

# NSBMC290™ Burst Mode Memory Controller—User Application Note

National Semiconductor  
Application Note 882  
July 1993



## TABLE OF CONTENTS

### 1.0 DISTINCTIVE FEATURES

- 1.1 Impact on System Design
- 1.2 Am29000 Access Protocol Support
- 1.3 Memory Interface Support
- 1.4 Bus Buffer Strategies

### 2.0 LOCAL CHANNEL PROTOCOL SUPPORT

- 2.1 Simple Access
- 2.2 Pipelined Access
- 2.3 Burst Access
  - 2.3.1 Establishing Burst Mode Access
  - 2.3.2 Burst Suspension
  - 2.3.3 Burst Pause
  - 2.3.4 Terminating a Burst Access

### 3.0 MEMORY REQUIREMENTS

- 3.1 Memory Refresh
- 3.2 Memory Performance Requirements
  - 3.2.1 RAS Pre-charge Time
  - 3.2.2 Access Time from RAS
  - 3.2.3 CAS Pre-charge Time
  - 3.2.4 Access Time from CAS
- 3.3 Write Cycle Restrictions

### 4.0 SYSTEM INTERCONNECT

- 4.1 Signal Description
  - 4.1.1 Local Channel Interface
  - 4.1.2 Memory Interface
  - 4.1.3 Buffer Control Signals
    - 4.1.3.1 Control Signal Interpretation—Mode 0
    - 4.1.3.2 Control Signal Interpretation—Mode 1
    - 4.1.3.3 Control Signal Interpretation—Mode 2
    - 4.1.3.4 Control Signal Interpretation—Mode 3
- 4.2 Bus Buffer Interconnect Strategies
  - 4.2.1 Nibble Wide Memories with “245” Style Buffers
  - 4.2.2 Bit Wide Memories with “245” Style Buffers
  - 4.2.3 Enhanced Performance using Nibble Wide Devices
  - 4.2.4 Single Cycle Burst Write with Bit Wide Devices
  - 4.2.5 Using Bit Wide Devices with the Am29C983
  - 4.2.6 Expanding Memory Size

### 5.0 SOFTWARE DESIGN

- 5.1 General Considerations
- 5.2 Dynamic separation of Instruction and Data spaces
- 5.3 Software Configuration of the NSBMC290

### 6.0 APPENDIX 1—SAMPLE CONFIGURATION PROGRAM

### 7.0 APPENDIX 2—NSBMC290 TO Am29000 INTERCONNECT (PGA)

### 8.0 APPENDIX 3—PIN LIST SORTED BY SIGNAL NAME (PGA)

### 9.0 APPENDIX 4—PIN LIST SORTED BY PIN NUMBER (PGA)

### 10.0 APPENDIX 5—QUAD FLATPAK PIN-OUT

### 11.0 APPENDIX 6—RECOMMENDED SYSTEM CLOCK GENERATOR

#### 1.0 DISTINCTIVE FEATURES

The NSBMC290 is a memory controller designed specifically for use in Am29000 based systems. The NSBMC290 is functionally equivalent to the V29BMC. The specific nature of such systems preclude the NSBMC290 from being a general purpose device. However, within the constraints of the targeted applications, it imposes few architectural restrictions. Every effort has been made to preserve application flexibility while still achieving the principal design objective—support of the Local Channel Protocol.

#### 1.1 Impact on System Design

By supporting both the Local Channel and fast page mode memory protocols, the NSBMC290 enables the system designer to incorporate very large memory sub-systems into high speed processing systems. The fact that little performance penalty is incurred is a key feature. In many application areas such as intelligent peripheral controllers, high speed communication controllers, test instrumentation and high performance work-stations, the use of large memory for code and/or data storage is not only desirable, but very necessary.

Given the low cost per bit of dynamic RAMs, total system cost can be maintained at levels approaching those of currently popular systems that offer performance rates lower by a factor of five or more. Cost/performance ratios of this magnitude are not possible if lower density static memory is substituted, or alternate implementations using power and space consumptive discrete devices are used.

