different link layer devices may not meet the setup and hold timing requirements over the full range of the output propagation delay. For revision C and revision D TEMUX devices, an external delay circuit of approximately 5 nS can be implemented to use opposite edge timing. Revisions E , F and G of the TEMUX device implement appropriate clock gapping such that same edge timing can be used between high density link layer devices without the use of external circuitry.

2.5. Switching Performance Monitors between the Transmit Path and the Receive Path

While the TEMUX is capable of monitoring the performance of the tributaries in either the transmit or receive directions, it is not recommended to switch from transmit performance monitoring to receive performance monitoring or vice versa when the tributary is in service. It is possible to corrupt one or two bits of data in the receive direction when the performance monitoring switch occurs. This was fixed in the revisions E , F and G of the TEMUX device so that switching between transmit performance monitoring and receive performance monitoring will not cause any data corruption.

2.6. Operating in T1 Mode with a 2MHz Backplane Clock

When operating revision C or revision D TEMUXs using the clock and data interfaces on the system side with the device in T1 mode using a jittered 2Mhz backplane clock instead of a 1.544 MHz backplane clock, it is possible to see corruption of data on the T1 tributaries. If the 2Mhz backplane clock is jitter-free, no data corruption will occur. While it has not been possible to verify that this issue exists in TEMUX, it has been verified for other PMC T1 products. To eliminate any possibility that TEMUX is affected, the T1 circuitry in revisions E , F and G of TEMUX has been modified to accept a 2Mhz backplane clock in T1 mode that is not jitter free.

2.7. INSBI/EXSBI Depth Check Interrupts

It is possible within the revision C and revision D TEMUX devices that when the depth check enable bits, DC_RSTEN, in the EXSBI and INSBI Control registers (1710H and 1720H) are cleared, continuous depth check interrupts will be created. Reading the INSBI or EXSBI depth check interrupt status registers will not clear the interrupts. In asynchronous SBI mode with the DC_RSTEN bit set, reading the depth check registers will clear any pending interrupts. In synchronous SBI mode with the DC_RSTEN bit set, reading the depth check registers only clears the interrupts on the EXSBI. The interrupts persist on the INSBI. These interrupts have absolutely no effect on the datapath. In revisions E , F and G of the TEMUX, the SBI circuitry has been modified eliminate the depth check interrupts from going off continuously.

2.8. Data Corruption with TXGAPEN=1 and TDATIFALL=0

When operating revision C and revision D TEMUXs in DS3 Framer only mode using serial clock and data, both the TXGAPEN and TDATIFALL bits in Register 1002H: DS3 Master Unchannelized Interface Options must be set to 1. If TDATIFALL is set to 0, the data in the transmit path can get corrupted. Revisions E , F and G of the TEMUX has been modified so that setting TDATIFALL to 0 in this mode will not cause any data corruption.

2.9. Maximum Read Cycle When Reading INSBI/EXSBI Diagnostic Register

When reading INSBI and EXSBI status registers in the revision C and revision D TEMUX devices, the maximum read cycle must be less than 102nS. When CSB and RDB are held active longer than this, false data can be read back by the microproocessor. This maximum read cycle only applies when reading the registers listed below (please note that these registers are used for diagnostic purposes only whose use may not be required). Revisions E, F and G of the TEMUX device will only require the maximum read cycle specified.

Register 1711H: EXSBI FIFO Underrun Interrupt Status
Register 1712H: EXSBI FIFO Overrun Interrupt Status
Register 1717H: SBI Parity Error Interrupt Status
Register 171EH: EXSBI Depth Check Interrupt Status
Register 1721H: INSBI FIFO Underrun Interrupt Status
Register 1722H: INSBI FIFO Overrun Interrupt Status
Register 1731H: INSBI Depth Check Interrupt Status

3. Errata Common to Revision C, Revision D, Revision E Devices

3.1. Scope

This section of the document is a notification of additional information to Issue 5 of the TEMUX Datasheet, PMC-1981125, dated January 10, 2000. This section of the errata applies to issues common in revision C, revision D and revision E TEMUX devices. The following information was full and complete as of March 8, 2001. Please feel free to contact PMC-Sierra's marketing department if any further information is required.

3.2. Error Generation in VT/TU Mapping Mode on Initialization

In the VT/TU tributary mapping mode, there is a chance that one or more of the tributaries can come up in a state where it is continuously generating bit errors. There is a workaround in place such that this state can be detected by looking at internal registers and can be corrected by running a TJAT (transmit jitter attenuator) centering routine.

This method of detection can be implemented by looking at the transmit mapper(TTMP) FIFO read and write pointers found in reserved registers that will be included in the next revision of the TEMUX register description:

Register 15E6H: TTMP FIFO Depth

Bit	Type	Function	Default
Bit 7		Unused	X
Bit 6		Unused	X
Bit 5		Unused	X
Bit 4		Unused	X
Bit 3	R/W	FIFO_DEPTH[3]	0
Bit 2	R/W	FIFO_DEPTH[2]	0
Bit 1	R/W	FIFO_DEPTH[1]	0
Bit 0	R/W	FIFO_DEPTH[0]	0

Register 15E6H configures the Depth of the FIFOs inside the TTMP block. FIFO_DEPTH increases the depth of the input FIFOs. The value in the register is coded as an unsigned integer and is valid from 0 through 8. Other values are illegal. The effective depth is 8 + the register value.

Reg 15E6H Value	FIFO Depth	Default
8	16	
7	15	
6	14	
5	13	
4	12	
3	11	
2	10	
1	9	
0	8	←

Please note that increasing this value will increase low frequency jitter tolerance at the expense of overall system latency.

Register 15E7H: TTMP MAP Capture Address

Bit	Type	Function	Default
Bit 7		Unused	X
Bit 6	R/W	map capture address[6]	0
Bit 5	R/W	map capture address [5]	0
Bit 4	R/W	map capture address [4]	0
Bit 3	R/W	map capture address [3]	0
Bit 2	R/W	map capture address [2]	0
Bit 1	R/W	map capture address [1]	0
Bit 0	R/W	map capture address [0]	0

This detection facility is used in conjunction with Reg 15E8H to capture specified locations in the TTMP state vector ram (SVRAM) during operation. This register sets the desired SVRAM address.

