Application Note AN-N21

Switching Using The MUSIC MUAA™ Routing CoProcessors (RCPs)

INTRODUCTION

The MUSIC MUAA Routing CoProcessors (RCPs) are specifically tailored for high performance MAC address filtering. The MUAA RCP is an ideal choice when designing multi-port switches because of their ability to quickly find and learn layer 2 MAC addresses.

The MUAA RCP is available with 2K, 4K, and 8K deep memory arrays but can be cascaded easily to provide a total table size of up to 32K. The MUAAs are chained transparently, which removes the need for configuration and delays caused by flag ripple-through. This adds to their ease of operation when a design requires deep memory.

The MUAA RCP contains an 80–bit wide address database which can be partitioned into separate CAM and RAM areas. In a typical switch application it is configured as 48 bits of CAM and 32 bits of RAM. The 48 bits of CAM stores MAC Source addresses and the 32 bits of RAM would hold associated data such as port ID and VLAN information. It also is possible to index external RAM if additional associated data are required.

The MUAA RCP filters 7.14 million packets per second, with one search and one learn operation per packet. It supports up to 48 100 Mb Ethernet ports because of its fast compare cycle time. The MUAA RCP contains internal logic that automatically time stamps learned entries and ages out older entries. This reduces the housekeeping tasks performed by a host processor. “Virtual queues” also provides the system to monitor recently aged and learned entries.

With the aid of block diagrams and software examples, this application note explains how the MUAA RCP is used in a switch application. This note also contains some background information such as filtering techniques and the use of forwarding tables. For a full explanation of the MUAA RCP please refer to the datasheet. The datasheet contains all the relevant timing diagrams and instruction set descriptions.

SWITCH BASICS

Switches are devices that provide an interface from one network segment to another. They intelligently filter traffic that crosses between segments preventing the unnecessary waste of bandwidth by traffic that does not need to be on a particular segment. Switches do this by automatically learning which network node resides on which segment. Traffic is forwarded only to those segments with one or more nodes that are traffic destinations.

Switches detect which node is on which segment by parsing received frames for the SA (Source address). That SA is written into a table along with useful associated data such as the port number on which the frame was received and a time stamp for aging entries (see section on time stamping and aging). Once the switch gains some knowledge about which nodes reside on which segments, it decides which traffic it will forward from one segment to another. The learning process is continual and entries are aged out (deleted) if a packet with the SA that is contained in the entry is not received for a certain length of time.

A switch decides what it will forward by examining the DA (Destination address) of a frame it receives on a port from a segment. It compares this DA with the table of SAs (and associated data) that it has learned. If it finds a match it forwards the frame onto the segment found in the table’s associated data field. If it does not find a match it forwards the frame to all enabled ports apart from the one on which it was received. Broadcast packets are always sent to all segments except for the segment originating the broadcast packet. A protocol called Spanning Tree is used to prevent loops and parallel paths forming on a switched network. This causes disabling of certain ports and prevents frames from being multiplied or from circulating forever.

