PM73121

AAL1GATOR II

REFERENCE DESIGN

ISSUE 3

CONTENTS

1 FEATURES 1

2 OVERVIEW..2

3 DEVELOPMENT DESCRIPTION 4

3.1 COMET VERSION OF REFERENCE DESIGN 4

3.2 TQUAD/EQUAD VERSION OF REFERENCE DESIGN 6

4 HIGH LEVEL DESIGN..8

4.1 AAL1GATOR II PLUS COMET DESIGN 8

4.1.1 MICROPROCESSOR AND MEMORY SYSTEM INTERFACE 11

4.1.2 FIELD PROGRAMMABLE GATE ARRAY (FPGA) 18

4.1.3 CLOCK AND POWER SUPPLY CIRCUITRY 22

4.1.4 UTOPIA INTERFACE 22

4.1.5 ADDITIONAL CONNECTIONS 24

4.1.6 LINE INTERFACE 26

4.1.7 ADDITIONAL COMET CONNECTIONS 26

4.1.8 COMET CONFIGURATION...29

4.1.9 AAL1GATOR II CONFIGURATION....................................29

4.1.10 THE JTAG PORT...29

4.2 AAL1GATOR II PLUS TQUAD/EQUAD DESIGN 30

4.2.1 MICROPROCESSOR AND MEMORY SYSTEM INTERFACE 30

4.2.2 LINE SIDE INTERFACE TO THE AAL1GATOR II..............30

4.2.3 LINE INTERFACE 30

4.2.4 ADDITIONAL CONNECTIONS FOR THE TQUAD DEVICE 34

4.2.5 TQUAD/EQUAD CONFIGURATION 35

5 MEMORY MAP AND REGISTER DEFINITIONS 36

6 SOFTWARE CONFIGURATION...39

6.1 AAL1GATOR II CONFIGURATION...39

6.2 COMET CONFIGURATION 43

6.3 TQUAD/EQUAD CONFIGURATION..45

6.4 FPGA CONFIGURATION..45

7 IMPLEMENTATION DESCRIPTION 46

7.1 COMET VERSION SCHEMATICS 46

7.1.1 ROOT DRAWING, SHEET 1 46

7.1.2 COMET BLOCK, SHEETS 2-9..46

7.1.3 LINE INTERFACE, SHEETS 10-13...................................46

7.1.4 FPGA BLOCK, SHEET 14 46

7.1.5 MEMORY SYSTEM BLOCK, SHEET 15 46

7.1.6 AAL1GATOR II BLOCK, SHEET 16 47

7.1.7 MICRO INTERFACE, SHEET 17......................................47

7.2 TQUAD VERSION SCHEMATICS...47

7.2.1 ROOT DRAWING, SHEET 1 47

7.2.2 TQUAD/EQUAD BLOCK, SHEETS 2,3.............................47

7.2.3 AAL1GATOR II BLOCK, SHEET 4 47

7.2.4 FPGA BLOCK, SHEET 5 47

7.2.5 MEMORY SYSTEM BLOCK, SHEET 6 48

7.2.6 MICRO INTERFACE BLOCK, SHEET 7 48

7.2.7 LIU INTERFACE BLOCK, SHEET 8.................................. 48

8 MODULARIZATION ISSUES.. 49

9 REFERENCES 55

10 APPENDIX A: BILL OF MATERIALS (COMET VERSION)..................... 56

11 APPENDIX B: BILL OF MATERIALS (TQUAD/EQUAD VERSION)........ 60

12 APPENDIX C: AAL1GATOR II PLUS COMET SCHEMATIC DIAGRAM. 62

13 APPENDIX D: SCHEMATIC DIAGRAM (TQUAD/EQUAD VERSION).... 63

14 APPENDIX E: SAMPLE VHDL CODE 64

15 DISCLAIMER.. 85

16 NOTES 86

LIST OF FIGURES

FIGURE 1. SYSTEM CONFIGURATION OPTIONS WITH AAL1GATOR II. 2

FIGURE 2. COMET/AAL1GATOR II BLOCK DIAGRAM. 4

FIGURE 3. THE COVERAGE OF THE COMET REFERENCE DESIGN. 5

FIGURE 4. TQUAD/EQUAD TO AAL1GATOR II BLOCK DIAGRAM................... 6

FIGURE 5. COVERAGE OF THE TQUAD/EQUAD REFERENCE DESIGN. 7

FIGURE 6. AAL1GATOR II TO COMET HIGH LEVEL DESIGN. 9

FIGURE 7. GLUELESS AAL1GATOR II TO COMET INTERCONNECTION..... 10

FIGURE 8. BLOCK DIAGRAM OF MICROPROCESSOR INTERFACE. 11

FIGURE 9. AAL1GATOR II CONNECTIONS TO MICROPROCESSOR. 14

FIGURE 10.COMET INTERFACE TO MICROPROCESSOR. 16

FIGURE 11.TERMINATION OF TL_CLK SIGNAL.. 19

FIGURE 12.TL_CLK SWITCHING TERMINATION OPTION............................. 20

FIGURE 13.INTERNAL FPGA CIRCUITRY.. 23

FIGURE 14. AAL1GATOR II PLUS TQUAD/EQUAD HIGH LEVEL DESIGN. ... 33

FIGURE 15.FLOWCHART FOR CONFIGURING THE AAL1GATOR II............. 40

FIGURE 16.AAL1GATOR II BOARD.. 50

FIGURE 17.COMET BOARD. 51

FIGURE 18.TQUAD BOARD. 52

LIST OF TABLES

TABLE 1. AAL1GATOR II TO COMET INTERCONNECTIONS. 8

TABLE 2. MICROPROCESSOR INTERFACE PIN ASSIGNMENT. 12

TABLE 3. ADDITIONAL AAL1GATOR II TO MICROPROCESSOR CONNECTIONS. 15

TABLE 4. ADDRESS RANGES OF THE COMET DEVICES.......................... 17

TABLE 5. TL_CLK OPTIONS. 18

TABLE 6. ADDITIONAL CONNECTIONS TO AAL1GATOR II. 24

TABLE 7. ADDITIONAL CONNECTIONS TO THE COMET DEVICE. 26

TABLE 8. LINE INTERFACE UNIT CONNECTIONS...................................... 31

TABLE 9. ADDITIONAL TQUAD DEVICE CONNECTIONS. 34

TABLE 10. AAL1GATOR II REFERENCE DESIGN MEMORY MAP. 36

TABLE 11. SOURCE_SELECT REGISTER (400000H). 36

TABLE 12. N_CLK_SOURCE REGISTER (400001H). 37

TABLE 13. XCLK_SELECT REGISTER (400002H). 37

TABLE 14. TL_CLK OPTIONS FOR THE AAL1GATOR II. 41

TABLE 15. SUMMARY OF MINIMUM AAL1GATOR II LINE CONFIGURATIONS.42

TABLE 16. SUMMARY OF THE MINIMUM COMET DEVICE CONFIGURATIONS. 44

TABLE 17. SUMMARY OF FPGA CONFIGURATIONS. 45

TABLE 18. AAL1GATOR II TO FRAMER INTERFACE. 53

TABLE 19. COMET VERSION BILL OF MATERIALS....................................... 56

TABLE 20. TQUAD/EQUAD VERSION BILL OF MATERIALS......................... 60

1 FEATURES

- Implementation strategy for AAL1gator II in a Multi Service Access Concentrator environment using the PM4351 COMET.

- Implementation strategy for AAL1gator II in a Multi Service Access Concentrator environment using the PM4344 TQUAD or PM6344 EQUAD.

- Implementation strategy for AAL1gator II in a DS3 system.

- Complete implementation of 8 link Circuit Emulation Service.

- Supports structured and unstructured (both with and without CAS) CES for T1/E1 links.

- Supports independently clocked links.

- Microprocessor interface for configuration and monitoring.

- Allows 8 T1/E1 lines (1.544 Mbit/s/2.048 Mbit/s) to be serviced.

- Supports enabling/disabling of Synchronous Residual Time Stamp (SRTS) clocking.

- Modular design

- Includes a software driven clock multiplexer to allow customers to select one of several possible TL_CLK sources.

2 OVERVIEW

This document describes the hardware and software configuration required for the PM73121 AAL1gator II Reference Design. This will be a paper only reference design, intended to assist customers in their efforts to build a Circuit Emulation Service (CES) card. The purpose of a CES card is to emulate circuit oriented transmission technology characteristics in order to support Constant Bit Rate (CBR) traffic. One example of CBR traffic is time sensitive voice applications.

The purpose of the AAL1gator II is to provide T1/E1, or DS3/E3/J2 line interfaces access to an AAL1 CBR ATM network. Figure 1 below indicates the ways in which an AAL1gator II can be used to connect to T1/E1 or DS3/E3/J2 line interfaces. Not shown is the interface to the UTOPIA bus, and associated routing tables, and switching elements.

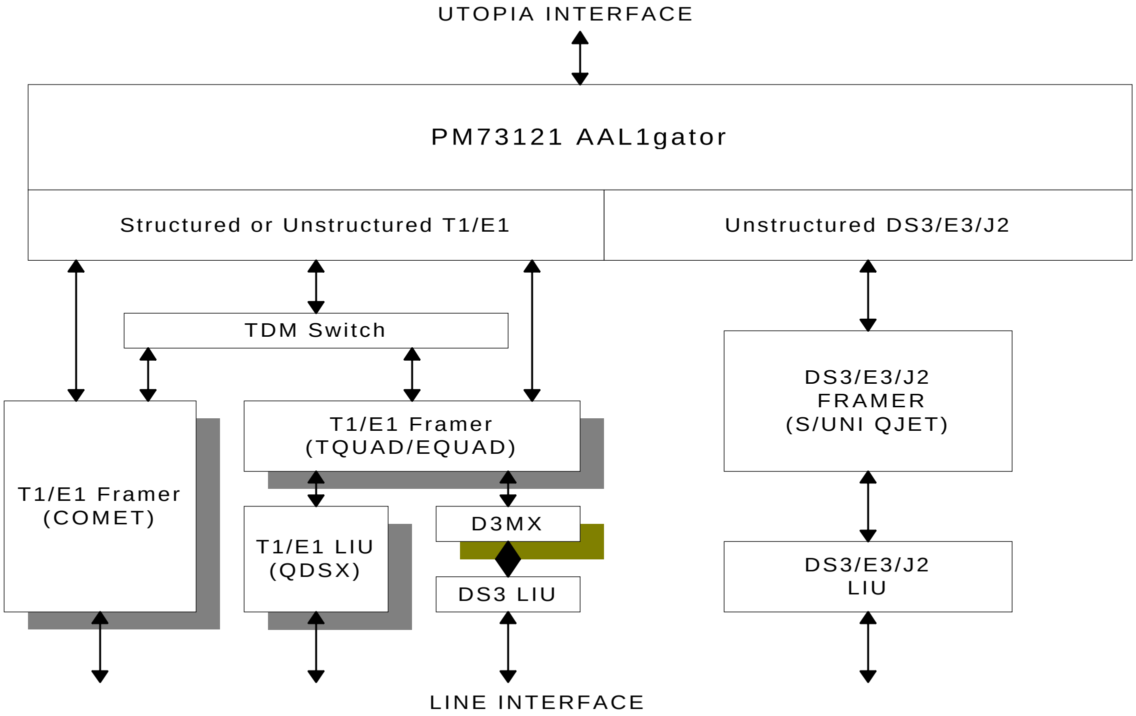


Figure 1. System Configuration options with AAL1gator II.

The AAL1gator II is capable of supporting eight data streams, where a single data stream cannot exceed 15 Mbit/s and the aggregate total cannot exceed 20 Mbit/s , or one data stream which can support up to 45 Mbit/s.

The reference design will consist of two separate AAL1gator II implementations:

1. AAL1gator II with 8 COMET'S
2. AAL1gator II with 2 TQUAD's or EQUAD's

The first implementation will target the Multi Service Access Concentrator market, while the second will provide a building block for using the AAL1gator II in a DS3 application.

The reference design will further support a modularized implementation, should the need arise in the future to build prototypes based on the design. A modular implementation was chosen to allow either a board consisting of 8 COMET devices or 2 TQUAD/EQUAD devices to be connected over a common interface to a main board consisting of a single AAL1gator II along with associated memory and microprocessor interface.

3 DEVELOPMENT DESCRIPTION

3.1 COMET Version of Reference Design

Figure 2 illustrates the block diagram of the PM73121 Reference Design using 8 COMET devices. The hardware allows full access to the AAL1gator II device via the microprocessor interface. Each COMET device acts as a line interface unit with an integrated long haul LIU, and T1/E1 framer/deframer. In the receive path (from T1 or E1 line), the COMET converts the incoming line data (in the form of channels) to a serial bit stream. The AAL1gator II then receives this data (and clocking information) from the COMET, and builds ATM cells to be sent to the UTOPIA bus.

In the transmit path (to T1 or E1 line), the AAL1gator II receives ATM cells from the UTOPIA bus. The AAL1gator II retrieves the data and signaling information, and places the data to be transmitted over the T1 (or E1) line (via the COMET) in the appropriate port and time slot.

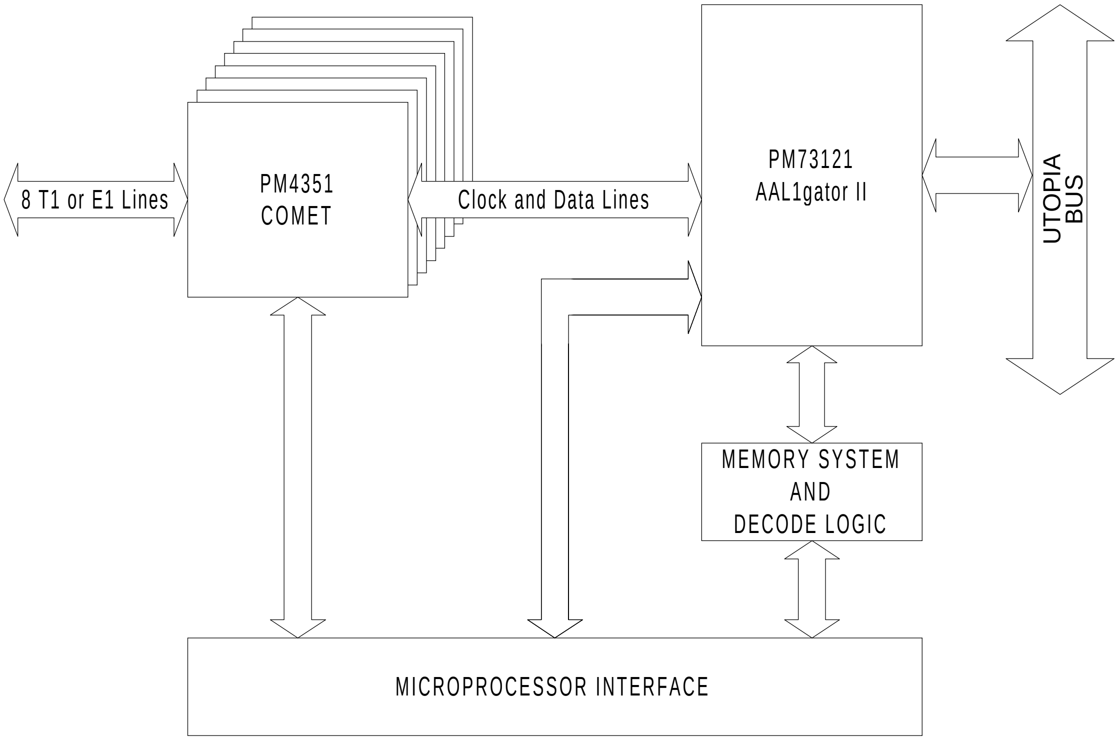


Figure 2. COMET/AAL1gator II Block Diagram.

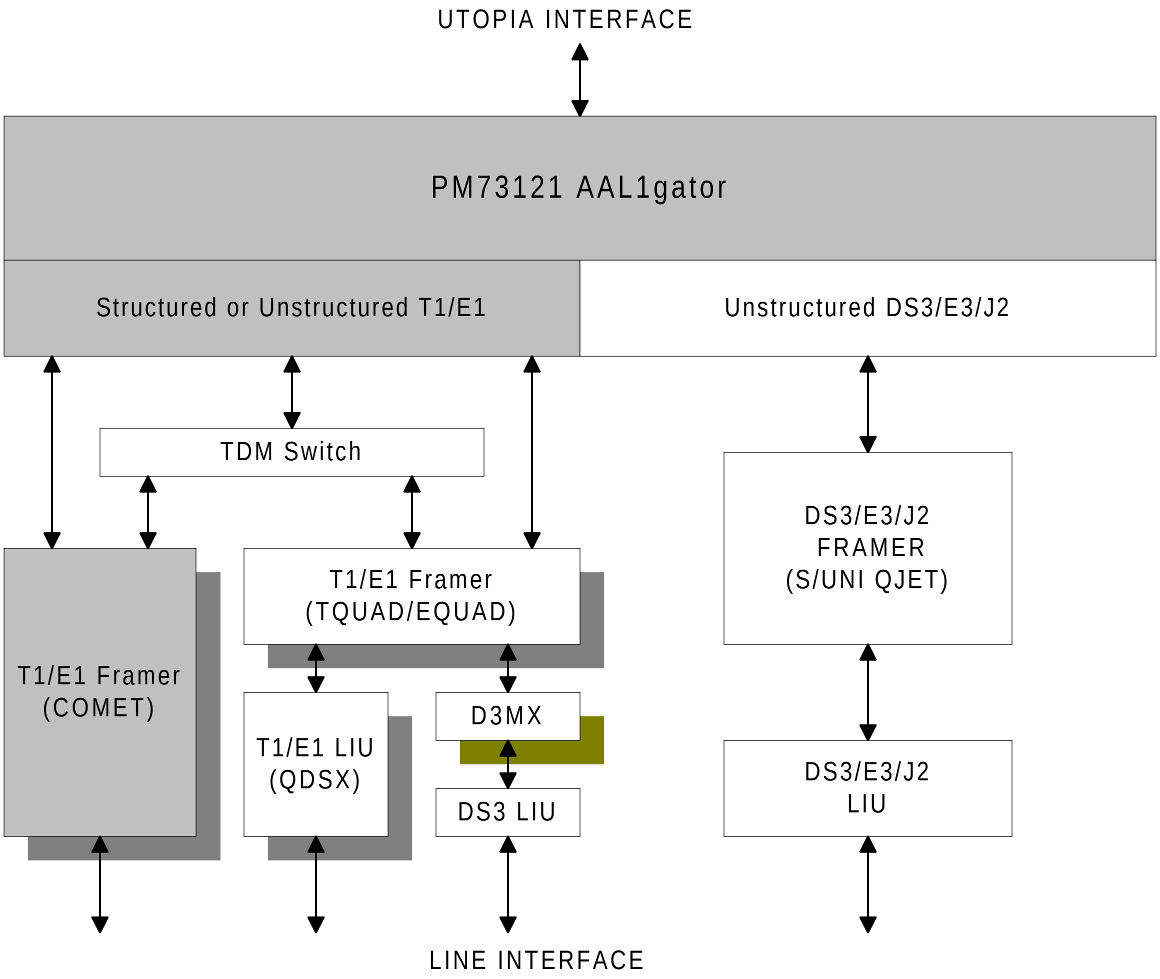


Figure 3. The coverage of the COMET reference design.

With reference to Figure 1, Figure 3 shows how a T1 or E1 line can be interfaced to an ATM network via the UTOPIA bus by using the COMET and AAL1gator II devices.

3.2 TQUAD/EQUAD Version of Reference Design

Figure 4 illustrates the AAL1gator II in an application involving the use of 2 TQUAD (or the pin compatible EQUAD) devices. This design could form a building block for the use of AAL1gator II in a DS3 system. The operation is similar to that of Figure 2.

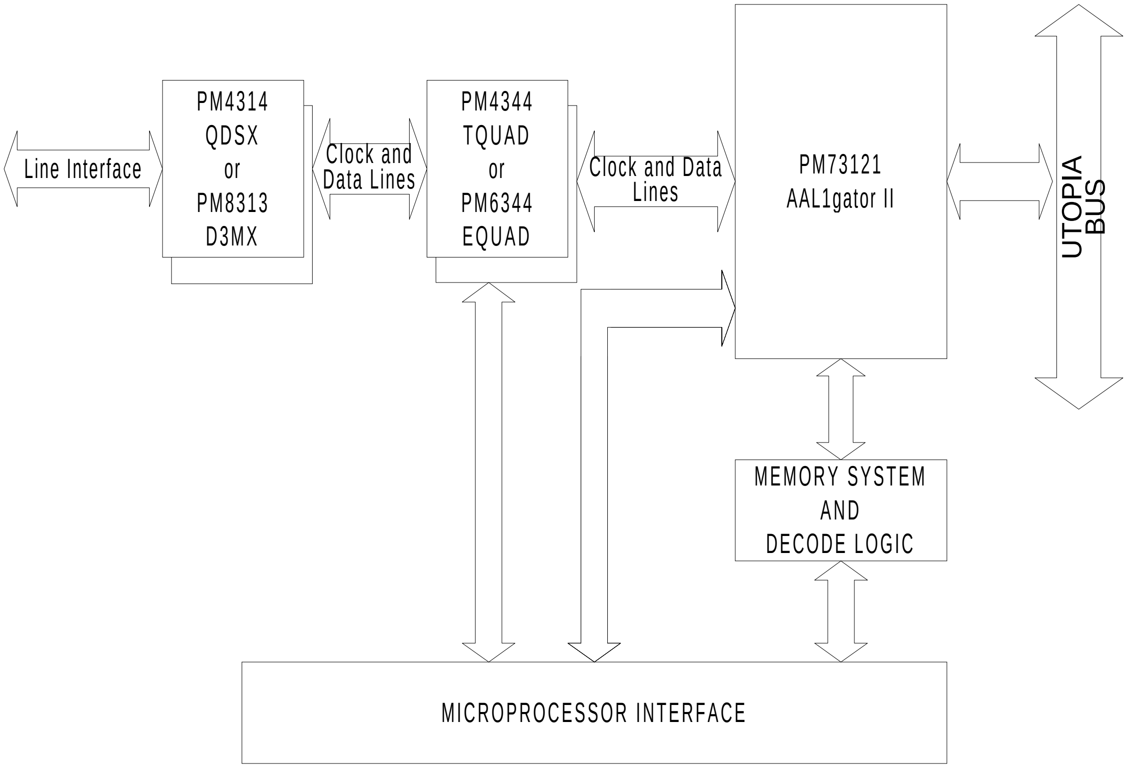


Figure 4. TQUAD/EQUAD to AAL1gator II Block Diagram.

With reference to Figure 1, Figure 5 illustrates the coverage of the TQUAD/EQUAD design.

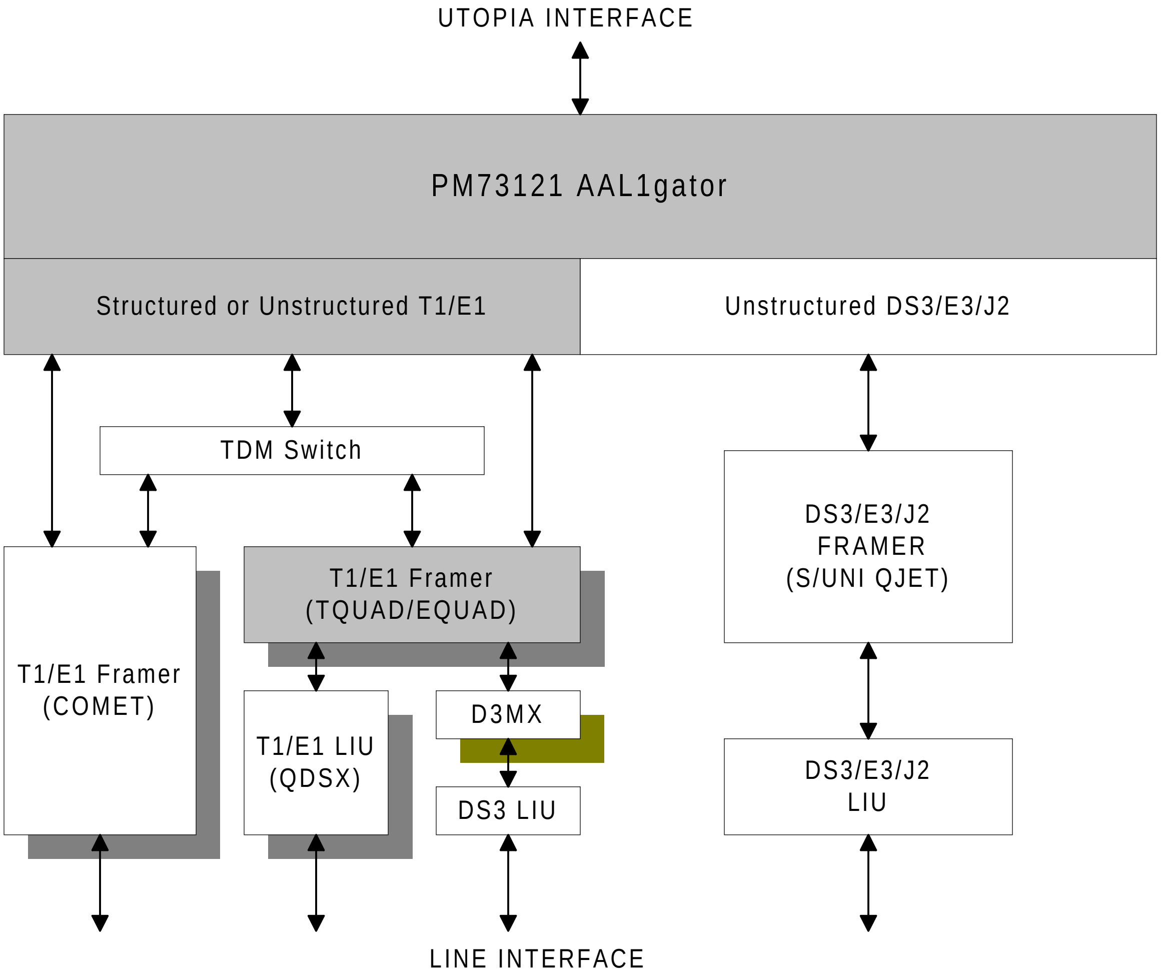


Figure 5. Coverage of the TQUAD/EQUAD reference design.

4 HIGH LEVEL DESIGN

4.1 AAL1gator II plus COMET Design

Figure 6 on the following page represents the high level design of the AAL1gator II plus COMET reference design. As can be seen, the design contains the following functional blocks:

1. AAL1gator II Integrated Circuit
2. 8 COMET Devices
3. Microprocessor and Memory System Interface
4. Field Programmable Gate Array
5. Clock Sources
6. UTOPIA Interface
7. Line Interface

The AAL1gator II communicates with the COMET devices via the framer bus signals listed in Table 1. As noted in Figure 9, one bit of each signal group connects to its associated COMET device. For instance, TL_FSYNC(0) connects to COMET0, while TL_FSYNC(1) connects to COMET1.

Table 1. AAL1gator II to COMET Interconnections.

SIGNAL	DESCRIPTION
TL_FSYNC(7..0)	The FPGA generates this signal for both the AAL1gator II and COMET. In T1 mode, this signal consists of a pulse once every 193 bit periods.
TL_CLK(7..0)	This is a clock signal at the transmit line rate. Its source is determined by the configuration of the FPGA.
RL_CLK(7..0)	Receive line clock at either 1.544 MHz or 2.048 MHz, derived from the recovered line rate timing.
RL_FSYNC(7..0)	Carries receive frame synchronization from the COMET devices.
RL_SIG(7..0)	Carries the CAS signaling information from the COMET devices.
RL_SER(7..0)	Carries the receive data from the COMET devices.
TL_SIG(7..0)	Carries the CAS signaling outputs to the COMET devices.
TL_SER(7..0)	Carries the serial data to the COMET devices.

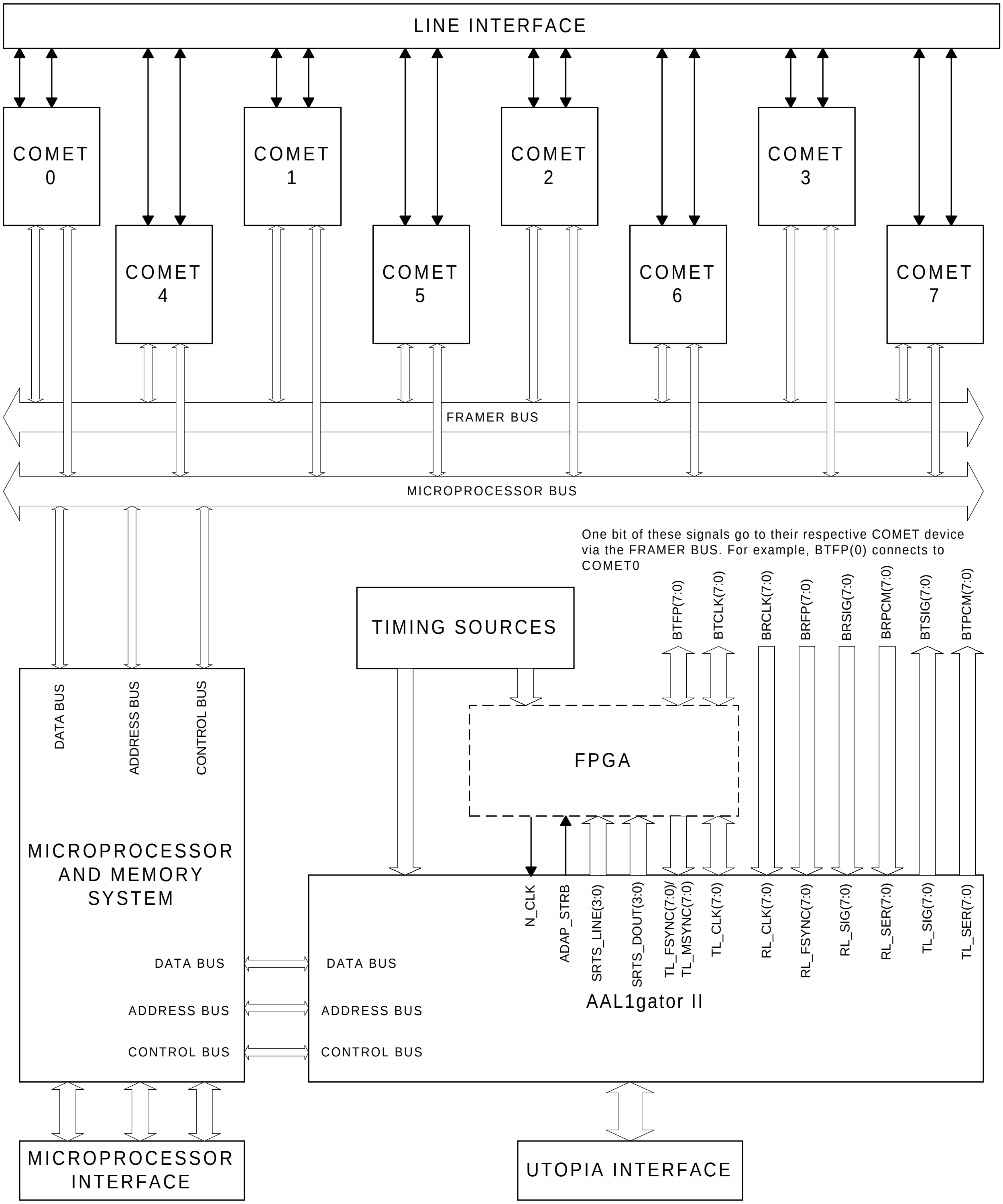


Figure 6. AAL1gator II to COMET High Level Design.

In Figure 6, the FPGA is shown with a dotted line. This is because it is possible to directly connect the AAL1gator II to the COMET device as illustrated in Figure 7. In this way, it is possible to align signaling bits to the multiframe boundary.

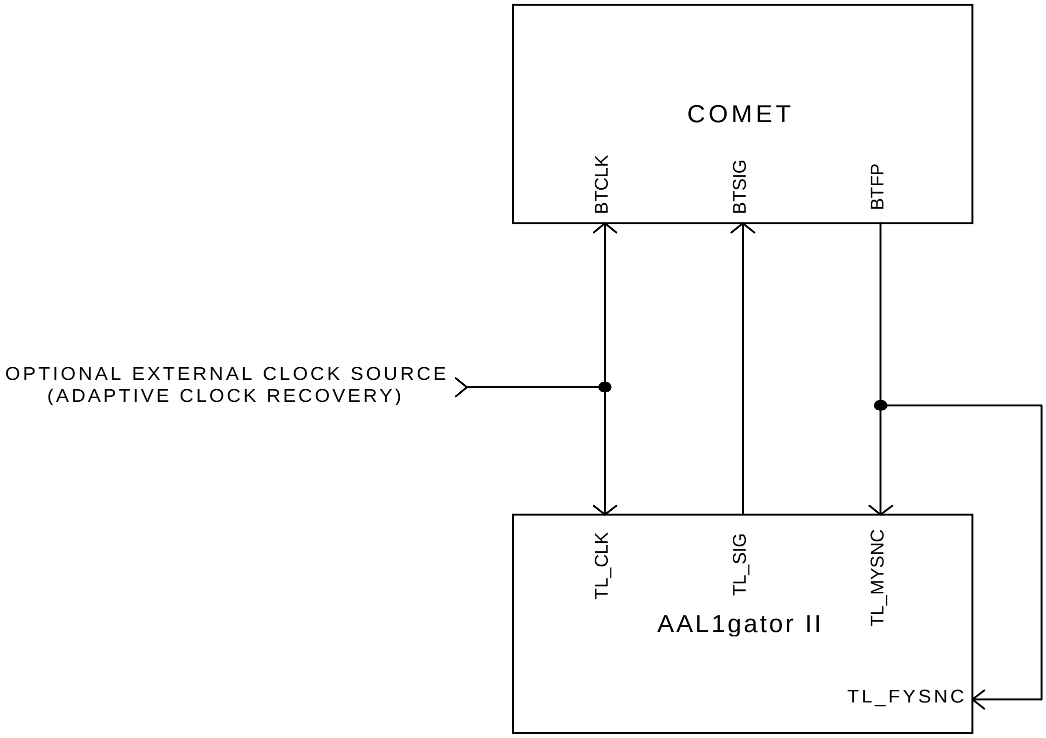


Figure 7. Glueless AAL1gator II to COMET interconnection.