Register 15E8H: TTMP MAP Control Signals and Bistint Abort

Bit	Type	Function	Default
Bit 7	R/W	map capture enable	0
Bit 6	R/W	map read new	1
Bit 5		Unused	X
Bit 4		Unused	X
Bit 3		Unused	X
Bit 2		Unused	X
Bit 1		Unused	X
Bit 0	R/W	Bistinit_abort	0

When map capture enable is set high, data from the SVRAM address set by 15E7H is available in the registers 15E9H-15F5H. When enabled, data is captured by these registers every time the tributary column is processed. When map read new is high, the current data is available, when low the data from the previousread is available (there are two sets of registers - shift register style).

Bisinit_abort is placed here for convenience. This bit, when set high, will cause the ram initialization on startup to be aborted.

Using the pseudo-code contained in Appendix A: Pseudo Code for Detection of Tributary Errors as a reference, the microprocessor can compare the pointer values of these registers by calculating the difference between them. If a 0 or 1 value is computed, there are errors on the tributary. Once detected, the TJAT centering procedure will move the pointer difference away from the near empty state in the TTMP FIFO, thus clearing the errors in the tributary

Please note that this issue can only occur when TEMUX is being used in VT/TU tributary mapping mode. It cannot be seen when operation in DS3 LIU or DS3 mapping mode. Revision F and Revision G of the TEMUX has been modified so that error generation in the VT/TU mapping mode will not be possible.

3.3. Error Generation in VT/TU Mapping Mode in Normal Operation

If CTCLK or XCLK in TEMUX suffers from a burst of jitter outside specified limits, there may be an extremely small chance that a tributary can come up in the errored state described in the previous section. The detection method described in Appendix A: Pseudo Code for Detection of Tributary Errors can be used to monitor the tributaries, and take any corrective action if required. Alternatively, any downstream bit error indications on a tributary can be used to initiate TJAT re-centering procedure to clear up the bit errors.

Please note that this issue can only occur when TEMUX is being used in VT/TU tributary mapping mode. It cannot be seen when operation in DS3 LIU or DS3 mapping mode. Revision F and Revision G of the TEMUX has been modified so that consistent error generation in the VT/TU mapping mode will not be possible.

4. Errata Common to Revision C, Revision D, Revision E, Revision F Devices

4.1. Scope

This section of the document is a notification of additional information to Issue 5 of the TEMUX Datasheet, PMC-1981125, dated January 10, 2000. This section of the errata applies to issues common in revision C, revision D, revision E and revision F TEMUX devices. The following information was full and complete as of March 8, 2001. Please feel free to contact PMC-Sierra's marketing department if any further information is required.

4.2 Possible Demapper Errors

Due to an issue in the TEMUX demapper there is potential for a loss of data bits, and resulting loss of frame on one or more T1 or E1 links. The effects of this issue are only manifested in the specific case when the demapper receives VT level pointer movements in combination with high-density bit stuffing. This issue will only effect VT/TU mapping mode and does not apply to any DS3 applications. Revision G of the TEMUX has been modified so that demapper errors in VT/TU mapping applications will not be possible.

There is no possibility of errors due to this demapper issue if pointer movements are eliminated. Figure 1 shows the recommended system and board clock configuration to minimize pointer movements. This configuration will not eliminate the potential for VT-level pointer movements already in the receive stream.

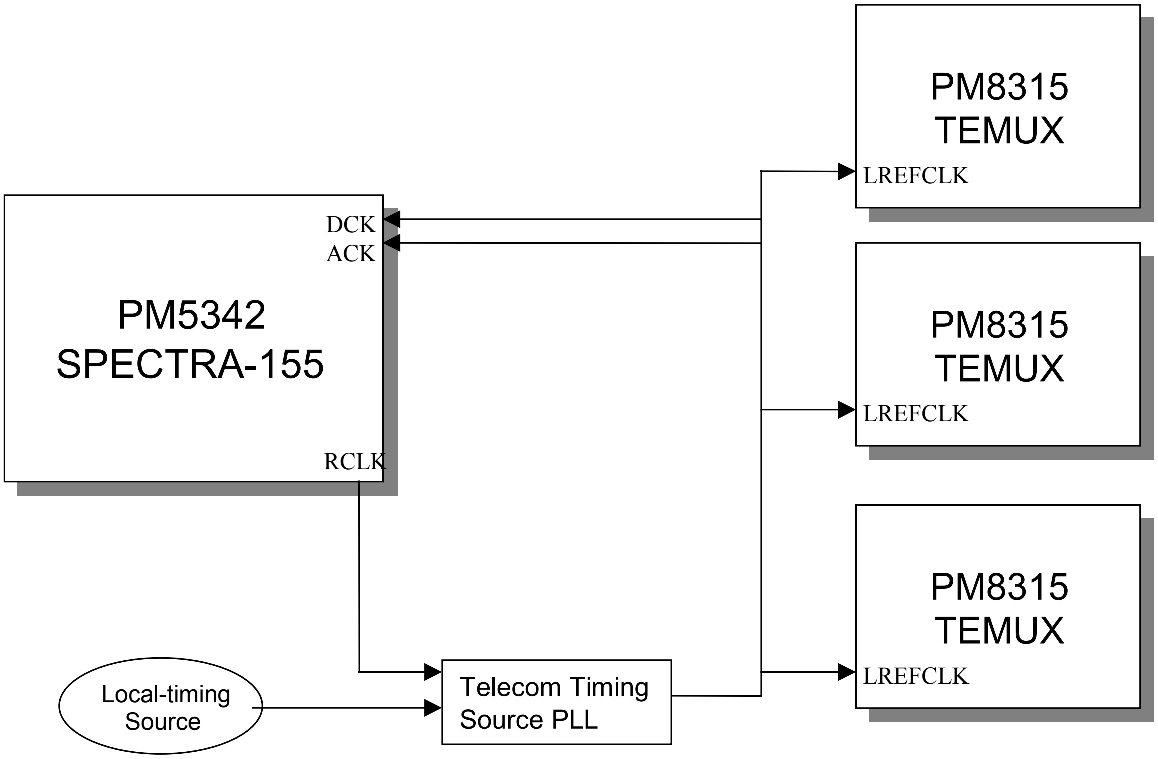


Figure 1 Recommended System and Board Clocking

The required clocking configuration must have both the telecombus looptimed as well as the SPECTRA. To looptime the telecombus the LREFCLK input of the TEMUX must be sourced from the RCLK output of the SPECTRA-155. RCLK also needs to be connected to DCK(D15) and ACK(C12) of the SPECTRA so the entire Telecombus remains synchronous. To looptime the SPECTRA to prevent the generation of outgoing pointer movements, the LTE bit of register 00H must be set. An alternative to looptiming SPECTRA is to input the network clock into TRCLK on the SPECTRA and the Telecom Timing Source PLL.