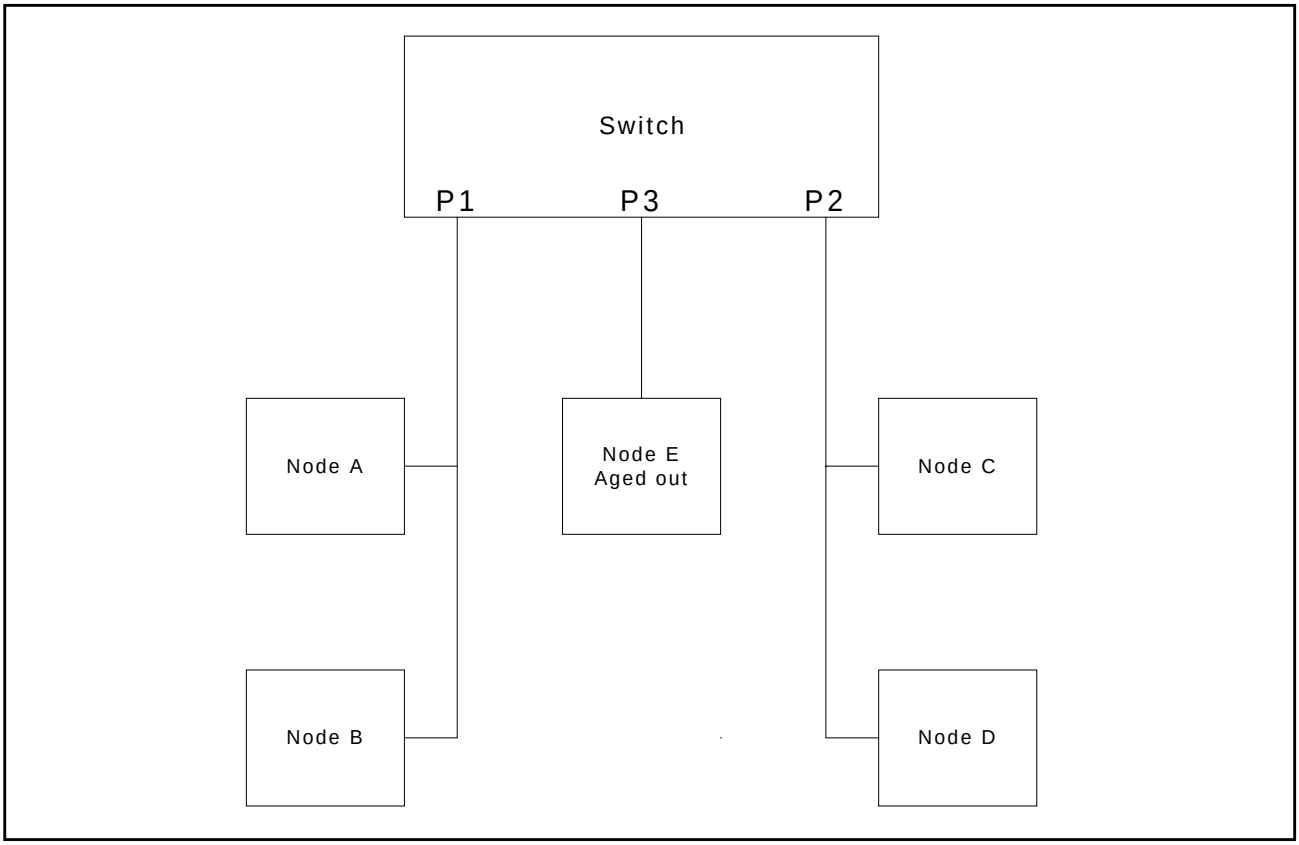

Figure 1: Theoretical 3 Port Switch Arrangement

Figure 1 shows a theoretical 3 port switch with nodes A and B connected to port 1, nodes C and D connected to port 2, and node E connected to port 3, which has not transmitted recently and has been aged out. The switch builds a (simplified) forwarding table similar to the one shown in Table 1.

Node	Port
A	1
B	1
C	2
D	2

Table 1: Simplified Forwarding Table

If node A sends a frame to node B the switch parses the incoming frame and finds that the DA is node B. It searches the table for B and discovers B is on port 1. As it receives the frame on port 1 it does not forward it to any other ports.

If node A sends a frame to node C, the switch finds that the DA is node C. When it searches the table it finds C on port 2, so it forwards the frame to port 2. If node C sends a frame to node E, the switch searches the table for E. It does not find it so it forwards the frame to ports 1 and 3, but not to its originating port 2.

If node E now sends a frame to node D, the switch searches for D and finds it on port 2. The switch also notes the SA of the frame is E and inserts it into the table with its port number of 3.

The switch always examines the SA for each frame it receives in order to learn new entries and/or update the age and port number of existing entries to prevent them from being aged out while a node is active.

THE FORWARDING TABLE

Traditionally a microprocessor and/or finite state machine using software techniques, such as hashing or binary searches, maintains and manipulates the forwarding table. Hashing makes less efficient use of memory and requires chaining and/or double hash techniques if two MAC addresses hash to the same memory location; the lookup time is variable. Binary searches require the table to be sorted by MAC address, which makes insert time long and variable as the table entries are sorted to accept the new address.

As networks increase in speed and more switch ports and features evolve, these techniques are proving too slow, which affects the throughput and latency performance of a switch. One solution is to move this function into a RCP. Because of the RCP's ability to search all memory locations in parallel and in a short time, the DA lookup is performed in a fast deterministic fashion. The RCP also locates the next free address to allow fast deterministic address insertion.

Switch Hardware

Figure 2 shows a typical system with one to four MUAA RCPs in a chain. See the device chaining section for an explanation of how MUAA RCPs are connected together to provide a deeper forwarding table. The FSM in the controller and the processor both can communicate with the MUAA RCP independently. The MUAA has two ports for data transfers to and from the device. The FSM uses the synchronous port and the processor uses the processor port. Operations may occur on both ports at the same time; the port with the highest priority gains access first if both ports require a read or write into the RCP address database simultaneously. The port priority is set during the configuration routine. For a typical switch application, the synchronous port is given the higher priority.

Port data is received from the port physical layer devices on the RXDATA lines clocked by the recovered clock and qualified as valid receive data by the QUAL signal that may come from the physical layer and/or generated by the user. This is dependent on the architecture of the physical and MAC layer controllers. If necessary the control logic converts the serial data stream to parallel. Some devices may provide access to the parallel receive data from the MAC. The controller then parses the frame for the DA and SA. These are latched and an arbiter arbitrates for the RCP address database. Once access is granted, the DA is looked up. If found, the port number is found and if need be the MATCH signal asserted to allow the appropriate MAC to forward the frame into the switch/frame buffer that then forwards the frame on to the appropriate output port when it becomes available.

The SA also is looked up in the RCP address database. If it is not found, then the SA is learned and a new time stamp is given. If found, then the age of the entry is updated. Time stamps normally are added or updated by the FSM once the match result is known. The MUAA RCP incorporates internal logic that performs these operations automatically. Depending on the system architecture and any FIFO in the MAC device, there may be a FIFO after the MAC before the switch/buffer. This stores the start of the frame during the time the forwarding decision is made. The frame may start to be written into the buffer and if rejected, the buffer pointers are returned to the previous position.

The MUAA synchronous ports allow the forwarding and learning operations to be performed at port wire speed. The host processor can be used to initialize the MUAA and access it for background management tasks such as aging and reading out learned MAC addresses and associated ports.

Arbiter

Figure 3 shows an 802.3 Ethernet frame. At 100 Mb/s the DA takes 480 ns to accumulate from the end of preamble and the SA takes 960 ns. Both these accumulated values must be manipulated in the RCP address database. In a multi-port system the maximum bandwidth into the RCP is required when a frame arrives at all ports simultaneously, requiring multiple DA lookups. To accommodate this the DA lookup routine must have a higher priority than the SA lookup routine, which in turn has a higher priority than the processor management routines.