Using the approach of Figure 7, the FPGA depicted in Figure 6 is no longer necessary. The main purpose of the FPGA is to provide maximum clock distribution flexibility by allowing for independently clocked links. Using the FPGA , it is possible to use Adaptive Clock Recovery for the generation of TL_CLK (AAL1gator II) and BTCLK (COMET). In addition, the FPGA generates an appropriate signal for N_CLK (at 2.43 MHz), and also distributes XCLK signals to the 8 COMET devices from only two clock oscillators (1.544 MHz and 2.048 MHz). Therefore, without the FPGA, some other method will have to be found to generate a 2.42 MHz signal for N_CLK (if SRTS is in use) and to distribute XCLK signals to the COMET devices. Please see section 4.1.2 for further discussion of the FPGA.

The other difference between the two drawings is that in Figure 7, both TL_FSYNC and TL_MSYNC of the AAL1gator II connects to the COMET BTFP pin (configured as an output), rather than just TL_FSYNC. As mentioned above, this allows for alignment of signaling bits on multiframe boundaries.

4.1.1 Microprocessor and Memory System Interface

The interface between the system microprocessor can be divided into two sections. The first section interfaces the system microprocessor to the AAL1gator II, while the second section interfaces the system microprocessor to the COMET devices and the FPGA. The following figure (Figure 8) represents a high level view the microprocessor interface.

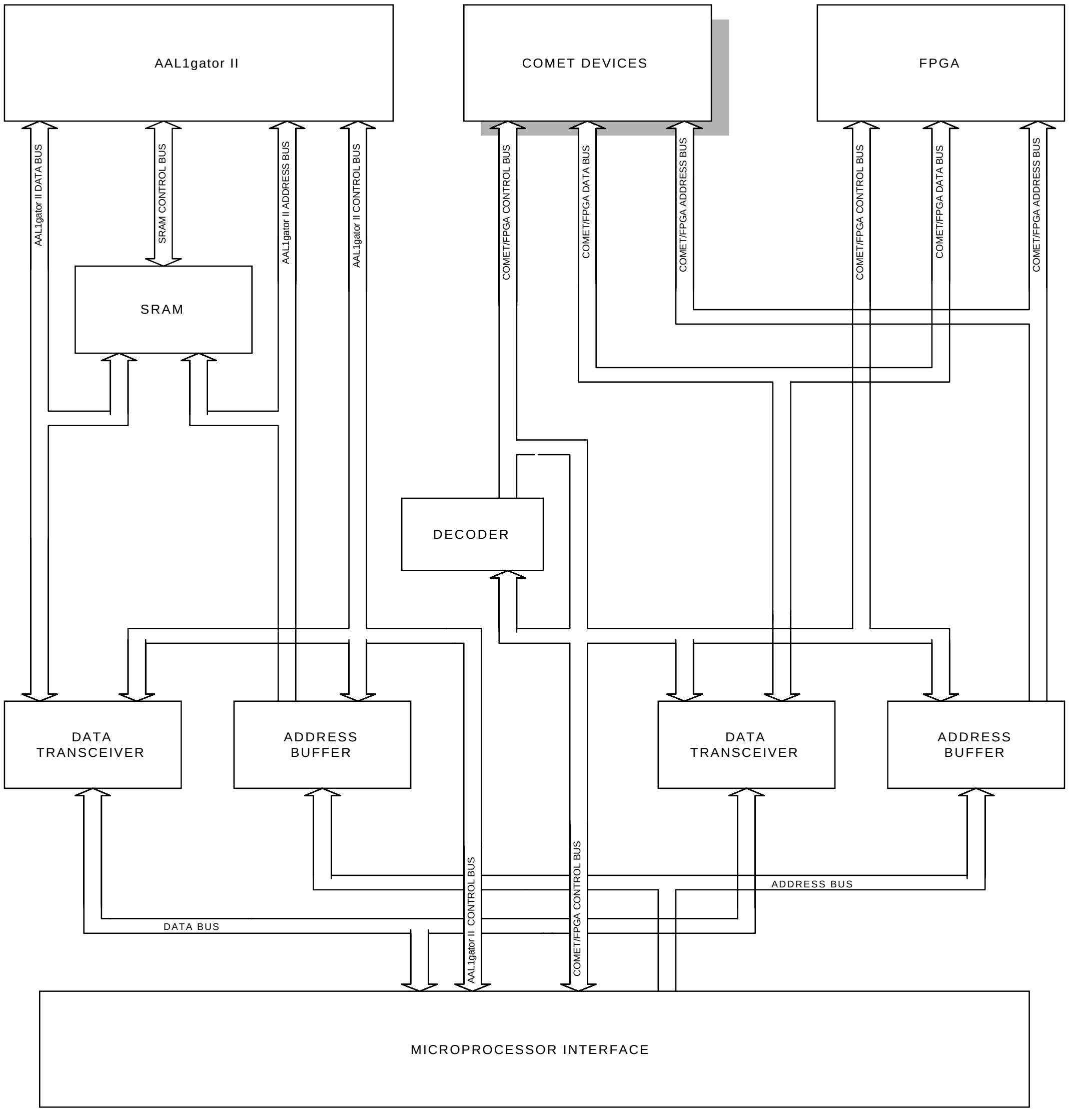


Figure 8. Block Diagram of microprocessor interface.

In order to provide maximum system implementation flexibility, a particular microprocessor has not been specified. However, in order to implement the microprocessor and memory system as shown in this document, the system microprocessor must have the following capabilities:

1. Minimum 22 bit address bus
2. Minimum 16 bit data bus
3. Minimum 3 programmable chip selects
4. Minimum 2 independent interrupt request lines

An example of a device that meets these minimum requirements is the Motorola MC68330 microprocessor. Another scenario would see the design implemented in a PCI or compact PCI system.

Table 2 lists the pin assignment of a potential microprocessor interface (96 pin DIN). Note that this interface includes all connections from the microprocessor to the AAL1gator II, COMET devices, and FPGA.

Table 2. Microprocessor Interface Pin Assignment.

PIN NAME	PIN TYPE	PIN NUMBER	FUNCTION

D(15)
D(14)
D(13)
D(12)
D(11)
D(10)
D(9)
D(8)
D(7)
D(6)
D(5)
D(4)
D(3)
D(2)
D(1)
D(0)
	I/O	A17
		A18
		A19
		A20
		A21
		A22
		A23
		A24
		A25
		A26
		A27
		A28
		A29
		A30
		A31
		A32
			16 bit data bus

A(21)
A(20)
A(19)
A(18)
A(17)
	Input
	(from uP)
		C11
		C12
		C13
		C14
		C15
			22 bit address bus

PIN NAME	PIN TYPE	PIN NUMBER	FUNCTION

A(16)
A(15)
A(14)
A(13)
A(12)
A(11)
A(10)
A(9)
A(8)
A(7)
A(6)
A(5)
A(4)
A(3)
A(2)
A(1)
A(0)
		C16
		C17
		C18
		C19
		C20
		C21
		C22
		C23
		C24
		C25
		C26
		C27
		C28
		C29
		C30
		C31
		C32	

RDB	Input	C7	Active Low read signal
WRB	Input	C8	Active Low write signal.
PROC_ACK	Input	C1	Active Low acknowledge signal to uP.
IRQ1B	Output (to uP)	C5	Active low interrupt request to uP. from AAL1gator II
IRQ2B	Output	C7	Active low interrupt request to uP from Framer
RESETB	Input	A1	Active low global reset.
CS1B	Input	C2	Active low chip select. When asserted, the AAL1gator II is selected.
CS2B	Input	C3	Active low chip select. When asserted, the COMET or TQUAD/EQUAD is selected.
CS3B	Input	C4	Active low chip select. When asserted, the FPGA is selected.
HOLDOFF	Output	A2	Prevents uP from gaining too many cycles.
GND	n/a	B1 – B26	GND. Ground Reference

Figure 9 on the following page represents the design of the interface from the AAL1gator II to its associated memory system and the system microprocessor. As can be noted, the interface consists of 2 128K x 8 static ram devices. In the transmit direction (from the line via RL_SER and RL_SIG, to the ATM network), the devices store queued data while cells for transmission are built. In the receive direction (from the ATM network via the UTOPIA bus to the line side interface) the SRAM holds data while frames are built for transmission on the T1 or E1 line. The selected SRAM must be 12 ns or faster, with a write setup time of 7 ns or faster.

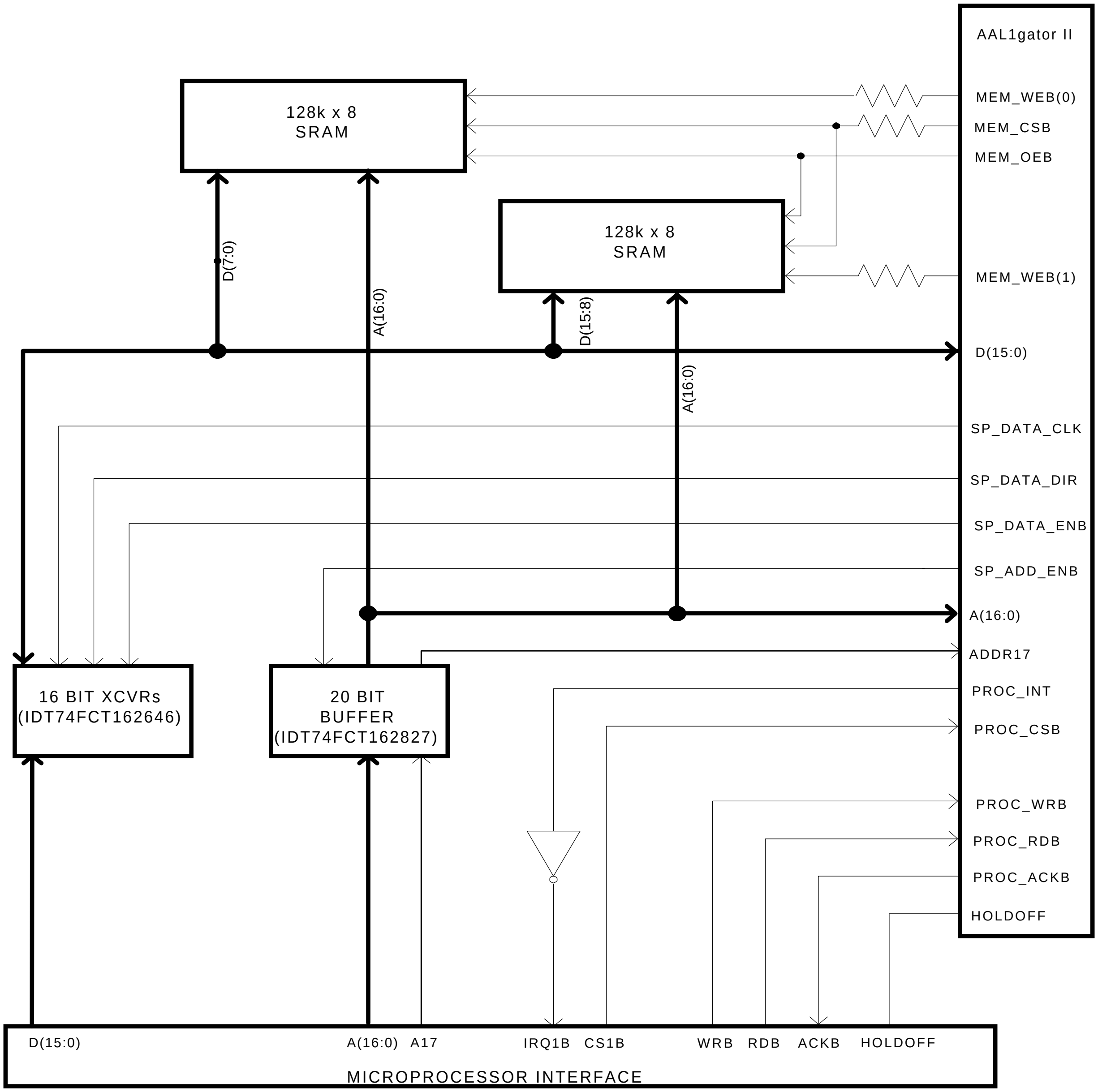


Figure 9. AAL1gator II Connections to Microprocessor.

Figure 9 also indicates the usage of external address buffers and data transceivers. In order for the system to operate at the maximum frequency of 40.00 MHz, the address buffers must have a worst case propagation delay of 8ns, while the data transceivers must have a worst case delay of 10ns. For these reasons the IDT74FCT162827CT was chosen as the address buffer. This 20 bit device has a maximum propagation delay of 4.4ns (50pF, 500Ω load). The IDT74FCT162646 was chosen as the data transceiver. This 16 bit device has a worst case propagation delay of 5.4ns under the same loading conditions. As can be seen, serial termination resistors are placed on the MEM_WEB(x) and MEM_CSB signals. The resistor value was chosen in accordance with the discussion in section 8.6 of the AAL1gator II data sheet. Please refer to that section for additional information.

Table 3 below describes additional connections between the AAL1gator II, the microprocessor and the memory system.

Table 3. Additional AAL1gator II to Microprocessor Connections.

SIGNAL NAME	DESCRIPTION
MEM_WEB(0)	Active low write enable for low order byte SRAM
MEM_WEB(1)	Active low write enable for high order byte SRAM
MEM_CSB	Active low chip select for SRAM devices
MEM_OEB	Active low output enable of SRAM devices
SP_DATA_CLK	Memory or register read data is written into the FCT162646 data transceiver on the rising edge of this signal. The microprocessor may then access the data. Signal is driven by the AAL1gator II.
SP_DATA_DIR	Controls the direction of data movement in the FCT162646. When high, data flows toward the microprocessor, when low, towards the AAL1gator II. Signal is driven by the AAL1gator II.
SP_DATA_ENB	AAL1gator II driven signal that enables the FCT162646 data transceiver.
SP_ADD_ENB	AAL1gator II driven signal that enables the FCT162827 address buffer.
PROC_INT	Active high interrupt from the AAL1gator II to the microprocessor.
PROC_CSB	Active low chip select from the microprocessor (via CS1B).
ADDR17	When high, the CMD_REG of the AAL1gator II is selected.
PROC_WRB	Active low write signal for the AAL1gator II from the microprocessor.
PROC_RDB	Active low read signal for the AAL1gator II from the microprocessor.
PROC_ACKB	Active low acknowledge signal from the AAL1gator II.
HOLDOFF	When high, prevents the microprocessor from additional accesses until 20 SYS_CLK cycles from the last processor access.

Figure 10 on the following page illustrates the connections between the microprocessor, and the 8 COMET devices plus the FPGA.

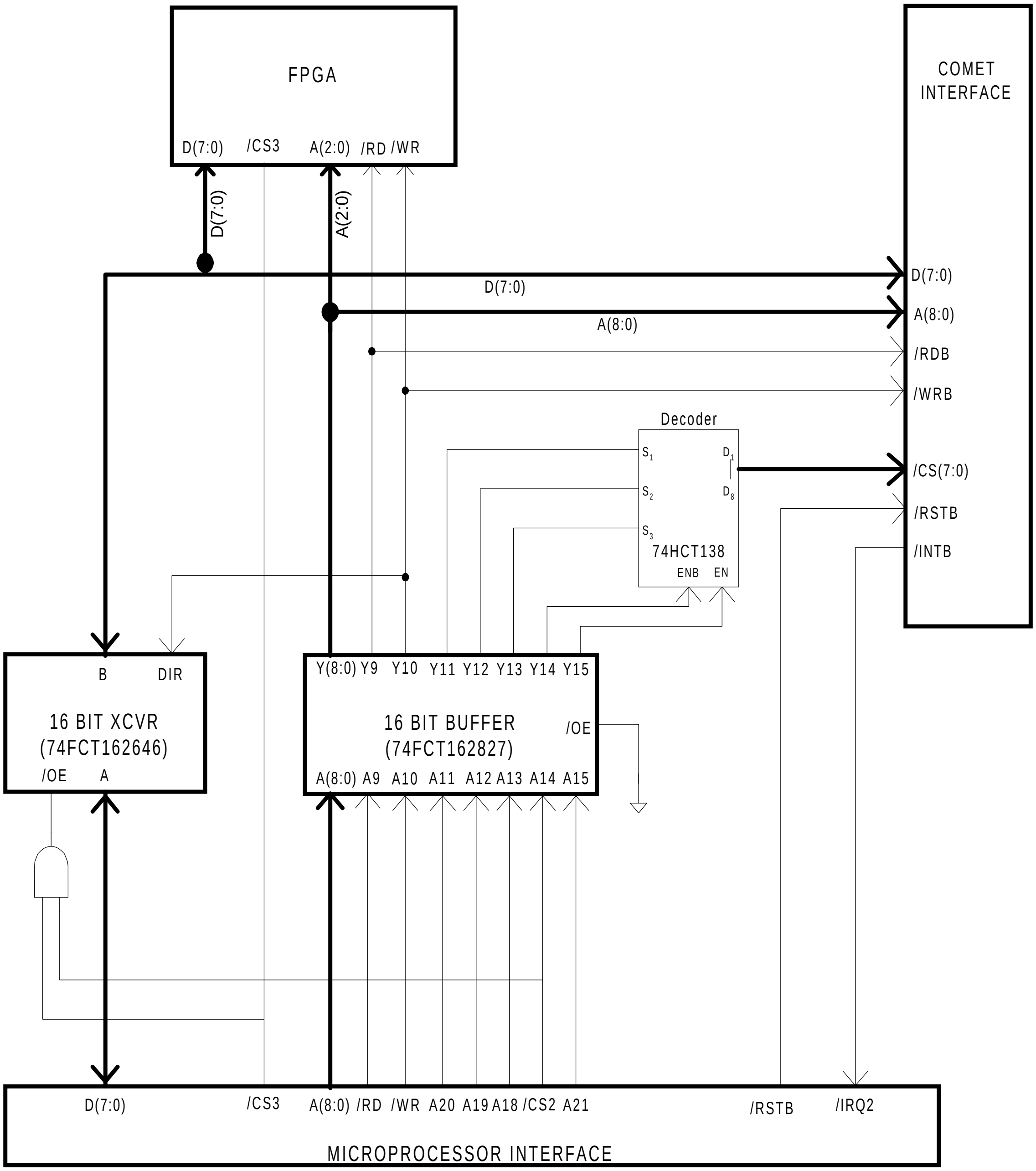


Figure 10. COMET Interface to Microprocessor.

When the microprocessor wishes to communicate with a COMET device, it asserts an address as listed in the following table (Table 4). When an appropriate address is driven onto the bus, the microprocessor simultaneously asserts CS2B. Since A21 is high, the decoder is then active. Address bits A(20..18) determine which output of the decoder is driven low. One decoder output connects to the CSB input of each COMET device. For example, if A(20..18) are 000, then decoder output Y0 is driven low, which also asserts CSB of COMET0. No other COMET device is selected at this time. Address bits A(8..0) determine which register of the COMET the microprocessor is communicating with.

In order to meet the timing requirements of the COMET devices (see PM4351 COMET data sheet), the 3-8 decoder (74HCT138) must have a minimum propagation delay of 10ns. With a 15pF load, the device has a typical delay of 13ns. At 50pf (VCC = 4.5), the delay increases to 27ns, and therefore will meet the specifications.

Table 4. Address Ranges of the COMET devices.

COMET	BASE ADDRESS	ADDRESS RANGE
COMET0	200000h	200000 – 2001FFh
COMET1	240000h	240000 – 2401FFh
COMET2	280000h	280000 – 2801FFh
COMET3	2C0000h	2C0000 – 2C01FFh
COMET4	300000h	300000 – 3001FFh
COMET5	340000h	340000 – 3401FFh
COMET6	380000h	380000 – 3801FFh
COMET7	3C0000h	3C0000 – 3C01FFh

Since the address and data buses are shared among many devices, a 20 bit buffer and transceiver is used. This insures that clean signals are present on the inputs of the devices, and that no data collisions occur. The buffer (FCT162827) is not only placed on the address lines, but the various control signals such as WRB and RDB as well. The 16 bit transceiver (FCT162646) is used in flow through mode to control data bus access. The transceivers output enable is controlled by the result of a logical AND of CS2 and CS3. In this way, whenever the microprocessor needs to communicate with either a COMET or the FPGA, either the CS2, or CS3 signal must be driven low, which drives the active low output enable signal of the transceiver low. The transceivers direction is controlled by the WRB signal.

4.1.2 Field Programmable Gate Array (FPGA)

The FPGA (see Figure 13 page 23) performs the following functions on a per line basis:

1. Allows fully software selectable TL_CLK source.

2. Allows a fully selectable COMET XCLK source (1.544 or 2.048 MHz).

3. Generates framing pulses at the 8 kHz rate for the TL_FSYNC (AAL1gator II) and BTFP (TQUAD or COMET) signals.

4. Produces a software selected N_CLK signal (gnd, or 2.43 MHz).

In order to provide maximum system flexibility, the AAL1gator II provides several options for the TL_CLK signal. Table 5 lists the options that the user may select. As can be seen, the FPGA permits the use of all options on a per line basis. The options must also be set in the LIN_STR_MODE register (CLK_SOURCE bits) of the AALI1gator II.

Table 5. TL_CLK options.

CLK_SOURCE(5..4)	FUNCTION	TL_CLK I/O	SUPPORTED
00	External Clock Source	input	YES
01	Recovered: based on RL_CLK	output	YES
10	Nominal T1 or E1 Clock Synthesis	output	YES
11	SRTS	output	YES

When the CLK_SOURCE(5..4) bits are set to 00, TL_CLK is configured as an input. In this mode, the FPGA will allow a 1.544 MHz, 2.048 MHz, or adaptive clock source to be supplied to the TL_CLK input (and also to the BTCLK input of the COMET or TQUAD device). In adaptive clock mode, an adaptive clock generator block provides a queue depth difference for control of an external clock. If the queue depth is low, the clock frequency is reduced, while if the queue depth is high, the clock frequency is increased. Adaptive clock data from the AAL1gator II appears on the SRTS port when the ADAP_STRB signal is asserted by the AAL1gator II.

In recovered mode, (CLK_SOURCE set to 01), the clock signal output on TL_CLK is based on the received RL_CLK.

When the CLK_SOURCE(5..4) bits are set to 10, the AAL1gator II is configured to generate a nominal T1 or E1 clock derived from a 38.88 MHz SYS_CLK. SYS_CLK must be network derived in order to lock the synthesized clock to the network clock.

When CLK_SOURCE(5..4) are set to 11, the AAL1gator II is configured to use the Synchronous Residual Time Stamp (SRTS) option. In this mode, the AAL1gator II compares locally generated SRTS values with received SRTS values. A 4 bit difference code is then generated. If the code value is higher than that previously generated, the remote clock is running faster than the local clock. Alternatively, if the code is lower, the remote clock is running slower than the local clock. The AAL1gator II uses this information to synthesize TL_CLK.

In this implementation, the interconnection between the AAL1gator II TL_CLK pin and the FPGA is bidirectional. There are many possible ways to alleviate the complexity of terminating a bidirectional signal. The first method, illustrated below in Figure 11, utilizes two separate FPGA ports for TL_CLK. The TL_CLKI port is an input port from the AAL1gator II, while the TL_CLKO port is an output port to the AAL1gator II. In additon two series terminating resistors (value between 33 and 50Ω) and one jumper per line are required. When the AAL1gator II is sourcing TL_CLK (to the TL_CLKI input of the FPGA), the jumper is not installed, and the resistor nearest the AAL1gator II is installed. Likewise, when the TL_CLKO port of the FPGA is sourcing TL_CLK, the jumper is installed, and the resistor nearest he AAL1gator II is not installed. This, in essence, creates a unidirectional transmission path for each direction.

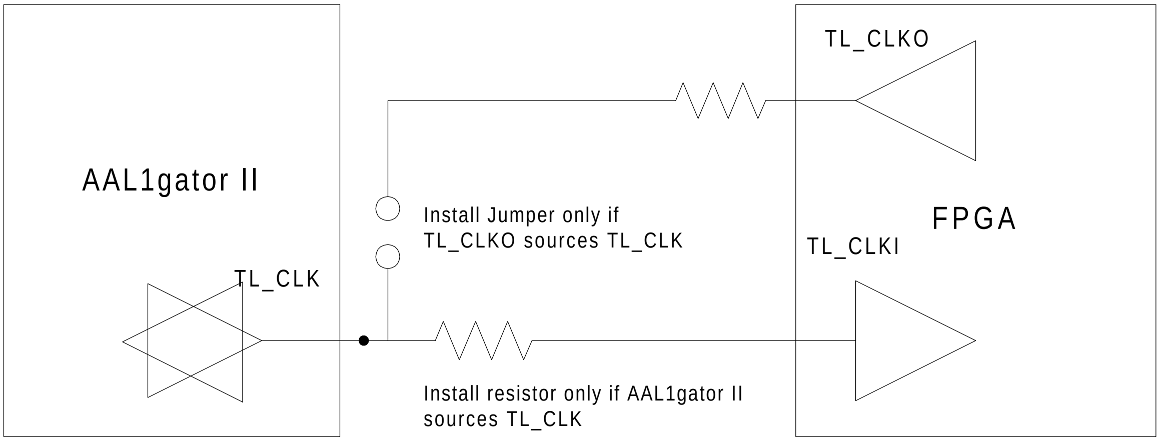


Figure 11. Termination of TL_CLK signal.

However, there is a disadvantage to the above method. It is no longer possible to interactively switch between having the AAL1gator II source TL_CLK, or have an external source provide TL_CLK to the AAL1gator II. Therefore, an alternative approach would be to combine the TL_CLKI and TL_CLKO ports of the FPGA into one bidirectional port (for each line). The signal path distance between the AAL1gator II and the FPGA must then be kept as short as possible. If this is not possible, both ends of the signal path may be serially terminated with an appropriate value resistor.

One last alternative is a hybrid between the previous two approaches. As illustrated in Figure 12 for one of the eight TL_CLK lines, this method utilizes a switching arrangement to change between having the AAL1gator II source TL_CLK, to having the FPGA source TL_CLK to the AAL1gator II.

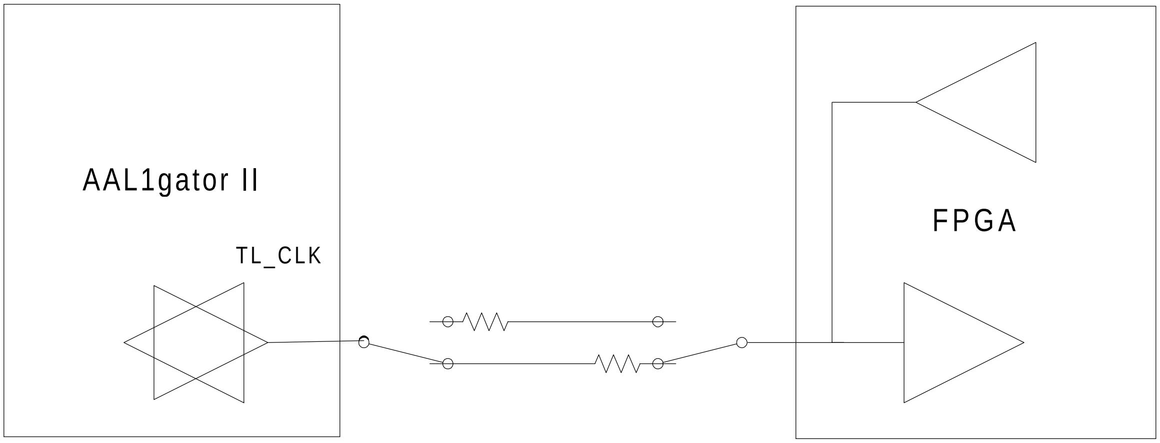


Figure 12. TL_CLK switching termination option.

The switches could take the form of dip switches, requiring manual interchange, or could be digitally controlled analog swiches. In addition, the switches could be replaced by jumpers.

PMC-Sierra Inc. has secured an agreement with Bellcore (patent holders for SRTS clock recovery technique). Please refer to the AAL1gator II data sheet for important information regarding this agreement.

In a typical application employing SRTS, N_CLK must be a 2.43 MHz signal derived from the ATM network (for example, from the 155.52 MHz SONET clock). To disable SRTS, N_CLK is connected to GND, whereas to enable SRTS, N_CLK must be connected to a 2.43 MHz signal. This is accomplished via a divide by 16 block in the FPGA whose input is the 38.88 MHz network clock signal.

Another function of the FPGA is to generate the 8 kHz framing pulse from the transmit line clock output (BTCLK) to the framer. This 8 kHz signal connects to the AAL1gator II TL_FSYNC input, and the framer’s BTFP input. In T1 mode, a pulse one BTCLK period wide is generated every 193 bits, while in E1 mode, the pulse is generated every 256 bits.

In an implementation involving the AAL1gator II and TQUAD or EQUAD device, each BTCLK output from the FPGA must be inverted. Whereas if the design involves AAL1gator II to COMET devices, the BTCLK output must not be inverted with respect to TL_CLK. The reason for this is the timing requirements of the line side interface of the

AAL1gator II. In the receive direction (from the ATM network to the line) , the AAL1gator II updates signals (TL_SIG, TL_SER) on the rising edge of TL_CLK. In addition, the TQUAD samples the corresponding signals on the rising edge of its BTCLK (analogous to TL_CLK) signal. If the output of the FPGA is not inverted, this would cause the TQUAD or EQUAD device to sample the TL_SIG and TL_SER (BTSIG and BTPCM) signals at the same time that they are in transition. Inverters shift the sample point to a time in which all signals are stable.

The COMET version does not require inverters because it allows users to select the active edge (on which it samples BTSIG and BTPCM) of its BTCLK signal.

Please see sections 5 and 6 for FPGA register and programming information.

4.1.3 Clock and Power Supply Circuitry

This is a relatively simple block. It consists of two crystal oscillators: One at 2.048 MHz (50 ppm), and one at 1.544 MHz (50 ppm). These crystals provide the XCLK signal via the FPGA to the individual COMET devices on a per line basis. For instance, if COMET 0 is configured to interface to a T1 line, a 1.544 MHz clock is applied to its XCLK input. Alternatively, if it is to interface to an E1 line, a 2.048 MHz clock signal is applied to its XCLK input.

The AAL1gator II requires a 38.88 MHz clock signal on its SYS_CLK input. Ideally, this is a network derived signal (for example, the 155.52 MHz SONET clock divided by four) with a duty cycle variance of ± 2.5 %. In applications employing SRTS, SYS_CLK need not be network derived, provided N_CLK is network derived.

In applications that do not employ SRTS (for example, synthesizing a nominal T1 or E1 clock), the accuracy of the synthesized clock is dependent on the accuracy of SYS_CLK. For example, to generate a T1 clock with 50 ppm accuracy, SYS_CLK must have 50 ppm accuracy. To lock this synthesized clock to a network clock, SYS_CLK must be derived from the network clock.

To accomplish these requirements in this reference design, the network clock signal (ideally, the 155.52 MHz SONET clock) is input to the FPGA via an SMA connector. The FPGA then provides a 38.88 MHz SYS_CLK signal for the AAL1gator II. As mentioned previously, the FPGA also generates the 2.43 MHz N_CLK signal from this same source.

In addition to the clock circuitry, this block also generates the required +5V/+3.3V supply for the board.

4.1.4 UTOPIA Interface

The AAL1gator II communicates with various ATM devices, such as the ATLAS or the QRT/QSE via the UTOPIA interface. Please refer to section 4.3 of the AAL1gator II data sheet for a discussion of the individual signals.

There are two possible UTOPIA modes of operation for the AAL1gator II: ATM mode and PHY mode. In ATM mode, the AAL1gator II is configured with an ATM layer UTOPIA interface. In PHY mode, it is configured with a PHY layer UTOPIA interface.

When PHY_ENABLE is high, PHY mode is enabled, while if low, the device is in ATM mode. To allow for application specific implementations, an optional jumper will be specified in the schematic diagram. If a PHY mode application is required, the jumper may be installed, otherwise it may be left unconnected, due to the internal pulldown resistor within the device (device defaults to ATM mode).

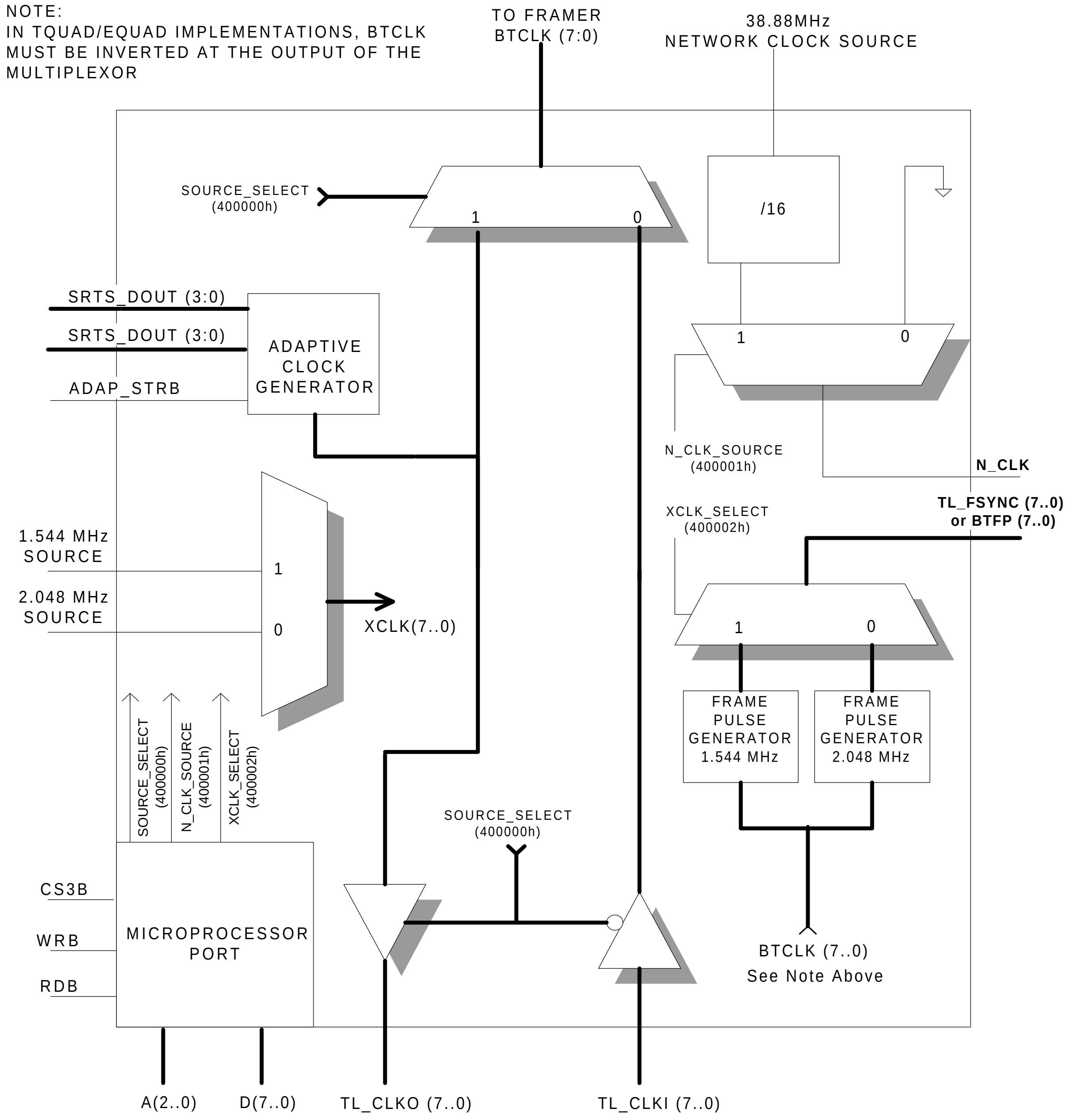


Figure 13. Internal FPGA Circuitry.