#### 1.2 Am29000 Access Protocol Support

The claim of support of the Am29000 local channel protocol is based on two assertions. The first is that the NSBMC290 implements the full channel protocol described in the “Am29000 Streamlined Instruction Processor Users Manual”. All access methods are supported.

TRI-STATE® is a registered trademark of National Semiconductor Corporation.  
NSBMC290 is functionally equivalent to the V29BMC.  
V29BMC™ is a trademark of V3 Corporation.

Secondly, the interconnect to the Local Channel interface is direct. The Local Channel naming convention has been used in designating the pins related to this interface. In order to connect the NSBMC290 to the Am29000 one simply connects like named signal pins together. An example of the manner in which these signals may be routed is shown in Appendix 2.

### 1.3 Memory Interface Support

Support for dynamic RAM fast page mode protocols is no less complete than that of the Am29000 Local Channel protocol. All outputs to the memory array have been designed with high current drive in order to avoid the necessity of external drivers. In addition, care has been taken during design to minimize problems associated with ground bounce and Simultaneously Switching Outputs.

### 1.4 Bus Buffer Strategies

The organization of Instruction and Data busses are very much a function of system target performance and cost. The designers of lower cost systems may elect to not retain instruction and data bus separation and be willing to accept lowered performance levels in exchange. Similarly, memory speed (and cost) can be traded against system clock rate. The architecture of the NSBMC290 allows the system designer to exercise these options in order to achieve a desired cost/performance ratio.

## 2.0 LOCAL CHANNEL PROTOCOL SUPPORT

The three memory and peripheral access methods specified by the Am29000 local channel protocol are:

- Simple
- Pipelined
- Burst

All of these access methods are defined in an orthogonal manner for both instruction and data accesses.

This section focuses on the way in which the NSBMC290 implements these access methods, individually, and in combination with each other.

### 2.1 Simple Access

The Simple Access method is similar to that characteristic of a conventional processor to memory interface with synchronous hand shake. It is a baseline access type and represents the starting point of all other access methods.

A simple access begins when a request strobe (IREQ or DREQ) is asserted while the address bus presents a value within the range for which a NSBMC290 has been configured. It is additionally required that the option field (OPT2..0) contain a value in the range from 0 to 2 and DREQT1..0 = 0 for data access, or IREQT = 0 for instruction access. If all of these conditions are met, selection of the NSBMC290 occurs, and a memory access begins.

Figure 1 diagrams the sequence of events. After the NSBMC290 has been selected, a row address is transferred to the address lines of the appropriate bank, and RAS for that bank is asserted. Memory bank selection is based on the state of address line 2 (A2) of the processor address bus. The interleave factor of the memory is two. The bank of memory selected when A2 = 0 is referred to as Bank A, the other (A2 = 1) as Bank B.