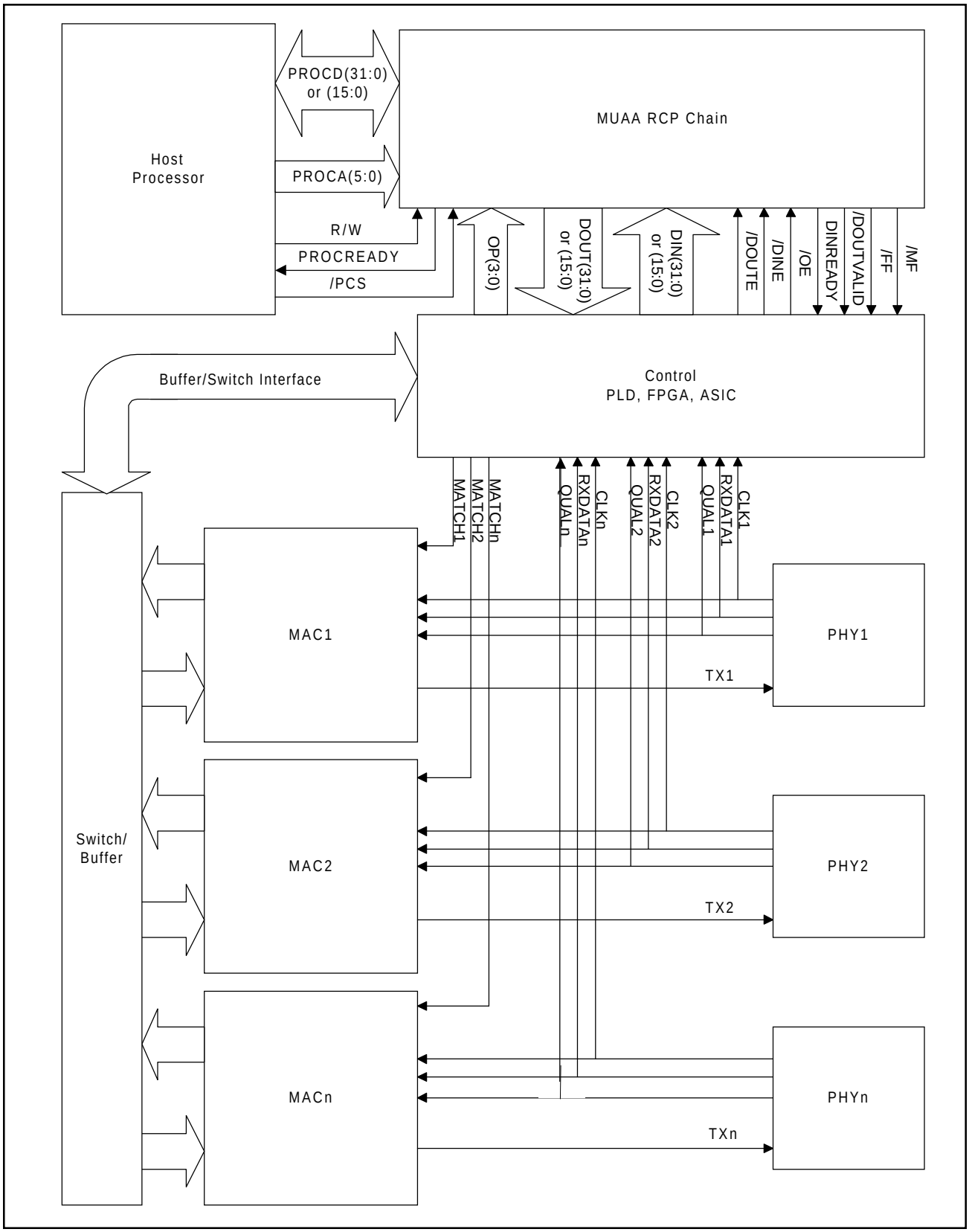

Figure 2: Interfacing a MUAA RCP to a System

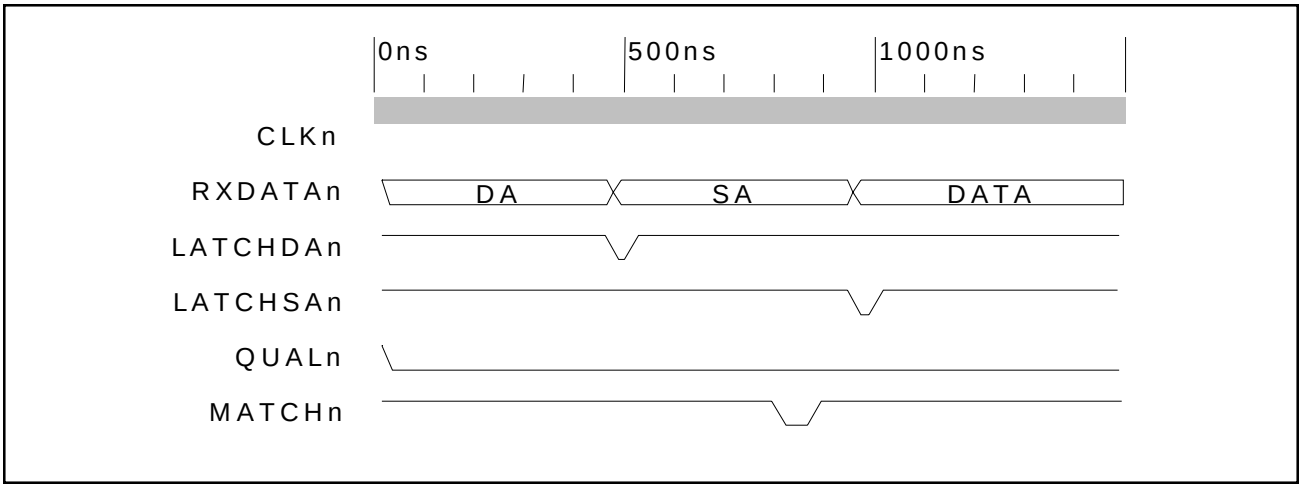

Figure 3: Ethernet Frame Arriving at a Port

The latency through the switch depends on how quickly all the ports are serviced in this situation. These requirements depend on the length of time it takes to run the DA lookup, SA lookup routines, and also any background tasks and how well these tasks are interleaved. Another consideration is the throughput of the switch. The worst case requirements are when a switch is forwarding minimum size frames on all ports with minimum inter-frame gap. To calculate the performance of the system and the number of ports per MUAA it is necessary to know the length of the various routines that are run.

If the system CLK input is 50 MHz, one search and one learn operation requires 140 ns (7 CLK periods @ 20 ns). Assuming a minimum packet size and inter-frame gap the MUAA RCP can support 48 100 Mb Ethernet ports or 4 Gigabit ports at wire speed. The MUAA RCP can filter 7.14 million packets per second, with one search, and one learn sequence per packet.