4.1.5 Additional Connections

The following table (Table 6) lists the remaining connections to the AAL1gator II not already discussed.

Table 6. Additional Connections to AAL1gator II.

PIN NAME	AAL1GATOR II PIN NUMBER	DESCRIPTION
PHY_ENABLE	56	Determines which UTOPIA mode the UTOPIA interface is configured for. Connect optional resistor to VCC for PHY mode, otherwise leave unconnected for ATM mode operation.
TL_MSYNC(x)	154, 141, 129, 113, 101, 82, 70, 51	Transmit Line Multiframe Synchronization (TL_MSYNC). An edge an these signals indicate the start of a multiframe. Connect to GND if unused.
RL_MSYNC(x)	160, 147, 135, 123, 107, 95, 76, 64	Receive Line Multiframe Synchronization (RL_MSYNC). RL_MSYNC carries the signaling information from the framer. Indicates the start of a multiframe. Connect to GND if unused..
/OE	184	Output Enable (/OE). Active low signal to enable outputs of the device. Connect to GND.
/SCAN_TRST	167	Scan Test Reset (/SCAN_TRST). Reset signal for boundary scan logic. Connect to GND (if JTAG port unused) or header.
SCAN_TMS	166	Scan Test Mode Select (SCAN_TMS). Mode select signal for boundary scan logic. Connect to header.
SCAN_TDI	165	Scan Test Data Input (SCAN_TDI). Serial data input for boundary scan.

PIN NAME	AAL1GATOR II PIN NUMBER	DESCRIPTION
		Connect to header.
SCAN_TDO	168	Scan Test Data Output (SCAN_TDO). Serial data output from boundary scan. Connect to header
SCAN_TCLK	169	Scan Test Clock (SCAN_TCLK). Clock for boundary scan logic. Connect to 330Ω resistor to GND (if JTAG port unused) or to header.
P_TEST	170	Process Test (P_TEST). Used to measure process test parameters. N.C.
PULLUP_DISABLE	238	Pullup Disable. Used to disable pullup resistors during manufacturing tests. Connect to GND
VDD	multiple	+5V supply
GND	multiple	Ground Reference

N.C. = No Connection

Note: The JTAG port signals (eg. SCAN_TDO) will connect to a header to allow for connection to other devices that have a JTAG port to allow for comprehensive boundary scan testing. On the schematic diagram, the SCAN_TRST pin is connected to a header and pulled up via a 4.7K resistor to VCC, to allow for external connection to a JTAG controller. If the JTAG port is not being used, the SCAN_TRST pin should be connected to ground via a 330Ω resistor.

4.1.6 Line Interface

Although the COMET device has an integral Line Interface Unit, additional circuitry is required before connecting to the T1 or E1 line. Transformers must be placed between the tip and ring lines. In addition, transient voltage suppressors must be placed in series with each line. Each set of tip and ring lines connect to the T1 or E1 line via a bantam connector.

4.1.7 Additional COMET Connections

The following table (Table 7) lists additional connections to the COMET device not previously discussed.

Table 7. Additional Connections to the COMET Device.

PIN NAME	DESCRIPTION
TDAT	Transmit Digital PCM Data (TDAT). Provides the line side NRZ PCM data (when transmit digital interface is enabled). N.C.
TFP	Transmit Digital Frame Pulse (TFP). When enabled, indicates the frame alignment of the line side transmitted PCM stream.
RDAT	Receive Digital Line Data (RDAT). When enabled, samples the line side recovered NRZ PCM data. N.C.
RCLKI	Receive Digital Line Clock (RCLK). When the transmit digital interface is enable, the externally recovered line rate clock is provided on this pin. Connect to GND.
RSYNC	Recovered Clock Synchronization Signal (RSYNC). Provides a 8 kHz timing reference signal for T1 and E1 applications. N.C.
TCLKI	Transmit Clock Reference (TCLKI). May be used a reference for transmit line rate generation. May be any multiple of 8 kHz. Connect to GND.
XCLK	Crystal Clock Input (XCLK). Jitter free, 50% duty cycle clock at 1.544

PIN NAME	DESCRIPTION
	MHz (if T1) or 2.048 MHz if E1. 50 ppm.
ATB	Analog Test Bus (ATB). Reserved for production test. Connect to GND.
TRIMF	Trim Fuse. Reserved for production purposes. Connect to GND
TCK	Test Clock (TCK) Boundary scan clock. Connect via 330Ω resistor to GND (if JTAG port unused) or header.
TMS	Test Mode Select (TMS). Boundary scan test select signal. Connect to header.
TDI	Test Data Input (TDI). Boundary scan input serial data. Connect to header.
TDO	Test Data Output (TDO). Boundary scan serial test data output. Connect to header.
/TRSTB	Test Reset (TRSTB). Boundary scan test reset. Connect to header and VCC via 4.7K resistor, or to RSTB (if JTAG port unused).
VDDOx	Output Power Pins Connect to well decoupled +3.3V supply (in common with VDDI.
VSSOx	Output Ground Pins. Connect to GND in common with VSSI.
VDDIx	Internal Power Pins Connect to well decoupled +3.3V supply in common with VDDOx
VSSIx	Internal Ground Pins Connect to GND in common with VSSOx
BIAS	+5 V Bias. Facilitates 5 V tolerance on inputs. Connect to well decoupled +5 V rail.
TAVD1	Transmit Analog Power. Power for transmit LIU reference circuitry.

PIN NAME	DESCRIPTION
	Connect to analog +3.3V
TAVD2, TAVD3	Transmit Analog Power. Transmit power for transmit LIU output drivers. Connect to analog +3.3V.
TAVD4	Transmit Analog Power. Supplies power for transmit clock synthesis. Connect to analog +3.3 V.
TAVS1	Transmit Analog Ground Connect to analog GND.
TAVS2, TAVS3	Transmit Analog Ground. Connect to analog GND.
TAVS4	Transmit Analog Ground. Connect to analog GND.
RAVD1	Receive Analog Power. Supplies power for receive LIU input equalizer. Connect to analog +3.3V.
RAVD2	Receive Analog Power. Supplies power for analog peak detect and slicer. Connect to analog +3.3 V.
RAVS1, RAVS2	Receive Analog Ground. Connect to analog GND.
QAVD	Quiet Analog Power. Supplies power for core analog circuitry. Connect to analog +3.3 V
QAVS	Quiet Analog Ground. Connect to analog GND.

N.C. = No Connection

Note: The JTAG port signals (eg. TCK) will connect to a header to allow for connection to other devices that have a JTAG port to allow for comprehensive boundary scan testing.

4.1.8 COMET Configuration

Soon after power up, a hardware reset should be performed.

The following minimum additional COMET software configurations are required:

1. Set active edge of BTCLK to falling edge.

2. Set active edge of BRCLK to rising edge.

3. Set BRCLK as output.

4. Set BRFP as output.

For additional information, including COMET configuration data, please refer to Section 6 and the COMET data sheet (PMC-970624).

4.1.9 AAL1gator II Configuration

Please refer to Section 6 and the AAL1gator II data sheet (PMC-980620) for additional information regarding configuration of the AAL1gator II.

4.1.10 The JTAG Port

The JTAG port is connected among all devices to allow for boundary scan testing. The signals are connected in the following way:

1. TMS The Test Mode Select signal is connected in parallel among all COMET devices and the AAL1gator II.

2. TCK The Test Clock signal is connected in parallel among all COMET devices and the AAL1gator II.

3. TRSTB The Test Reset Select signal is connected in parallel among all COMET devices and the AAL1gator II. The source of this signal may either be the JTAG controller, or from a pushbutton activation

4. TDI/TDO The Test Data Input/Test Data Output signal is connected serially among all COMET devices and the AAL1gator II, beginning with the AAL1gator II, and ending with the last COMET device.

The JTAG port signals connect to an externally accessible header.

4.2 AAL1gator II Plus TQUAD/EQUAD Design

The high level design of the AAL1gator II plus TQUAD/EQUAD is illustrated in Figure 14 on page 33. As can be seen, the COMET design and the TQUAD/EQUAD design are nearly identical. However, they do have the following differences:

1. Only 2 TQUAD/EQUAD are devices used in the TQUAD/EQUAD design.

2. The address bus connecting to each individual TQUAD/EQUAD is 10, rather than 9 bits wide.

3. The TQUAD/EQUAD device does not contain an integral Line Interface Unit, therefore a separate LIU such as the QDSX or D3MX must be used.

4. The TQUAD or EQUAD does not require a 1.544 or 2.048 MHz oscillator. Instead, a 37.56 MHz 32/50 ppm (49.152 MHz 32/50 ppm) if EQUAD) oscillator is used.

4.2.1 Microprocessor and Memory System Interface

The same microprocessor and memory system interface as that used in the AAL1gator II plus COMET reference design is used in the AAL1gator II plus TQUAD reference design.

4.2.2 Line Side Interface to the AAL1gator II

As discussed in section 4.1.2, the AAL1gator II plus TQUAD requires additional inverters on the line side interface. This is because the AAL1gator II samples RL_SER and RL_SIG on the falling edge of RL_CLK while the TQUAD asserts the corresponding signals on the falling edge of RCLKO. Therefore, inverters are placed on the RCLKO output to the AAL1gator II device.

4.2.3 Line Interface

Also mentioned previously, the TQUAD does not contain an integral Line Interface Unit. Therefore, an LIU such as the QDSX or D3MX (if used in a channelized DS3 application) must be used. Table 8 lists the pin information for the connections to the line interface unit.

Table 8. Line Interface Unit Connections.

SIGNAL NAME	FUNCTION
XCLK	XCLK. Crystal Clock signal. Nominally 37.056 MHz if T1, 49.152 MHz if E1.

TDP(7)
TDP(6)
TDP(5)
TDP(4)
TDP(3)
TDP(2)
TDP(1)
TDP(0)
	Transmit Digital Positive Line Pulse (TDP). Available when the TQUAD/EQUAD is configured to transmit dual rail data.

TDN(7)
TDN(6)
TDN(5)
TDN(4)
TDN(3)
TDN(2)
TDN(1)
TDN(0)
	Transmit Digital Negative Line Pulse (TDN). Available when the TQUAD/EQUAD is configured to transmit dual rail data.

TCLKO(7)
TCLKO(6)
TCLKO(5)
TCLKO(4)
TCLKO(3)
TCLKO(2)
TCLKO(1)
TCLKO(0)
	Transmit Clock Output (TCLKO) Signals TDN and TDP are updated on the active edge of this signal.

RDP(7)
RDP(6)
RDP(5)
RDP(4)
RDP(3)
RDP(2)
RDP(1)
RDP(0)
	Receive Digital Positive Line Pulse (RDP). Available when the TQUAD/EQUAD is configured to receive dual rail data.

RDN(7)
RDN(6)
RDN(5)
RDN(4)
RDN(3)
	Receive Digital Negative Line Pulse (RDN). Available when the TQUAD/EQUAD is configured to receive dual rail data.

SIGNAL NAME	FUNCTION

RDN(2)
RDN(1)
RDN(0)	

RCLKI(7)
RCLKI(6)
RCLKI(5)
RCLKI(4)
RCLKI(3)
RCLKI(2)
RCLKI(1)
RCLKI(0)
	Receive Line Clock Input (RCLKI). Each input is an externally recovered 1.544 MHz (or 2.048 MHz) line clock that may be enabled to sample the RDN and RDP inputs, when the device is configured to receive dual rail data.

GND	GND. Ground Reference.

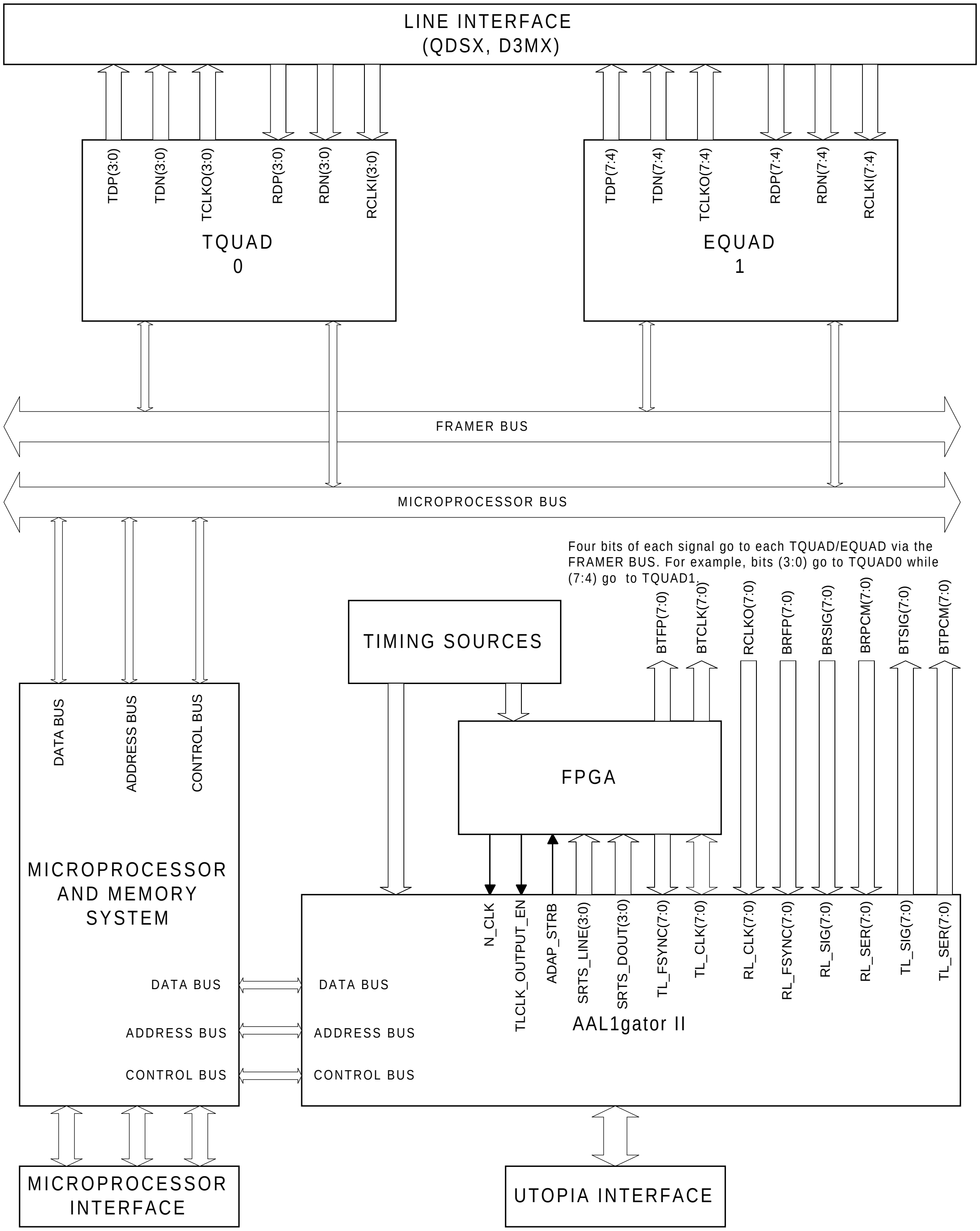


Figure 14. AAL1gator II plus TQUAD/EQUAD High Level Design.

4.2.4 Additional Connections for the TQUAD Device

The following table (Table 9) lists the additional connections for the TQUAD/EQUAD device not previously discussed.

Table 9. Additional TQUAD Device Connections.

PIN NAME	TQUAD PIN NUMBER	DESCRIPTION
BRCLKx	94	Backplane Receive Clock. Either a 1.544 or 2.048 MHz clock signal. This signal is common to all framers within the TQUAD/EQUAD device. Connect to GND..
MENB	45	Multiplex Enable. When asserted low, the four sets of data per TQUAD are combined into a single bit interleaved 12.352 MHz stream. Connect to VCC via 4.7K resistor
RDLSIGx	125, 126, 127, 128	Receive Data Link Signal. Signals are available when the HDLC receiver is enabled. Connect to header.
RDLCLKx	119, 120, 123, 124	Receive Data Line Clock. Clock signals are available on these pins when the HDLC receiver is enabled. Connect to header.
MRD	59	Multiplexed Receive Data (MRD). When MENB is low, four sets of data and signaling information is bit interleaved into a single 12.352 MHz data stream. Connect to header.
TDLSIGx	113, 114, 117, 118	Transmit Data Link Signal. When the associated HDLC transmitter is enabled, signaling data is available on these pins. Connect to header.
TDLCLKx	109, 110, 111, 112	Transmit Data Link Clock. When HDLC is enabled, used to sample the TDLSIG signal. Connect to header.
XCLK	60	XCLK. Crystal Clock Input.

PIN NAME	TQUAD PIN NUMBER	DESCRIPTION
		If TQUAD, connect to 37.056 MHz crystal. If EQUAD, connect to 49.152 MHz crystal. 32/50 ppm
ALE	41	Address Latch Enable. Allows TQUAD to be connected to a multiplexed address data bus. Connect to VCC via 4.7K resistor.
PHAx	18, 52, 89, 105, 121	AC Power Pins. Connect to well decoupled +5 V supply together with PHD.
PHDx	20, 50, 85, 115	DC Power Pins. Connect to well decoupled +5 V supply together with PHA.
PLAx	19, 53, 90, 106, 122, 1	AC Ground Pins Connect to common ground with PLD
PLDx	21, 51, 86, 116	DC Ground Pins Connect to common ground with PLA.

N.C. = No Connection

4.2.5 TQUAD/EQUAD Configuration

A software reset should occur soon after power up to restore the TQUAD or EQUAD to its default configuration state. Please refer to Section 6 and the TQUAD or EQUAD data sheet (PMC-940910, PMC-951013) for additional configuration information.

5 MEMORY MAP AND REGISTER DEFINITIONS

The memory map for the AAL1gator II reference design is as indicated in Table 10.

Table 10. AAL1gator II Reference Design Memory Map.

DEVICE	BASE ADDRESS	RANGE
AAL1gator II	000000h	000000h – 01FFFFh
AAL1gator II CMD_REG	020000h	020000h – 03FFFFh
COMET0	200000h	200000h – 2001FFh
TQUAD0	200000h	200000h – 2003FFh
COMET1	240000h	240000h – 2401FFh
TQUAD1	240000h	240000h – 2403FFh
COMET2	280000h	280000h – 2801FFh
COMET3	2C0000h	2C0000h – 2C01FFh
COMET4	300000h	300000h – 3001FFh
COMET5	340000h	340000h – 3401FFh
COMET6	380000h	380000h – 3801FFh
COMET7	3C0000h	3C0000h – 3C01FFh
FPGA SOURCE_SELECT Register	400000h	400000h
FPGA N_CLK_SOURCE Register	400001h	400001h
FPGA XCLK_SELECT Register	400002h	400002h

To communicate with the FPGA and its internal registers, a programmable chip select on the system microprocessor should be configured to be asserted when an address in the range 400000h – 400002h is driven onto the address bus. The VHDL code within the FPGA will then examine the WRB and RDB to determine if a register read or write is being performed. The FPGA then either drives the data bus with the selected registers contents, or updates one of the following registers with the data driven on the data bus by the system microprocessor.

As mentioned previously, the FPGA allows the support of several configurable options. This is accomplished through the use of the following registers within the FPGA:

Table 11. SOURCE_SELECT Register (400000h).

bit 7	6	5	4	3	2	1	0
ss7	ss6	ss5	ss4	ss3	ss2	ss1	ss0

The SOURCE_SELECT determines whether the signal transmitted to the COMET or TQUAD device via the BTCLK pin is from the AAL1gator II TL_CLK pin or from an external source. If an external source is selected, adaptive clock mode is enabled, and the FPGA synthesizes an external clock at the line rate. The synthesized clock appears at the TL_CLKIO<7..0> and BTCLK<7..0> pins

ssX:

0 = AAL1gator II internal clock (nominal, recovered, or SRTS) sources clock signal via the FPGA to the COMET or TQUAD BTCLK<X> pin.

1 = External Clock mode is selected. Adaptive clock recovery mode is in use. The FPGA provides a line rate clock to both the AAL1gator II and the COMET or TQUAD device.

Table 12. N_CLK_SOURCE Register (400001h).

Bit 7	6	5	4	3	2	1	0
srts_enable	x	x	x	x	x	x	x

Table 12 presents the structure of the N_CLK_SOURCE Register. If the srts_enable bit is set to 0, N_CLK is connected to GND. Otherwise, if the bit is set to 1, a 2.43MHz (38.88 MHz divided by 16) signal is presented on N_CLK. The affect of this register is to allow for disabling of Synchronous Residual Time Stamp (SRTS) on the AAL1gator II. If the N_CLK signal is connected to GND, SRTS is disabled. Bits 6 through 0 are unused.

srts_enable:

1 = 2.43MHz signal presented on N_CLK. SRTS is enabled.

0 = N_CLK is set to a logic 0. SRTS is disabled.

Table 13. XCLK_SELECT Register (400002h).

Bit 7	6	5	4	3	2	1	0
xclk7	xclk6	xclk5	xclk4	xclk3	xclk2	xclk1	xclk0

Table 13 lists the details of the XCLK_SELECT register. The purpose of this register to allow support of both T1 and E1 lines. If a T1 line is used, that lines bit (for example xclk7) must be set to 1. Otherwise, if an E1 line is in use, the associated lines bit must be set to 0. This feature is available on a per line basis.

The effect of this register is seen in two places on a per line basis:

1. The register determines which clock signal (1.544 MHz or 2.048 MHz) is output from the FPGA to each individual COMETs XCLK pin.
2. The register determines which framing pulse format (T1 or E1) is used. The effect is seen on the outputs the AAL1gator II's TL_FSYNC<7..0> pins and the COMET or TQUAD/EQUAD's BTFP pins.

xclk:

1 = Line is T1. If using a COMET device, the COMET receives a 1.544MHz clock signal on its XCLK pin. In addition, a pulse one BTCLK period wide is generated every 193 bits, and output to the AAL1gator II and COMET or TQUAD device.

0 = Line is E1. If using a COMET device, the COMET receives a 2.048MHz clock signal on its XCLK pin. In addition, a pulse one BTCLK period wide is generated every 256 bits, and output to the AAL1gator II and COMET or EQUAD device.

6 SOFTWARE CONFIGURATION

This section describes how to configure the AAL1gator II to work with T1 and E1 lines. A hypothetical scenario consisting of the AAL1gator II interfacing to 4 T1 and 4 E1 lines will be explored. Lines 0 through 3 will be T1, while lines 4 through 7 will be E1.

6.1 AAL1gator II Configuration

The basic configuration steps for the AAL1gator II are as outlined in Figure 15 (next page). As can be seen, the first step is to enter a SW_RESET state. There are two possible ways to enter the SW_RESET state:

1. By performing a hardware reset. Once the hardware reset is removed, the AAL1gator II enters the SW_RESET state.
2. By asserting bit 5 (SW_RESET) of the CMD_REG (base address 20000h).

Once in the SW_RESET state, various configurable options are set by writing to the data structures in memory. Among many others, these options include:

1. Type of line (T1 or E1).
2. Source of the TL_CLK signal.
3. The value of empty bytes in partially filled cells.
4. OAM cell format.

When the AAL1gator II has been set up in the desired format, the CMD_REG_ATTN bit (bit 3 in the CMD_REG) should be set to 1. Finally, to complete the configuration, the SW_RESET bit in the CMD_REG should be deasserted.

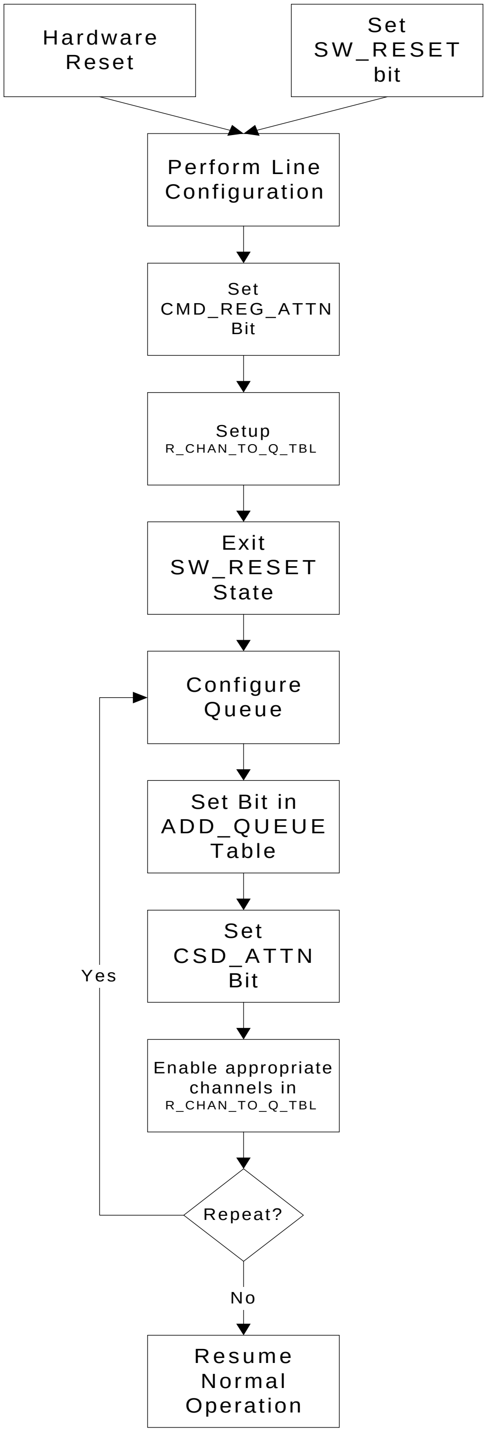


Figure 15. Flowchart for configuring the AAL1gator II.

To configure the AAL1gator II to operate with the conditions given above, the following operations should be performed:

1. Enter the SW_RESET state

2. Write a 1 to bit 6 (MIXED_MODE_EN) of the COMP_LIN_REG (base address 1h). This permits the device to operate with a mix of T1 and E1 lines.

3. Write to the remaining bits of the COMP_LIN_REG as necessary for each individual implementation.

4. Write to the LINE_STR_MODE_X registers. These eight registers store the per line configuration. In these registers, users may set each line in T1 or E1, and determine the source of TL_CLK.

   To set lines 0 through 3 for T1 mode, write a 1 to bit 13 (T1_MODE) of LINE_STR_MODE_0 through LINE_STR_MODE_3 (addresses 10h through 13h).

   To set lines 4 through 7 for E1 mode, write a 0 to bit 13 of LINE_STR_MODE_4 through LINE_STR_MODE_7.

   The TL_CLK source for each line may be set at the same time that the line mode is set.

   The remaining bits may be set as required, and as indicated in the AAL1gator II data sheet.

Table 14. TL_CLK options for the AAL1gator II.

CLK_SOURCE(5:4)	DESCRIPTION
00	Use external clock (TL_CLK is an input)
01	Looped: Loop the recovered RL_CLK as the source.
10	Nominal: Generate a clock of the nominal T1 or E1 frequency based on SYS_CLK
11	SRTS: Generate a clock based on the received SRTS values.

5. Set CMD_REG_ATTN (bit 3) in the CMD_REG (20000h)

6. Configure the R_CHAN_TO_Q_TBL data structure (base address 08200h) as required. See the AAL1gator II data sheet, section 7.8.6 for further details.

7. Exit the SW_RESET state.

8. Configure a queue via the T_QUEUE_TBL (base address 2000h) as required. See the AAL1gator II data sheet, section 7.6.8 for further details.

9. Set corresponding bit for the queue just added in the ADD_QUEUE table (base address 30h). See the AAL1gator II data sheet, section 7.6.8 for further details.

10. Set CSD_ATTN (bit 4) in the CMD_REG (base address 20000h).

11. Perform steps 8 through 10 as necessary, adding only queue at a time (helps minimize CDV due to clumping).

Table 15. Summary of minimum AAL1gator II line configurations.

REGISTER	BASE ADDRESS	VALUE	FUNCTIONAL DESCRIPTION
COMP_LIN_REG	1h	0040h	AAL1gator II is set to operate with mixed lines.
LINE_STR_MODE_0	10h	2000h	Line 0 is in T1 mode, using adaptive clock recovery (external TL_CLK source).
LINE_STR_MODE_1	11h	2010h	Line 1 is in T1 mode. TL_CLK outputs a looped RL_CLK.
LINE_STR_MODE_2	12h	2020h	Line 2 is in T1 mode. AAL1gator II generates a nominal T1 clock, and outputs it on TL_CLK.
LINE_STR_MODE_3	13h	2030h	Line 3 is in T1 mode. AAL1gator II is using SRTS clock recovery on this line.
LINE_STR_MODE_4	14h	0000h	Line 0 is in E1 mode, using adaptive clock recovery (external TL_CLK source).
LINE_STR_MODE_5	15h	0010h	Line 1 is in E1 mode. TL_CLK outputs a looped RL_CLK.
LINE_STR_MODE_6	16h	0020h	Line 2 is in E1 mode. AAL1gator II generates a nominal T1 clock, and outputs it on TL_CLK.
LINE_STR_MODE_7	17h	0030h	Line 3 is in E1 mode. AAL1gator II is using SRTS clock recovery on this line.

The minimum line configuration for the AAL1gator II has now been performed to enable it to work with 4 T1 and 4 E1 lines. For additional configuration options, please refer to the AAL1gator II data sheet.

6.2 COMET Configuration

The next step is to perform the necessary configurations on the COMET or TQUAD/EQUAD devices. First, let us assume that 8 COMET devices are being used. Let us further assume that COMET devices 0 though 3 will operate in T1 mode, while COMET devices 4 through 7 operate in E1 mode.

The following steps are required to configure the COMET devices:

1. Write a 0 to bit 0 (E1/T1B) of the Global Configuration register (000h) of COMETs 0 through 3.
2. Write a 1 to bit 0 (E1/T1B) of the Global Configuration register (000h) of COMETs 4 through 7.
3. For COMETs 0 through 3, write 18h to the BRIF Configuration register (030h). This configures the COMET in the following way:
    - Full frame mode
    - BRCLK is an output
    - BRPCM and BRSIG are updated on the rising edge of BRCLK.
    - BRFP is updated on the rising edge of BRCLK.
    - BRCLK is at the backplane rate (1.544 MHz).
4. For COMETs 4 through 7 write 19h to the BRIF Configuration register. This action configures the COMET in the same way as step 3, with the exception that BRCLK is at the E1 rate of 2.048 MHz.
5. For all COMET devices, write 00h to the BRIF Frame Pulse Configuration register (031h). This action sets the BRFP pin as an output, and causes BRFP to pulse high for one BRCLK period every 193 (T1) or 256 (E1) bits.
6. For COMETs 0 through 3, write 20h to the BTIF Configuration register (040h). This action configures the COMETs in the following way:

- Full Frame mode.
- BTCLK is an input.
- BTSIG and BTPCM are sampled on the falling edge of BTCLK.
- BTFP is sampled on the falling edge of BTCLK.
- BTCLK is at the backplane rate (1.544 MHz).

7. For COMETs 4 through 7, write 21h to the BTIF Configuration register (040h). This action configures the COMET in the same way as step 6, with the exception that BTCLK is at the E1 rate of 2.048 MHz.

Table 16. Summary of the minimum COMET device configurations.

COMET	REGISTER	BASE ADDRESS	VALUE	FUNCTIONAL DESCRIPTION
0-3	Global Configuration	000h	10h	Sets the COMET devices to operate in T1 mode.
4-7	Global Configuration	000h	00h	Sets the COMET devices to operate in E1 mode.
0-3	BRIF Configuration	030h	18h	COMET configured for: • Full Frame Mode • BRCLK is an output • BRPCM, BRSIG, BRFP updated on rising edge of BRCLK. • BRCLK is at the backplane rate (1.544 MHz).
4-7	BRIF Configuration	030h	19h	COMET configured for: • Full Frame Mode • BRCLK is an output • BRPCM, BRSIG, BRFP updated on rising edge of BRCLK. • BRCLK is at 2.048 MHz.
0-7	BRIF Frame Pulse Configuration	031h	00h	BRFP is an output

COMET	REGISTER	BASE ADDRESS	VALUE	FUNCTIONAL DESCRIPTION
0-3	BTIF Configuration	040h	20h	COMET configured for: • Full Frame Mode • BTCLK is an input • BTPCM, BTSIG, BTFP sampled on rising edge of BTCLK. BTCLK is at the backplane rate (1.544 MHz).
4-7	BTIF Configuration	040h	21h	COMET configured for: • Full Frame Mode • BTCLK is an input • BTPCM, BTSIG, BTFP sampled on rising edge of BTCLK. BTCLK is at the backplane rate (2.048 MHz).