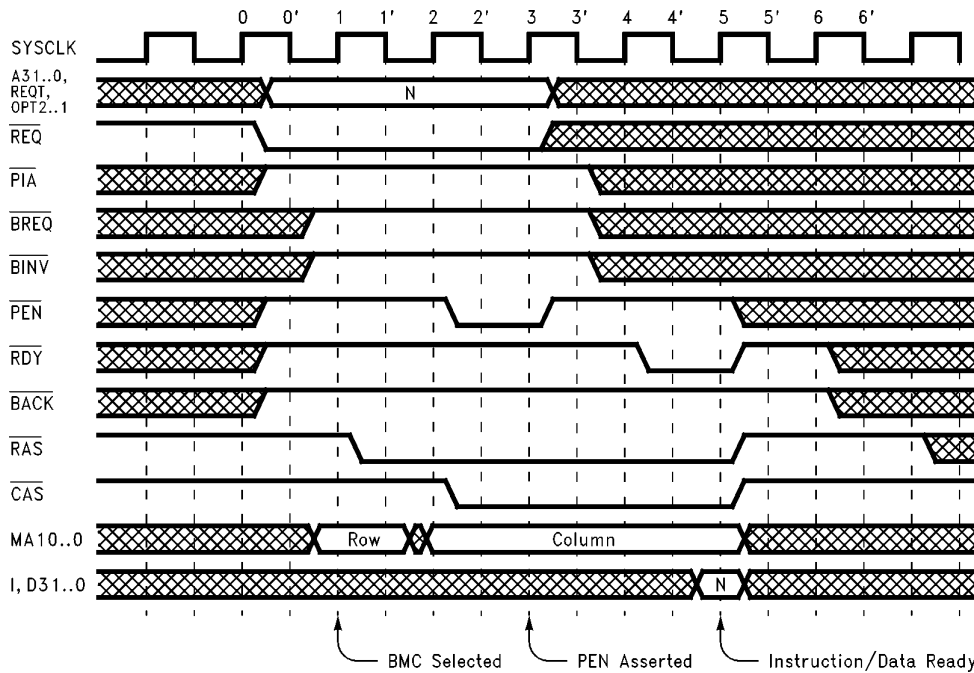


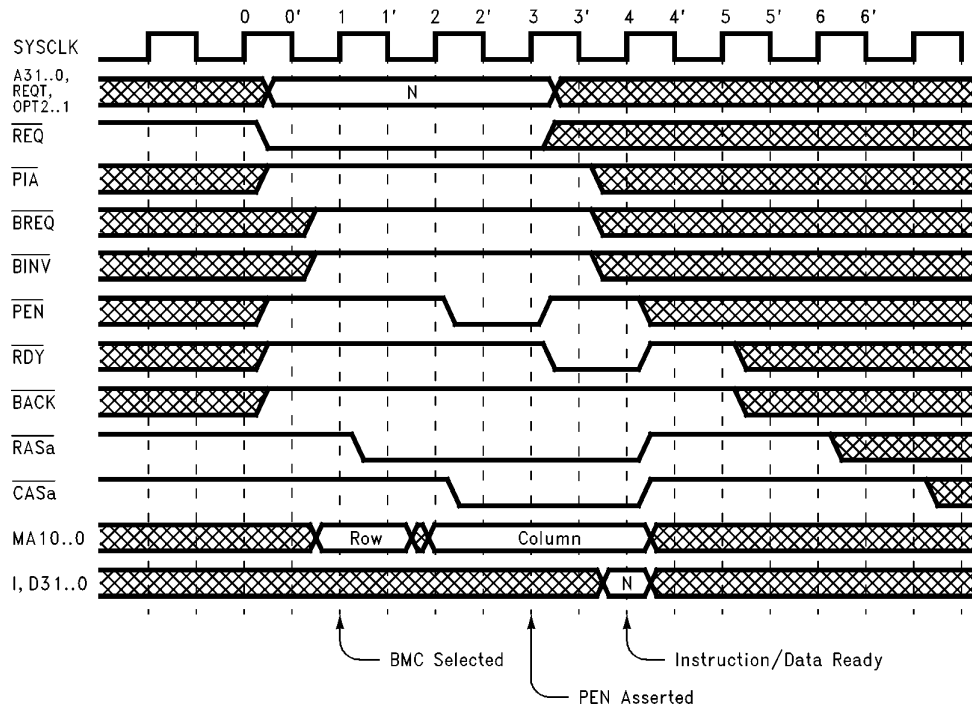
FIGURE 1. Simple Access with Four Clock RAS Cycle

TL/V/11802-1

If an access request immediately follows a cycle in which RAS has been active, additional wait states will be inserted to guarantee RAS de-assertion for a minimum of 2 clocks. This ensures that the RAS pre-charge time is not violated.

After strobing the memory column address by CAS, the NSBMC290 no longer requires the processor address. The PEN signal is asserted to indicate that the address bus is available for use elsewhere. A Simple Cycle completes when the NSBMC290 returns a ready strobe to the processor (IRDY for instruction access, DRDY for data access).

The overall cycle time is programmable via configuration bit 17. This bit determines the number of clock cycles for which RAS is asserted. The timing of *Figure 1* reflects the sequence of events that occur when configuration bit 17 is set to 0 and RAS is asserted for 4 cycles. *Figure 2* shows the same access cycle when 3 clock RAS cycle is selected (bit 17 set to 1). The timing remains the same as *Figure 1* except that (I, D)RDY is returned one cycle earlier.