USING THE MUAA RCP

The MUSIC MUAA RCP has an address database with a partitionable CAM/RAM format. The CAM partition is the part of the field that any DA/SA searches are performed on. The RAM partition is the part of the field that contains associated data. The MUAA RCP has an 80-bit wide address database. It is programmable from 32 bits of CAM and 48 bits of RAM, to 80 bits of CAM and 0 bits of RAM. In a typical switch application (layer 2) it is partitioned as 48 bits of CAM and 32 bits of RAM. This sets the CAM partition wide enough to accommodate the MAC addresses that are 48 bits wide. The remaining 32 bits of the word are used for associated data such as port number, VLAN information, etc. There is no need to use any of the associated data field for time stamping as this is done automatically by the RCP (see the section on time stamping and aging).

VLAN (Virtual LAN) information represents an alternative solution to using routers for broadcast containment as a VLAN allows switches also to restrict broadcast traffic. The physical LAN can be configured to have nodes located in departments such as accounts, sales, and engineering grouped together. The nodes that are grouped together need not actually be part of the same physical segment or switch. If an explicit tagging scheme is used, the departments would have their own group VLAN tags, which would identify where each VLAN node belongs. This tag is transmitted as part of the MAC frame and stored as associated data with each entry. The associated data partition of the address database can be used for the VLAN information (if space allows). Otherwise the MUAA RCP provides the address index of each MAC address to enable more associated data to be incorporated in software or external hardware.

As previously mentioned, the MUAA has two ports for data transfers to and from the device. These are the synchronous port and the processor port. Operations may occur on both ports simultaneously; the port with the highest priority gains address database access first if both ports require a read or write into the address database simultaneously. The port priority is set during the configuration routine. For a typical switch application, the synchronous port is given the higher priority.

Synchronous Port

The synchronous port consists of DIN(31:0), DOUT(31:0), and the associated inputs and outputs required for successful data transfer. Setting or clearing bits 1 and 2 in the Configuration register can configure it for either 16-bit or 32-bit use. If the FSM has to transfer information to the MUAA it loads data onto the DIN(31:0) pins of the MUAA and takes /DINE LOW. At the same time, the correct instruction is given by loading the OP(3:0) pins with the appropriate data. There is a range of instructions possible such as learn, search, delete, load, and age. The full list of all the possible instructions with an explanation of their uses can be found in the MUAA RCP datasheet.

Data is transferred from the MUAA to the FSM using the DOUT(31:0) output of the synchronous port. The MUAA loads the DOUT(31:0) pins with the relevant data and takes /DOUTVALID LOW. Note that the timing of /DOUTVALID with respect to DOUT(31:0) can be configured to be active on the same CLK or one CLK period earlier. This allows the MUAA RCP to be interfaced with devices that have registered I/O.

The output port is used to transfer the associated data and address index. The address index allows the user to implement further associated data in software or hardware. This can be configured to be available before or after the associated data when a DA search results in a MATCH condition. At this point both the associated data and the address index become available. If the address index is not required, the associated data should be configured to become available first.

The user has high impedance control of the output data from the synchronous port by asserting /OE when necessary. The /DOUTE input is the DOUT(31:0) enable control. When the associated data is configured to be wider than the output port, this strobe enables the next word(s) of the DOUT data onto the DOUT(31:0) pins. The input and output ports can share the same 16-bit or 32-bit bus, allowing the pin count of surrounding hardware to be reduced if required. Refer to the datasheet for a comprehensive description of all the pins found in the MUAA RCP.

Processor Port

The processor can transfer data to the MUAA by loading the PROCD(31:0) pins with the necessary data. These pins are bidirectional and are used for both reading from and writing to the MUAA. The internal registers are accessed by the use of the PROCA(5:0) inputs. The relevant address is loaded on bits (5:1), with bit 0 being used for only the 16-bit mode of operation. When the 16-bit mode is selected, a logic 0 indicates the lower 16 bits of the register and a logic 1 indicates the upper 16 bits. If, on the other hand, the 32-bit mode is selected, bit 0 must be connected to a valid logic level. The processor has other pins that are required to ensure successful data transfer. These are R/W, /PCS, and PROCREADY.

R/W is the read or write input and controls the direction of data to and from the MUAA. R/W is held LOW to write data to the MUAA and it is held HIGH to read data from the MUAA. /PCS is the processor port chip select input and is held LOW to indicate that the cycle is to the processor port. The relevant timing information for read and write cycles with respect to /PCS is given in the data sheet. PROCREADY is an output from the MUAA and indicates that the processor may read data or data was read successfully by the device.

Interface Pins

The MUAA RCP has other pins that allow it to be interfaced simply with surrounding hardware. /MF is the Match Flag output and is asserted LOW by the device to indicate that there was a match after the previous search or learn operation. /FF is the Full Flag output and is asserted LOW by the device to indicate that the device is full and is unable to learn new addresses. CHAINUP, CHAINDN, CHAINCS, and CHAIN(3:0) allow up to four MUAAs to be chained together transparently. See the section on device chaining for more information.

The MUAA has an INT output pin that is asserted HIGH or LOW (this is configurable to suit the users application). This allows hardware to be notified of any internal interrupts. The four interrupts possible are: the aged or learn queue has at least one entry or a processor or

synchronous write exception has occurred. The individual interrupts can be enabled or disabled during system configuration. When enabled, the interrupt facility allows the system to keep track of aged and learned entries and to deal with exceptions.

The CLK input is the main system clock and registers all data read and writes to the device. /RESET is the system reset input. This must be asserted LOW for at least three CLK periods to reset the MUAA RCP. Note that the DINREADY output may stay LOW for up to 800 CLK periods after /RESET has been taken HIGH. /TRST, /TCLK, TMS, TDI, and TDO are the JTAG IEEE Standard test pins.

AQUEUE and LQUEUE

The MUAA RCP provides "virtual queues" to enable the system to keep track of entries that are aged or learned. The AQUEUE contains all entries that have been aged from the MUAA address database. The LQUEUE contains all the entries that have been learned or added to the MUAA address database. Both "virtual queues" work in the same way with only minor differences. The LQUEUE is always enabled whereas the AQUEUE must be enabled, if it is required, by setting bit 14 of the Configuration register to 1. Both share the same hardware interrupt output (INT) to inform the processor that a new entry has been written into the relevant queue.