6.3 TQUAD/EQUAD Configuration

An alternative to using COMET devices as the line interface is to use a PM4344 TQUAD for lines 0 through 3 and a PM6344 EQUAD for lines 4 through 7. The device should be placed in a default state (either by asserting RSTB, or asserting the RESET register bit. Once this is completed, the RCLKOSEL bit (bit 7) of the Receive Backplane Options register (001h, 081h, 101h, 181h) should be set to a logic 1. This causes BRPCMx and BRSIGx to be updated on the falling edge of RCLKOx. Once this is completed, user and application specific configurations may then be performed. Please refer to the TQUAD or EQUAD data sheet for further information.

6.4 FPGA Configuration

Table 17. Summary of FPGA Configurations.

REGISTER	ADDRESS	VALUE	FUNCTIONAL DESCRIPTION
SOURCE_SELECT	400000h	11h	Lines 0, 4 external clock source.
N_CLK_SOURCE	400001h	80h	SRTS enabled.
XCLK_SELECT	400002	0Fh	Lines 0-3 are T1, 4-7 are E1.

7 IMPLEMENTATION DESCRIPTION

Both reference design schematics were captured using Cadence software Concept Schematics Capture tool.

7.1 COMET Version Schematics

7.1.1 ROOT DRAWING, Sheet 1

This sheet provides an overview of the major functional blocks of the AAL1gator II plus COMET reference design. In addition it shows the interconnection between the various blocks.

7.1.2 COMET BLOCK, Sheets 2-9

These sheets show the COMET device and its power circuitry. The power circuitry includes a schottky diode for protection while powering up the COMET device and separate filtering circuitry for the analog and digital power pins. In addition, the JTAG port is connected among the 8 COMET devices, and the AAL1gator II.

7.1.3 LINE INTERFACE, Sheets 10-13

These schematics show the termination, magnetics and protection circuitry for the line interface. A 1:2.42 transformer is used to couple the COMET transmit and receive line to the connectors. The LC01-6 transient voltage suppressor (TVS) and the Raychem PTC provide over voltage protection. A single footprint is provided for both the bantam and RJ48C connectors.

7.1.4 FPGA BLOCK, Sheet 14

This sheet shows the interconnection of the FPGA between the AAL1gator II and the COMET devices. Two oscillators are present to supply the XCLK signal to the COMET devices. 0.1uF bulk capacitors are specified, and should be placed at the corners of the FPGA.

7.1.5 MEMORY SYSTEM BLOCK, Sheet 15

This sheet indicates the connections between the system microprocessor, and the AAL1gator II, COMET devices, and FPGA. 20 bit buffers and 16 bit data transceivers are used to minimize part count.

7.1.6 AAL1gator II BLOCK, Sheet 16

This sheet shows how the AAL1gator II is connected into the system. A connector is provided to provide access to the UTOPIA bus externally.

7.1.7 MICRO INTERFACE, Sheet 17

This page shows the connections between the system microprocessor, and the board. The LT1528 low drop out voltage regulator provides up to 3A at 3.3V to the 8 COMET devices. The LT1528 should be in the DD package, so that no additional heat sink is required.. Included is a pushbutton to provide a hardware reset.

7.2 TQUAD Version Schematics

7.2.1 ROOT DRAWING, Sheet 1

This sheet provides an overview of the major functional blocks of the AAL1gator II plus TQUAD/EQUAD reference design. In addition it shows the interconnection between the various blocks.

7.2.2 TQUAD/EQUAD BLOCK, Sheets 2,3

These sheets show how to connect a TQUAD or EQUAD into the system. If a T1 line is in use, the PM4344 TQUAD should be used (sheet 2). If an E1 line is in use, the PM6344 EQUAD should be used. Included in the schematics are 0.1 µF bulk capacitors, and 0.01µF decoupling capacitors. These should be placed at the corners of each device. Also included is 37.056MHz or 49.152MHz oscillator.

7.2.3 AAL1gator II BLOCK, Sheet 4

This schematic diagram is the same as that for the COMET version.

7.2.4 FPGA BLOCK, Sheet 5

This sheet is similar to the COMET version with the following exceptions. First, the 1.544MHz and 2.048MHz oscillator are not present since the TQUAD and EQUAD operate with 37.056MHz and 49.152MHz oscillators. Second, in the COMET version there is an XCLK bus which supplies the individual COMET devices with a 1.544MHz or 2.048MHz clock signal. This signal is not required by the TQUAD or EQUAD and is not supplied.

7.2.5 MEMORY SYSTEM BLOCK, Sheet 6

This sheet is identical (with the exception of signal names) to the COMET version.

7.2.6 MICRO INTERFACE BLOCK, Sheet 7

This sheet shows the interconnection between the system microprocessor and the board. Included is a pushbutton to provide a hardware reset.

7.2.7 LIU INTERFACE BLOCK, Sheet 8

This sheet shows how the TQUAD or EQUAD could interface with a Line Interface Unit such as the PM4314 QDSX.

8 MODULARIZATION ISSUES

Some implementations may require a modular deployment of this reference design. This section describes one possible strategy for creating a modular system based on either the AAL1gator II plus COMET or AAL1gator II plus TQUAD/EQUAD reference design.

The strategy developed involves separating the AAL1gator II from the desired framer device (COMET or TQUAD/EQUAD). In this way, one card would contain the AAL1gator II, the FPGA, and microprocessor and memory system as indicated in Figure 16 on the next page. As can be seen, this modularization requires breaking the connections between the AAL1gator II and COMET or TQUAD/EQUAD and inserting a 96 pin DIN connector. Figure 17 on page 51 indicates the suggested layout for the COMET board, while Figure 18 on page 52 shows the suggested layout for the TQUAD/EQUAD board. The interface between the AAL1gator II board and the board containing either the COMET or TQUAD/EQUAD was standardized as listed in Table 18 on page 53. This table lists the interface connections from the AAL1gator II point of view. The associated signals on the COMET or TQUAD/EQUAD board can be determined using either Figure 17 or Figure 18.

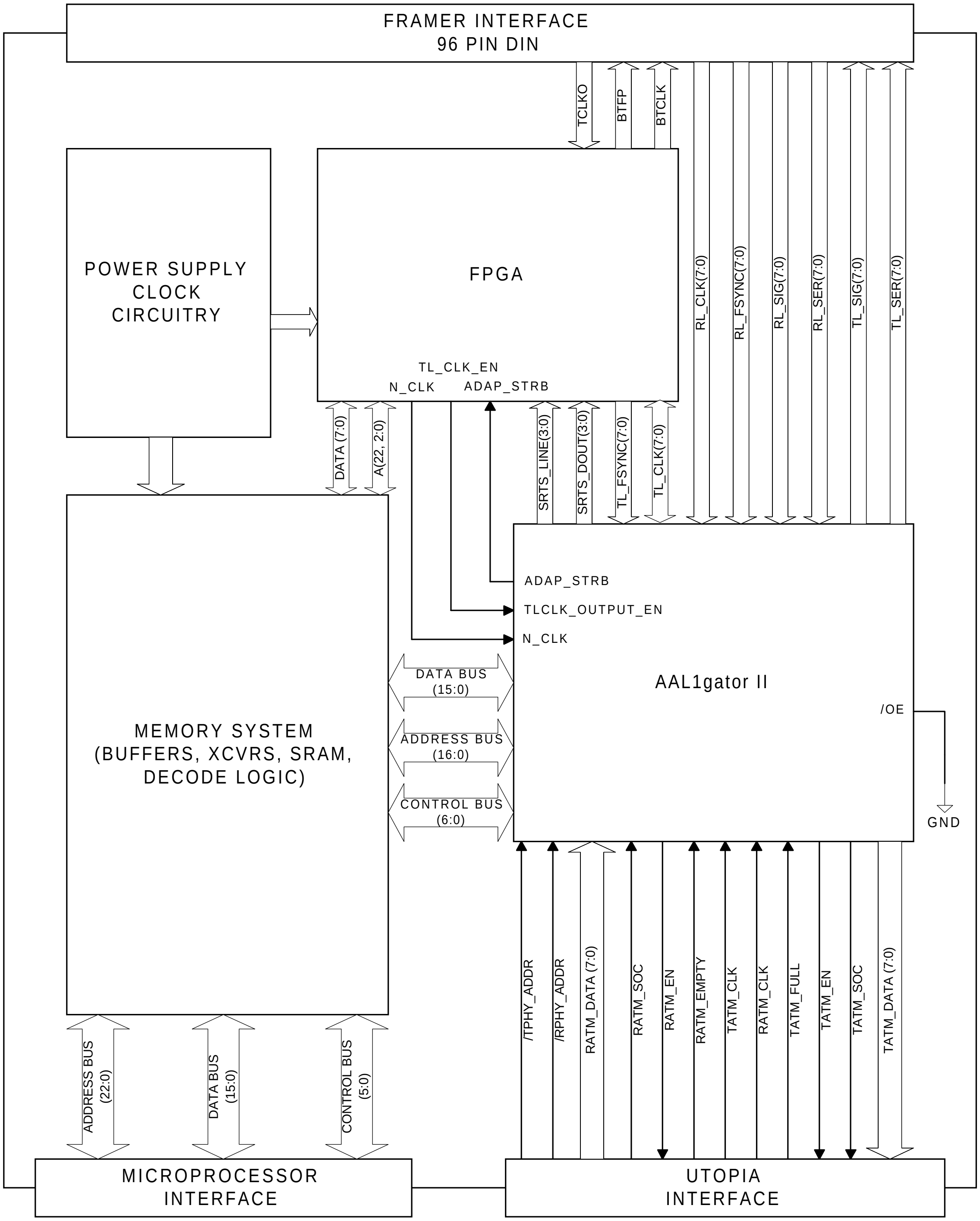


Figure 16. AAL1gator II board.

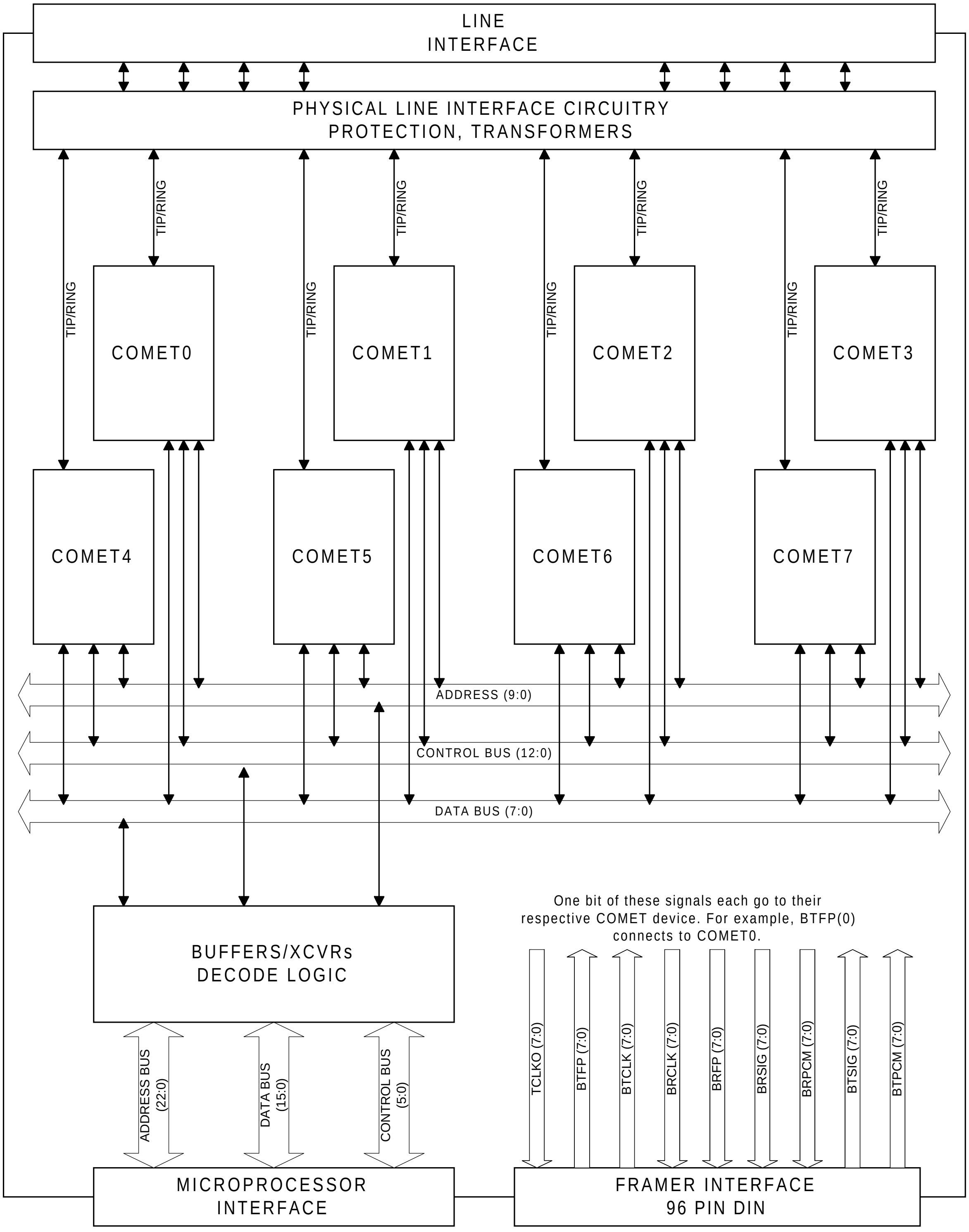


Figure 17. COMET board.

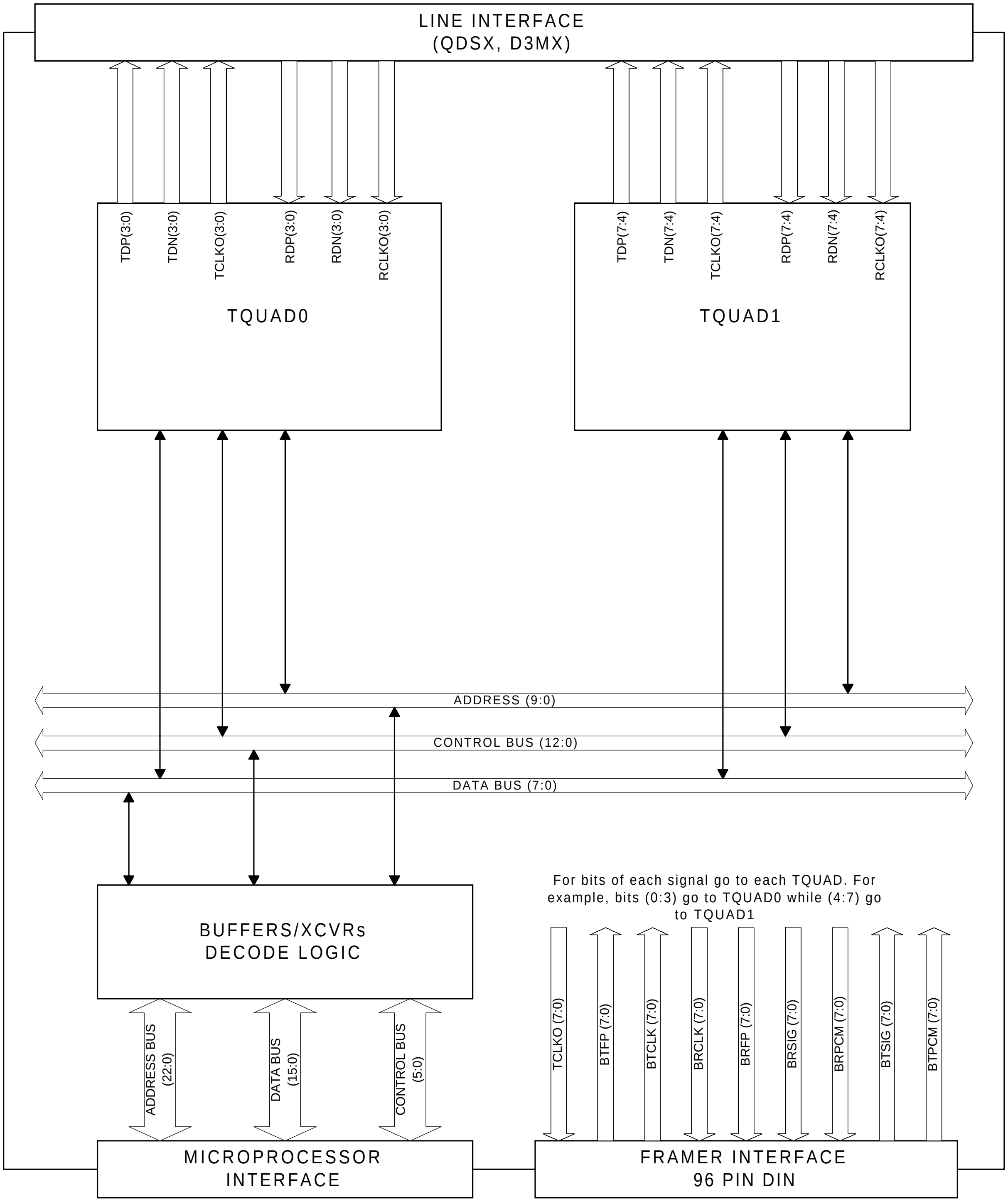


Figure 18. TQUAD board.

Table 18. AAL1gator II to Framer Interface.

PIN NAME	PIN TYPE	PIN NUMBER	FUNCTION

TCLK0(7)
TCLK0(6)
TCLK0(5)
TCLK0(4)
TCLK0(3)
TCLK0(2)
TCLK0(1)
TCLK0(0)
	Input (from Framer)	B(31)
		B(27)
		B(23)
		B(19)
		B(15)
		B(11)
		B(7)
		B(3)
			Transmit Clock Output. TCLKO is a clock at the transmit line rate. The FPGA uses this signal to generate the BTFP signal.

BTFP(7)
BTFP(6)
BTFP(5)
BTFP(4)
BTFP(3)
BTFP(2)
BTFP(1)
BTFP(0)
	Output (to Framer)	A(7)
		A(6)
		A(5)
		A(4)
		A(3)
		A(2)
		A(1)
		A(0)
			Backplane Transmit Frame Pulse (BTFP). In T1 mode, a pulse at least one BTCLK period wide must occur every 193 bits. In E1 mode, it must occur every 256 bits. This signal is generated by the FPGA after passing through either the T1 or E1 pulse generator.

BTCLK(7)
BTCLK(6)
BTCLK(5)
BTCLK(4)
BTCLK(3)
BTCLK(2)
BTCLK(1)
BTCLK(0)
	Output	A(15)
		A(14)
		A(13)
		A(12)
		A(11)
		A(10)
		A(9)
		A(8)
			Backplane Transmit Clock (BTCLK). The signal may be 1.544 MHz, or a multiple of 2.048 MHz. The source of the signal is determined by register settings of the FPGA.

RL_CLK(7)
RL_CLK(6)
RL_CLK(5)
RL_CLK(4)
RL_CLK(3)
RL_CLK(2)
RL_CLK(1)
RL_CLK(0)
	Input	C(7)
		C(6)
		C(5)
		C(4)
		C(3)
		C(2)
		C(1)
		C(0)
			Receive Line Clock. A 1.544 or 2.048 MHz clock signal derived from the recovered line rate timing.

GND	n/a	B(30) – B(28), B(26) – B(24), B(22) – B(20), B(18) – B(16), B(14) – B(12), B(10) – B(8), B(6) – B(4),	GND. Ground Reference.

PIN NAME	PIN TYPE	PIN NUMBER	FUNCTION
		B(2) – B(0)	

RL_FSYNC(7)
RL_FSYNC(6)
RL_FSYNC(5)
RL_FSYNC(4)
RL_FSYNC(3)
RL_FSYNC(2)
RL_FSYNC(1)
RL_FSYNC(0)
	Input	C(15)
		C(14)
		C(13)
		C(12)
		C(11)
		C(10)
		C(9)
		C(8)
			Receive Line Frame Synchronization (RL_FSYNC) carry receive frame information from the framer.

RL_SIG(7)
RL_SIG(6)
RL_SIG(5)
RL_SIG(4)
RL_SIG(3)
RL_SIG(2)
RL_SIG(1)
RL_SIG(0)
	Input	C(23)
		C(22)
		C(21)
		C(20)
		C(19)
		C(18)
		C(17)
		C(16)
			Receive Line Signaling (RL_SIG) bits. RL_SIG carries the CAS signaling information from the framers.

RL_SER(7)
RL_SER(6)
RL_SER(5)
RL_SER(4)
RL_SER(3)
RL_SER(2)
RL_SER(1)
RL_SER(0)
	Input	C(31)
		C(30)
		C(29)
		C(28)
		C(27)
		C(26)
		C(25)
		C(24)
			Receive Line Serial Data (RL_SER) bits. RL_SIG carries the receive data from the framers.

TL_SIG(7)
TL_SIG(6)
TL_SIG(5)
TL_SIG(4)
TL_SIG(3)
TL_SIG(2)
TL_SIG(1)
TL_SIG(0)
	Output	A(23)
		A(22)
		A(21)
		A(20)
		A(19)
		A(18)
		A(17)
		A(16)
			Transmit Line Signaling (TL_SIG) bits. TL_SIG carries the CAS signaling outputs to the framer devices.

TL_SER(7)
TL_SER(6)
TL_SER(5)
TL_SER(4)
TL_SER(3)
TL_SER(2)
TL_SER(1)
TL_SER(0)
	Output	A(31)
		A(30)
		A(29)
		A(28)
		A(27)
		A(26)
		A(25)
		A(24)
			Transmit Line Serial Data (TL_SER). TL_SER carries the serial data to the framer devices.

9 REFERENCES

1. PMC-Sierra, PMC-970624, “PM4351 COMET Combined E1/T1 Transceiver/Framer”, November 1998, Issue 5.

2. PMC-Sierra, PMC-940910, “PM4344 TQUAD Quadruple T1 Framer”, June 1998, Issue 7.

3. PMC-Sierra, PMC-980620, “PM73121 AAL1gator II AAL1 Segmentation and Reassembly Processor Data Sheet”, December 1998, Issue 2.

4. PMC-Sierra, PMC-951013, “PM6344 EQUAD Quadruple E1 Framer”, June 1998, Issue 5.

10 APPENDIX A: BILL OF MATERIALS (COMET VERSION)

Table 19. COMET version Bill of Materials.

DESCRIPTION	PART NUMBER	MANUFACTURER	REF DES	QTY
CAPACITOR, 0.01UF, 50V, X7R_805	ECU-V1H102KBN	PANASONIC	C2, C3, C5, C6, C9-C11, C13, C16-C18, C20,C21, C23, C24, C27-C29,C31, C34-C36, C38, C39,C41, C42, C45-C47, C49, C52-C54, C56, C57,C59, C60, C63-C65, C67, C69-C71, C74, C75,C77, C78, C81-C83, C85, C88-C90, C92, C93,C95, C96, C99-C101, C103, C106-C108, C110,C111, C113, C114, C117-C119, C121, C124-C126, C128, C129, C131, C132, C135-C137, C139, C142- C144, C155-C162, C171-C173, C190-C194	104
CAPACITOR, 0.1UF, 50V, X7R_805	ECJ-2VB1E104K	PANASONIC	C8, C14, C15, C26, C32, C33, C44, C50, C51, C62, C68, C72, C80, C86, C87, C98, C104, C105, C116, C122, C123, C134, C140, C141, C163-C170, C174-C180, C185-C189	44
CAPACITOR, 0.47UF, 25V, TANT TEH	ECS-H1EY474R	PANASONIC	C1, C19, C37, C55, C73, C91, C109, C127, C183	9
CAPACITOR, 1.0UF, 16V, Y5V_805	ECJ-2VF1C105Z	PANASONIC	C182	1
CAPACITOR, 22PF, 50V, NPO_805	ECU-V1H220JCN	PANASONIC	C153, C154	2
CAPACITOR, 22UF, 6.3V, TANT TEH	EEV-FC0J221P	PANASONIC	C4, C12, C22, C30, C40, C48, C58, C66, C76, C84, C94, C102, C112, C120, C130, C138	16
CAPACITOR, 4.7UF, 10V,	ECS-H1AX475R	PANASONIC	C145-C152	8

DESCRIPTION	PART NUMBER	MANUFACTURER	REF DES	QTY
TANT TEH				
CAPACITOR, 47UF, 10V, TANT TEH	ECS-H1AD476R	PANASONIC	C181, 184, C195-C197	5
CAPACITOR, 68UF, 6.3V, TANT TEH	ECS-H0JD686R-	PANASONIC	C7, C25, C43, C61, C79, C97, C115, C133	8
DIODE, 1N5817, MELF	1N5817MCT	DIODES INC	D1-D8	8
LED SUPERGREEN, SURFACE MOUNT	7002X5	INDUSTRIAL DEVICES	D9, D10	2
FUSE, 1.500A	F1T66CT-ND	LITTLEFUSE	F1	1
RJ-48	95001-9841	MOLEX	J2, J5, J8, J11, J14, J17, J20, J23	8
SMA-BASE	901-144	AMPHENOL	J26	1
BANTAM-BASE, BLACK	PC834	ADC	J1, J3, J4, J6, J7, J9, J10, J12, J13, J15, J16, J18, J19, J21, J22, J24	16
HEADER2 100 MIL BASE	640452-2	AMP	JP1, JP2	1
HEADER5 100 MIL BASE	640452-5	AMP	J25, J29	2
HEADER6 100 MIL BASE	640452-6	AMP	J30	1
HEADER10 100 MIL BASE	1-640452-0	AMP	J27, J28	2
DIN96 MALE-BASE	650473-5	AMP	P1	1
2x20 MALE, BASE	87331-4020	MOLEX	P2	1
RESISTOR, 1.0, 1%, 805	ERJ-6RQF1.0V	PANASONIC	R13-R15, R19-R21, R25-R27, R31-R33, R37-R39, R43-R45, R49-R51, R55-R57	24
RESISTOR, 100, 5%, 603	ERJ-3GSY100	PANASONIC	R67-R70, R77-R80, R88-R91, R100-R103	16
RESISTOR, 100K, 5%, 805	ERJ-6GEY100K	PANASONIC	R17, R23, R29, R35, R41, R47, R53, R59, R66	9
RESISTOR, 10K, 5%, 805	ERJ-6GEY10K	PANASONIC	R8	1
RESISTOR, 12.7, 1%, 603	ERJ-3EK712.7V	PANASONIC	R60, R61, R63, R64, R71-R74, R82, R84, R85, R92-R94, R96, R97	16
RESISTOR,	ERJ-3EK718.2V	PANASONIC	R62, R65, R75, R81,	8

DESCRIPTION	PART NUMBER	MANUFACTURER	REF DES	QTY
18.2, 1%, 603			R83, R86, R95, R98	
RESISTOR, 1K0, 5%, 805	ERJ-6GEY1K0	PANASONIC	R3	1
RESISTOR, 270, 5%, 805	ERJ-6GEY270	PANASONIC	R1, R2	2
RESISTOR, 330, 5%, 805	ERJ-6GEY330	PANASONIC	R5, R7, R104-R108, R115, R116, R119, R122, R125, R128, R131, R134, R148	17
RESISTOR, 4.7, 5%, 805	ERJ-6GEY4.7	PANASONIC	R12, R18, R24, R30, R36, R42, R48, R54	8
RESISTOR, 4.7K, 5%, 805	ERJ-6GEY4K7	PANASONIC	R4, R6, R16, R22, R28, R34, R40, R46, R52, R58, R117, R118, R120, R138-143, R149	20
RESISTOR, 49.9, 1%, 805	ERJ-6ENF49.9V	PANASONIC	R145-R147	3
RESISTOR ARRAY 4K7, SMD	766161472G	CTS	RN6	1
RESISTOR ARRAY, 10K, SMD	742163103J	CTS	RN3-RN5	3
RESISTOR ARRAY, 68, SMD	742163680J	CTS	RN1, RN2	2
SWITCH, NO PB, 6MM, VERT	EVQ-PAG04K	PANASONIC	SW1	1
TRANSFORMER, 1:2.42	MI50436	MIDCOM	T1-T4, T7-T10, T13-T16, T19-T22	16
TRANSFORMER	PE68624	PULSE ENGINEERING	T5, T6, T11, T12, T17, T18, T23, T24	8
THERMISTOR, PTC	TR250-180U	RAYCHEM	TR1-TR32	32
TEST_POINT	PZC36SAAN	SULLINS ELECTRONICS	TP1-TP4	4
IC, AAL1GATOR II QFP	PM73121-RI-P	PMC-SIERRA	U1	1
IC, COMET, CABGA	PM4351-NI	PMC-SIERRA	U2-U9	8
IC, CMOS QUAD NAND GATE, SOIC	SN74HCT08	TEXAS INSTRUMENTS	U10	1
IC, CMOS 3 to 8 DECODER, SOIC	MC74HC T138AD	MOTOROLA	U11	1
IC, FPGA, PQFP	A1425A–PQ–160C	ACTEL	U12	1
IC, CMOS HEX INVERTER, SOIC	SN74HCT 04D	TEXAS INSTRUMENTS	U13	1
IC, POWER SUPPLY MONITOR,	MAX701ESA	MAXIM	U14	1

DESCRIPTION	PART NUMBER	MANUFACTURER	REF DES	QTY
SOIC8				
IC, TVS DIODE ARRAY, SOIC8	SRDA3.3_4	SEMTECH	U18, U19, U24, U25, U30, U31, U36, U37	8
IC, TVS, SO-16W	LC01-6	SEMTECH	U20-U23, U26-U29, U32-U35, U38-U41	16
IC, 3.3V PRECISON VOLTAGE REGULATOR, DD PACKAGE	LT1528	LINEAR TECHNOLOGIES	U42	1
IC, FAST 20 BIT BUFFER, SOP	IDT74FCT162827CT	IDT	U44, U46	2
IC FAST 16 BIT TRANSCEIVER, SOP	IDT74FCT162646CT	IDT	U45, U49	2
IC 1MB SRAM, 10NS, TSOP	KM681002B-10	SAMSUNG	U47, U48	2
TTL OSC, 1.544MHZ, 32PPM	K1125BA	CHAMPION	Y2	1
TTL OSC, 2.048MHz, 50PPM	K1150BA	CHAMPION	Y5	1

11 APPENDIX B: BILL OF MATERIALS (TQUAD/EQUAD VERSION)

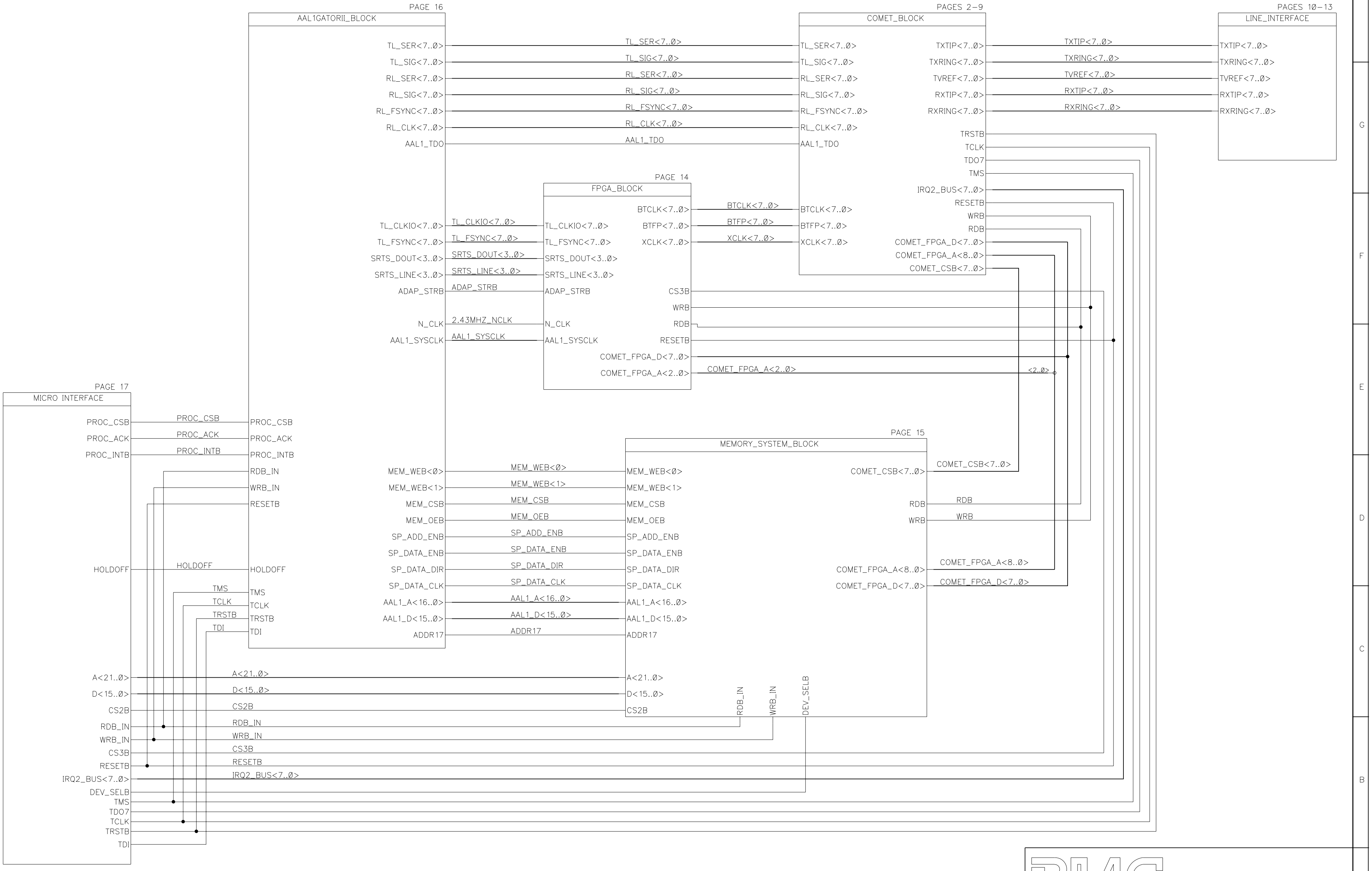
Table 20. TQUAD/EQUAD version Bill of Materials.