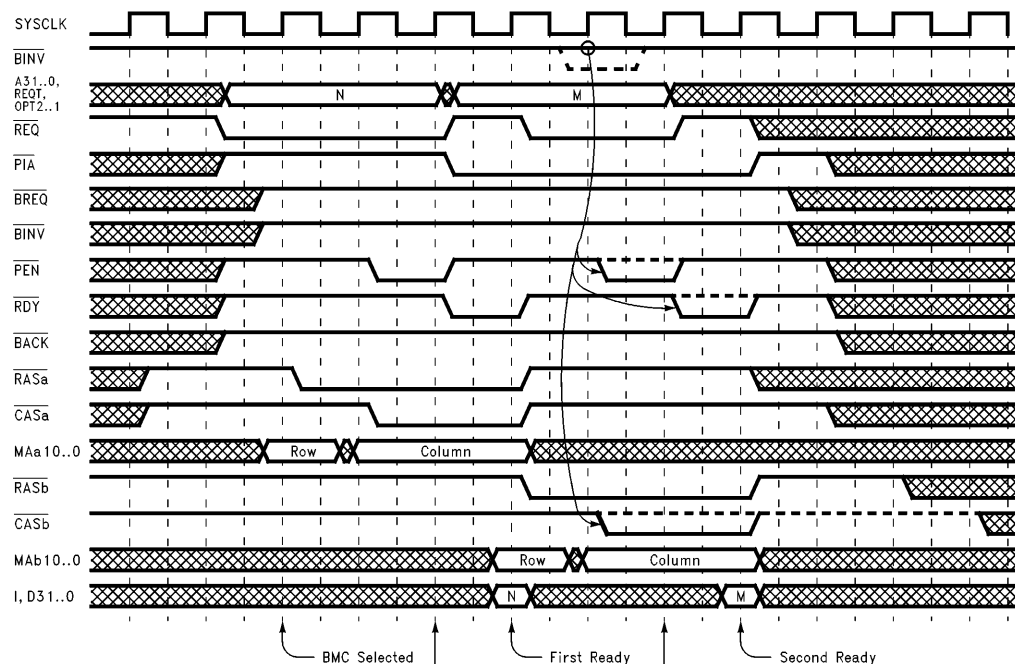
TL/V/11802-2

**FIGURE 2. Simple Access with Three Clock RAS Cycle**

## 2.2 Pipelined Access

The NSBMC290 asserts PEN one cycle after a primary access has been initiated. Another access can begin immediately following the assertion of PEN. When the primary access completes (I/DRDY asserted), the pipelined access becomes the primary access and the pipeline process can start again. If the pipelined access is to the same memory block but to a memory bank alternate to the first (A followed by B or vice versa), the two accesses can overlap as shown in *Figure 3*.

*Figure 3*, shows a pipelined access that begins with a simple access to memory Bank A. After PEN is asserted for the first time, an access to memory Bank B starts (before the access to Bank A completes). The timing diagram is drawn with the assumption that configuration bit 17 set to 1—three cycle RAS access. If four cycle RAS mode is selected, RDY assertion is delayed by one cycle. If the pipelined access were to address memory Bank A, a RAS pre-charge time of two cycles would be inserted between accesses.



TL/V/11802-3

FIGURE 3. Pipelined Access on Alternate Memory Bank

### 2.3 Burst Access

The burst access method of the Am29000 local channel protocol is the most important of all the methods specified if near theoretical performance is to be achieved. It is also, by far, the most difficult to implement. The NSBMC290 fully supports the slave device specification of this access method for both instruction and data accesses.

During execution of a burst access, the state machine of the NSBMC290 cycles through the following phases:

- Established:** The processor and NSBMC290 have successfully initiated a burst access. The initial primary access has been completed and burst start validated.
- Active:** During this phase data or instruction transactions occur every clock cycle as long as IBREQ or DBREQ remains asserted.

