PM9315

Enhanced TT1 Chipset

Enhanced Port Processor

Revision A DEVICE ERRATA

PRELIMINARY

Issue 2: June 2001

THE DOCUMENT MAY CONTAIN TECHNICAL INACCURACIES OR TYPOGRAPHICAL ERRORS. CHANGES ARE REGULARLY MADE TO THE INFORMATION CONTAINED IN THE DOCUMENTS. CHANGES MAY OR MAY NOT BE INCLUDED IN FUTURE EDITIONS OF THE DOCUMENT. PMC-SIERRA, INC.OR ITS SUPPLIERS MAY MAKE IMPROVEMENTS AND/OR CHANGES IN THE PRODUCTS(S), PROCESS(ES), TECHNOLOGY, DESCRIPTION(S), AND/OR PROGRAM(S) DESCRIBED IN THE DOCUMENT AT ANY TIME.

THE DOCUMENT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTY OR MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT.

REVISION HISTORY

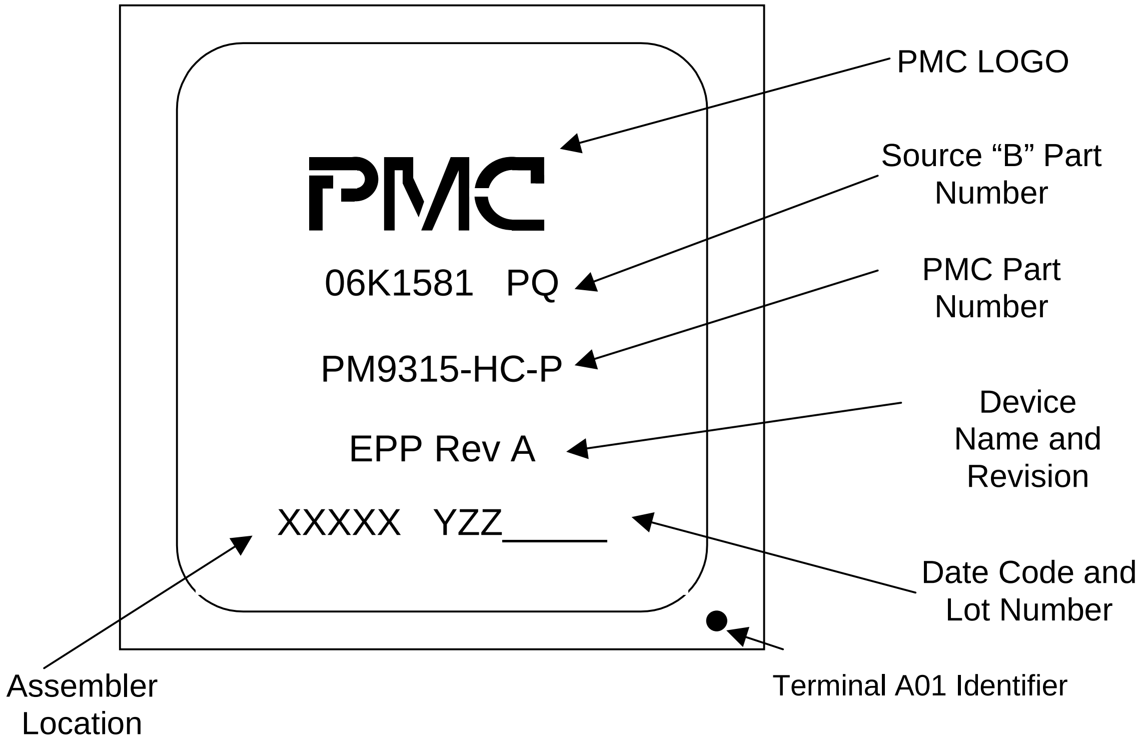
Issue Number	Issue Date	Details Of Change
1	June 2000	Creation of document
2	June 2001	Added section on EPP Revision A/Revision B mixed operation

CONTENTS

1 INTRODUCTION

The information contained in this document applies to PM9315 ETT1 Enhanced Port Processor Revision A, only.

Figure 1. Device Marking



TOP VIEW
SCALE : 3:1 (APPROX.)