DESCRIPTION	PART NUMBER	MANUFACTURER	REF DES	QTY
CAPACITOR, 0.01UF, 50V, X7R_805	ECU-V1H102KBN	PANASONIC	C1, C6-C10, C15-C22, C31-C33, C37-C40, C54-C58	26
CAPACITOR, 0.1UF, 50V, X7R_805	ECJ-2VB1E104K	PANASONIC	C2-C5, C11-C14, C23-C30, C34, C43-C53	28
CAPACITOR, 1.0UF, 16V, Y5V_805	ECJ-2VF1C105Z	PANASONIC	C42	1
CAPACITOR, 47UF, 10V, TANT TEH	ECS-H1AD476R	PANASONIC	C41	1
LED SUPERGREEN, SURFACE MOUNT	7002X5	INDUSTRIAL DEVICES	D1	1
SMA-BASE	901-144	AMPHENOL	J1	1
HEADER2 100 MIL-BASE	640452-2	AMP	JP1, JP2	2
HEADER5 100 MIL-BASE	640452-5	AMP	J2	1
HEADER6 100 MIL-BASE	640452-6	AMP	J3, J7	2
HEADER10, 100 MIL BASE	1-640452-0	AMP	J4-J6	3
DIN96 MALE-BASE	650473-5	AMP	P2, P3	2
2x20 MALE, BASE	87331-4020	MOLEX	P1, P4, P5	3
RESISTOR, 1K, 5%, 805	ERJ-3GSY1K0	PANASONIC	R1	1
RESISTOR, 270, 5%, 805	ERJ-3GSY270	PANASONIC	R18	1
RESISTOR, 10K, 5%, 805	ERJ-6GEY10K	PANASONIC	R3	1
RESISTOR, 330, 5%, 805	ERJ-6GEY330	PANASONIC	R4, R6, R11-R15	7
RESISTOR, 4.7K, 5%, 805	ERJ-6GEY4K7	PANASONIC	R2, R5, R8-R10, R16, R17, R19-R21, R25-R29	15
RESISTOR,	ERJ-6ENF49.9V	PANASONIC	R22-R24	3

DESCRIPTION	PART NUMBER	MANUFACTURER	REF DES	QTY
49.9, 1%, 805				
RESISTOR ARRAY, 10K, SMD	742163103J	CTS	RN2-RN4	3
RESISTOR ARRAY, 4K7, SMD	766161472G	CTS	RN5	1
RESISTOR ARRAY, 68, SMD	742163680J	CTS	RN1	1
SWITCH, NO PB, 6MM, VERT	EVQ-PAG04K	PANASONIC	SW1	1
IC, AAL1GATOR II QFP	PM73121-RI-P	PMC-SIERRA	U1	1
IC, TQUAD	PM4344 –R1	PMC-SIERRA	U2	1
IC, EQUAD	PM6344 –R1	PMC-SIERRA	U3	1
IC, CMOS QUAD NAND GATE, SOIC	SN74HCT08	TEXAS INSTRUMENTS	U5	1
IC, CMOS 3 to 8 DECODER, SOIC	MC74HC T138AD	MOTOROLA	U13	1
IC, FPGA, PQFP	A1425A–PQ–160C	ACTEL	U12	1
IC, CMOS HEX INVERTER, SOIC	SN74HCT 04D	TEXAS INSTRUMENTS	U4, U14, U15	3
IC, FAST 20 BIT BUFFER, SOP	IDT74FCT162827CT	IDT	U7, U10	2
IC FAST 16 BIT TRANSCEIVER, SOP	IDT74FCT162646CT	IDT	U6, U11	2
IC, POWER SUPPLY MONITOR	MAX701ESA	MAXIM	U16	1
IC 1MB SRAM, 10NS, TSOP	KM681002B-10	SAMSUNG	U8, U9	2
TTL OSC, 37.056 MHz, 32/50	K1125BA	CHAMPION	Y1	1
TTL 0SC 49.152 MHz, 32/50	K1125BA	CHAMPION	Y2	1

12 APPENDIX C: AAL1GATOR II PLUS COMET SCHEMATIC DIAGRAM

REVISIONS				
ZONE	REV	DESCRIPTION	DATE	APPR



DRAWING
TITLE=AAL_ROOT
ABBREV=AAL_ROOT
LAST_MODIFIED=Wed May 19 07:48:49 1999

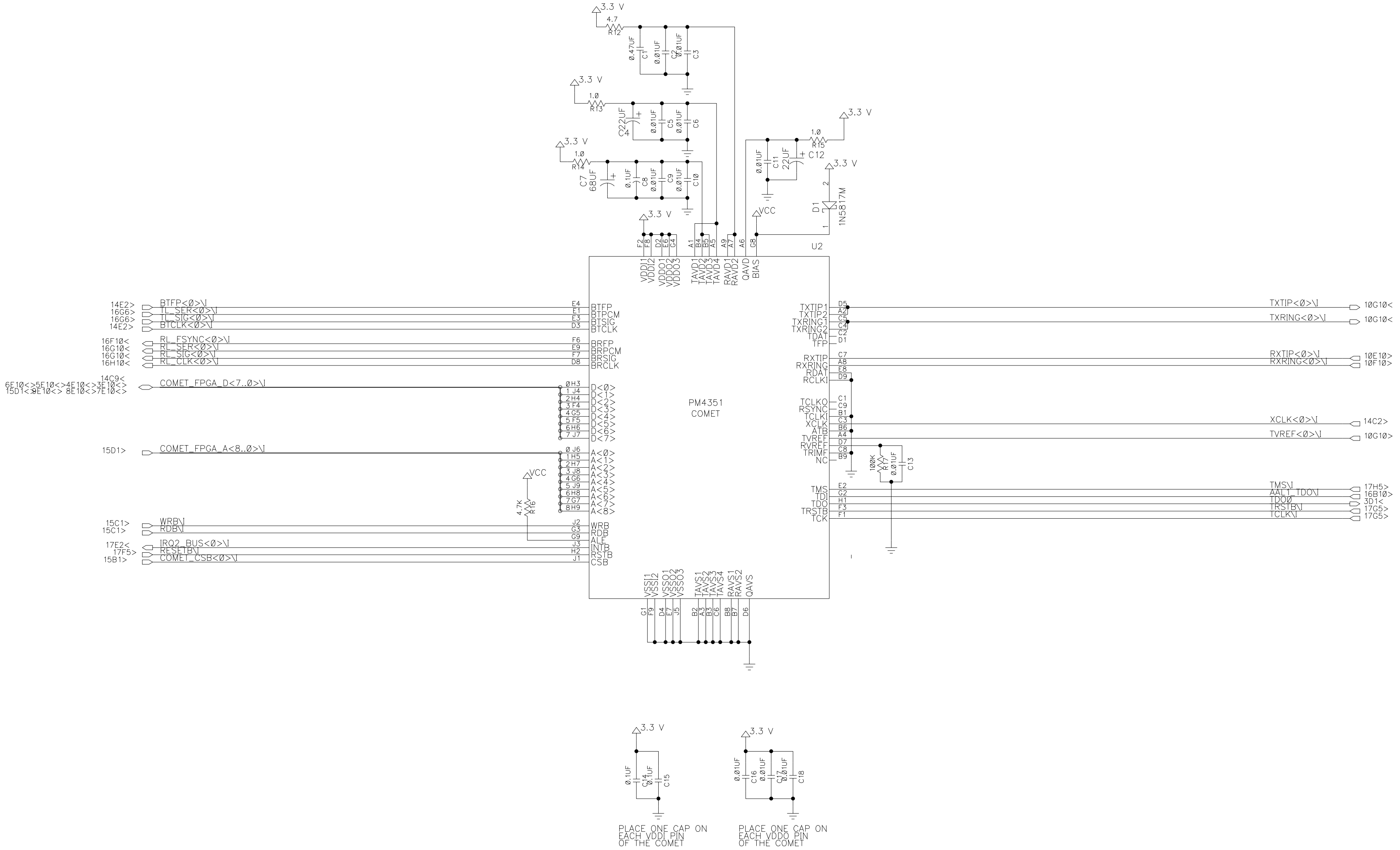
PMC-Sierra, Inc.

DOCUMENT NUMBER: PMC-99Ø2Ø6	ISSUE: 3.Ø
TITLE: AAL1GATOR II PLUS COMET REF DESIGN ROOT DRAWING	DATE: 99/Ø3/Ø1
ENGINEER: SW	PAGE: 1 OF 17

TRUE

REVISIONS

ZONE	REV	DESCRIPTION	DATE	APPR



PLACE ONE CAP ON
EACH VDDI PIN
OF THE COMET

PLACE ONE CAP ON
EACH VDDO PIN
OF THE COMET

DRAWING

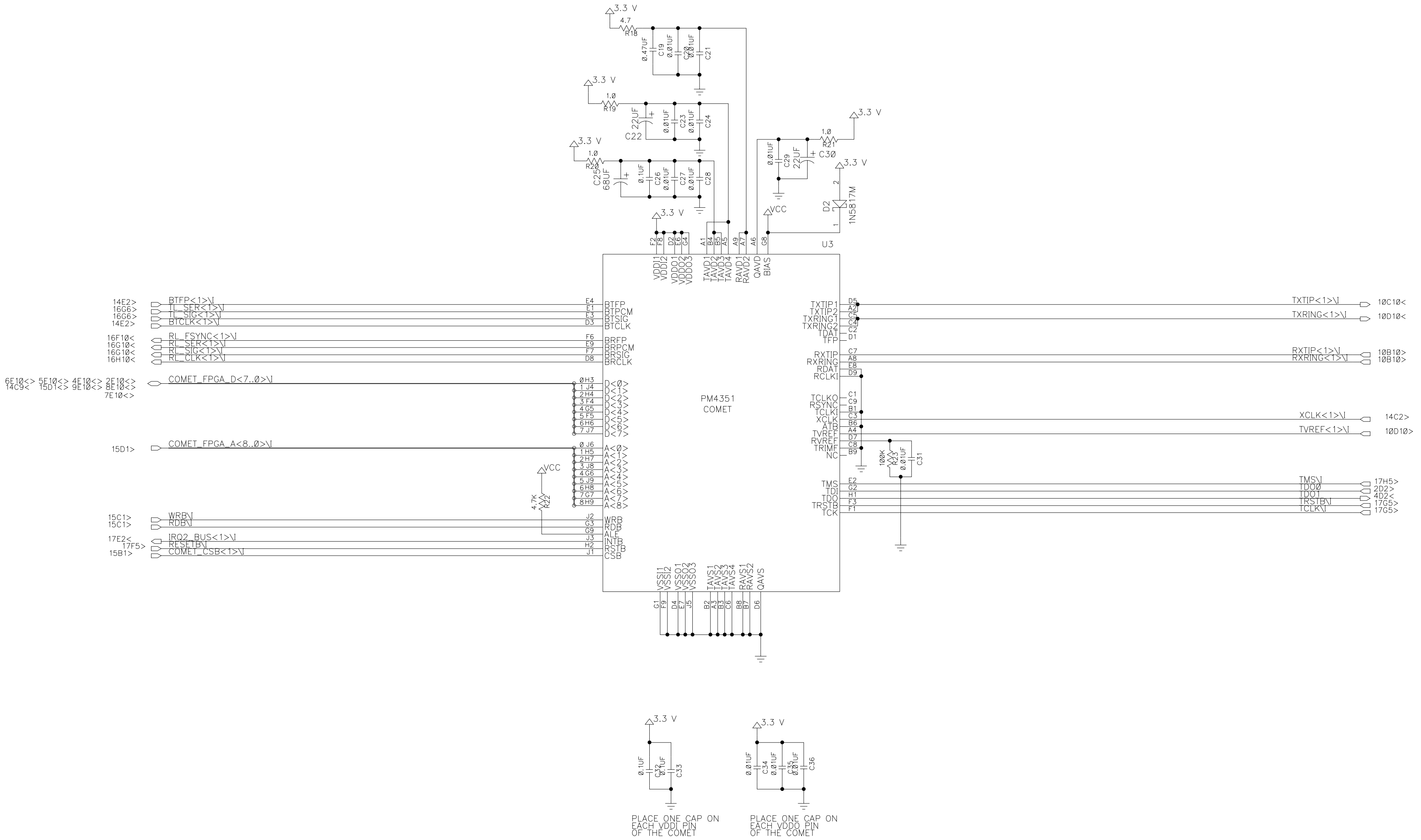
TITLE=COMET_BLOCK
ABBREV=COMET_BLOCK
LAST_MODIFIED=Wed May 19 10:37:46 1999

PMC-Sierra, Inc.

DOCUMENT NUMBER: PMC-990206	ISSUE: 3.0
TITLE: AAL1GATOR II PLUS COMET REF DESIGN COMET BLOCK 1	DATE: 99/02/25
ENGINEER: SW	PAGE: 2 OF 17

TRUE

REVISIONS				
ZONE	REV	DESCRIPTION	DATE	APPR



PLACE ONE CAP ON
EACH VDDI PIN
OF THE COMET

PLACE ONE CAP ON
EACH VDDO PIN
OF THE COMET

DRAWING

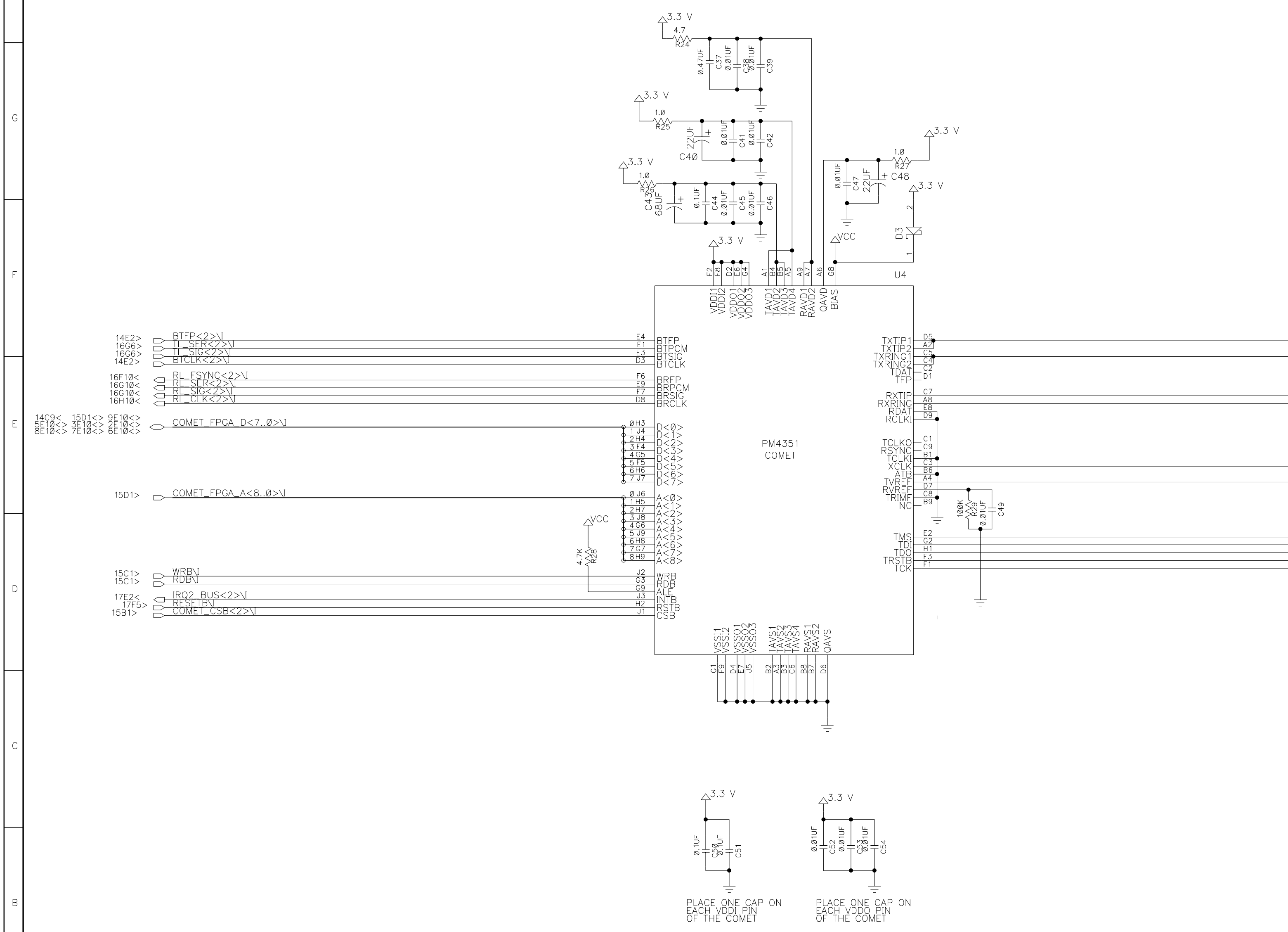
TITLE=COMET_BLOCK
ABBREV=COMET_BLOCK
LAST_MODIFIED=Wed May 19 10:37:48 1999

PMC-Sierra, Inc.

DOCUMENT NUMBER: PMC-990206	ISSUE: 3.0
TITLE: AAL1GATOR II PLUS COMET REF DESIGN COMET BLOCK 2	DATE: 99/02/06
ENGINEER: SW	PAGE: 3 OF 17

REVISIONS

ZONE	REV	DESCRIPTION	DATE	APPR



PLACE ONE CAP ON
EACH VDDI PIN
OF THE COMET

PLACE ONE CAP ON
EACH VDDO PIN
OF THE COMET

DRAWING

TITLE=COMET_BLOCK
ABBREV=COMET_BLOCK
LAST_MODIFIED=Wed May 19 10:37:50 1999

PMC-Sierra, Inc.

DOCUMENT NUMBER: PMC-990206
ISSUE: 3.0

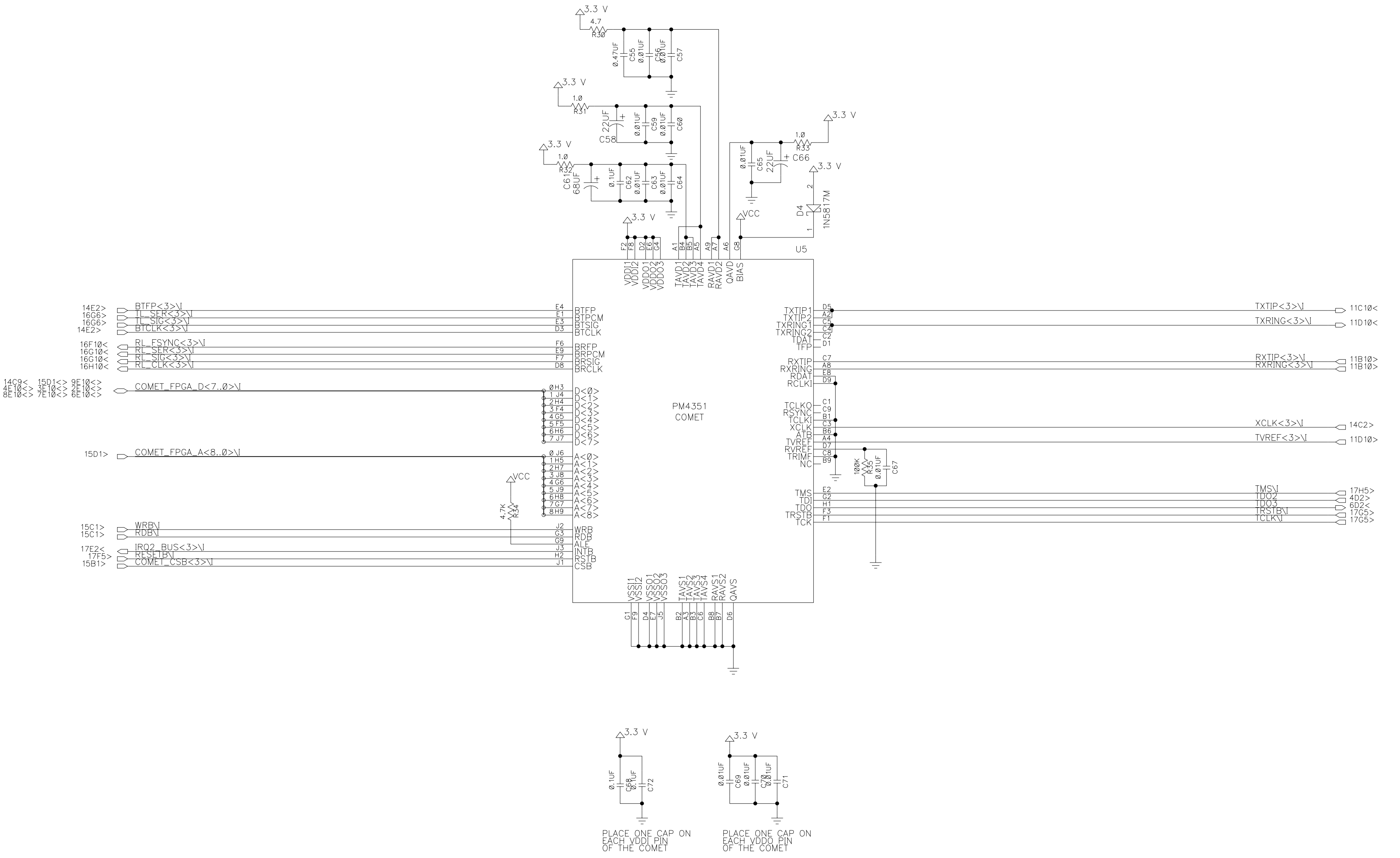
TITLE: AAL1GATOR II PLUS COMET REF DESIGN
COMET BLOCK 3

DATE: 99/02/25

ENGINEER: SW

PAGE: 4 OF 17

REVISIONS				
ZONE	REV	DESCRIPTION	DATE	APPR



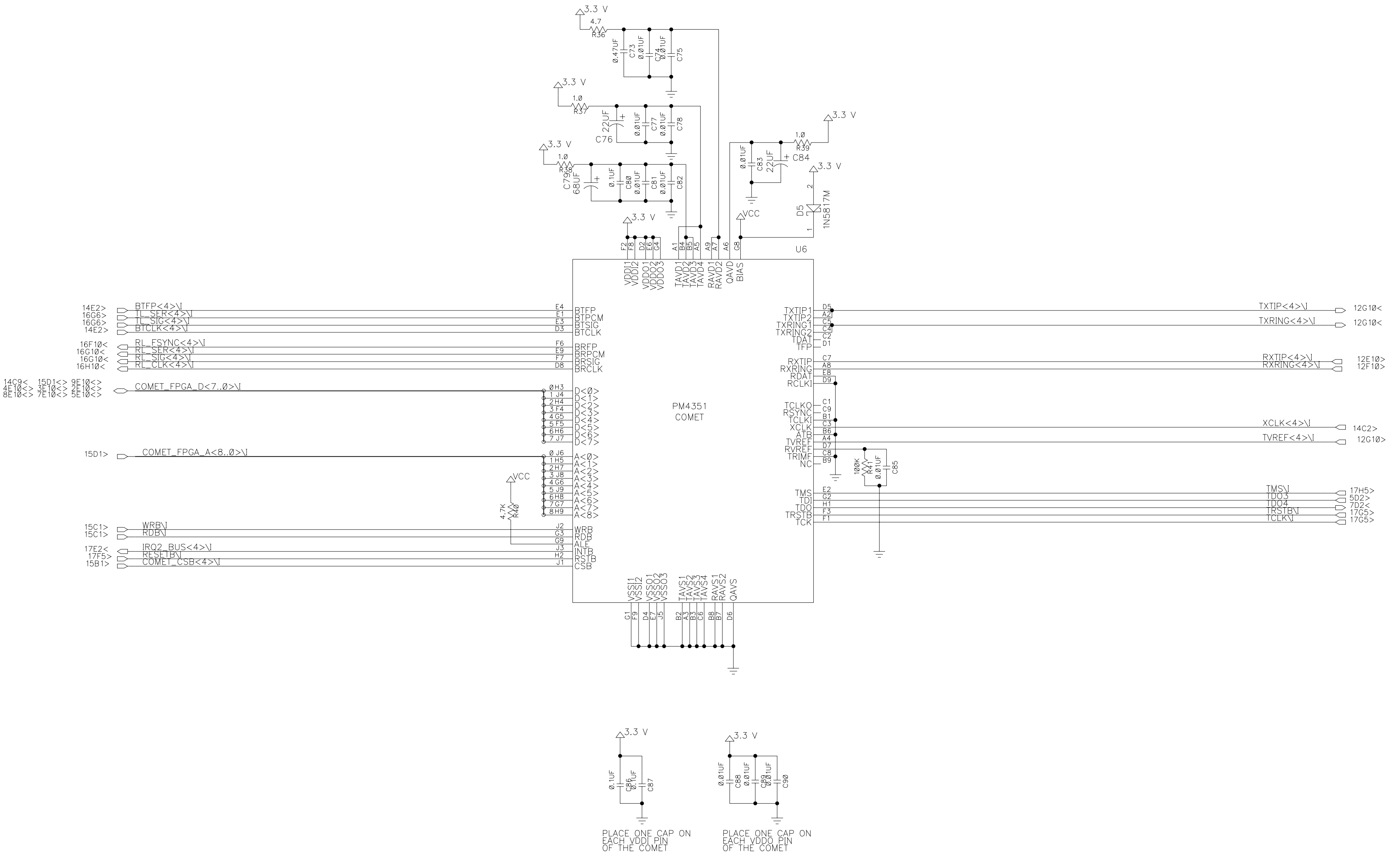
DRAWING

TITLE=COMET_BLOCK
ABBREV=COMET_BLOCK
LAST_MODIFIED=Wed May 19 10:37:53 1999

PMC-Sierra, Inc.

DOCUMENT NUMBER: PMC-990206	ISSUE: 3.0
TITLE: AAL1GATOR II PLUS COMET REF DESIGN COMET BLOCK 4	DATE: 99/02/25
ENGINEER: SW	PAGE: 5 OF 17

REVISIONS				
ZONE	REV	DESCRIPTION	DATE	APPR



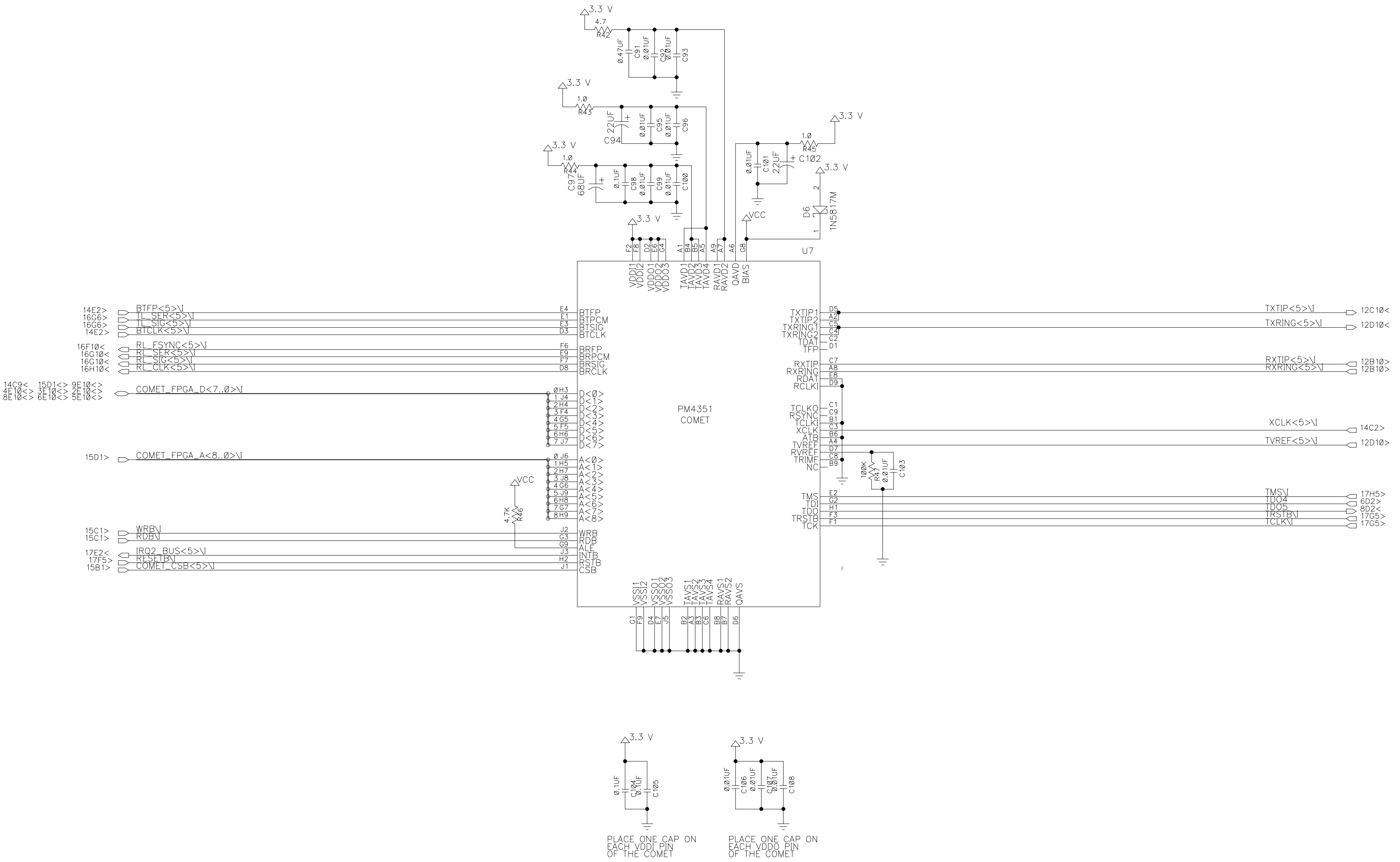
DRAWING

TITLE=COMET_BLOCK
ABBREV=COMET_BLOCK
LAST_MODIFIED=Wed May 19 10:37:57 1999

PMC PMC-Sierra, Inc.

DOCUMENT NUMBER: PMC-990206	ISSUE: 3.0
TITLE: AAL1GATOR II PLUS COMET REF DESIGN COMET BLOCK 5	DATE: 99/02/25
ENGINEER: SW	PAGE: 6 OF 17

REVISIONS				
ZONE	REV	DESCRIPTION	DATE	APPR



PLACE ONE CAP ON
EACH VDDI PIN
OF THE COMET

PLACE ONE CAP ON
EACH VDDO PIN
OF THE COMET

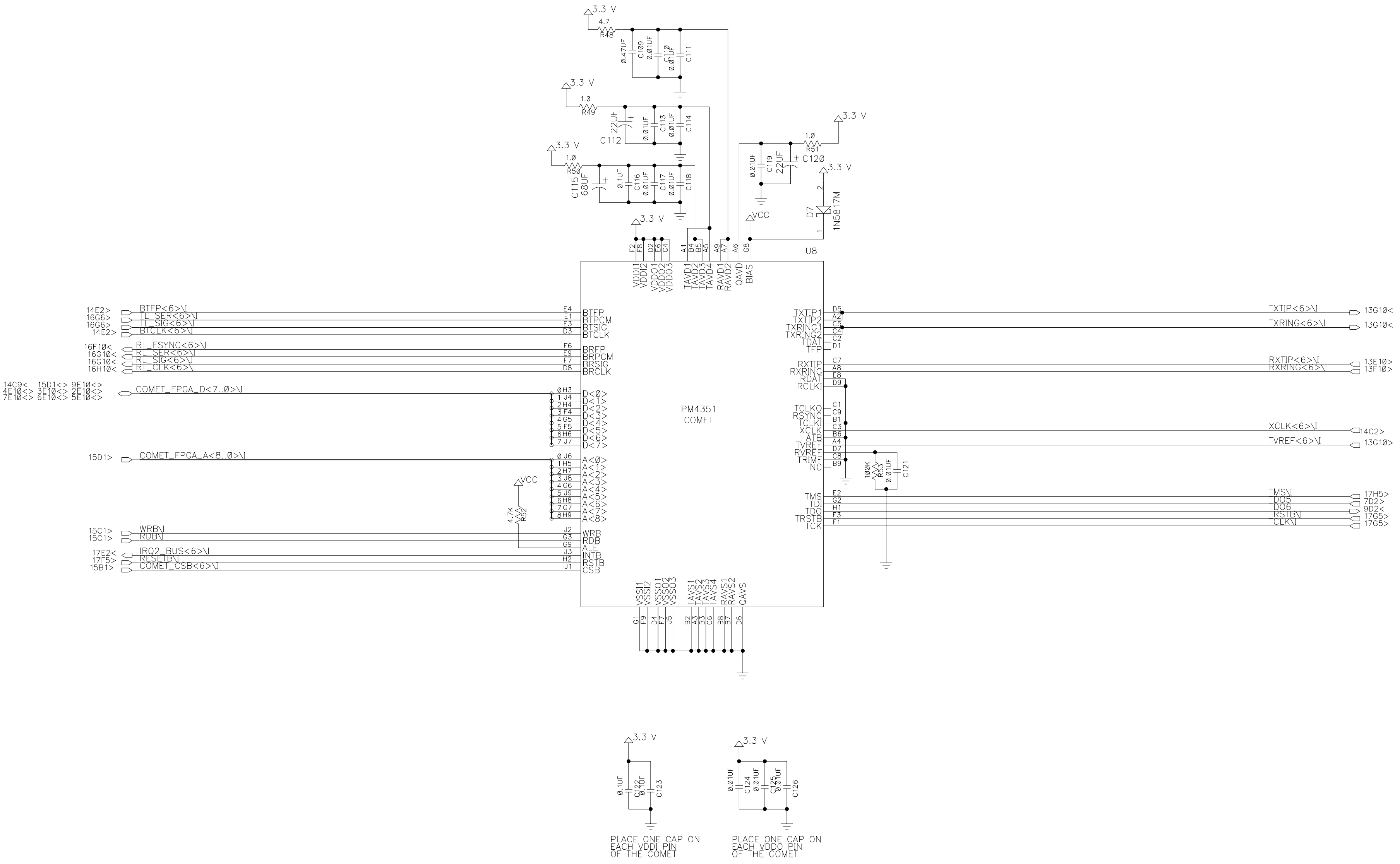
DRAWING

TITLE=COMET_BLOCK
ABBREV=COMET_BLOCK
LAST_MODIFIED=Wed May 19 10:38:01 1999

PMC-Sierra, Inc.