Note: RCLK is pin L17 of the SPECTRA and is enabled by setting the RCLKEN bit in register 80H. LREFCLK is pin W12 of the TEMUX.

The TEMUX demapper issue is aggravated by high density bit stuffing. High density bit stuffing can appear when the TEMUX mapper is mapping a synchronous tributary, where the transmit T1/E1 clock (SREFCLK, CTCLK, CECLK or CMV8MCLK) is frequency locked to the transmit telecom bus clock (LREFCLK). This issue may also be manifested when the mapped T1/E1 clock has a minute PPM offset relative to the SONET/SDH virtual container **and** there are VT level pointer movements. The error condition is such that the lower the frequency of pointer movements the smaller the permissible PPM offset of the T1/E1 clocks for error-free operation. VT level pointer movements are required for this error case; if pointer movements are minimized with the configuration as shown in Figure 1, the potential for errors will be greatly reduced.

5. Errata Common to Revision C, Revision D, Revision E, Revision F Devices

5.1. Scope

This section of the document is a notification of additional information to Issue 5 of the TEMUX Datasheet, PMC-1981125, dated January 10, 2000. This section of the errata applies to issues common in revision C, revision D, revision E, revision F and revision G TEMUX devices. The following information was full and complete as of March 8, 2001. Please feel free to contact PMC-Sierra's marketing department if any further information is required.

5.2. Switching Modes When Using SBI

When switching a link between synchronous mode and asynchronous modes on the SBI bus, the link must first be disabled. The SBI interface can then be switched to the appropriate mode and the link enabled again. When switching between synchronous and asynchronous modes on the SBI, it is possible to corrupt some of the links if they are not first disabled.

5.3. Duplication of APRM Messages

Triggering a manual performance report generation can intermittently cause a duplication of the 13-byte performance report. This is caused by the manual generation of the report not setting the FIFO empty bit high after reading the message. This is not an issue when setting the TEMUX device to generate the performance report automatically, which is its normal operational mode.

5.4. C-Bit Parity Errors in DS3 Payload Loopback Mode

It is possible to inadvertently generate C-Bit parity errors in DS3 Payload loopback mode by the way its operation is sequenced. When configuring for DS3 framer mode before payload loopback, C-Bit parity errors can occur. To avoid this, configure the TEMUX for payload loopback before configuring for DS3 framer only mode.

5.5. Connecting to the Telecom ADD bus via an External MUX

The pin description for LAC1J1V1 may need clarification. When connecting the TEMUX to the telecom bus via an external MUX (instead of simply tri-stating the bus), the LAC1J1V1 signal should not be muxed. This means that when there are three TEMUXs connected to a Spectra's Telecom bus through the MUX, the LAC1J1V1 signal from only one of the three TEMUXs should be connected to the SPECTRA. The other two LAC1J1V1 signals should remain unconnected. All TEMUXs must have the LOCK0 bit in registers 1202H and 15E5H set the same.

5.6. AIS not inserted by PSL match of 000

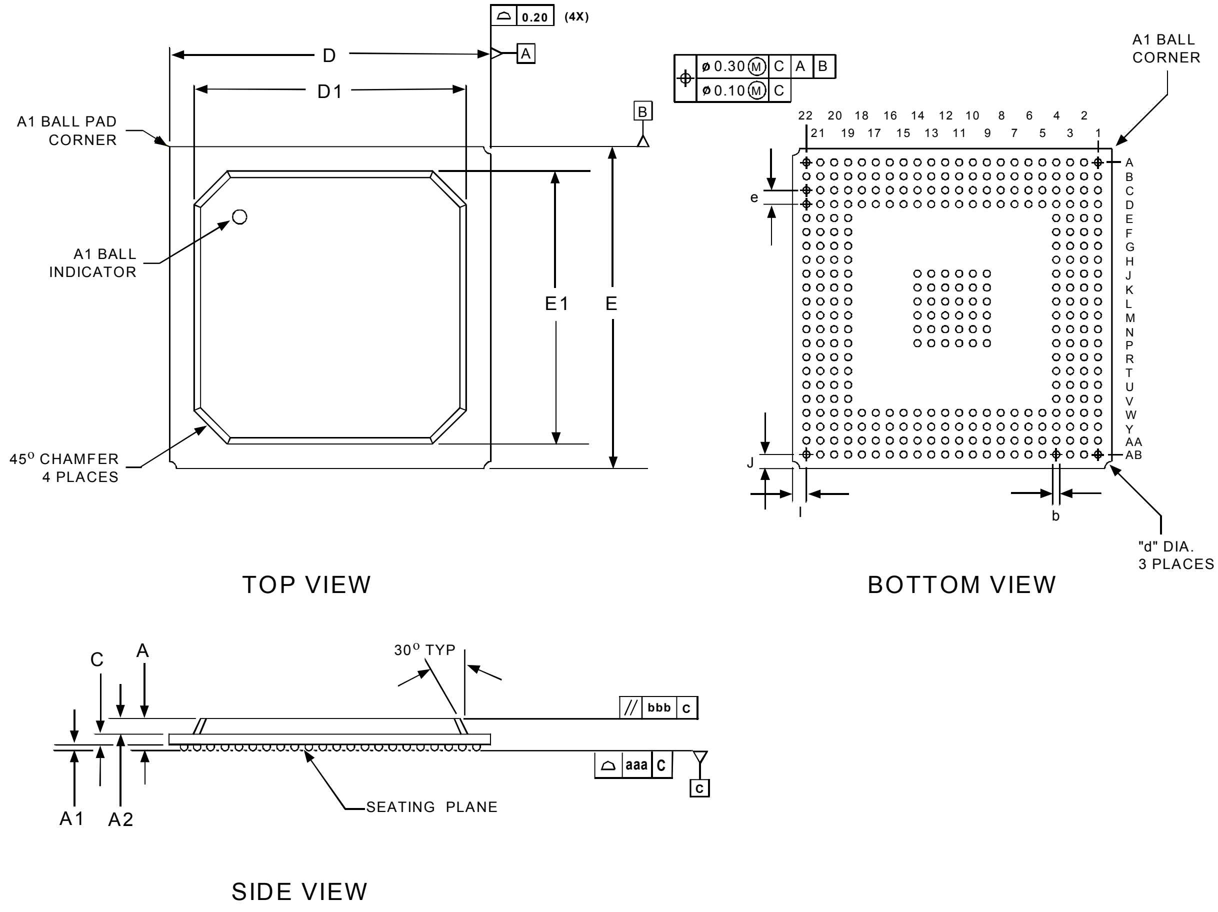
When there is a match of 000 between the expected PSL and the accepted PSL (please refer to table 2 in the datasheet), an AIS cannot be inserted, even if the UNEQAIS bit in Register 1206H: SONET/SDH Master Tributary Alarm AIS Control is set high.