These interrupts are enabled or disabled in the Configuration register. When an interrupt occurs, the MUAA RCP asserts the INT pin. The system finds the cause by reading the Address Index register and acknowledges it by reading the appropriate Flag register. The interrupt is not reasserted until the relevant queue is emptied and then receives another entry. The Flag registers can also be used in a software implementation when the INT pin is not enabled.

The entries in the queues can be read out from the processor port using the read LQUEUE and read AQUEUE instructions. When attempting to read queue entries from the processor port, there may be a delay if the synchronous port has higher priority. In a typical switch, the housekeeping functions carried out by the processor port would have a lower priority than the main forwarding table operations. Thus, if the processor port tries to gain access to read a queue entry, the PROCREADY output is held LOW by the MUAA RCP until the synchronous port is finished. The PROCREADY can then be taken HIGH to indicate that the data read operation can be completed. Both queues can also be completely emptied by using the appropriate clear instruction.

The processor may use the LQUEUE data to maintain a management database of MAC addresses and associated port IDs. The LQUEUE can also be used to measure the rate that entries are being learned in order to incorporate a dynamic time stamping scheme (see section on dynamic time stamping).

SWEX and PWEX

The MUAA RCP contains two interrupts that signal write exceptions. The exceptions indicate that a lower priority port attempted to write a new entry into the MUAA at the same time as the higher priority port, when the higher priority port already has a write operation in the pipeline and the MUAA has two entries left. The SWEX exception indicates that the synchronous port was the offending lower priority port and PWEX indicates the processor port.

Both interrupts cause the INT pin to be asserted if they are enabled in the Configuration register. The interrupt service routine should read the Address Index register to identify the cause of the interrupt. The appropriate Flag register should then be read to acknowledge the interrupt and return the INT pin to its normal state. Appropriate action can then be taken to rectify the subsequent problems caused by the exception.

For instance, assuming that the synchronous port is configured to be the highest priority, the INT pin is asserted if the processor port causes an exception. Once the PWEX Flag register has been read to acknowledge the interrupt and reset the INT output, the following action can be taken. The learn or insert instruction initiated by the processor prior to the exception must be attempted once the MUAA RCP is no longer full. The /FF output can be monitored to notify the processor when the MUAA RCP is no longer full and hence allow it to reattempt the failed instruction. Issuing a manual age instruction may be required to delete some of the older entries and therefore produce some free memory space.

TIME STAMPING AND AGING

Normally software techniques are used to mark the entries in the forwarding table with a time stamp in order to keep track of the age of an entry. As new entries are added, the associated data field is used to store the time stamp as well as port ID and other necessary information. Entries can also be marked as static or permanent. The software routines normally use simple "old" and "new" time stamps or a more sophisticated approach with multiple time stamps. The more sophisticated approach gives the entries a range of time stamps and ages out the oldest when necessary.

When a search (in order to learn the SA) produces a MATCH condition, the associated data field is updated to contain the most recent time stamp. If the search produced a NO-MATCH condition the SA is added to the memory array with its time stamp. After a predefined aging interval, the processor then invokes a purge routine to remove all the entries that contain the oldest time stamp (permanent entries are not removed). This helps keep forwarding tables short and therefore reduces the amount of memory required. It also helps network management as nodes that no longer exist or have moved are automatically removed from the forwarding table.

Automatic Aging with the MUAA RCP

The MUAA RCP has an IEEE 802.1D Compliant automatic time stamp and aging facility that greatly increases the usability of the device. The time stamps are stored internally when entries are learned and are invisible to the user. The auto-aging function is selected by setting bit 13 of the Configuration register to 1. This causes the MUAA to automatically delete all entries in the address database that have the oldest time stamp (and are not marked permanent). This can also be done manually by disabling the auto-aging facility and using the age instruction instead.

The auto-age interval is the time that elapses between purges and is configured by the user by setting the Auto-age Interval register. The register is 32 bits wide and the value contained in it is divided by the CLK input to produce the time interval.

The register resets to 02FAF080h, which gives an interval of 1 second with a 50 MHz CLK. The priority of this auto-aging function is selected at configuration. The value of bit 18 in the Configuration register is changed to make automatic aging higher or lower priority than synchronous or processor port operations. In a typical switch application, any auto-aging facility is normally given the lowest priority. The time stamps are automatically added or updated when a SA learn operation is carried out. The **learn<DIN>** instruction is used in conjunction with **load<DIN>** to add SAs to the address database. The way that the MUAA RCP would learn SAs and update time stamps is as follows:

The least significant 32 bits of the SA is loaded on the synchronous input port (DIN(31:0)) and the **load<DIN>** instruction is given. The remaining 16 bits are loaded on bits 15–0 of the port and another **load<DIN>** instruction is given. Bits 31–16 are discarded by the MUAA RCP at this point and can be ignored. Finally the associated data is placed on the port and a **learn<DIN>** instruction is given. The associated data usually contains information such as the identification of the port where the SA was located.

The MUAA RCP is normally configured to partition the address database as 48 bits of CAM and 32 bits of RAM. The **learn<DIN>** instruction causes the 48-bit SA to be compared with the contents of the 48-bit CAM partition. If a MATCH condition is produced, /MF flag is asserted and the associated data is moved to the 32-bit RAM partition. At the same time, the time stamp is automatically updated to the most recent.

If there is a NO-MATCH condition, the SA and the associated data are both added to the next free address (/MF is held HIGH). At this point the SA is also marked with the most recent time stamp. This routine is explained with the aid of Table 6 in the MUAA routines section. Table 2 shows the different settings required for automatic and manual aging. If the user does not want to use the time stamp and aging facility, the MUAA is configured as for no automatic aging and the manual age instruction is never given.