DOCUMENT NUMBER: PMC-990206	ISSUE: 3.0
TITLE: AAL1GATOR II PLUS COMET REF DESIGN COMET BLOCK 6	DATE: 99/02/25
ENGINEER: SW	PAGE: 7 OF 17

REVISIONS				
ZONE	REV	DESCRIPTION	DATE	APPR



PLACE ONE CAP ON
EACH VDDI PIN
OF THE COMET

PLACE ONE CAP ON
EACH VDDO PIN
OF THE COMET

DRAWING

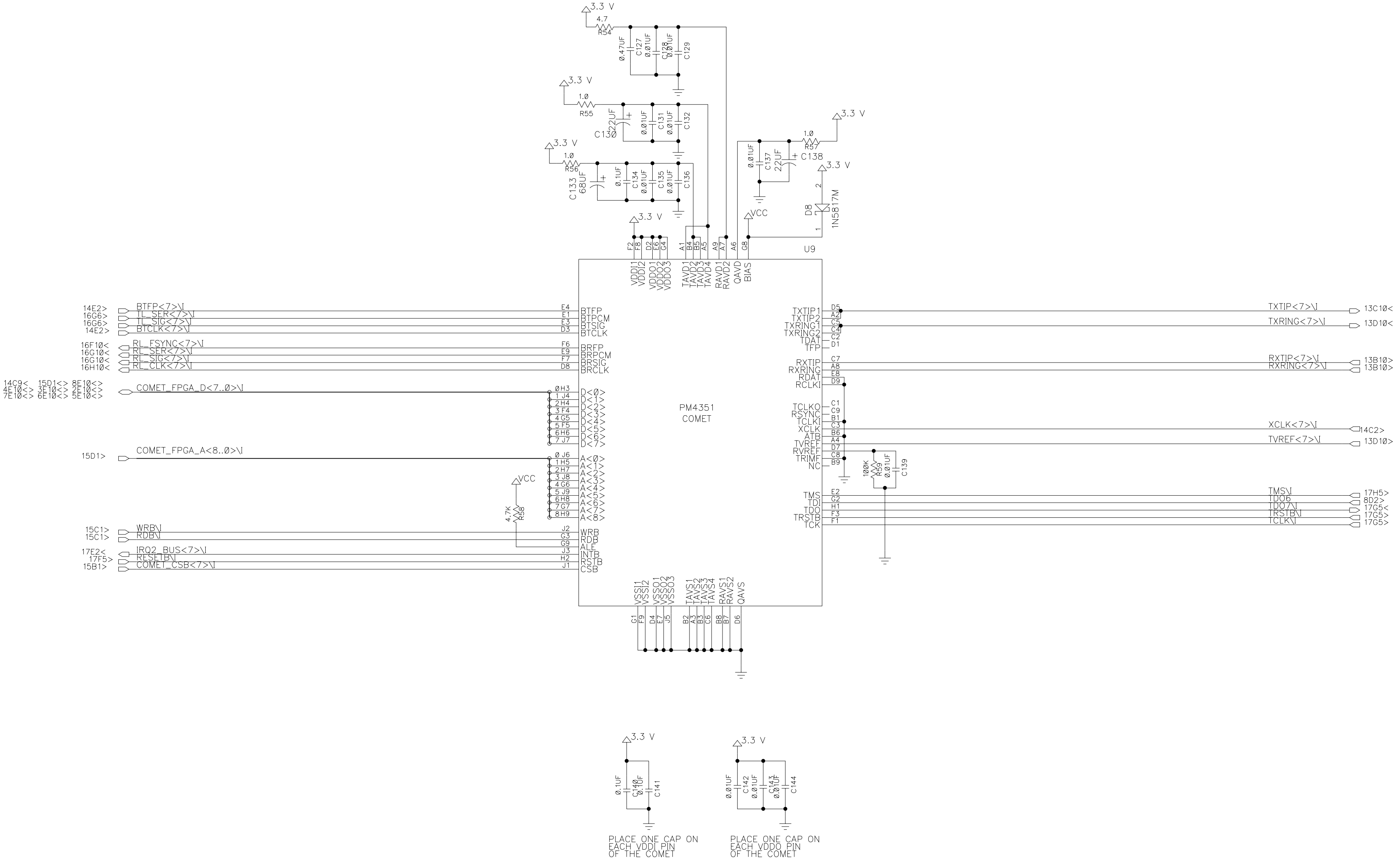
TITLE=COMET_BLOCK
ABBREV=COMET_BLOCK
LAST_MODIFIED=Wed May 19 10:38:03 1999

PMC-Sierra, Inc.

DOCUMENT NUMBER: PMC-990206	ISSUE: 3.0
TITLE: AAL1GATOR II PLUS COMET REF DESIGN COMET BLOCK 7	DATE: 99/02/25
ENGINEER: SW	PAGE: 8 OF 17

REVISIONS

ZONE	REV	DESCRIPTION	DATE	APPR



DRAWING

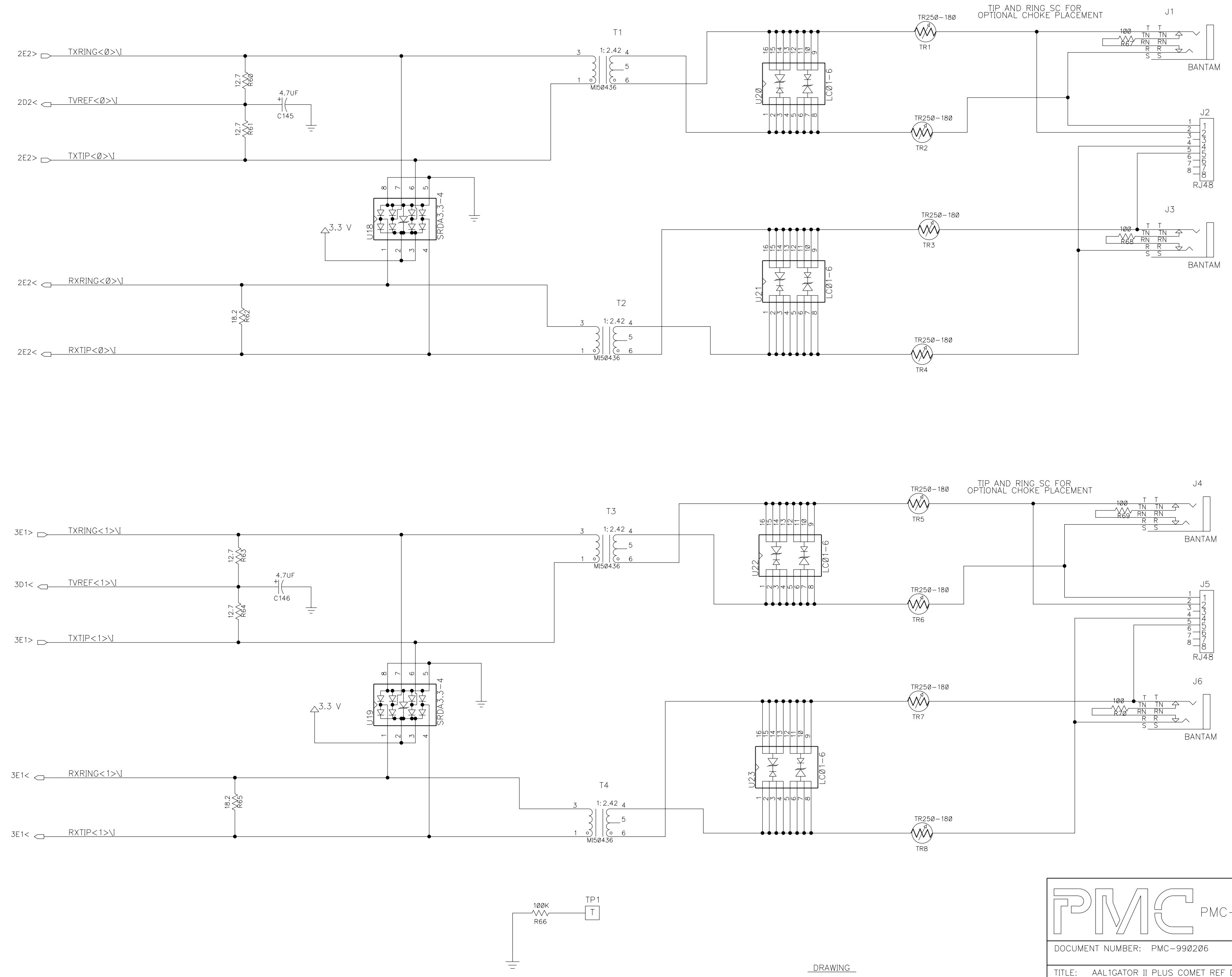
TITLE=COMET_BLOCK
ABBREV=COMET_BLOCK
LAST_MODIFIED=Wed May 19 10:38:05 1999

PMC-Sierra, Inc.

DOCUMENT NUMBER: PMC-990206	ISSUE: 3.0
TITLE: AAL1GATOR II PLUS COMET REF DESIGN COMET BLOCK 8	DATE: 99/02/25
ENGINEER: SW	PAGE: 9 OF 17

REVISIONS

ZONE	REV	DESCRIPTION	DATE	APPR



DRAWING
TITLE=LINE_INTERFACE
ABBREV=LINE
LAST_MODIFIED=Wed May 19 10:37:35 1999

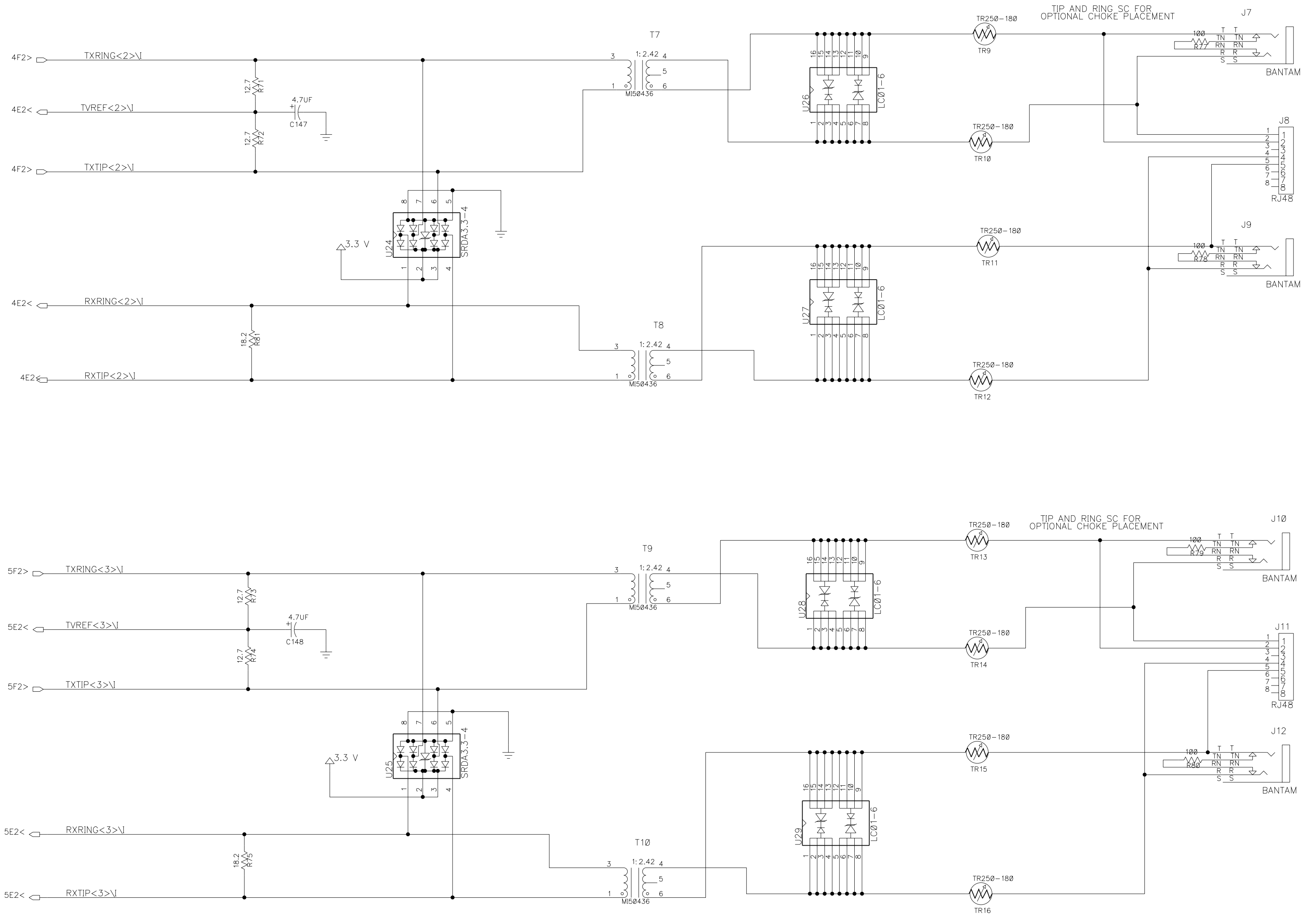
PMC PMC-Sierra, Inc.

DOCUMENT NUMBER: PMC-990206	ISSUE: 3.0
TITLE: AAL1GATOR II PLUS COMET REF DESIGN LINE INTERFACE 1	DATE: 99/02/25
ENGINEER: SW	PAGE: 10 OF 17

TRUE

REVISIONS

ZONE	REV	DESCRIPTION	DATE	APPR



DRAWING

TITLE=LINE_INTERFACE
ABBREV=LINE
LAST_MODIFIED=Wed May 19 10:37:37 1999

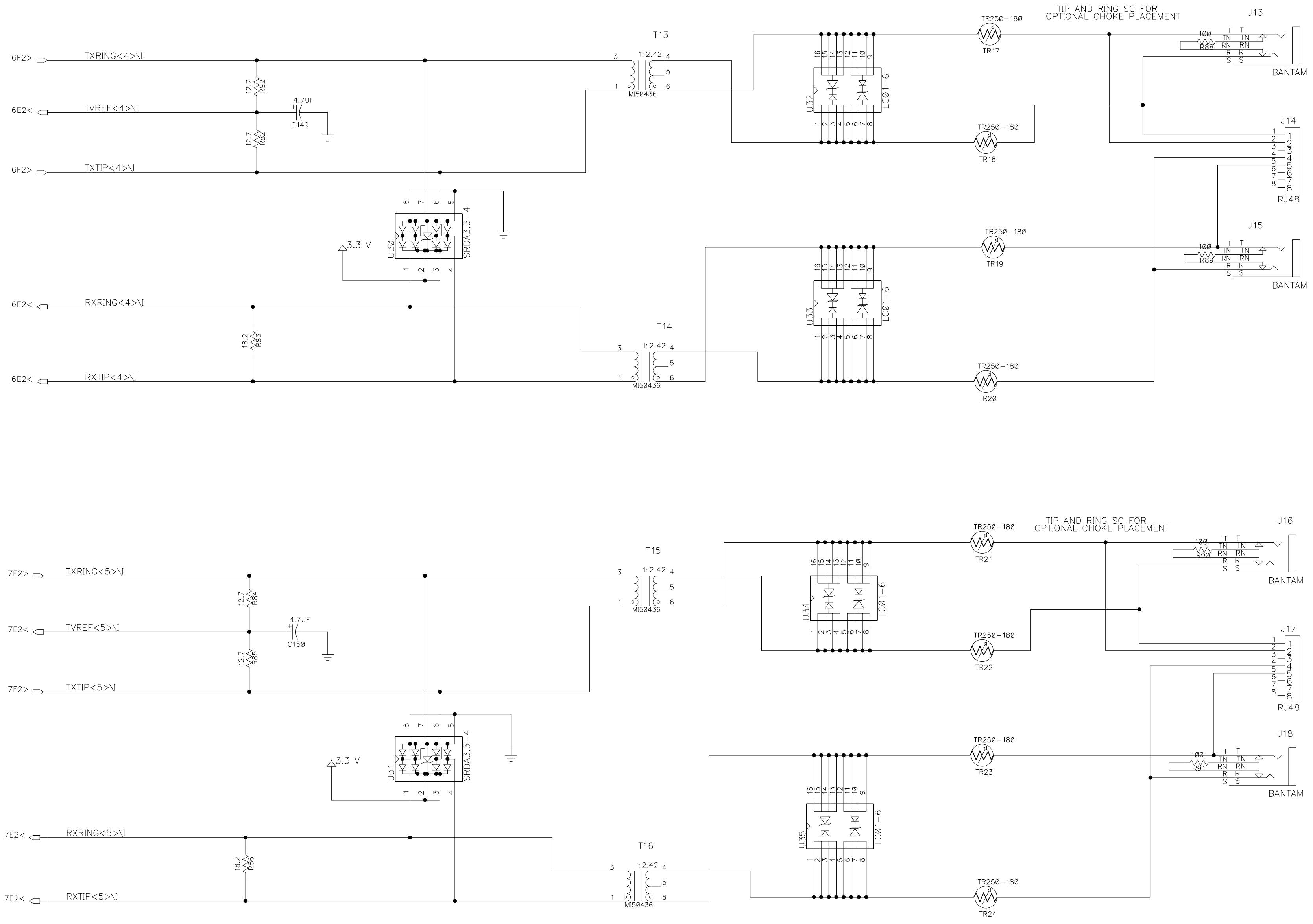
PMC-Sierra, Inc.

DOCUMENT NUMBER: PMC-990206	ISSUE: 3.0
TITLE: AAL1GATOR II PLUS COMET REF DESIGN LINE INTERFACE 2	DATE: 99/02/25
ENGINEER: SW	PAGE: 11 OF 17

TRUE

REVISIONS

ZONE	REV	DESCRIPTION	DATE	APPR



DRAWING

TITLE=LINE_INTERFACE
ABBREV=LINE
LAST_MODIFIED=Wed May 19 10:37:39 1999

PMC-Sierra, Inc.

DOCUMENT NUMBER: PMC-990206

ISSUE: 3.0

TITLE: AAL1GATOR II PLUS COMET REF DESIGN
LINE INTERFACE 3

DATE: 99/02/25

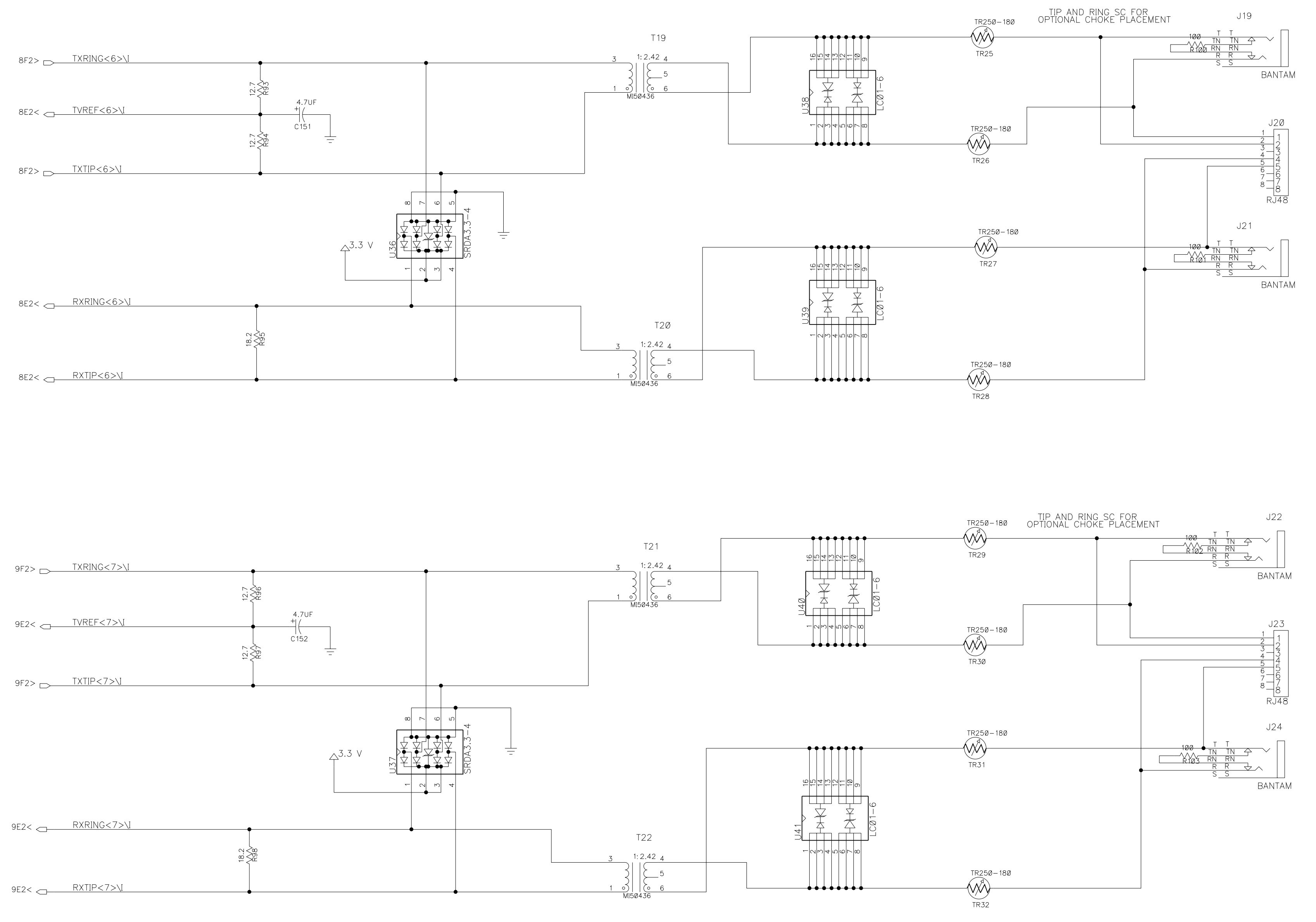
ENGINEER: SW

PAGE: 12 OF 17

TRUE

REVISIONS

ZONE	REV	DESCRIPTION	DATE	APPR



DRAWING

TITLE=LINE_INTERFACE
ABBREV=LINE
LAST_MODIFIED=Wed May 19 10:37:41 1999

PMC-Sierra, Inc.

DOCUMENT NUMBER: PMC-990206
ISSUE: 3.0

TITLE: AAL1GATOR II PLUS COMET REF DESIGN
LINE INTERFACE 4

DATE: 99/02/25

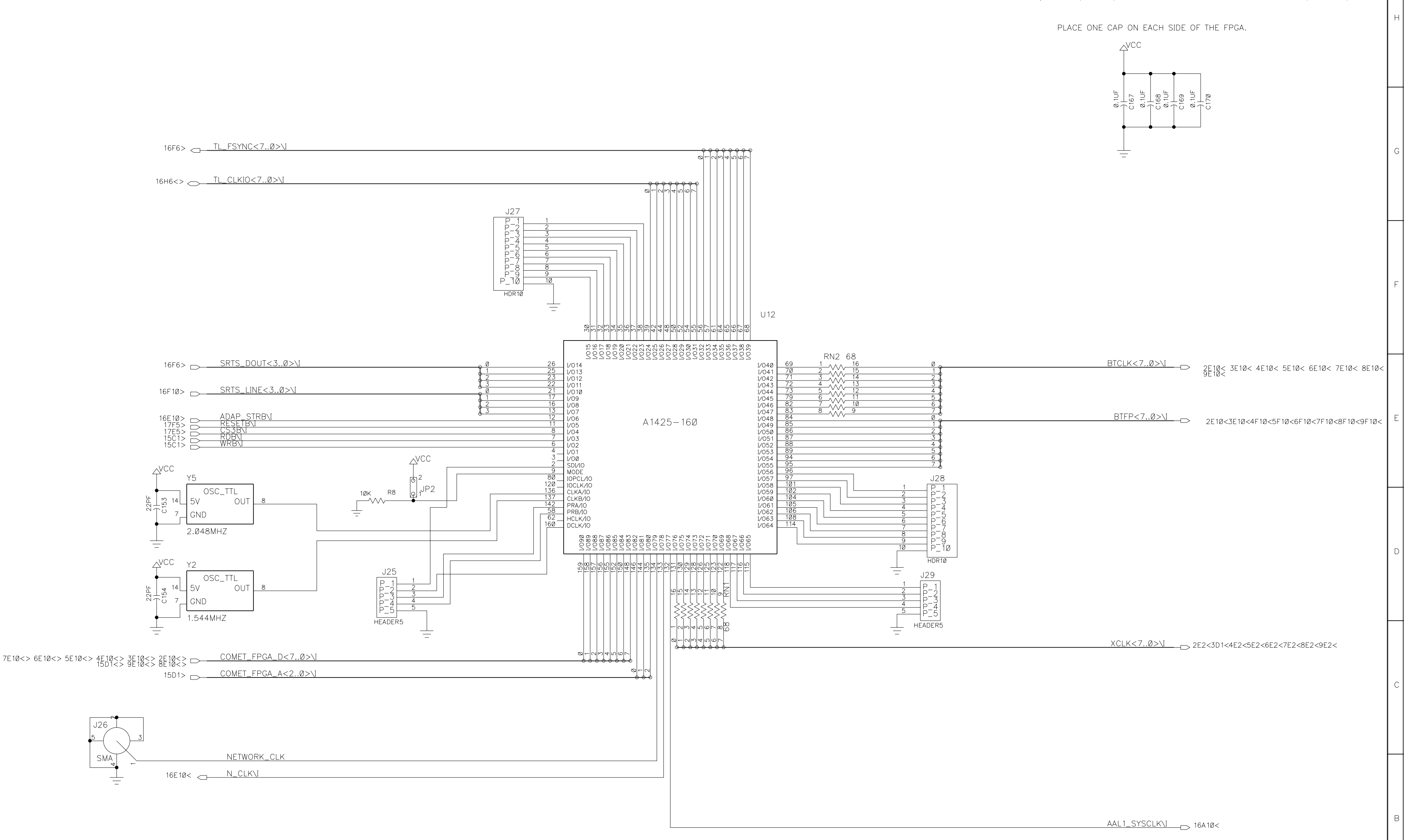
ENGINEER: SW

PAGE: 13 OF 17

TRUE

REVISIONS

ZONE	REV	DESCRIPTION	DATE	APPR



DRAWING
TITLE=FPGA_BLOCK
ABBREV=FPGA_BLOCK
LAST_MODIFIED=Wed May 19 10:38:10 1999

PMC-Sierra, Inc.

DOCUMENT NUMBER: 990206
ISSUE: 3.0

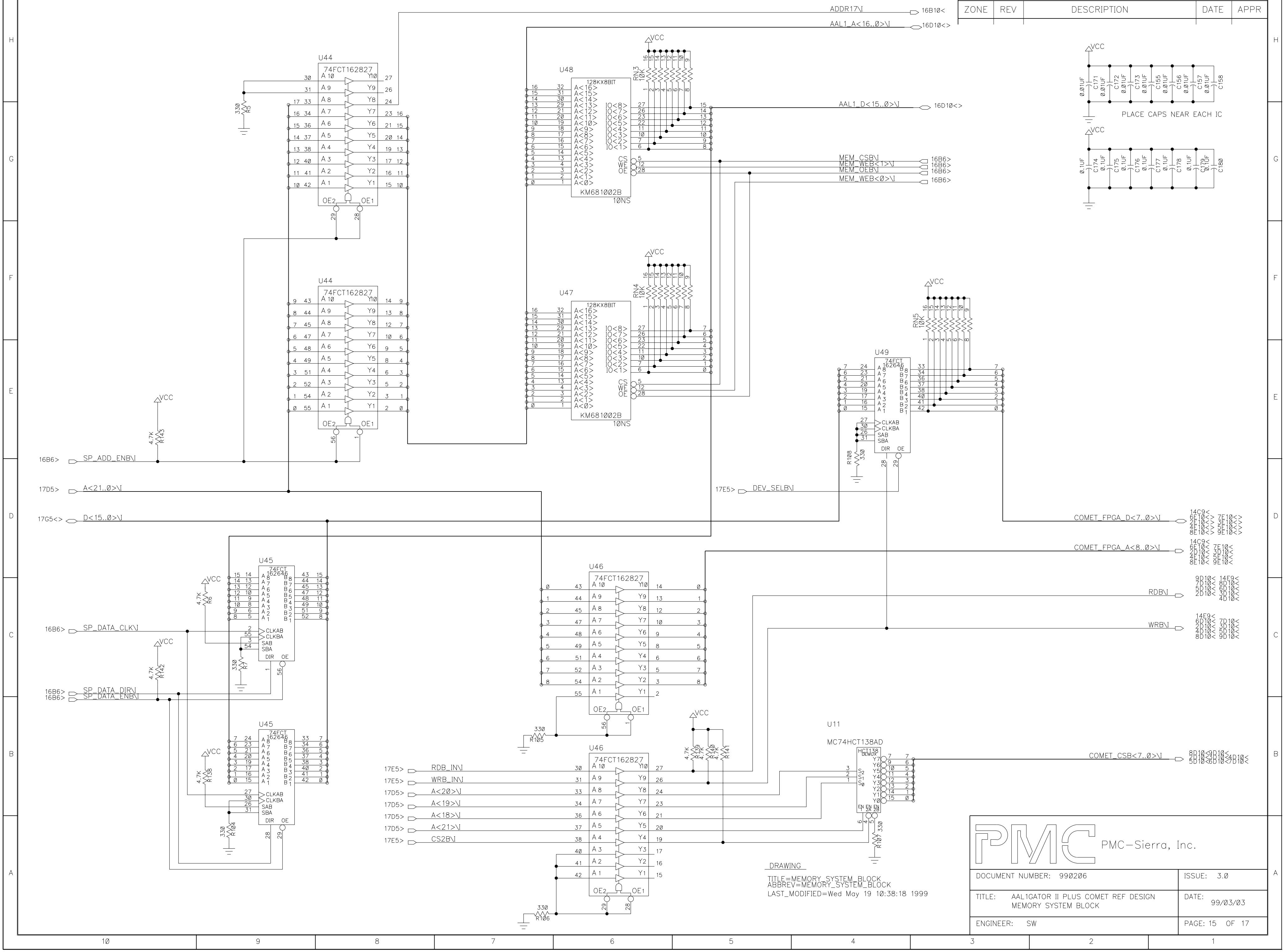
TITLE: AAL1GATOR II PLUS COMET REF DESIGN
FPGA BLOCK

DATE: 99/02/24

ENGINEER: SW

PAGE: 14 OF 17

ZONE	REV	DESCRIPTION	DATE	APPR



DRAWING

TITLE=MEMORY_SYSTEM_BLOCK
ABBREV=MEMORY_SYSTEM_BLOCK
LAST_MODIFIED=Wed May 19 10:38:18 1999

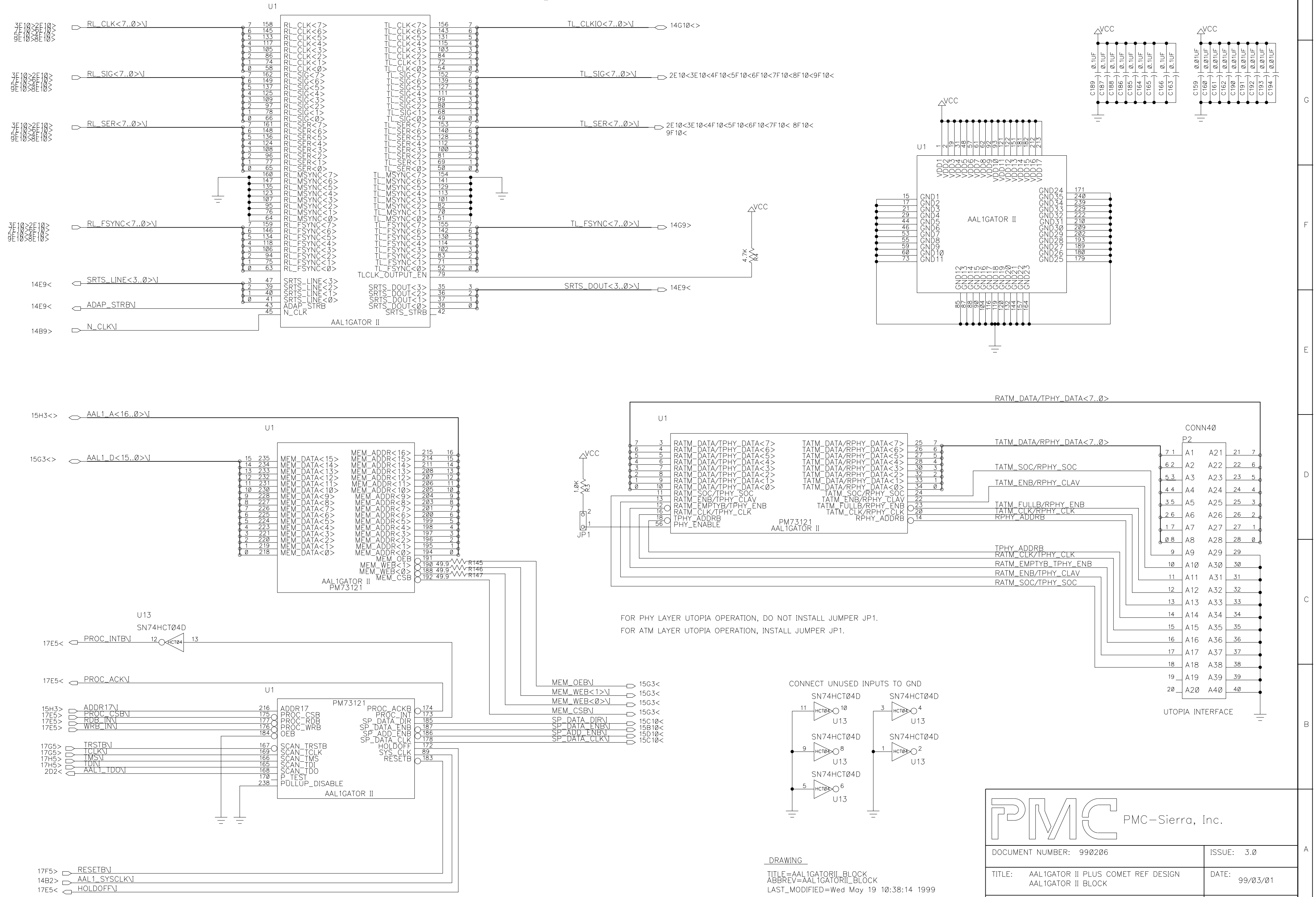
PMC-Sierra, Inc.