5.7. Payload Loopback in Asynchronous SBI Mode

In asynchronous SBI mode, T1/E1 payload loopback within the TEMUX devices can cause loss of frame and data errors. To allow the data to be looped back without corruption, simply disable the tributaries in the EXSBI SBI block while in payload loopback.

5.8. TEMUX Mechanical Drawing

The height dimensions listed in the Mechanical Information section of the datasheet are incorrect. The dimension in error is the thickness of the package. The following figure contains the correct dimensions.



TOP VIEW

BOTTOM VIEW

SIDE VIEW

NOTES: 1) ALL DIMENSIONS IN MILLIMETER.
2) DIMENSION aaa DENOTES COPLANARITY.
3) DIMENSION bbb DENOTES PARALLEL.

PACKAGE TYPE : 324 PLASTIC BALL GRID ARRAY - PBGA

BODY SIZE : 23 x 23 x 2.28 MM (4 layer)

Dim.	A (2 layer)	A (4 layer)	A1	A2	D	D1	C (2 layer)	C (4 layer)	E	E1	I	J	b	d	e	aaa	bbb
Min	1.82	2.07	0.40	1.12	-	19.00	0.30	0.55	-	19.00	-	-	0.50	-	-	-	-
Nom.	2.03	2.28	0.50	1.17	23.00	19.50	0.36	0.61	23.00	19.50	1.00	1.00	0.63	1.00	1.00	-	-
Max.	2.22	2.49	0.60	1.22	-	20.20	0.40	0.67	-	20.20	-	-	0.70	-	-	0.15	0.35

5.9. Framing up in DS3 Framer-Only Mode

In order to frame up properly in DS3 Frame-Only mode, the SBICLKMODE bit in Register 1001H must be set before the SBI tributaries are enabled. If this is not done first, out of frame alarms can occur.

5.10. Telecom ADD Bus Parity Generation

The TEMUX device cannot generate parity on the Telecom ADD bus when the LAC1J1V1 is set to participate in the egress parity generation. Parity generation only helps check integrity of the bus connections and have no effect on the data path or with control of the device. If egress parity needs to be generated, LAC1J1V1 should not be used.

5.11. Looptiming in Unchannelized DS3 Mode

When operating in DS3 Framer Only mode over the SBI bus, LOOPT in Register 1001H: DS3 Master Data Source does not disable TICLK as the reference timing to the SBI bus. If looptiming in DS3 framer mode is required when using the SBI bus, the receive clock from the external phase lock loop (required for de-jittering) should be multiplexed into the TICLK input and used as the reference for TEMUX in this mode.

5.12. Byte Deletion/Insertion in VT/TU Mapping Mode

While rare, it is possible for jitter on XCLK or CTCLK to cause an overflow or underflow in the transmit mapper (TTMP) FIFO. By design, this will result in the loss of a frame of data. The initialization procedure shown in Appendix A: Pseudo Code for Detection of Tributary Errors will push the FIFO closer to the center such that it can withstand more jitter from these clocks to reduce any impact on the datapath. If the FIFO is close to an underflow or overflow condition and jitter does push it over or under, the FIFO will insert or delete a byte of data, pushing itself away from these states and giving itself at least another 8 UI of margin against future jitter events on XCLK or CTCLK. It is recommended that the initialization sequence shown in Appendix A be used for TEMUX Revision D, E, F and G devices. This sequence is not applicable to Revision C devices.

5.13. TEMUX Initialization Sequencing in VT/TU Mapping Mode

When using TEMUX in VT/TU mapping mode with either version of FREEDM-84 over SBI, care should be taken in the initialization of both devices to make sure they come in the proper states. Please use the recommended initialization sequence below.

1. Initialization of framer timing options

In TEMUX Register 004H+80H*N: T1/E1 Egress Line Interface Configuration select CTCLK as the transmit timing source by setting PLLREF[1:0]=01. Do not bypass the TJAT. This will ensure that when the transmit timing source is referenced to the recovered timing for the framer the transition will be smooth.

2. Initialization of RTDM

Program Register 12E0H: RTDM Pointer Justification Rate Control for a T1 rate control of 0.4 seconds (T1RATE[1:0]=10) and an E1 rate control of 0.4 seconds (E1RATE[1:0]=10) depending if T1 or E1 mode is used.

3. Initialization of RJAT and TJAT for all TU11/TU12 T1/E1 framing configurations

In the initial setting of the RJAT (registers 0011+80*N, 0012+80*N and 0013+80*N) perform both PLL reset and FIFO centering with the following sequence. Note the following applies only to configurations in which CTCLK is equal to line rate (T1 or E1). For other configurations use TTMP Reprov workaround as outlined in Appendix A: Pseudo Code for Detection of Tributary Errors. The following sequence does not apply to Revision C devices. In the RJAT configuration register, set the CENT and LIMIT bits to 0 and the reserved and SYNC bits to 1. Program the RJAT N1 and N2 divisors to 2F and then go back to write a logic 0 to the CENT, LIMIT and SYNC bits and a logic 1 to the reserved bit.
This is done for both T1 and E1. The relative time between the changing of N1 and N2 has to be as short as possible in the RJAT case.

The TJAT for a specific framer should be initialized after the transmit tributary mapper's (TTMP) tributary is provisioned. Then for the initial setting of the TJAT (registers 0015+80*N, 0016+80*N and 0017+80*N) perform both PLL reset and FIFO centering with the following sequence. Program the TJAT N1 and N2 divisors to 2F and then write a logic 0 to the CENT, LIMIT and SYNC bits and a logic 1 to the reserved bit. Next the N2 divisor should be programmed to the value A specified in Table 2depending upon the mode selected (T1 or E1) frequency referenced to TJAT initially. Delay 75ms, then change the N1 value to B specified in Table 2. Delay another 75ms then change the N1 value to C specified in Table 2. Then in the TJAT configuration register, set the CENT and LIMIT bits to 0 and the reserved and SYNC bits to 1. Rewrite the value C into the N1 register and then clear the SYNC bit back to 0. Accurate delays need to be implemented in the TJAT case. The speed is important in order to create long frequency steps that are required to stabilize the TTMP.