AUTO-AGING

REGISTER	BIT	VALUE	DESCRIPTION	NOTES
Configuration register	13	1	Turns on the auto-aging function.	
Configuration register	18	1 or 0	Sets the priority of auto-age over I/0 ports.	1
Auto-age Interval register	all	32-bit value	Sets the automatic aging interval.	2
Learned Entry Life register	all	9-bit value	Sets the length of the entry life.	3

MANUAL AGING

REGISTER	BIT	VALUE	DESCRIPTION	NOTES
Configuration register	13	0	Turns off the auto-aging function.	
Configuration register	18	Don't care	Sets the priority of auto-aging. Because auto-aging is off, this can be either 1 or 0.	
Auto-age Interval register	all	Don't care	Sets the time between purges. Because auto-aging is off, this can be left with its initial value.	
Learned Entry Life register	all	9-bit value	Sets the length of the entry life.	3

NOTES:

1. When the auto-aging function is chosen, the priority must be set. At some point the auto-age logic may want to operate at the same time as the synchronous or processor ports. Setting bit 18 of the Configuration register to 1 gives the auto-age function higher priority. In a typical switch application this bit is set to 0 to give the function the lowest priority.
2. The Auto-age Interval register requires a 32-bit value to set the interval between purges or aging. This value is divided by the system input CLK signal to produce the time interval. The MUAA RCP resets with the value 02FAF080h in this register. If a 50MHz CLK signal is used, the aging interval is 1 second.
3. The Learned Entry Life register requires a 9-bit value to set the length of time each entry will exist in the address database if it is not updated. This is effectively the number of aging intervals that occur before the entry expires. The life of each entry restarts every time it is "touched" or the time stamp is updated. The system resets with the value 012Ch, giving each entry a life of 300 seconds with a 1 second aging interval.

Table 2: Register Settings for Automatic and Manual Aging

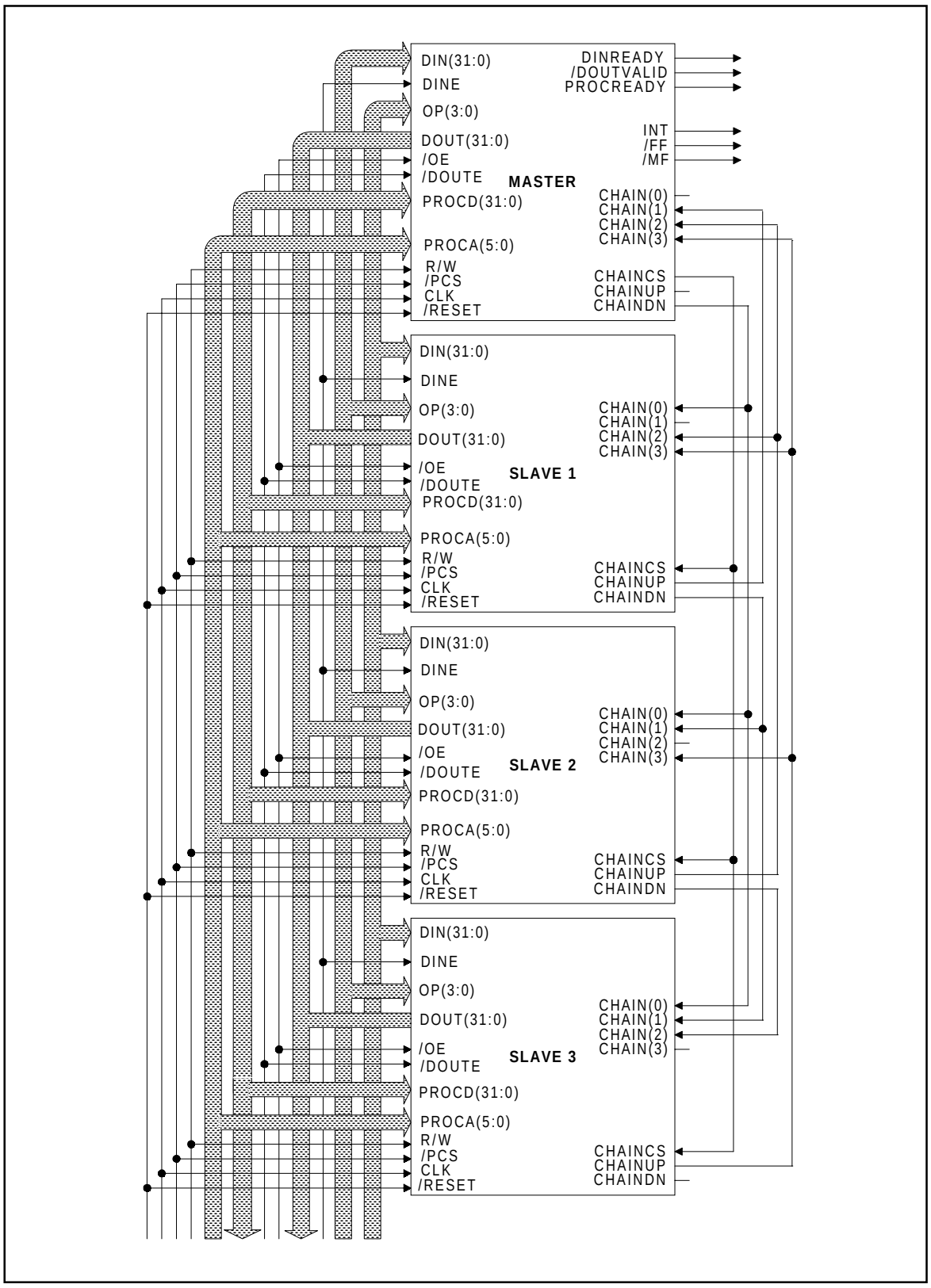

Figure 4: Four Devices Chained Together

Dynamic Aging

If automatic aging is required, the aging interval and time stamp resolution are set at configuration. The settings remain the same regardless of how busy the switch is. This means that the system is configured from an initial assumption of the possible traffic conditions. If this assumption is not accurate, the automatic aging facility may be deleting entries too frequently or not quickly enough.