DOCUMENT NUMBER: 990206	ISSUE: 3.0
TITLE: AAL1GATOR II PLUS COMET REF DESIGN MEMORY SYSTEM BLOCK	DATE: 99/03/03
ENGINEER: SW	PAGE: 15 OF 17

REVISIONS				
ZONE	REV	DESCRIPTION	DATE	APPR

NOTE: MINIMIZE TL_CLK TRACE LENGTH TO FPGA.

PLACE ONE OF EACH CAP ALTERNATING BETWEEN VDD PINS.



FOR PHY LAYER UTOPIA OPERATION, DO NOT INSTALL JUMPER JP1.
FOR ATM LAYER UTOPIA OPERATION, INSTALL JUMPER JP1.

CONNECT UNUSED INPUTS TO GND

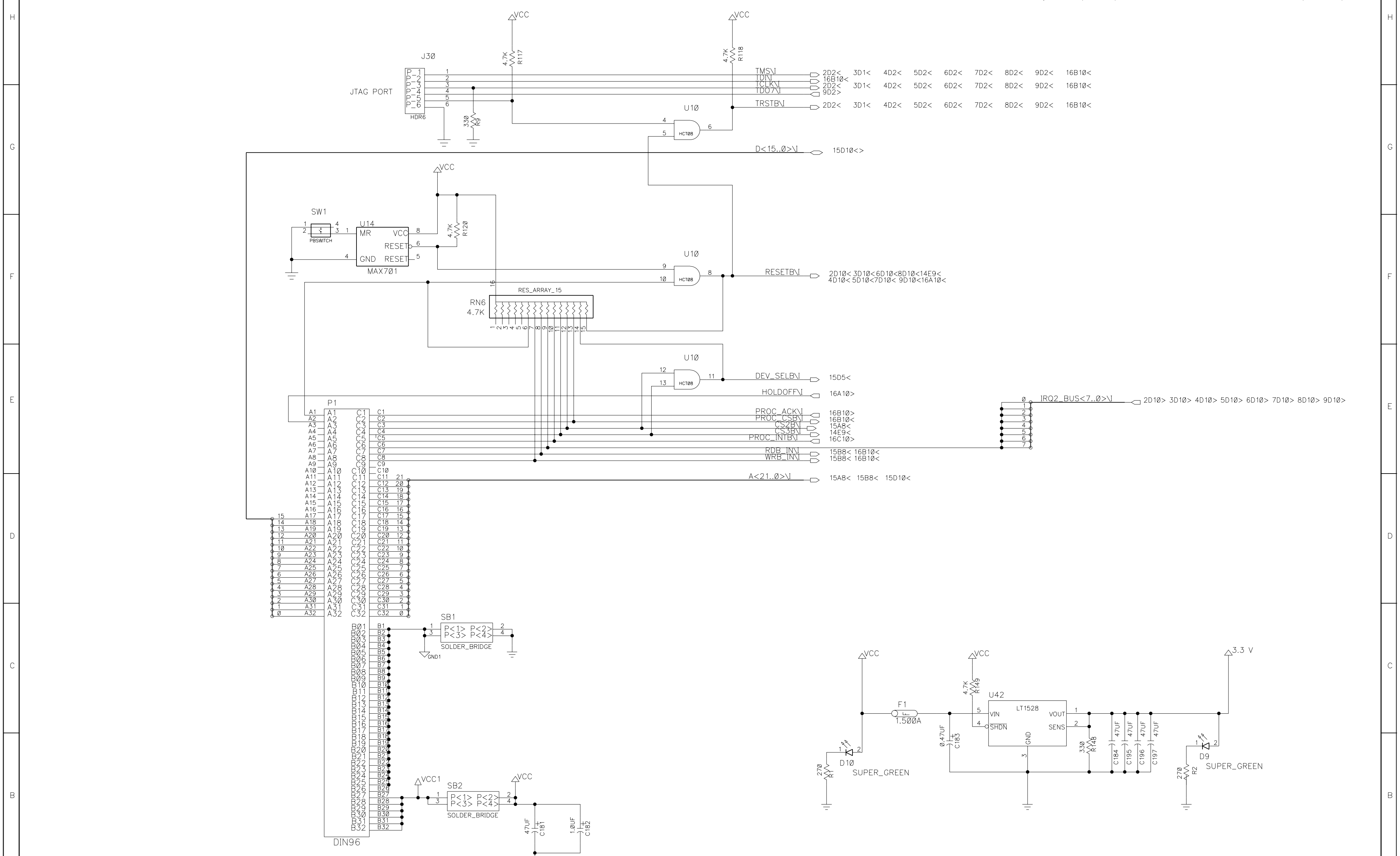
DRAWING

TITLE=AAL1GATORII_BLOCK
ABBREV=AAL1GATORII_BLOCK
LAST_MODIFIED=Wed May 19 10:38:14 1999

PMC-Sierra, Inc.

DOCUMENT NUMBER: 990206	ISSUE: 3.0
TITLE: AAL1GATOR II PLUS COMET REF DESIGN AAL1GATOR II BLOCK	DATE: 99/03/01
ENGINEER: SW	PAGE: 16 OF 17

REVISIONS				
ZONE	REV	DESCRIPTION	DATE	APPR



CONNECT UNUSED INPUTS TO GND
U10
HCT08

DRAWING

TITLE=MICRO_INTERFACE
ABBREV=MICRO
LAST_MODIFIED=Wed May 19 10:37:44 1999

PMC-Sierra, Inc.

DOCUMENT NUMBER: PMC-990206

ISSUE: 3.0

TITLE: AAL1GATOR II PLUS COMET REF DESIGN
MICRO_INTERFACE

DATE: 99/03/01

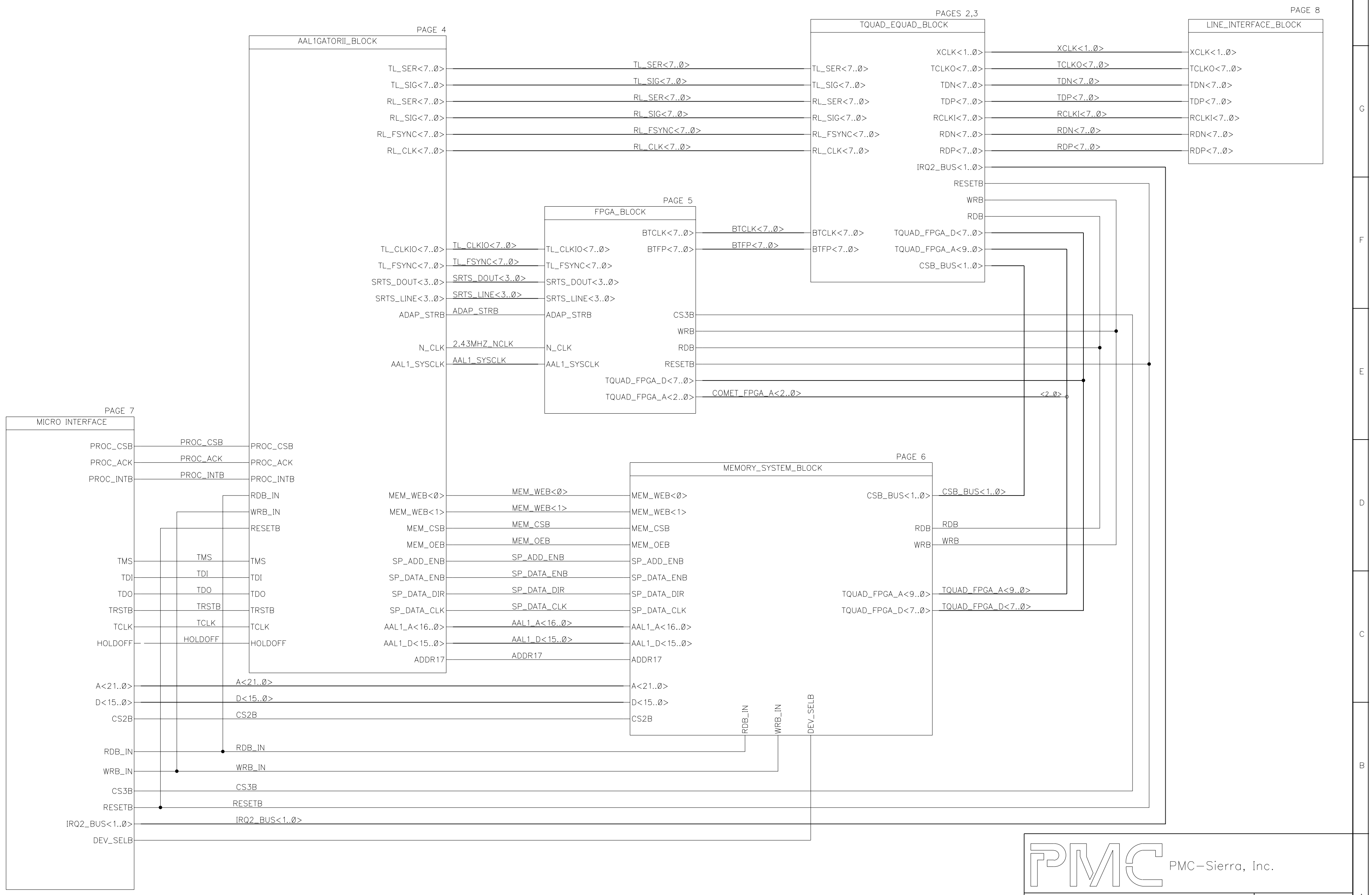
ENGINEER: SW

PAGE: 17 OF 17

TRUE

13 APPENDIX D: SCHEMATIC DIAGRAM (TQUAD/EQUAD VERSION)

REVISIONS				
ZONE	REV	DESCRIPTION	DATE	APPR



DRAWING
TITLE=TQUAD_AAL_ROOT
ABBREV=TQUAD_AAL_ROOT
LAST_MODIFIED=Wed May 19 10:48:03 1999

PMC-Sierra, Inc.

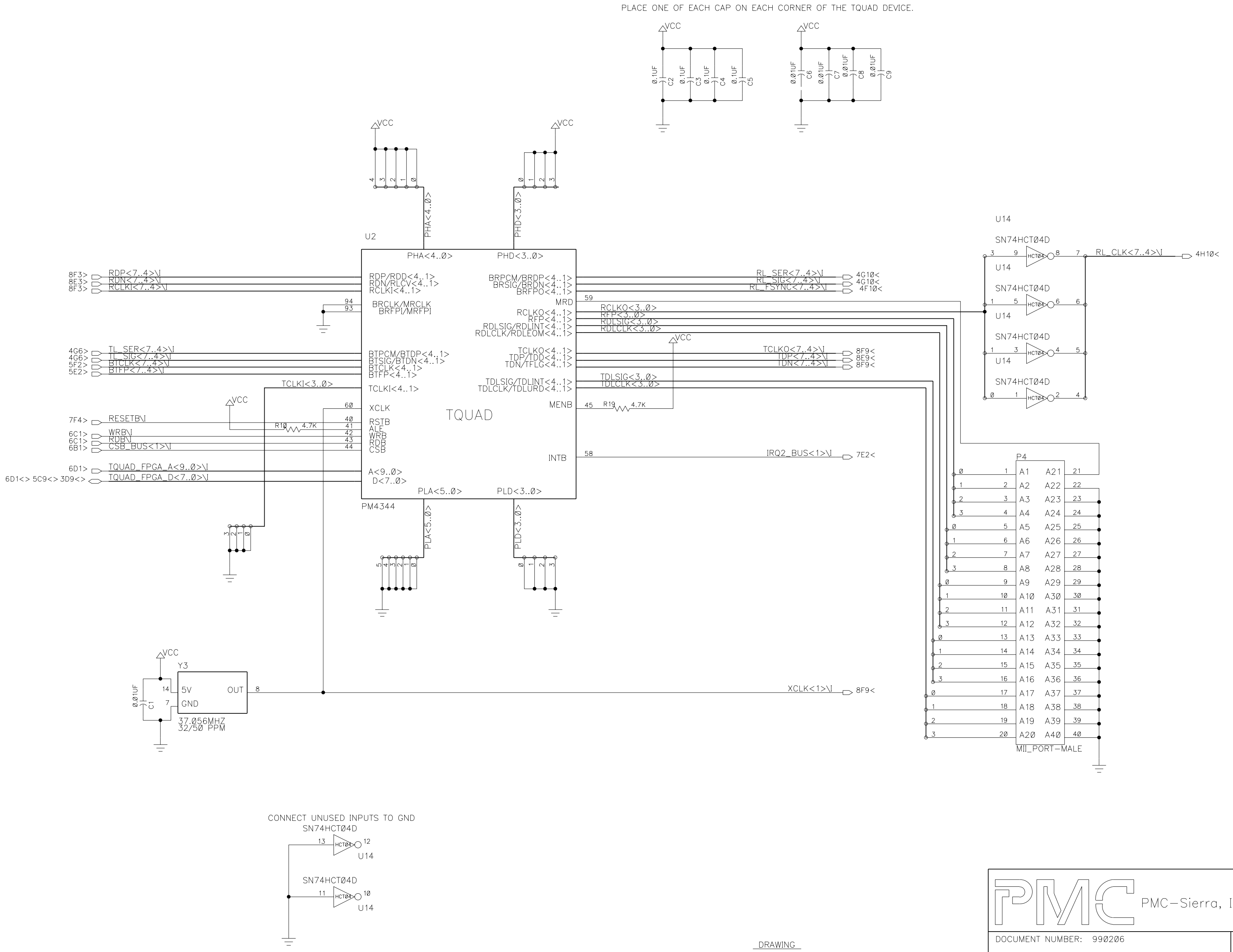
DOCUMENT NUMBER: PMC-990206	ISSUE: 3.Ø
TITLE: AAL1GATOR II PLUS TQUAD REF DESIGN ROOT DRAWING	DATE: 99/Ø3/Ø8
ENGINEER: SW	PAGE: 1 OF 8

TRUE

ZONE	REV	DESCRIPTION	DATE	APPR

REVISIONS

PLACE ONE OF EACH CAP ON EACH CORNER OF THE TQUAD DEVICE.



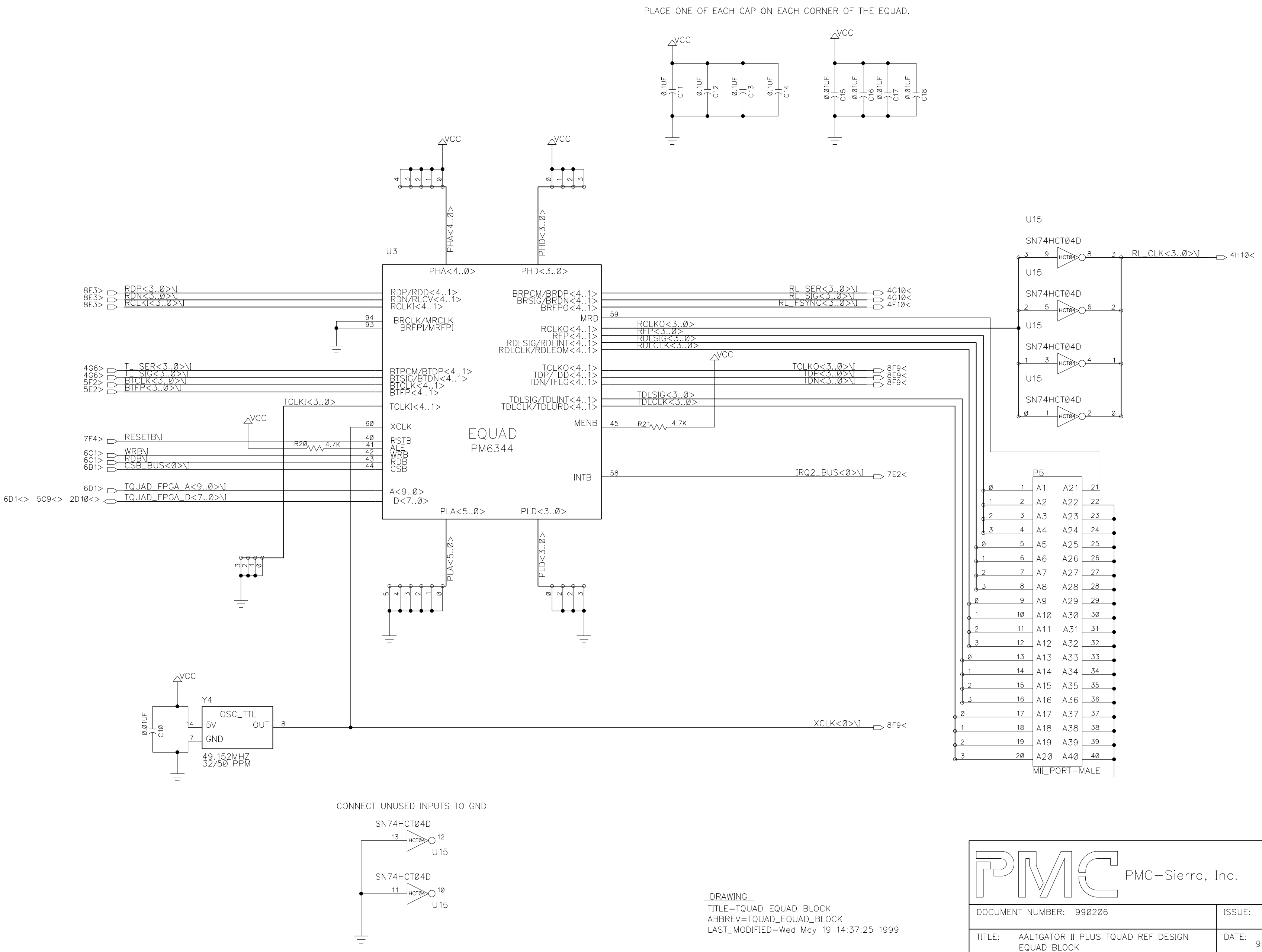
DRAWING
TITLE=TQUAD_EQUAD_BLOCK
ABBREV=TQUAD_EQUAD_BLOCK
LAST_MODIFIED=Wed May 19 14:37:23 1999

PMC-Sierra, Inc.

DOCUMENT NUMBER: 990206	ISSUE: 3.0
TITLE: AAL1GATOR II PLUS TQUAD REF DESIGN TQUAD BLOCK	DATE: 99/03/08
ENGINEER: SW	PAGE: 2 OF 8

REVISIONS

ZONE	REV	DESCRIPTION	DATE	APPR



DRAWING
TITLE=TQUAD_EQUAD_BLOCK
ABBREV=TQUAD_EQUAD_BLOCK
LAST_MODIFIED=Wed May 19 14:37:25 1999

PMC–Sierra, Inc.

DOCUMENT NUMBER: 99Ø2Ø6 | ISSUE: 3.Ø

TITLE: AAL1GATOR II PLUS TQUAD REF DESIGN
EQUAD BLOCK | DATE: 99/Ø3/Ø8

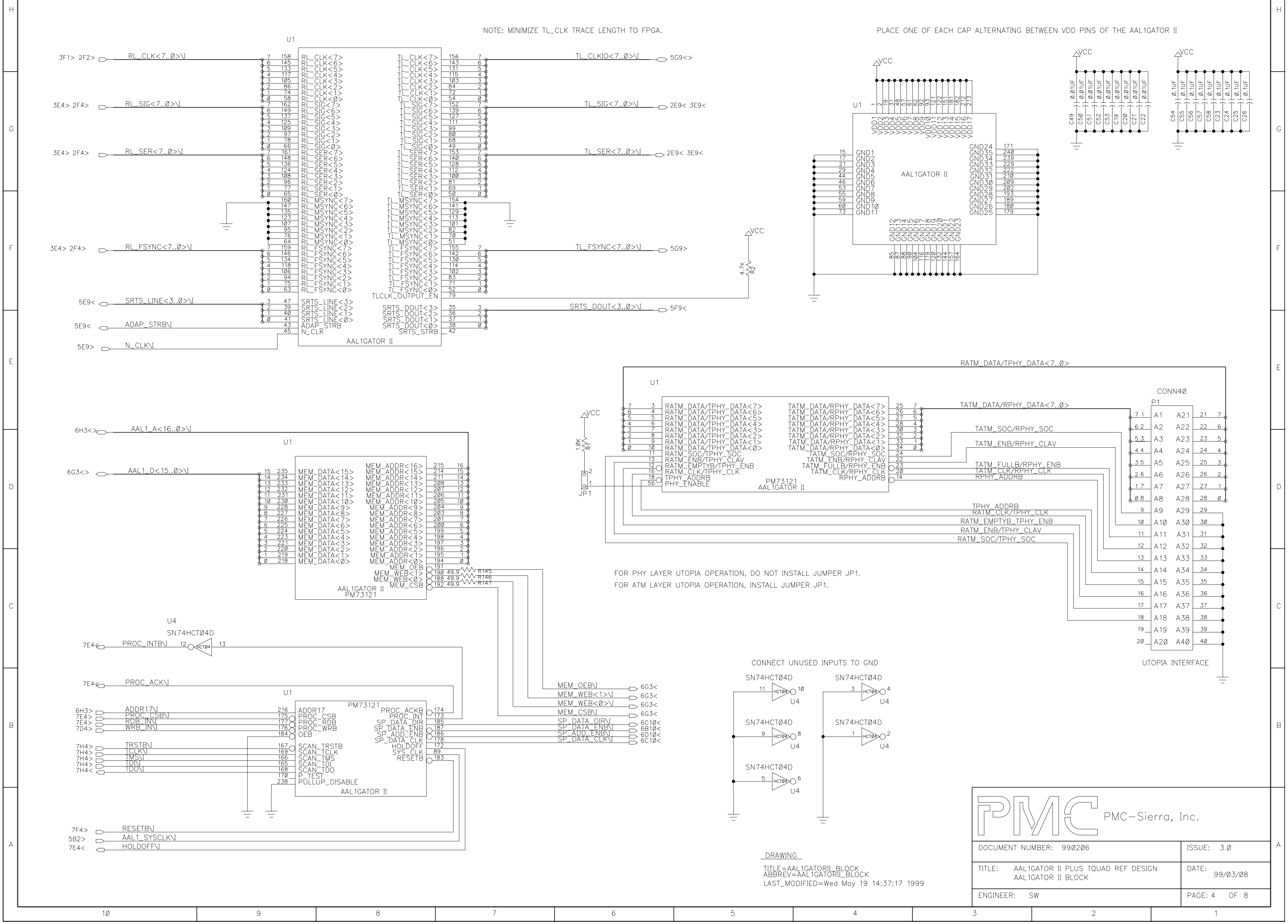
ENGINEER: SW | PAGE: 3 OF 8

ZONE	REV	DESCRIPTION	DATE	APPR

REVISIONS

NOTE: MINIMIZE TL_CLK TRACE LENGTH TO FPGA.

PLACE ONE OF EACH CAP ALTERNATING BETWEEN VDD PINS OF THE AAL1GATOR II



FOR PHY LAYER UTOPIA OPERATION, DO NOT INSTALL JUMPER JP1.
FOR ATM LAYER UTOPIA OPERATION, INSTALL JUMPER JP1.

CONNECT UNUSED INPUTS TO GND

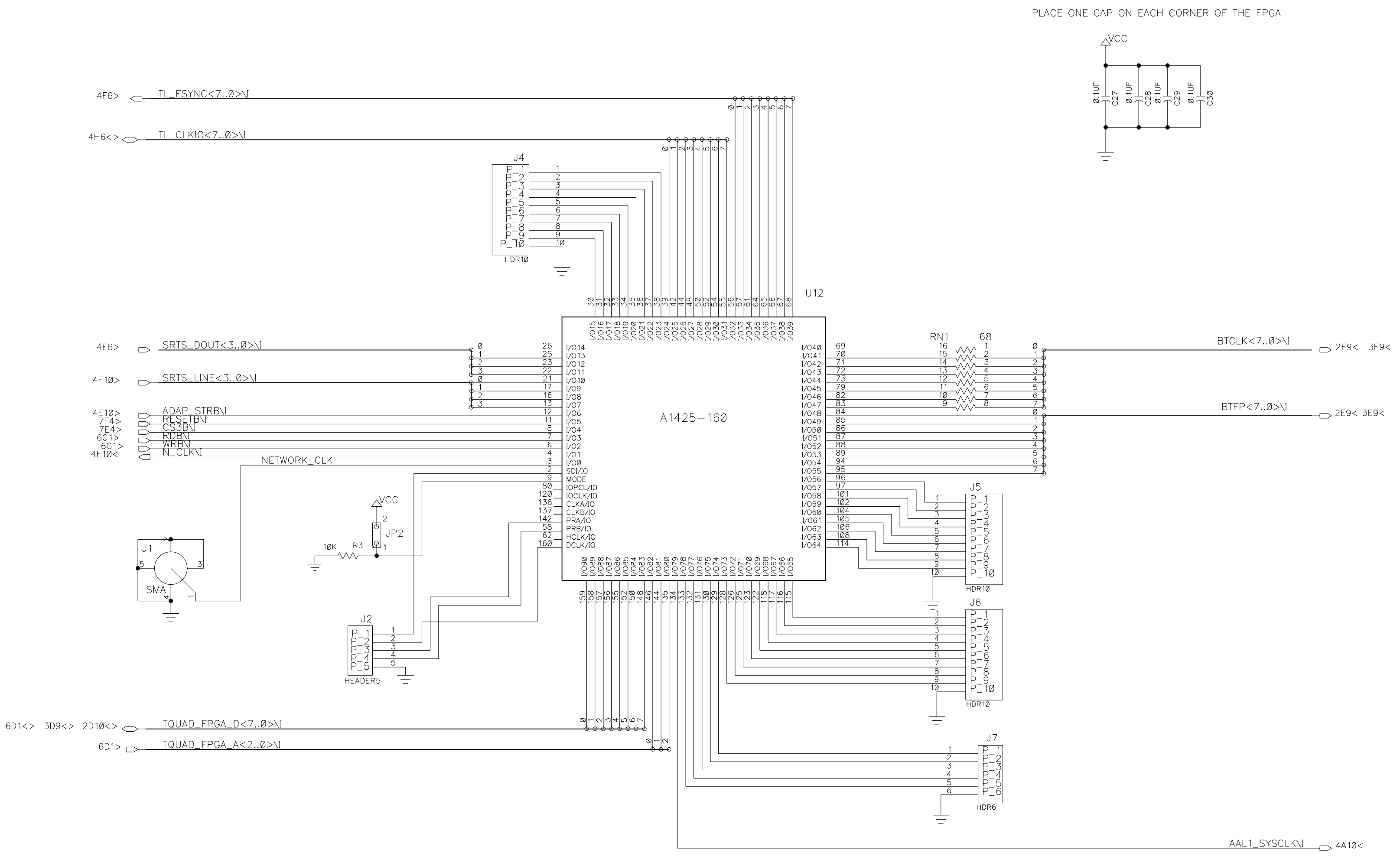
DRAWING

TITLE=AAL1GATORII_BLOCK
ABBREV=AAL1GATORII_BLOCK
LAST_MODIFIED=Wed May 19 14:37:17 1999

PMC-Sierra, Inc.

DOCUMENT NUMBER: 990206	ISSUE: 3.0
TITLE: AAL1GATOR II PLUS TQUAD REF DESIGN AAL1GATOR II BLOCK	DATE: 99/03/08
ENGINEER: SW	PAGE: 4 OF 8

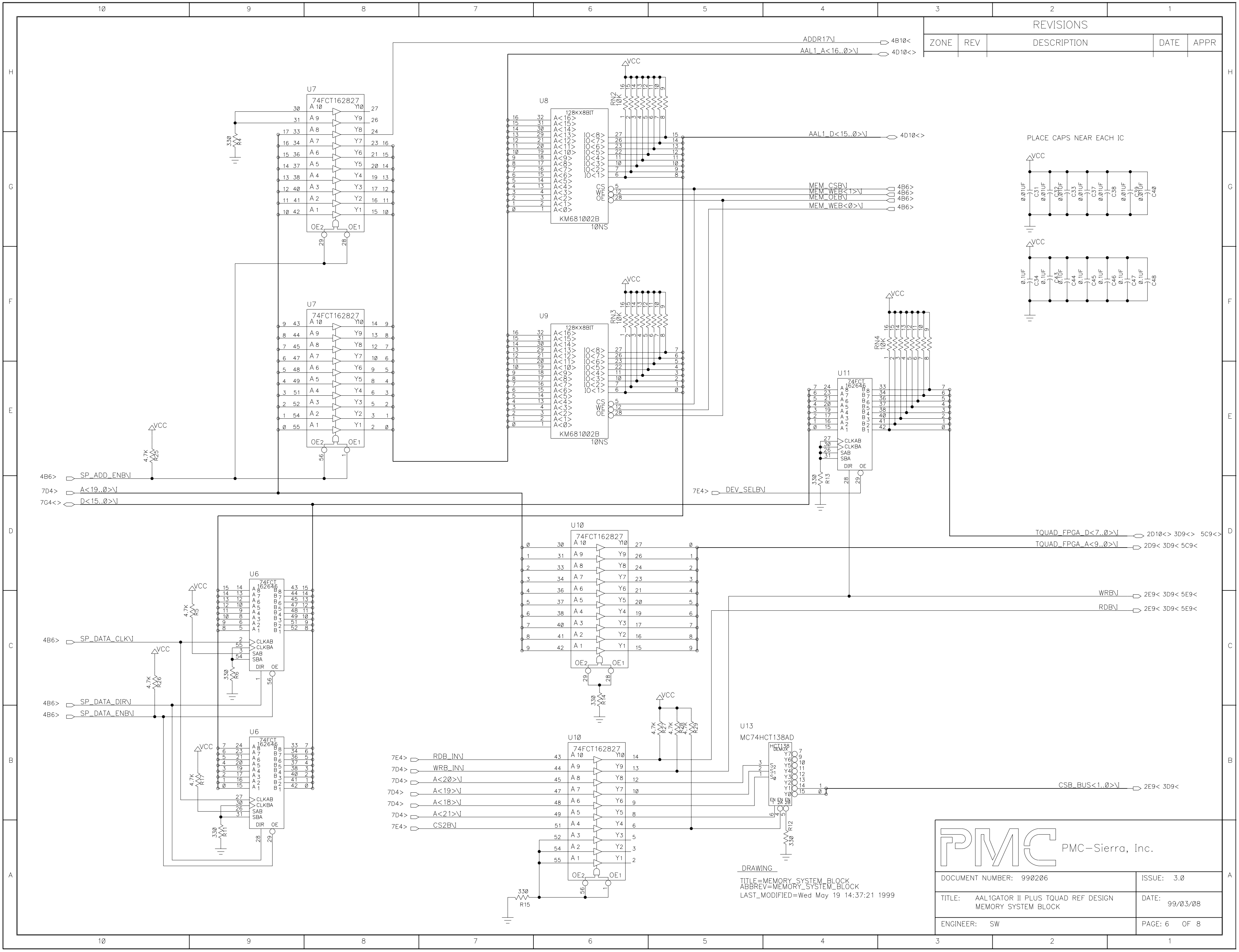
REVISIONS				
ZONE	REV	DESCRIPTION	DATE	APPR



DRAWING
TITLE=FPGA_BLOCK
ABBREV=FPGA_BLOCK
LAST_MODIFIED=Wed May 19 14:37:13 1999

PMC-Sierra, Inc.

DOCUMENT NUMBER: 990206
ISSUE: 3.0
TITLE: AAL1GATOR II PLUS TQUAD REF DESIGN
FPGA BLOCK
DATE: 99/03/08
ENGINEER: SW
PAGE: 5 OF 8

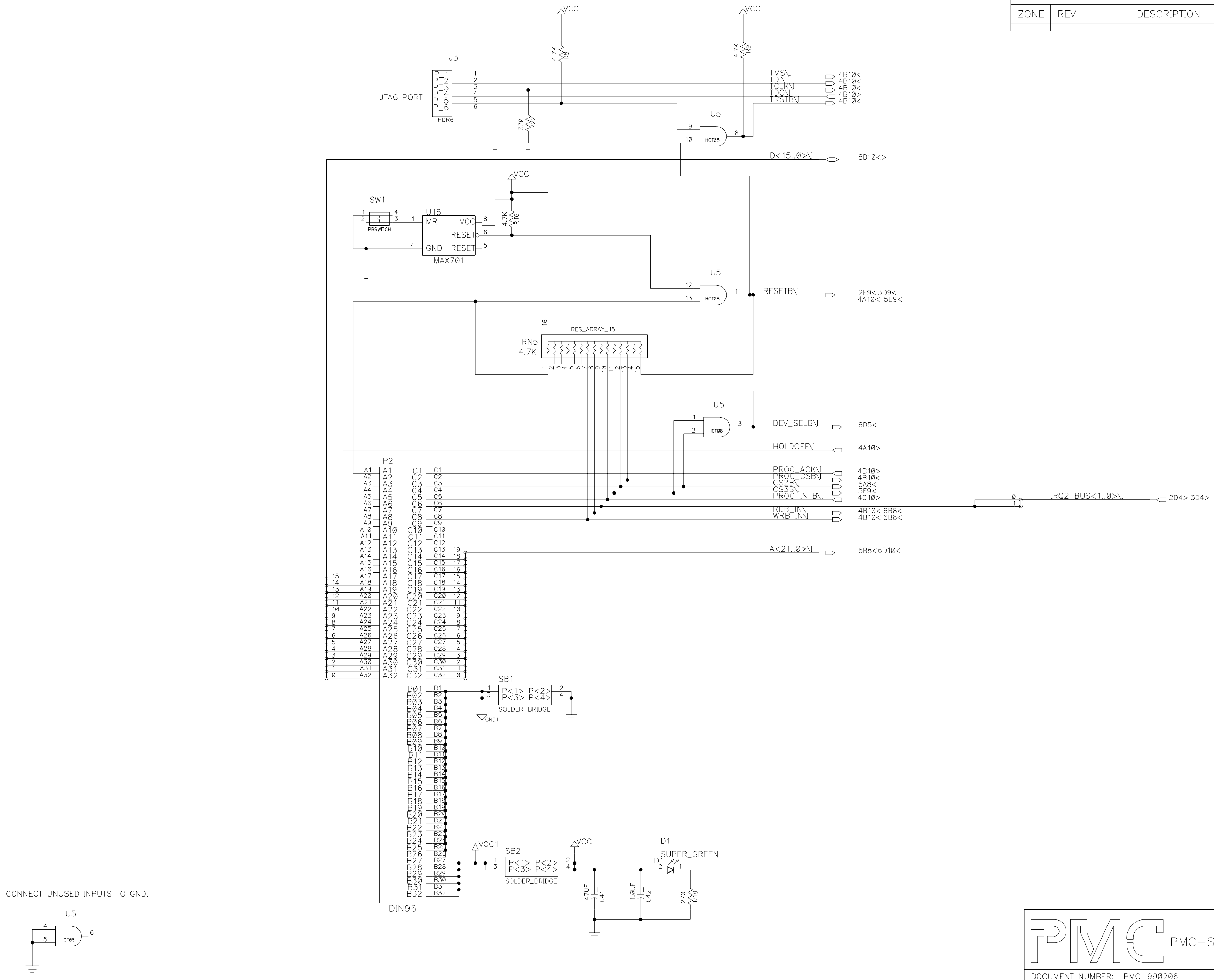


REVISIONS				
ZONE	REV	DESCRIPTION	DATE	APPR

PMC-Sierra, Inc.