It is a good practice to have the TTMP FIFO check from Appendix A: Pseudo Code for Detection of Tributary Errors in place following the TJAT centering procedure to make sure the centering was successful. In addition, keep the relative time between the changing N1 and N2 and clearing SYNC bit as short as possible for both TJAT and RJAT. It is also good practice to have the related FREEDM INSBI and EXSBI blocks deprovisioned at the time. Use Section 5.14as a guide for the TEMUX/FREEDM SBI initialization sequence that must occur after the TJAT and RJAT are properly initialized.

Example software scripts are as follows:

```
// RJAT initial centering procedure
write temux 0013+80*N 22 // RJAT Configuration register: CENT=LIMIT=0,
RESERVED=SYNC=1
write temux 0011+80*N 2F // RJAT N1
write temux 0012+80*N 2F // RJAT N2
write temux 0013+80*N 20 // RJAT Configuration register: CENT=LIMIT=SYNC=0,
RESERVED=1
```

```
// TJAT initial centering procedure
write temux 0017+80*N 20 // TJAT Configuration register: CENT=LIMIT=SYNC=0,
RESERVED=1
write temux 0015+80*N 2F // TJAT N1
write temux 0016+80*N 2F // TJAT N2
write temux 0016+80*N “A” // TJAT N2, see Table 2wait 75ms
write temux 0015+80*N “B” // TJAT N1, see Table 2
wait 75ms
write temux 0015+80*N “C” // TJAT N1, see
write temux 0017+80*N 22 // TJAT Configuration register: CENT=LIMIT=0,
RESERVED=SYNC=1
write temux 0015+80*N “C” // TJAT N1, see Table 2
write temux 0017+80*N 20 // TJAT Configuration register: CENT=LIMIT=SYNC=0,
RESERVED=1
```

Table 2: N1 and N2 values for TJAT initial centering procedure

CTCLK (kHz)	T1			E1		
	A	B	C	A	B	C
1544	0xff	0x70	0xff	N/A	N/A	N/A
2048	N/A	N/A	N/A	0xff	0xa0	0xff

Note: This only applies to configurations where CTCLK = Line Rate.

4. Procedure for Ingress datapath recovery.

Once the datapath has been restored, the RJAT should be re-centered.

In framed E1 mode use the removal of the E1 RED Alarm (located in register 0067+80*N) as the trigger for re-centering the RJAT FIFO. The re-centering procedure will cause a sudden bit slip and thus will cause the E1 framer to have a single change of frame alignment.

In unframed E1 mode use the removal of the TU-LOP alarm (located in register 1240H for TU#1 of TUG2#1) and the TU-AIS alarm (located in register 1241H for TU#1 of TUG2#1) as the trigger for re-centering the RJAT FIFO.

Once the re-centering procedure has been completed an interval of 50ms is required before, optionally, switching back to looptimed mode.

An example software script is as follows:

```
// RJAT re-centering procedure
write temux 0013+80*N 22 // RJAT Configuration register: CENT=LIMIT=0,
RESERVED=SYNC=1
```

```
write temux 0011+80*N 2F // RJAT N1
write temux 0013+80*N 20 // RJAT Configuration register: CENT=LIMIT=SYNC=0, RESERVED=1
```

After the RJAT initialization is complete the TEMUX INSBI should be reset followed by the FREEDM EXSBI.

```
// Ingress SBI Reset
write temux INSBI(trib) ENBL 1 // re-writes a logic 1 to the enable bit for the selected tributary causing a reset
write freedm EXSBI(trib) ENBL 1 // re-writes a logic 1 to the enable bit for the selected tributary causing a reset
```

5. Algorithm for datapath corruption restoration when link is internally timed:

With the E1 link intemally timed, upon detection of datapath corruption with either the E1 RED Alarm in E1 framed mode or with TU-LOP and TU-AIS alarms in E1 unframed mode, the RJAT centering procedure should be used before bringing the link back up.

A sample algorithm is as follows:

```
#define data_corruption_event // either framed or unframed as noted above processed by an interrupt service routine
#define re_centerRJAT() // Re-centering procedure as shown above
#define declare_link_up // Routine for bringing up a link
#define declare_link_down // Routine for tearing down a link
#define ingress_SBI_reset  // Routine for resetting a tributary on the ingress SBI bus

if (!data_corruption_event) {
  re_centerRJAT(framer)
  wait 50ms
  ingress_SBI_reset(trib)
  declare_link_up(framer)
  } else {
     declare_link_down(framer)
     }
```

6. Algorithms for switching from internally timed to looptimed and vice versa

When switching from looptimed to internally timed mode, or vice-versa, the initialization sequence is important. Once the desired mode has been set, the TJAT should be re-centered using the same procedure as noted above and the FREEDM should be re-enabled. The TTMP pointer procedure listed in Appendix A: Pseudo Code for Detection of Tributary Errors should also be followed.

Example software scripts are as follows:

```
// TJAT re-centering procedure
re_centerTJAT( framer ) {
write temux 0017+80*N 20 // TJAT Configuration register: CENT=LIMIT=SYNC=0,
RESERVED=1
write temux 0015+80*N 2F // TJAT N1
write temux 0016+80*N 2F // TJAT N2
write temux 0016+80*N “A” // TJAT N2, see Table 3
wait 75ms
write temux 0015+80*N “B” // TJAT N1, see Table 3
wait 75ms
write temux 0015+80*N “C” // TJAT N1, see Table 3
write temux 0017+80*N 22 // TJAT Configuration register: CENT=LIMIT=0,
RESERVED=SYNC=1
write temux 0015+80*N “C” // TJAT N1, see Table 3
write temux 0017+80*N 20 // TJAT Configuration register: CENT=LIMIT=SYNC=0,
RESERVED=1
}
```

Table 3: N1 and N2 values for TJAT re centering procedure

CTCLK (kHz)	**T1**			**E1**		
	A	**B**	**C**	**A**	**B**	**C**
1544	0xff	0x70	0xff	N/A	N/A	N/A
2048	N/A	N/A	N/A	0xff	0xa0	0xff