2 FUNCTIONAL DEFICIENCIES

This section lists the known functional deficiencies of Revision A of the Enhanced Port Processor device as of the publication date of this document. For each deficiency, the known workaround and operating constraints, if any, are described.

Please report any functional deficiencies observed to PMC-Sierra at:

PMC-Sierra Inc.
105-8555 Baxter Place
Burnaby, BC
Canada V5A 4V7
Tel: (604) 415-6000
App Support: (604) 415-4533
Fax: (604) 415-6001

Product Information:	info @pmc-sierra.com
Applications Information:	apps@pmc-sierra.com
Web Site:	http://www.pmc-sierra.com

2.1 EPP Issues Too Many TDM Requests to the Scheduler

When carrying TDM traffic, the EPP can issue too many TDM requests to the Scheduler. Under certain circumstances this may also result in TDM cell loss or cell misdirection.

Description

The linecard sends in TDM cells to the EPP. Each TDM cell has a TDM tag associated with it. The EPP stores these tags in a 96-deep tag FIFO.

The defect occurs when:

(a) the number of TDM tags in the FIFO mod 96 (24 in OC48 mode) is zero,

AND

(b) the TDM tag at entry 0 in the tag FIFO matches the current tag in the TDM ingress table,

AND

(c) a new TDM cell does not arrive before (b) happens.

Entry 0 in the tag FIFO will be the TDM tag that was requested 96 entries previously (or 24 for OC48 mode).

The effect of this is that the EPP thinks that the tag at entry 0 is a valid entry (even though it was used many cell times earlier) and so will try to match that tag with a tag in the TDM ingress table. If it does match, then the EPP will issue a TDM request to the Scheduler even though there is no TDM cell waiting to be sent.

The Scheduler will grant the request, since TDM has precedence over all other traffic. The EPP receives the grant and then looks at the actual TDM cell queue. It sees that the cell queue is empty and issues a Grant-to-empty-TDM-queue error (interrupt).

This interrupt in itself does not cause a problem (apart from being a spurious interrupt). However the pointers in the TDM tag FIFO are now incorrect. Subsequent TDM tags that arrive may be ignored because the EPP is now looking at the wrong point in the tag FIFO, and this can cause TDM cells to be discarded (lost).

The deficiency is in the EPP logic that decides whether to issue a TDM request to the Scheduler and has been has been demonstrated in our simulations of the system and on the actual hardware.

Workaround

The conditions under which the deficiency manifests itself are quite rare, and it may well be that the linecard can prevent those conditions for occurring:

Any one of the following four methods can be used to avoid the deficiency.

NOTE: If using OC-48 mode, substitute “24” for “96” below.

The first method does not restrict the usage of TDM tags, but instead tries to keep the tag FIFO continually busy, thus avoiding the error condition.

The other three methods restrict the usage of TDM tags to ensure that a false request is never issued.

1. When the linecard receives a TDM_Sync, it should send all TDM requests for the current frame in one continuous burst. It’s OK to send fewer requests than the number of valid slots in the TDM Frame Table; but all requests that are to be sent for the current frame must be sent in one burst.

    - The linecard must send a cell payload to the ETT1 for every TDM grant that it receives. (No dropping LCS-2 grants; send in a request only if you can send in a cell payload when you get the grant.)

    - Program the idle counters of the EPP and linecard such that no more than 1 idle can be sent by each end per TDM Sync period.

    - Limit the number of linecard-to-OOB cells requested to be only 1 per TDM frame, (or else increase the invalid slots to allow for the number of linecard-to-OOB cells that will be sent)

- Start the TDM frame with at least 2 invalid slots; **or** increase the value of the EPP's "TDM Offset Value" register by at least 2 and increase the TDM Sync Period by the same amount.
- The above conditions will prevent cell loss, but the "Scheduler Grant to Empty TDM Queue" interrupt can still happen if the number of TDM requests sent is a multiple of 96 **and** is less than the number of **valid** slots in the TDM Frame Table. If you would like to avoid the interrupt as well, either send as many requests as valid slots, or send an extra request if the number of requests to send is a multiple of 96.

2. Use each tag only once in the TDM Frame Table; for example, set the tag equal to the slot number for all slots.
3. Use each tag in blocks of 96 or fewer slots; for example, use tag A in slots 0-95, tag B in slots 96-191, and so on.
4. Use each tag only within an arbitrary window of 96 or fewer slots.