DOCUMENT NUMBER: 990206	ISSUE: 3.0
TITLE: AAL1GATOR II PLUS TQUAD REF DESIGN MEMORY SYSTEM BLOCK	DATE: 99/03/08
ENGINEER: SW	PAGE: 6 OF 8

REVISIONS				
ZONE	REV	DESCRIPTION	DATE	APPR



CONNECT UNUSED INPUTS TO GND.

DRAWING

TITLE=MICRO_INTERFACE
ABBREV=MICRO
LAST_MODIFIED=Wed May 19 14:37:11 1999

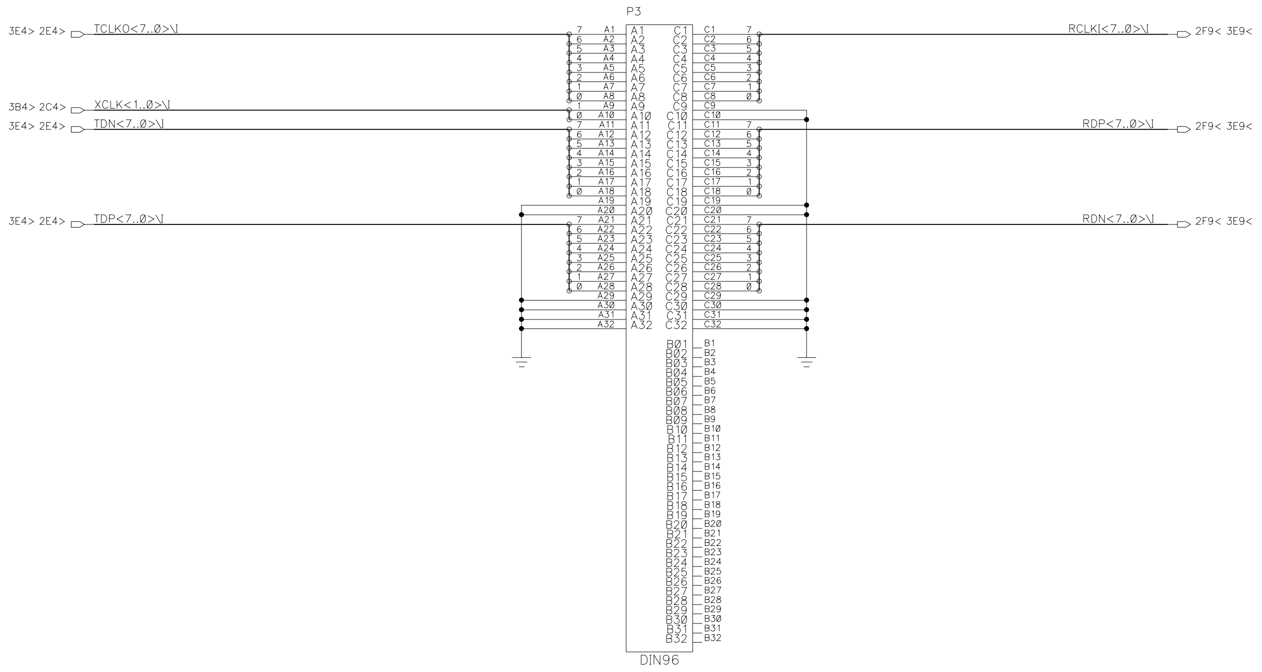
PMC-Sierra, Inc.

DOCUMENT NUMBER: PMC-990206	ISSUE: 3.0
TITLE: AAL1GATOR II PLUS TQUAD REF DESIGN MICRO_INTERFACE	DATE: 99/03/08
ENGINEER: SW	PAGE: 7 OF 8

TRUE

REVISIONS				
ZONE	REV	DESCRIPTION	DATE	APPR

CONNECT TO LINE INTERFACE UNIT
(EG. PM4314 QDSX)


P3
DIN96
3E4> 2E4> TCLKO<7..0>\I
3B4> 2C4> XCLK<1..0>\I
3E4> 2E4> TDN<7..0>\I
3E4> 2E4> TDP<7..0>\I
RCLKI<7..0>\I 2F9< 3E9<
RDP<7..0>\I 2F9< 3E9<
RDN<7..0>\I 2F9< 3E9<
A1 A2 A3 A4 A5 A6 A7 A8 A9 A10 A11 A12 A13 A14 A15 A16 A17 A18 A19 A20 A21 A22 A23 A24 A25 A26 A27 A28 A29 A30 A31 A32
C1 C2 C3 C4 C5 C6 C7 C8 C9 C10 C11 C12 C13 C14 C15 C16 C17 C18 C19 C20 C21 C22 C23 C24 C25 C26 C27 C28 C29 C30 C31 C32
B01 B02 B03 B04 B05 B06 B07 B08 B09 B10 B11 B12 B13 B14 B15 B16 B17 B18 B19 B20 B21 B22 B23 B24 B25 B26 B27 B28 B29 B30 B31 B32
B1 B2 B3 B4 B5 B6 B7 B8 B9 B10 B11 B12 B13 B14 B15 B16 B17 B18 B19 B20 B21 B22 B23 B24 B25 B26 B27 B28 B29 B30 B31 B32
7 6 5 4 3 2 1 0

DRAWING

TITLE=LINE_INTERFACE_BLOCK
ABBREV=LINE_INTERFACE_BLOCK
LAST_MODIFIED=Wed May 19 14:37:27 1999

PMC–Sierra, Inc.

DOCUMENT NUMBER: 990206	ISSUE: 3.0
TITLE: AAL1GATOR II PLUS TQUAD REF DESIGN LIU INTERFACE BLOCK	DATE: 99/03/08
ENGINEER: SW	PAGE: 8 OF 8

14 APPENDIX E: SAMPLE VHDL CODE

Note: The module adapt_clk is not provided in this document. The module is available from PMC-Sierra, Inc. on a special zero fee license. Please contact PMC-Sierra for additional information.

```
-- -----------------------------------------------------------------
-- -----------------------------------------------------------------
-- PMC-Sierra, Inc.
-- PROPRIETARY AND CONFIDENTIAL
--
-- Copyright 1999 PMC-Sierra, Inc.
--
-- All rights reserved. No part of this documentation or computer
-- program may be used, modified, reproduced, or distributed in any
-- form by any means without the prior written permission of
-- PMC-Sierra, Inc.
--
-- This documentation and computer program contains trade secrets,
-- confidential business information and commercial or financial
-- information (collectively, the "Information") of PMC-Sierra, Inc.,
-- or unlawful disclosure of any or all of the Information may cause
-- irreparable harm and result in significant commercial and
-- competitive loss to PMC-Sierra, Inc.
-- -----------------------------------------------------------------
-- -----------------------------------------------------------------
-- PMC-Sierra, Inc.
-- 105 - 8555 Baxter Place
-- Burnaby, B.C.
-- Canada   V5A 4V7
-- Tel: 604-415-6000
-- Fax: 604-415-6206
-- email: apps@pmc-sierra.com
-- -----------------------------------------------------------------
-- Project       : PMC-990206
-- File Name     : aal1gator_top.vhd
-- Path          :
-- Designer      : SW
--
-- Revision History
-- Issue         Date          Initials  Description
-- 1             03/23/99      SW             Initial Release
--
-- Function
-- This is the top level of the VHDL code required for the AAL1gator II
-- reference design.  The code instantiates several modules, and
-- creates various multiplexers.
-------------------------------------------------------------------


library IEEE;
use IEEE.std_logic_1164.all;
```

```
library aal1_lib;
use aal1_lib.all;

entity aal1gator_top is
        port (
        tl_fsync_btfp : out std_logic_vector (7 downto 0) := "ZZZZZZZZ";
        tl_clkio : inout std_logic_vector (7 downto 0) := "ZZZZZZZZ";
        btclk : out std_logic_vector (7 downto 0) := "ZZZZZZZZ";
        xclk : out std_logic_vector (7 downto 0);
        srts_dout : in std_logic_vector (3 downto 0) := "0000";
        srts_line : in std_logic_vector (3 downto 0) := "0000";
        adap_strb : in std_logic := '0';
        cs3b : in std_logic := '1';
        rdb : in std_logic := '1';
        resetb : in std_logic := '1';
        wrb : in std_logic := '1';
        network_clk : in std_logic := '0';
        Eclk : in std_logic := '0';     --2.048 MHz source
        Tclk : in std_logic := '0';     --1.544 MHz source
        n_clk : out std_logic := '0';
        aal1_sysclk : out std_logic := '0';
        fpga_data : inout std_logic_vector (7 downto 0) := "ZZZZZZZZ";
        fpga_add : in std_logic_vector (2 downto 0) := "ZZZ"
        );
end aal1gator_top;

architecture aal1gator_top_arch of aal1gator_top is

component div16
        port (
                clk_in   : in std_logic;
                resetb  : in std_logic;
                clk_out : out std_logic
        );
end component;

component frame_genE
        port (
                clk_in : in std_logic;
                resetb : in std_logic;
                frame_out : out std_logic;
                pre_frame_out : out std_logic
        );
end component;

component frame_genT
        port (
                clk_in : in std_logic;
                resetb : in std_logic;
                frame_out : out std_logic;
                pre_frame_out : out std_logic
        );
end component;

component microport
        port (
        CS3B : in std_logic;
                WRB : in std_logic;
                RDB : in std_logic;
                RESETB : in std_logic;
                IO: inout std_logic_vector (7 downto 0) := "ZZZZZZZZ";
                address: in std_logic_vector (2 downto 0) := "ZZZ";
                source_select :out std_logic_vector (7 downto 0);
                n_clk : out std_logic_vector (7 downto 0);
                xclk : out std_logic_vector (7 downto 0);
                clk : in std_logic
        );
end component;

-- Note 1. The following block is not provided with this code. It is
-- available from PMC-Sierra with a special zero fee license.
-- Please contact PMC-Sierra for further information.

-- Note 2. The module (as tested in this application) has been
-- modified such that the top level entity is as shown in the following
-- component declaration.  This was done to remove the SRTS code
-- which is not required in this example.

component adapt_clk
        port (
                clk3888: in std_logic;
                srts_dout: in std_logic_vector(3 downto 0);
                srts_line: in std_logic_vector(3 downto 0);
                adap_strobe: in std_logic;

                -- if 1, use T1 mode, else E1

                t1_mode: in std_logic_vector(7 downto 0);
                reset: in std_logic;                    -- when 1 reset
                tclk: out std_logic_vector(7 downto 0)
   );
end component;

signal tl_clki : std_logic_vector (7 downto 0);
signal tl_clko : std_logic_vector (7 downto 0);

--Source Mux out connects to BTCLK
signal source_mux_out : std_logic_vector (7 downto 0);
signal reset : std_logic := '0';

--Pulse generator outputs
signal Tpulse_out : std_logic_vector (7 downto 0);
signal Epulse_out : std_logic_vector (7 downto 0);
signal pre_Tpulse_out : std_logic_vector (7 downto 0);
signal pre_Epulse_out : std_logic_vector (7 downto 0);

--Source select determines
signal source_select : std_logic_vector(7 downto 0);

signal adapt_clk_out : std_logic_vector(7 downto 0);

-- connects output of pulse gens to output pins
signal tl_fsync_btfp_sig : std_logic_vector (7 downto 0);

-- control register that sets the XCLK source
-- only required in AAL1gator II plus COMET implementations
signal xclk_reg : std_logic_vector (7 downto 0);

-- controls the n_clk multiplexer
signal nclk_reg : std_logic_vector (7 downto 0);

-- Input signal to n_clk mux
signal n_clk_mux_in : std_logic;

signal btclk_sig : std_logic_vector (7 downto 0);


begin

-- divides the network_clk by 16 (out = 2.43MHz)
divider : div16
        port map (
        clk_in => network_clk,
        resetb => resetb,
        clk_out => n_clk_mux_in
        );

adaptive : adapt_clk
        port map (
        clk3888 => network_clk,
        srts_dout => srts_dout,
        srts_line => srts_line,
        adap_strobe => adap_strb,
        t1_mode => xclk_reg,
        reset => reset,
        tclk => adapt_clk_out
        );


Micro : microport
        port map (
        CS3B => cs3b,
        WRB => wrb,
        RDB => rdb,
        RESETB => resetb,
        IO => fpga_data,
        address => fpga_add,
        source_select => source_select,
        n_clk => nclk_reg,
        xclk => xclk_reg,
        clk => network_clk
        );

--The following 16 blocks produce a pulse every 193 bits
-- (if T1) or 256 bits if E1.

Eblock0 : frame_genE
        port map (
        clk_in => btclk_sig(0),
        resetb => resetb,
        frame_out => Epulse_out(0),
        pre_frame_out => pre_Epulse_out(0)
        );

Eblock1 : frame_genE
        port map (
        clk_in => btclk_sig(1),
        resetb => resetb,
        frame_out => Epulse_out(1),
        pre_frame_out => pre_Epulse_out(1)
        );

Eblock2 : frame_genE
        port map (
        clk_in => btclk_sig(2),
        resetb => resetb,
        frame_out => Epulse_out(2),
        pre_frame_out => pre_Epulse_out(2)
        );

Eblock3 : frame_genE
        port map (
        clk_in => btclk_sig(3),
        resetb => resetb,
        frame_out => Epulse_out(3),
        pre_frame_out => pre_Epulse_out(3)
        );

Eblock4 : frame_genE
        port map (
        clk_in => btclk_sig(4),
        resetb => resetb,
        frame_out => Epulse_out(4),
        pre_frame_out => pre_Epulse_out(4)
        );

Eblock5 : frame_genE
        port map (
        clk_in => btclk_sig(5),
        resetb => resetb,
        frame_out => Epulse_out(5),
        pre_frame_out => pre_Epulse_out(5)
        );

Eblock6 : frame_genE
        port map (
        clk_in => btclk_sig(6),
        resetb => resetb,
        frame_out => Epulse_out(6),
        pre_frame_out => pre_Epulse_out(6)
        );

Eblock7 : frame_genE
        port map (
        clk_in => btclk_sig(7),
        resetb => resetb,
        frame_out => Epulse_out(7),
        pre_frame_out => pre_Epulse_out(7)
        );

Tblock0 : frame_genT
        port map (
        clk_in => btclk_sig(0),
        resetb => resetb,
        frame_out => Tpulse_out(0),
        pre_frame_out => pre_Tpulse_out(0)
        );

Tblock1 : frame_genT
        port map (
        clk_in => btclk_sig(1),
        resetb => resetb,
        frame_out => Tpulse_out(1),
        pre_frame_out => pre_Tpulse_out(1)
        );

Tblock2 : frame_genT
        port map (
        clk_in => btclk_sig(2),
        resetb => resetb,
        frame_out => Tpulse_out(2),
        pre_frame_out => pre_Tpulse_out(2)
        );

Tblock3 : frame_genT
        port map (
        clk_in => btclk_sig(3),
        resetb => resetb,
        frame_out => Tpulse_out(3),
        pre_frame_out => pre_Tpulse_out(3)
        );

Tblock4 : frame_genT
        port map (
        clk_in => btclk_sig(4),
        resetb => resetb,
        frame_out => Tpulse_out(4),
        pre_frame_out => pre_Tpulse_out(4)
        );

Tblock5 : frame_genT
        port map (
        clk_in => btclk_sig(5),
        resetb => resetb,
        frame_out => Tpulse_out(5),
        pre_frame_out => pre_Tpulse_out(5)
        );

Tblock6 : frame_genT
        port map (
        clk_in => btclk_sig(6),
        resetb => resetb,
        frame_out => Tpulse_out(6),
        pre_frame_out => pre_Tpulse_out(6)
        );

Tblock7 : frame_genT
        port map (
        clk_in => btclk_sig(7),
        resetb => resetb,
        frame_out => Tpulse_out(7),
        pre_frame_out => pre_Tpulse_out(7)
        );

btclk_sig <= not source_mux_out;

-- If the application involves the TQUAD/EQUAD device, the following line
-- should be uncommented.

-- btclk <= not source_mux_out;

-- If the application involves the COMET device the following line
-- should be commented out
btclk <= source_mux_out;

-- if the application uses a mix of TQUAD/EQUAD and COMETS, invert the
-- source_mux_out bit lines that connect to TQUAD or EQUAD devices.  Leave
-- lines connecting to COMET lines uninverted.

tl_fsync_btfp <= tl_fsync_btfp_sig;


-- The following mux set the output signal that eventually connects
-- to the framer BTCLK signal.  The option is a signal from the
-- AAL1gator II (TL_CLKI) or an external source.

source_mux_out(7) <= adapt_clk_out(7) when source_select(7) = '1'
                        else tl_clkio(7);

source_mux_out(6) <= adapt_clk_out(6) when source_select(6) = '1'
                        else tl_clkio(6);

source_mux_out(5) <= adapt_clk_out(5) when source_select(5) = '1'
                        else tl_clkio(5);

source_mux_out(4) <= adapt_clk_out(4) when source_select(4) = '1'
                        else tl_clkio(4);

source_mux_out(3) <= adapt_clk_out(3) when source_select(3) = '1'
```

```
                        else tl_clkio(3);

source_mux_out(2) <= adapt_clk_out(2) when source_select(2) = '1'
                        else tl_clkio(2);

source_mux_out(1) <= adapt_clk_out(1) when source_select(1) = '1'
                        else tl_clkio(1);

source_mux_out(0) <= adapt_clk_out(0) when source_select(0) = '1'
                        else tl_clkio(0);


--The following signals set the output value of the bidir pin TL_CLKIO.
--When high Z, any signal driven onto the tl_clkio pin externally will
--be passed through by virue of the vhdl resolution function.

--IMPORTANT NOTE 1: When switching between having the AAL1gator II source, to
--having the FPGA source, the following steps must be taken:
--        1) Configure AAL1gator II to receive external clock (via clk_source(5:4))
--        2) Configure FPGA to source external clk via tl_clkio
--
--IMPORTANT NOTE 2: When switching between having the FPGA source, to
--having the AAL1gator II source, the following steps must be taken:
--        1) Configure FPGA to receive external clock tl_clkio
--        2) Configure AAL1gator II to source via clk_source(5:4)

tl_clkio(7) <= adapt_clk_out(7) when source_select(7) = '1'
                else 'Z';

tl_clkio(6) <= adapt_clk_out(6) when source_select(6) = '1'
                else 'Z';

tl_clkio(5) <= adapt_clk_out(5) when source_select(5) = '1'
                else 'Z';

tl_clkio(4) <= adapt_clk_out(4) when source_select(4) = '1'
                else 'Z';

tl_clkio(3) <= adapt_clk_out(3) when source_select(3) = '1'
                else 'Z';

tl_clkio(2) <= adapt_clk_out(2) when source_select(2) = '1'
                else 'Z';

tl_clkio(1) <= adapt_clk_out(1) when source_select(1) = '1'
                else 'Z';
```

```
tl_clkio(0) <= adapt_clk_out(0) when source_select(0) = '1'
                    else 'Z';



-- The following mux determines which pulse generator source should be used
-- on a per line basis.

tl_fsync_btfp_sig(7) <= Epulse_out(7) when xclk_reg(7) = '0'
                           else Tpulse_out(7);

tl_fsync_btfp_sig(6) <= Epulse_out(6) when xclk_reg(6) = '0'
                              else Tpulse_out(6);

tl_fsync_btfp_sig(5) <= Epulse_out(5) when xclk_reg(5) = '0'
                              else Tpulse_out(5);

tl_fsync_btfp_sig(4) <= Epulse_out(4) when xclk_reg(4) = '0'
                              else Tpulse_out(4);

tl_fsync_btfp_sig(3) <= Epulse_out(3) when xclk_reg(3) = '0'
                              else Tpulse_out(3);

tl_fsync_btfp_sig(2) <= Epulse_out(2) when xclk_reg(2) = '0'
                              else Tpulse_out(2);

tl_fsync_btfp_sig(1) <= Epulse_out(1) when xclk_reg(1) = '0'
                              else Tpulse_out(1);

tl_fsync_btfp_sig(0) <= Epulse_out(0) when xclk_reg(0) = '0'
                              else Tpulse_out(0);


-- The following mux is only required in applications involving the COMET
-- device.  It sets the XCLK source signal to each COMET.

xclk(7) <= Tclk when xclk_reg(7) = '1'
                    else Eclk;

xclk(6) <= Tclk when xclk_reg(6) = '1'
                    else Eclk;

xclk(5) <= Tclk when xclk_reg(5) = '1'
                    else Eclk;

xclk(4) <= Tclk when xclk_reg(4) = '1'
```

```
                    else Eclk;

xclk(3) <= Tclk when xclk_reg(3) = '1'
                    else Eclk;

xclk(2) <= Tclk when xclk_reg(2) = '1'
                    else Eclk;

xclk(1) <= Tclk when xclk_reg(1) = '1'
                    else Eclk;

xclk(0) <= Tclk when xclk_reg(0) = '1'
                    else Eclk;

-- Sets n_clk to be either 2.43 MHz (SRTS enabled) or GND
-- (SRTS disabled).

n_clk <= n_clk_mux_in when nclk_reg(7) = '1'
          else '0';


end aal1gator_top_arch;
```

```
-- ------------------------------------------------------------------
-- ------------------------------------------------------------------
-- PMC-Sierra, Inc.
-- PROPRIETARY AND CONFIDENTIAL
--
-- Copyright 1999 PMC-Sierra, Inc.
--
-- All rights reserved. No part of this documentation or computer
-- program may be used, modified, reproduced, or distributed in any
-- form by any means without the prior written permission of
-- PMC-Sierra, Inc.
--
-- This documentation and computer program contains trade secrets,
-- confidential business information and commercial or financial
-- information (collectively, the "Information") of PMC-Sierra, Inc.,
-- or unlawful disclosure of any or all of the Information may cause
-- irreparable harm and result in significant commercial and
-- competitive loss to PMC-Sierra, Inc.
-- ------------------------------------------------------------------
-- ------------------------------------------------------------------
-- PMC-Sierra, Inc.
-- 105 - 8555 Baxter Place
-- Burnaby, B.C.
-- Canada   V5A 4V7
-- Tel: 604-415-6000
-- Fax: 604-415-6206
-- email: apps@pmc-sierra.com
-- ------------------------------------------------------------------
-- Project        : PMC-990206
-- File Name      : frame_genE.vhd
-- Path           :
-- Designer       : SW
--
-- Revision History
-- Issue          Date          Initials  Description
-- 1              03/23/99      SW                Initial Release
--
-- Function
-- This module generates an output pulse 1 clk period wide every 256 bits
---------------------------------------------------------------------

library IEEE;
use IEEE.std_logic_1164.all;

entity frame_genE is
        port(
        clk_in   : in std_logic;
```

```
          resetb  : in std_logic;
          frame_out : out std_logic;
          pre_frame_out : out std_logic
          );

end frame_genE;

architecture frame_genE_arch of frame_genE is

          --every 256 bits output a frame pulse 1 clk_out cycle wide
          constant frame_count_max : integer := 256;

          begin

           --a frame pulse, 1 cycle clk_out wide is generated every 256 bits
          gen_framepulse : process(clk_in, resetb)
          variable count_frame: integer := 1;
begin
                              if resetb = '0' then
                                        count_frame := 1;
                                        frame_out <= '0';
                                        pre_frame_out <= '0';
                              elsif (clk_in'event and clk_in = '0') then  --falling edge

                                        if count_frame = (frame_count_max - 1) then
                                                  pre_frame_out <= '1';
                                                  frame_out <= '0';
                                                  count_frame := count_frame + 1;
                                        elsif count_frame = frame_count_max then
                                                  pre_frame_out <= '0';
                                                  frame_out <= '1';
                                                  count_frame := 1;
                                        else
                                                  count_frame := count_frame + 1;
                                                  frame_out <= '0';
                                                  pre_frame_out <= '0';
                                        end if;
                              end if;
                    end process;
end frame_genE_arch;
```

```
-- -----------------------------------------------------------------
-- -----------------------------------------------------------------
-- PMC-Sierra, Inc.
-- PROPRIETARY AND CONFIDENTIAL
--
-- Copyright 1999 PMC-Sierra, Inc.
--
-- All rights reserved. No part of this documentation or computer
-- program may be used, modified, reproduced, or distributed in any
-- form by any means without the prior written permission of
-- PMC-Sierra, Inc.
--
-- This documentation and computer program contains trade secrets,
-- confidential business information and commercial or financial
-- information (collectively, the "Information") of PMC-Sierra, Inc.,
-- or unlawful disclosure of any or all of the Information may cause
-- irreparable harm and result in significant commercial and
-- competitive loss to PMC-Sierra, Inc.
-- -----------------------------------------------------------------
-- -----------------------------------------------------------------
-- PMC-Sierra, Inc.
-- 105 - 8555 Baxter Place
-- Burnaby, B.C.
-- Canada   V5A 4V7
-- Tel: 604-415-6000
-- Fax: 604-415-6206
-- email: apps@pmc-sierra.com
-- -----------------------------------------------------------------
-- Project       : PMC-990206
-- File Name     : frame_genT.vhd
-- Path          :
-- Designer      : SW
--
-- Revision History
-- Issue         Date            Initials  Description
-- 1             03/23/99        SW                Initial Release
--
-- Function
-- This module generates an output pulse 1 clk period wide every 193 bits
--------------------------------------------------------------------

library IEEE;
use IEEE.std_logic_1164.all;

entity frame_genT is
        port(
        clk_in   : in std_logic;
        resetb  : in std_logic;
        frame_out : out std_logic;
        pre_frame_out : out std_logic
        );

end frame_genT;

architecture frame_genT_arch of frame_genT is

        --every 193 bits output a frame pulse 1 clk_out cycle wide
        constant frame_count_max : integer := 193;

        begin

         --a frame pulse, 1 cycle clk_out wide is generated every 193 bits

        gen_framepulse : process(clk_in, resetb)
variable count_frame: integer := 1;
                begin
                        if resetb = '0' then
                                count_frame := 1;
                                frame_out <= '0';
                                pre_frame_out <= '0';
                        elsif (clk_in'event and clk_in = '0') then  --falling edge

                                if count_frame = (frame_count_max - 1) then
                                        pre_frame_out <= '1';
                                        frame_out <= '0';
                                        count_frame := count_frame + 1;
                                elsif count_frame = frame_count_max then
                                        pre_frame_out <= '0';
                                        frame_out <= '1';
                                        count_frame := 1;
                                else
                                        count_frame := count_frame + 1;
                                        frame_out <= '0';
                                        pre_frame_out <= '0';
                                end if;
                        end if;
                end process;
end frame_genT_arch;
```

```
-- -----------------------------------------------------------------
-- -----------------------------------------------------------------
-- PMC-Sierra, Inc.
-- PROPRIETARY AND CONFIDENTIAL
--
-- Copyright 1999 PMC-Sierra, Inc.
--
-- All rights reserved. No part of this documentation or computer
-- program may be used, modified, reproduced, or distributed in any
-- form by any means without the prior written permission of
-- PMC-Sierra, Inc.
--
-- This documentation and computer program contains trade secrets,
-- confidential business information and commercial or financial
-- information (collectively, the "Information") of PMC-Sierra, Inc.,
-- or unlawful disclosure of any or all of the Information may cause
-- irreparable harm and result in significant commercial and
-- competitive loss to PMC-Sierra, Inc.
-- -----------------------------------------------------------------
-- -----------------------------------------------------------------
-- PMC-Sierra, Inc.
-- 105 - 8555 Baxter Place
-- Burnaby, B.C.
-- Canada   V5A 4V7
-- Tel: 604-415-6000
-- Fax: 604-415-6206
-- email: apps@pmc-sierra.com
-- -----------------------------------------------------------------
-- Project        : PMC-990206
-- File Name      : micrport.vhd
-- Path           :
-- Designer       : SW
--
-- Revision History
-- Issue          Date          Initials  Description
-- 1              03/23/99      SW             Initial Release
--
-- Function
-- This module creates a port for which the microprocessor communicates
-- with in order to set the function of the FPGA.
-------------------------------------------------------------------

library IEEE;
use IEEE.std_logic_1164.all;

entity MICROPORT is
```

```
port (
        CS3B : in std_logic;
        WRB : in std_logic;
        RDB : in std_logic;
        RESETB : in std_logic;
        IO: inout std_logic_vector (7 downto 0) := "ZZZZZZZZ";
        address: in std_logic_vector (2 downto 0) := "ZZZ";
        source_select :out std_logic_vector (7 downto 0);
        n_clk : out std_logic_vector (7 downto 0);
        xclk : out std_logic_vector (7 downto 0);
        clk : in std_logic
);

end MICROPORT;

architecture MICROPORT_ARCH of MICROPORT is

-- setup the control registers:

        type regtype is array (0 to 2) of std_logic_vector(7 downto 0);
        signal fpga: regtype := (("11111111"), ("11111111"), ("11111111"));

-- utility function to convert std_logic_vectors to intergers
function vec2int(vec1: std_logic_vector)
 return integer is
variable retval: integer:= 0;
alias vec : std_logic_vector(vec1'length -1 downto 0) is vec1;
begin
  for i in vec'high downto 1 loop
        if (vec(i) = '1') then
                retval:= (retval+1)*2;
        else
                retval := retval*2;
        end if;
  end loop;
if vec(0) = '1' then
        retval := retval + 1;
end if;
return retval;
end vec2int;

begin
        process (CS3B)
        begin
             if CS3B = '1' then IO <= "ZZZZZZZZ";
                  else
                         if WRB = '0' then
```

```
                                fpga (vec2int(address)) <= IO;


                         elsif RDB = '0' then
                             IO <= fpga(vec2int(address));

                  end if;
                  end if;
        end process;

source_select <= fpga(0) after 1 ns;
n_clk <= fpga(1) after 1 ns;
xclk <= fpga(2) after 1 ns;

end MICROPORT_ARCH;
```

```
-- ------------------------------------------------------------------
-- ------------------------------------------------------------------
-- PMC-Sierra, Inc.
-- PROPRIETARY AND CONFIDENTIAL
--
-- Copyright 1999 PMC-Sierra, Inc.
--
-- All rights reserved. No part of this documentation or computer
-- program may be used, modified, reproduced, or distributed in any
-- form by any means without the prior written permission of
-- PMC-Sierra, Inc.
--
-- This documentation and computer program contains trade secrets,
-- confidential business information and commercial or financial
-- information (collectively, the "Information") of PMC-Sierra, Inc.,
-- or unlawful disclosure of any or all of the Information may cause
-- irreparable harm and result in significant commercial and
-- competitive loss to PMC-Sierra, Inc.
-- ------------------------------------------------------------------
-- ------------------------------------------------------------------
-- PMC-Sierra, Inc.
-- 105 - 8555 Baxter Place
-- Burnaby, B.C.
-- Canada   V5A 4V7
-- Tel: 604-415-6000
-- Fax: 604-415-6206
-- email: apps@pmc-sierra.com
-- ------------------------------------------------------------------
-- Project       : PMC-990206
-- File Name     : div16.vhd
-- Path          :
-- Designer      : SW
--
-- Revision History
-- Issue         Date          Initials  Description
-- 1             03/23/99      SW              Initial Release
--
-- Function
-- This module divides a clock signal by 16
--------------------------------------------------------------------

library IEEE;
use IEEE.std_logic_1164.all;

entity div16 is
        port(
        clk_in   : in std_logic;
```

```
    resetb  : in std_logic;
    clk_out : out std_logic
    );

end div16;

architecture div16_arch of div16 is

    constant clk_count_max : integer := 8;
   signal div_clk : std_logic := '0';
    begin

    gen_clock : process(clk_in, resetb)
    variable clk_count: integer := 0;

        begin
            if (resetb = '0') then
                clk_count := 0;
                div_clk <= '0';

            elsif (clk_in'event and clk_in = '1') then  --rising edge
               clk_count := clk_count + 1;

                if (clk_count = clk_count_max) then
                    div_clk <= not div_clk;
                    clk_count := 0;
                end if;
            end if;
        end process;

    clk_out <= div_clk;
end div16_arch;
```

15 DISCLAIMER

This reference design has not been built or tested.

16 NOTES

CONTACTING PMC-SIERRA, INC.

PMC-Sierra, Inc.
105-8555 Baxter Place Burnaby, BC
Canada V5A 4V7

Tel: (604) 415-6000

Fax: (604) 415-6200

Document Information: document@pmc-sierra.com
Corporate Information: info@pmc-sierra.com
Application Information: apps@pmc-sierra.com
(604) 415-4533
Web Site: http://www.pmc-sierra.com

None of the information contained in this document constitutes an express or implied warranty by PMC-Sierra, Inc. as to the sufficiency, fitness or suitability for a particular purpose of any such information or the fitness, or suitability for a particular purpose, merchantability, performance, compatibility with other parts or systems, of any of the products of PMC-Sierra, Inc., or any portion thereof, referred to in this document. PMC-Sierra, Inc. expressly disclaims all representations and warranties of any kind regarding the contents or use of the information, including, but not limited to, express and implied warranties of accuracy, completeness, merchantability, fitness for a particular use, or non-infringement.

In no event will PMC-Sierra, Inc. be liable for any direct, indirect, special, incidental or consequential damages, including, but not limited to, lost profits, lost business or lost data resulting from any use of or reliance upon the information, whether or not PMC-Sierra, Inc. has been advised of the possibility of such damage.

PMC-990206 Issue date: May 1999