Note: This only applies to configurations where CTCLK = Line Rate.

```
//FREEDM INSBI re-enabling procedure
re_enable_frd_INSBI(tributary) {

set trib framer
toggle_ENBL(trib)

}

looptime(framer) {

  write temux 0x04+(0x80*$framer) = 0x62
  re_centerTJAT(framer)
  re_enable_frd_INSBI(tributary)
  check_TTMP_pointers(framer)
}

internaltime(framer) {

  write temux 0x04+(0x80*$framer) = 0x61
  re_centerTJAT(framer)
  re_enable_frd_INSBI(tributary)
  check_TTMP_pointers(framer)
}
```

7. Algorithm for datapath corruption and restoration when link is looptimed:

With the E1 link looptimed, upon detection of datapath corruption with either the E1 RED Alarm in E1 framed mode or with TU-LOP and TU-AIS alarms in E1 unframed mode, the transmit timing reference to TJAT must be changed to the CTCLK source. This occurs because the recovered clock for that link is not valid. Thus when the datapath is restored not only must the RJAT re-centering procedure be used before bringing the link back up but the TJAT re-centering procedure should be used after referencing it back to the recovered clock.

A sample algorithm is as follows:

```
#define data_corruption_event // either framed or unframed as noted above processed by an
interrupt service routine
#define re_centerRJAT() // Recentering procedure as shown above
#define declare_link_up() // Routine for bringing up a link
#define declare_link_down() // Routine for tearing down a link
#define looptime() // write PLLREF[1:0]=10 of register 004H+80H*N
#define internaltime() write PLLREF[1:0]=01 of register 004H+80H*N

if (!data_corruption_event) {
  re_centerRJAT(framer)
  wait_50ms
  looptime(framer)  // includes TJAT re-center, see above
  declare_link_up(framer)
    } else {
      wait 15ms
      internaltime(framer) // includes TJAT re-center, see above
      declare_link_down(framer)
      }
```

5.14. SBI Initialization Sequencing

When using TEMUX with SBI over layer 2 devices like either version of FREEDM-84, care should be taken in the initialization of both devices to make sure they come in the proper states. Please use the recommended initialization sequence below.

5.14.1. Initialization of the SBI Extract Threshold

The SBI Extract Depth must be programmed correctly. In FREEDM-84, Register 0x5EC and 0x5E8 sets the fill levels of the SBI FIFO before data is read out for E1 and T1 modes respectively. Depending on the mode, E1 or T1, the corresponding register (0x5EC or 0x5E8) should be programmed to 78H.

Register 0x5EC : SBI EXTRACT MIN_THR and MAX_THR for E1

Bit	Type	Function	Default
Bit 31 to 8	R/W	Reserved	0000H
Bit 7	R/W	MIN_THR_E1[3]	0
Bit 6	R/W	MIN_THR_E1[2]	0
Bit 5	R/W	MIN_THR_E1[1]	1
Bit 4	R/W	MIN_THR_E1[0]	0
Bit 3	R/W	MAX_THR_E1[3]	1
Bit 2	R/W	MAX_THR_E1[2]	1
Bit 1	R/W	MAX_THR_E1[1]	0
Bit 0	R/W	MAX_THR_E1[0]	1

MIN_THR_E1[3:0]:

The Minimum Threshold for E1 bits (MIN_THR_E1[3:0]) specify the FIFO depth below which a slow down request is made from the EXSBI to the PISO block.

MAX_THR_E1[3:0]:

The Maximum Threshold for E1 bits (MAX_THR_E1[3:0]) specify the FIFO depth which when exceeded will cause a speed up request from the EXSBI to the PISO block to be made.

Register 0x5E8 : SBI EXTRACT MIN_THR and MAX_THR for T1

Bit	Type	Function	Default
Bit 31 to 8	R/W	Reserved	0000H
Bit 7	R/W	MIN_THR_T1[3]	0
Bit 6	R/W	MIN_THR_T1[2]	0
Bit 5	R/W	MIN_THR_T1[1]	1
Bit 4	R/W	MIN_THR_T1[0]	0
Bit 3	R/W	MAX_THR_T1[3]	1
Bit 2	R/W	MAX_THR_T1[2]	1
Bit 1	R/W	MAX_THR_T1[1]	0
Bit 0	R/W	MAX_THR_T1[0]	1

MIN_THR_T1[3:0]:

The Minimum Threshold for T1 bits (MIN_THR_T1[3:0]) specify the FIFO depth below which a slow down request is made from the EXSBI to the PISO block.

MAX_THR_T1[3:0]:

The Maximum Threshold for T1 bits (MAX_THR_T1[3:0]) specify the FIFO depth which when exceeded will cause a speed up request from the EXSBI to the PISO block to be made.

5.14.2. SBI initialization sequence

When initially configuring the Egress path of the TEMUX and layer 2 device (like FREEDM-84) combination, the INSBI of the layer 2 device should be the last portion of the tributary path to be enabled. If the layer 2 device INSBI tributary is enabled before the TEMUX EXSBI tributary, there is chance that the layer 2 device INSBI could generate inadvertent errors (ie packet errors). Enabling the layer 2 device INSBI after the TEMUX EXSBI ensures correct operation.

5.15. Use of HDLC Controller in E1 Mode

When using the internal HDLC controllers in E1 mode there are some restrictions to be aware of:

- If data is inserted into a timeslot from the internal HDLC controller and the previous timeslot has an idle code byte inserted from the TPSC, the last two bits of the idle code can be corrupted. This means that if timeslot 4 has data inserted from the HDLC controller, and an idle code has been inserted in timeslot 3 from the TPSC, the least significant two bits of timeslot 3 can be corrupted. It is recommended that HDLC traffic be inserted from the backplane rather than the internal controller if idle codes are being transmitted in the preceding timeslot.

- If data is inserted into timeslot 1 from the internal HDLC controller, the least significant bit in timeslot 0 for NFAS frames only (i.e., Sa8) can be corrupted if configured to come from the backplane. The National Use Bits codeword, however, operates correctly on Sa8 if enabled. It is recommended that timeslot 1 not be used for HDLC traffic inserted from the internal controller.

- In normal operation, if a timeslot is configured for both HDLC transmission and idle code insertion, HDLC is supposed to be transmitted and the idle code ignored. It is possible in TEMUX for the last two bits of the HDLC data to be corrupted. To get around this, simply disable idle code insertion for that timeslot when HDLC data is being transmitted.

- Inserting HDLC data into the National bits Sa8, Sa7, Sa6, or Sa5 can cause the neighboring more significant bit to be corrupted. For example, if HDLC is inserted into Sa7, then Sa6 can be corrupted, but only if Sa6 is inserted from the backplane. The National bits are always inserted error-free when they are generated via the National Bits Codeword register of the E1-TRAN block.