2.2 EPP Revision A TDM Frame Timing Inconsistent with EPP Revision B

Two bugs are fixed in EPP Revision B: the bug described in section 2.1 of this document, and the bug described in section 2.2 of the PM9311 Scheduler Revision A Device Errata document (PMC-2001247).

The fix for the latter bug causes revision B to begin the active portion of its TDM frame one celltime earlier than revision A. Since all EPPs must step through their TDM Frame tables in perfect synchronization, this means that revision A and revision B EPPs are incompatible for TDM if their TDM Frame tables are programmed normally.

There is a straightforward software workaround for this incompatibility: in a "mixed" revision A/revision B switch, when programming the TDM frame table of a revision B EPP, shift all entries one timeslot later so that input and output requests will be sent to the Scheduler in sync with revision A EPPs. This has a side-effect of making the last slot unavailable to revision B EPPs because revision B EPPs will reach the end of their TDM frame tables one celltime earlier than revision A EPPs. Thus the maximum TDM Frame table size is 1023, not 1024, for a "mixed" revision A/revision B switch.

Customers who send TDM traffic through a switch fabric that contains both revision A and revision B EPPs need to implement this fix. We anticipate that customers who use TDM will go to production with only revision B EPPs in their production switch fabrics, so this workaround will not be required in production software. However, evaluation/reference systems may include both revision A and revision B EPPs, so customers may need this workaround for such systems.

Description

When the Scheduler chip broadcasts the TDM Sync to all EPPs, it sends 2 consecutive TDM Syncs (actually 3 consecutive syncs for the very first frame, then 2 consecutive syncs for all subsequent frames). In EPP Revision A, each consecutive TDM Sync will reset the EPP's countdown of TDM_OFFSET to the beginning of the TDM frame. In EPP Revision B, only the first (consecutive) TDM Sync will reset the counter.

Therefore when 2 consecutive TDM Syncs are broadcasted to all EPPs, revision B will start its TDM_ OFFSET countdown one celltime earlier than revision A. Then revision B will start counting through the timeslots of the TDM Frame table one celltime earlier than revision A, and will always be one timeslot ahead.

If all EPPs in a switch have the same revision, they will all start the TDM Frame at the same time and this incompatibility will not exist.

Workaround

Since the problem is a 1-celltime/timeslot difference between revision A and revision B TDM frame timing, it can be solved by shifting the TDM frame table entries of revision B EPPs by one celltime and disallowing connections to or from revision B EPPs in the last entry of the TDM Frame table.

NOTE: This assumes that the TDM Sync from the Scheduler has been allowed to reach a steady state before TDM traffic is sent. This workaround will not work for the very first TDM Sync sent by the Scheduler after initialization, because it is a "burst" of 3 consecutive Syncs and will cause a 2-celltime timing difference between revision A and revision B. Each subsequent TDM Sync will be a "burst" of 2, which is what the workaround compensates for.

1. Detect the rev of the EPP on each port - determine if the switch is "mixed" revision A/revision B: For each port, read the value of the EPP Status register (ESTS, offset 0x0). Bits 27:24 contain the EPP revision code. If the value of those bits is 0, the EPP is revision A. If greater than 0, the EPP is revision B (or later, which should have the same TDM Frame timing behavior).

If revision A ports are present and revision B (or later) ports are present, then the switch is **"mixed"** and software should perform the following steps:

2. **If "mixed":** Assign TDM connections as usual, **except** do not allow connections to or from any revision B port in the last slot (slot 1023, counting up from 0) of the TDM frame. The easiest way to do this is to disallow any TDM connections in the last slot, though revision As are able to send traffic to other revision As in the last slot.
3. **If "mixed":** When translating TDM connections to an EPP TDM frame table memory image for a revision B port, write the entry for slot N to the address corresponding to slot N+1. Write a value of 0 to the address for slot 0 in each revision B EPP. For example, if a revision B EPP has a connection assigned in slot 0, write that entry to the address for slot 1. This shifts the revision B TDM frame table one celltime later, to compensate for revision B starting its TDM frame one

celltime earlier than revision A.