The solution to this problem is to use a dynamic aging scheme. A processor monitors the speed at which the device is learning SAs and alters the auto-age interval to suit. Dynamic aging allows the MUAA to store a consistent number of entries. Problems caused by the MUAA being completely filled or emptied become less likely. To incorporate dynamic aging, the LQUEUE facility is used. The MUAA RCP places all learned entries in a "learned queue" or LQUEUE. This allows the system to keep track of entries as they are learned and added to memory array. As entries are learned they are tagged as being in the LQUEUE. They can be read from the queue by using the read LQUEUE instruction. This instruction reads the next entry from the queue and makes it available on the processor port.

To incorporate dynamic aging, the system must keep track of the number of entries being learned during each learning interval. Therefore, at the end of each period, the LQUEUE entries are all read out using successive read LQUEUE instructions and the total number is noted. As traffic through the switch increases or decreases, the number of learned entries varies accordingly.

The total number of learned entries after each interval can then be used to alter the aging interval. As the number of learned entries increases, the aging interval can be reduced. Conversely, the interval can be increased as a fall in network traffic causes the number of learned entries to decrease. If the auto-age function is selected, the Auto-age Interval register can be altered to age out older entries more quickly or slowly than before. If aging is done manually by the processor, the time between aging commands can be varied.

DEVICE CHAINING

The MUAA RCPs consist of 2K, 4K, and 8K deep devices. If an application requires a table that is larger than 8K, it is possible to chain MUAAs together transparently. The MUAA RCP is designed so that up to four can be connected together to provide a total memory depth of 32K without the need for any external logic. Figure 4 shows how four devices can be connected. Each MUAA has a set of I/O pins that allow them to be connected in a chain.

The pins are CHAIN(3:0), CHAINCS, CHAINUP, and CHAINDN. Only the device known as the MASTER provides the DOUTVALID, PROCREADY, INT, /FF, and /MF outputs. They are left unconnected on the other MUAAs. This scheme allows devices to be designed and included on the PCB but not fitted. The fit order is MASTER, SLAVE1, SLAVE2, SLAVE3. The devices can be chained with no need to worry about timing delays caused by internal flag ripple-through. Using a single MUAA RCP is the same as using four of them chained together. If the application requires a table depth greater than 32K, external logic is required to do parallel operations on each group of up to four MUAAs.

Routine	Priority	Performed By
Configuration	Power up only	Processor
DA Forwarding	Priority 1	FSM in controller
SA Learning	Priority 2	FSM in controller
Aging-out Older Entries	Priority 3	Processor (if auto-age not selected)

Table 3: MUAA Routines

MUAA ROUTINES

Table 3 shows the control routines needed for the functions performed in a typical switch. It shows the relevant priorities of each of the operations needed with reference to the switch diagram in Figure 2.

Tables 4 through 8 show the instructions needed for: RCP configuration, DA lookup, SA learning, Manual aging of older entries, and Dynamic aging. The system can age out older entries automatically but it also can be done manually by the processor if needed. This routine is shown although the configuration specified in Table 4 is for auto-aging.

Configuration (performed by processor)

Table 4 shows the code required to configure the MUAA RCP in the following way:

- 32-bit I/O for both the synchronous and processor ports.
- 48 bits CAM and 32 bits RAM. This allows the MAC addresses to be loaded in two clock cycles and the associated data read out in one clock cycle.
- /DOUTVALID is on the same CLK signal as the data.
- INT is active HIGH.
- LQUEUE, AQUEUE, PWEX, and SWEX interrupts are enabled.
- The auto-aging function is enabled.
- AQUEUE is enabled.
- The synchronous port is given higher priority than the processor port.
- The port address index is available to be read from DOUT after the associated data.
- The auto-aging function has the lowest priority during internal arbitration.

DA Forwarding (performed by FSM)

The FSM performs the DA forwarding routine described in Table 5. This involves loading the synchronous port of the MUAA with the DA of a recently received frame. It is done in two steps:

1. The least significant 32 bits of the DA are loaded using the load<DIN> instruction.
2. The remaining 16 bits are placed on DIN and the learn<DIN> instruction is issued.

The 48-bit DA is compared with the 48-bit CAM partition of the address database.

If the search finds a MATCH, (/MF is LOW) the associated data (port ID) can be read from DOUT to route the frame to the correct port. If the result of the search is a NO-MATCH, (/MF is HIGH) the frame is forwarded to all ports except the receiving port. The search operation takes three CLK periods after the instruction is given. The /DOUTVALID output is asserted LOW for one CLK period when the results are available.

LINE	/PCS	R/W	PROCA (5:1)	PROCD (31:0)	DESCRIPTION
1	L	L	10000b	00007F10h	Set up Configuration register
2	L	L	11000b	017D7840h	Sets Auto-age Interval register as 0.5 seconds at 50MHz
3	L	L	11001b	00000064h	Sets up the Learned Entry Life register as 100 decimal

Table 4: Signal Values to Configure System

LINE	MNEMONIC	/DINE	OP(3:0)	DIN(31:0)	DESCRIPTION	NOTES
1	Load<DIN>	L	0001b	DA bits <31:0>	Load DIN with first 32 bits of DA	
2	Search<DIN>	L	0011b	DA bits <47:32>	Load DIN with remaining DA bits and initiate a search of the CAM partition of the address database for DA MATCH.	1

If /MF is asserted to indicate a MATCH and /DOUTVALID is asserted to indicate valid data, the following reads the associated data (and address index if required):

LINE	/OE	/DINE	/DOUTE	DOUT(31:0)	DESCRIPTION	NOTES
3	L	H	H	assoc. data <31:0>	Read the associated data from MATCH location to find port ID etc.	2
4	L	H	L	index <31:0>	Read the address index if required.	3

NOTES:

1. Bits <47:32> of the DA are placed on DIN [15:0]. DIN [31:16] bits are discarded and not used during the search.
2. The MUAA RCP is configured during the configuration routine to place the associated data (RAM partition) onto the DOUT pins before the address index. This can be configured to allow the address index to be placed onto the pins first.
3. /DOUTE is asserted LOW by the host system to clock out the next word from the MUAA. In this case it is the address index of the matched location. If the address index is not required, line 4 can be ignored.