- Inserting HDLC data into the Si bit in TS0 (i.e., the international bit) can cause the least significant two bits of an IDLE code in TS31 to be corrupted.

5.16 VT-AIS Tributary Corruption

An issue in the TEMUX demapper results in the VT-AIS causing corruption to the previous tributary of SPE#3 only. To circumvent this issue apply the software workaround outlined in Appendix B: SPE Configuration to Prevent TU-AIS Data Corruption.

6. Errata to the PMC-1990145 TEMUX Register Description

6.1. Scope

This section of the document is a notification of additional information to Issue 3 of the TEMUX Register Description, PMC-1990145, dated December 22, 1999. The following information was full and complete as of April 14, 2000. Please feel free to contact PMC-Sierra's marketing department if any further information is required.

6.2. Register Description Issues

These registers are stated incorrectly in the Register Description and should be changed as follows:

- Bit 2 and bit 3 Register 0026: Master Interrupt Source SBI should read as defaulting to 0 and not to 1 as stated.

- Bit 2 in Registers 1309H, 1311H, 1319H, 1321H, 1329H, 1331H: RTOP, TU #1 in TUG2 #2 to TUG2 #7, Configuration and Alarm Status should read as defaulting to X and not to 1 as stated.

- Bit 6 in Register 12E3H: RTDM Indirect Time Switch Tributary RAM Status and Control should read as defaulting to 1 and not to 0 as stated.

- Bit 5 in Register 006H+80H*N: T1/E1 Egress Serial Interface Mode Select should read ETS15EN/ECCSEN instead of just ECCSEN to reflect that it controls insertion from CCSED into timeslot 15 when in E1 mode. Similarly, bit 6 should read ETS16EN instead of ETS15EN to reflect that it controls the insertion from CCSED into timeslot 16 in E1 mode.

- The IIDLE bit in VTPP Egress Configuration and Status Registers 1400H, 1402H, 1404H, 1406H, 1408H, 140AH, 140CH, 1410H, 1412H, 1414H, 1416H, 1418H, 141AH, 141CH 1420H, 1422H, 1424H, 1426H, 1428H, 142AH, 142CH, 1430H, 1432H, 1434H, 1436H, 1438H, 143AH and 143CH should not be used and should be programmed to their default value of 0. Insertion of all ones in the egress direction is accomplished using the IPAIS bit in these registers.

- The TAISEN bit in Register 003H+80H*N: T1/E1 Alarm Configuration states that the TAISEN bit enables generation of an all ones AIS alarm in the egress tributary. This bit description should read RESERVED and should not be used as AIS is enabled in the appropriate MX12 AIS Insert Registers or the appropriate TRAP AIS enable registers for the different TUGs.

- The EGRALMEN bit (in Register 1206H: SONET/SDH Master Tributary Alarm AIS Control) will not force the egress data stream to an all ones AIS alarm as stated even when set high. This will be fixed in the next revision of the device.

- When calculating parity, the INCLDC1J1V1 bit (in Register 1201H: SONET/SDH Master Ingress Configuration) and the INCLAC1J1V1 bit (in Register 1202H: SONET/SDH Master Egress Configuration) should be set low so that parity is calculated without regard to the state of LDC1J1V1 or LAC1J1V1.

- For serial clock and data mode with CCS H-MVIP interface mode, SYSOPT[2:0] in Normal MODE Register 0001H should be programmed to 100 instead of 1xx.

- When the interrupt enable bit BOCSMPE in Register 0064H+80H*N: XBOC Control is set to logic 1, interrupts are not generated, even when the BOCSMPI bit toggles. This will be dedocumented in the next revision of the data sheet.

- When the SBI Collision Detect registers are reset in registers 002C through 002E, they are not necessarily cleared. They will accept new values from subsequent collisions.

- PLLREF [1:0] in Register 004H+80H*N: T1/E1 Egress Line Interface Configuration PLLREF should not be set to 11 to set the tributaries into slave mode. Slave mode is enabled by programming PLLREF to 00. 11 is not a valid mode and will be documented as reserved in the next issue of the register document. Figure 1 – Transmit Timing Options Diagram will also be corrected to reflect this change.

- When using the SBI bus in DS3-framer-only mode, the TXMFPI and RXMFPO in Register 1002H: DS3 Master Unchannelized Interface Options must be programmed to 1. The TXMFPI bit controls which of the inputs TMFPI or TFPI is valid and the RXMFPO bit controls which of the outputs RMFPO or RFPO is valid.

7. Revision ID Bits Summary

Register 0002H: Revision/Global PMON Update identifies the TEMUX revision using the type identification bits TYPE[3:0] as shown in the following table:

Table 4 Revision ID Bits Summary

TEMUX Revision	TYPE[3:0]
A	0000
C	0001
D	0010
E	0011
F	0100
G	0101

Appendix A: Pseudo Code for Detection of Tributary Errors

```
# pseudo Code for TEMUX TTMP Pointer detection

checkTributaries {} {
    fifoSize = read (0x1580 + 0x66) + 0x08
    for {set tug3 0} {tug3 <=2} {incr tug3} {
        for {set tug2 0} {tug2 <= 6} {incr tug2}
            for {set tu 0} {tu <= 3} {incr tu}
                trib = getTrib{tug3,tug2,tu}
                if {PROVbitSet(trib)} {
                    if {checkTributary (trib fifoSize) == BAD_POINTER} {
                        re_centerTJAT(trib) OR toggleProv (trib)
    # If CTCLK = Line Rate use re_centerTJAT(trib) else use toggleProv(trib)
                    }
                }
            }
        }
    }
    wait 250msec
}


# Build build trib address from trib information.
getTrib (tug3, tug2, tu) {
    tribAddr = ((tug3 && 0x03) << 5) | ((tu && 0x03) << 3) |(tug2 && 0x07)
}

# Checks if the trib's PROV bit is set
PROVBitSet (Trib){
    tribCfg = read (0x1580 + Trib)
    return (tribCfg & 0x20)
}

# Checks a tributary's pointer values 1000 times
# Pointers difference should not be less than 1
checkTributary (trib fifoSize) {
    for {set iter 0} {iter <= 1000} {incr iter} {
        pointerDifference = checkPointers (trib fifoSize)
        if {pointerDifference <= 1 || pointerDifference >= fifoSize - 2} {
            return "BAD_POINTER"
        }
    }
    return "POINTERS_OKAY"
```

```
}

check_pointers {trib fifoSize}
        write (0x1580 + 0x67) trib; #Set up RAM Capture Address
        write (0x1580 + 0x68) 0xC0; #Enable ctrl to capture RAM Vectors
        wait at least 6microseconds
        write (0x1580 + 0x68) 0x40; #Disable ctrl to stop RAM Vector capture
        readPointer = getReadPointer (trib)
        writePointer = getWritePointer (trib)
        pointerDifference = writePointer - readPointer
        if {pointerDifference < 0} {
                pointerDifference = pointerDifference + fifoSize
        }
        return pointerDifference
}

getWritePointer {fifoSize} {
        # Write pointer overlaps byte boundary into readPointer byte
        readPointer = {read (0x1580 + 0x73)}            # Read portion of RAM Vector that
contains Write Pointer
        writePointer = {read (0x1580 + 0x72)}           # Read portion of RAM Vector that
contains Write Pointer
        readPointer = (readPointer << 3) & 0x08
        writePointer = (writePointer >> 5) & 0x07
        writePointer = readPointer | writePointer # Make write pointer from two portions
        writePointer = (writePointer)mod fifoSize
        return writePointer
}

getReadPointer {} {
        readPointer = {read (0x1580 + 0x73)}            # Read portion of RAM Vector that
contains Read Pointer
        readPointer = (readPointer >> 1) & 0x0F# Adjust read pointer data
        return readPointer
}
#If CTCLK = Line Rate use re_centerTJAT(trib)
re_centerTJAT(trib) {
write temux 0017+80*N 20 // TJAT Configuration register: CENT=LIMIT=SYNC=0,
RESERVED=1
write temux 0015+80*N 2F // TJAT N1
write temux 0016+80*N 2F // TJAT N2
write temux 0016+80*N “A” // TJAT N2, see Table 5
wait 75ms
write temux 0015+80*N “B” // TJAT N1, see Table 5
wait 75ms
```

```
write temux 0015+80*N “C” // TJAT N1, see Table 5
write temux 0017+80*N 22 // TJAT Configuration register: CENT=LIMIT=0,
RESERVED=SYNC=1
write temux 0015+80*N “C” // TJAT N1, see Table 5
write temux 0017+80*N 20 // TJAT Configuration register: CENT=LIMIT=SYNC=0,
RESERVED=1
}
```

Table 5: N1 and N2 values for TJAT centering procedure

CTCLK (kHz)	T1			E1		
	A	B	C	A	B	C
1544	0xff	0x70	0xff	N/A	N/A	N/A
2048	N/A	N/A	N/A	0xff	0xa0	0xff

#If CTCLK ≠ Line Rate use toggleProv(trib)

```
toggleProv(trib)
        tribCfg = read (0x1580 + trib)
        write (0x1580 + trib) (tribCfg &0xDF)
        wait 1 usec
        write (0x1580 + trib) tribCfg
}
```

Appendix B: SPE Configuration to Prevent VT-AIS Data Corruption

When the TEMUX is setup to receive SPE #3(LDROPSEL[1:0] of register 1200H=11) of a given STS3 you must use the RTDM's time switch RAM to look at the copy of SPE#3 that exists on SPE#2. Otherwise you will see data corruption in the framer adjacent to the framer receiving TU-AIS. The internal RAM makes a copy of the data in SPE3# in all three TUG3s in this configuration so to prevent this problem from occurring, the RTDM Time Switch should be setup to switch the copy of SPE#3's data in TUG3 #2 to TUG3 #3.

The following is a script to switch TU #1 of TUG2 #1 of TUG3 #2 to TU#1 of TUG2 #1 of TUG3 #3. It should be repeated for all 28 VT1.5/TU11s for T1 mode or all 21 VT2/TU12s for E1 mode.

a) Register 12E2H RTDM Time Switch Page Control

APAGE	Set to 0 when configuring page 1 Set to 1 when configuring page 0

b) Register 12A0H RTDM TU#1 in TUG2 #1 of TUG3#2, Control

PROV	Set to 1

c) Register 12C0H RTDM TU#1 in TUG2 #1 of TUG3#3, Control

PROV	Set to 1

d) Register 12E4H RTDM Indirect Time Switch Internal Link Address

INT_SPE[1:0]	Set to '11'
INT_LINK[4:0]	Set to '00001'

e) Register 12E5H RTDM Indirect Ingress Tributary Data

ING_TUG3[1:0]	Set to '10'
INT_TUG2[2:0]	Set to '001'
ING_TU[2:0]	Set to '001'

f) Register 12E3H RTDM Indirect Time Switch Tributary RAM Status and Control

RWB	Set to 0
PAGE	Set to 0 if APAGE of 12E2H was set to 1 Set to 1 if APAGE of 12E2H was set to 0

g) Repeat a) through f) for the other 27 VT1.5/TU11s or 20 VT2/TU12s.

h) Register 12E2H RTDM Time Switch Page Control

APAGE	Set to 0 if PAGE 0 was just configured Set to 1 if PAGE 1 was just configured

i) Register 1201H SONET/SDH Master Ingress Configuration

ITSEN	Set to 1

Contacting PMC-Sierra

PMC-Sierra, Inc.
105 - 8555 Baxter Place
Burnaby, BC V5A 4V7
Tel: (604) 415-6000
Fax: (604) 415-6200

TEMUX Product Marketing Manager:	Jillian Mallory
e-mail:	jillian_mallory@pmc-sierra.com
Product information:	info@pmc-sierra.bc.ca
Applications information:	apps@pmc-sierra.bc.ca
Internet:	http://www.pmc-sierra.com

NOTES

Seller will have no obligation or liability in respect of defects or damage caused by unauthorized use, mis-use, accident, external cause, installation error, or normal wear and tear. There are no warranties, representations or guarantees of any kind, either express or implied by law or custom, regarding the product or its performance, including those regarding quality, merchantability, fitness for purpose, condition, design, title, infringement of third-party rights, or conformance with sample. Seller shall not be responsible for any loss or damage of whatever nature resulting from the use of, or reliance upon, the information contained in this document. In no event will Seller be liable to Buyer or to any other party for loss of profits, loss of savings, or punitive, exemplary, incidental, consequential or special damages, even if Seller has knowledge of the possibility of such potential loss or damage and even if caused by Seller’s negligence.

Issue date: March, 2001

Printed in Canada