Table 5: Signal Values to Search MUAA for DA and Read Results

SA Learn (performed by FSM)

The FSM performs the SA learn routine described in Table 6. This involves loading the synchronous port of the MUAA with the SA of a recently received frame. It is done in three steps:

1. Load bits <31:0> of the SA.
2. Load the remaining 16 bits of the SA on bits <15:0> of the DIN input. The most significant bits on DIN at this point are discarded.
3. Load the associated data and initiate the Learn operation.

The associated data is tailored to suit the individual application but typically it will be the port ID.

The 48-bit SA is compared with the contents of the 48-bit CAM partition of the address database. If the compare results in a MATCH condition, (/MF is LOW) only the 32-bit associated data is written to the memory array and the time stamp is updated automatically. If the compare results in a NO-MATCH (/MF is HIGH), both the 48-bit SA and the 32 associated data is written to the next free address. The time stamp is added automatically. The SA Learn operation takes four CLK periods after the instruction is given. The /DOUTVALID output asserted when the results are available.

Aging Out Older Entries (performed by processor if manual aging is selected)

The MUAA RCP can age out older entries automatically when the auto-age function is selected. When this option is not configured, the processor can perform the task instead. Table 7 shows the instruction required to remove old entries from the address database. This reduces the required RCP size and also simplifies network management by removing nodes from the forwarding table that are no longer active. When the command is given, all non-permanent entries are compared with the age counter. The resulting MATCH locations are deleted (or added to the AQUEUE) and therefore aged out or purged.

LINE	MNEMONIC	/DINE	OP(3:0)	DIN(31:0)	DESCRIPTION	NOTES
1	Load<DIN>	L	0001b	SA bits <31:0>	Load DIN with first 32 bits of SA	
2	Load<DIN>	L	0001b	SA bits <47:32>	Load DIN with last 16 bits of SA	1
3	Learn<DIN>	L	0101b	associated data	Load DIN with the 32 bits of associated data and add all 80 bits to address database	2

NOTES:

1. Bits <47:32> of the SA are placed on DIN [15:0]. DIN [31:16] bits are discarded and not used during the load.
2. The 32 bits placed on DIN are used for associated data. They are written to the RAM partition of the address database and are used for the port ID and any other relevant information.

Table 6: Signal Values to Add a SA to the MUAA

LINE	/PCS	R/W	PROCA (5:1)	PROCD (31:0)	DESCRIPTION
1	L	L	00111b	xxxxxxxxh	Age out older entries

NOTES:

1. xxxxxxxxh = “Don’t care”

Table 7: Signal Values to Purge Older Entries

Dynamic Aging (performed by processor)

The MUAA RCP can be used with a dynamic aging scheme. If the older entries are removed from the address database too quickly or slowly, the interval between deletions can be altered dynamically. The processor must read the contents of the LQUEUE at regular intervals to count the number of entries being learned. From this information, the value in the Age Interval register can be increased or decreased.

Table 8 shows the instructions required to read an entry from the LQUEUE. Line 1 initiates a data read from the LQUEUE. If the MUAA RCP is unable to allow the processor access at this time, it holds PROCREADY LOW. When the cycle can be completed, the MUAA RCP asserts PROCREADY HIGH. The least significant 32 bits of the CAM partition of the first entry in the queue is placed on the bus. Please note that the entries are read from the LQUEUE in internal priority order, lowest address first and not in the order that they were learned. Line 2 reads the next 16 bits of the CAM partition. They are returned on bits 15–0 of the PROCD bus with bits 31–16 being unused. Once the address database is accessed in line 1, the two subsequent reads of the DATA register in lines 2 and 3 can be completed even if the synchronous port is busy. Line 3 completes the LQUEUE read by reading the last 32 bits of the entry (RAM partition).

At this point, when the cycle completes, the processor must increment the counter that it is using to count the number of entries learned. At line 4 the processor reads the LQUEUE Flag register. This indicates if the queue has at least one entry left. If the flag is set to 1 (entries remaining), the processor continues to read the next entries until they all have been read. Repeating lines 1 through 4 until the flag is set to 0 does this. When all the entries in the LQUEUE are read, the processor knows how many entries were learned during the most recent dynamic aging period. This can be compared with previous numbers and the aging interval can be altered to suit.

LINE	/PCS	R/W	PROCA (5:1)	PROCD (31:0)	DESCRIPTION
1	L	H	01011b	bits <31:0>(CAM)	Read LQUEUE register
2	L	H	01101b	bits <47:32>(CAM)	Read DATA register
3	L	H	01101b	bits <79:48>(RAM)	Read DATA register
4	L	H	10100b	bit 0	Read LQUEUE Flag register

NOTES:
Please refer to the MUAA Routing CoProcessor (RCP) datasheet for the processor port read cycle timing.

Table 8: Signal Values to Read One Entry from LQUEUE

NOTES

MUSIC Semiconductors Agent or Distributor:

MUSIC SEMICONDUCTORS

http://www.music-ic.com
email: info@music-ic.com

USA Headquarters
MUSIC Semiconductors
254 B Mountain Avenue
Hackettstown, New Jersey 07840
USA
Tel: 908/979-1010
Fax: 908/979-1035
USA Only: 800/933-1550 Tech. Support
888/226-6874 Product Info.

Asian Headquarters
MUSIC Semiconductors
Special Export Processing Zone 1
Carmelray Industrial Park
Canlubang, Calamba, Laguna
Philippines
Tel: +63 49 549 1480
Fax: +63 49 549 1023
Sales Tel/Fax: +632 723 62 15

European Headquarters
MUSIC Semiconductors
Torenstraat 28
6471 JX Eygelshoven
Netherlands
Tel: +31 45 5462177
Fax: +31 45 5463663