- Suspended:** Burst access remains established but no instruction/data transfers occur. This phase is processor initiated and is concluded either by the re-assertion of the IBREQ/DBREQ signal or the assertion of I/DREQ.
- Paused:** This is a suspended phase that is initiated by the NSBMC290. It is entered only during two clock write cycles in order to extend data hold times. Burst access remains established.
- Preempted:** The NSBMC290 will preempt an active or suspended burst only if a long burst delays a memory refresh access for longer than 80% of the refresh period.
- Terminated:** All required burst access cycles are complete and a new access is requested. This operation is initiated by the processor.

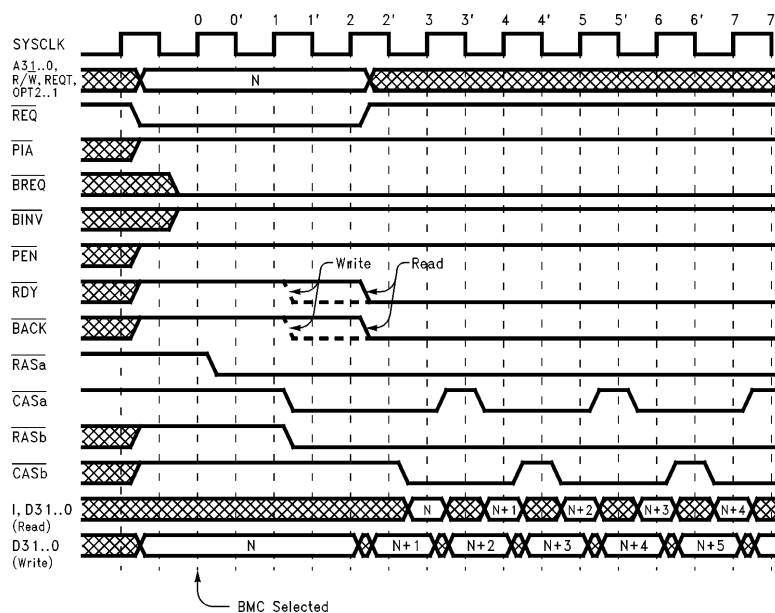


FIGURE 4. Establishing Burst Access

TL/V/11802-4

### 2.3.1 Establishing Burst Mode Access

If the NSBMC290 is selected as for a simple access and the (I,D)BREQ signal is asserted, burst access establishment begins. See Figure 4. The device, however, is not limited to simple access startup. Burst establishment can also be initiated from a pipeline access in progress if the processor asserts (I,D)BREQ. The complete set of conditions that must be met before time T0 are:

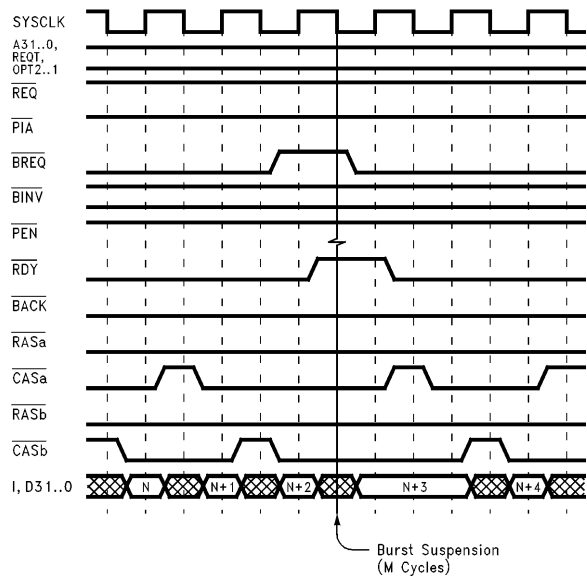
- An address on the address bus in range of the configuration address and block size.
- A request signal with an accompanying burst request (IREQ, IBREQ or DREQ, DBREQ).
- A OPT field value of 0 for data access (indicating word length access)
- A request type of 0 (DREQ = 0 for data access, IREQT = 0 for Instruction access).

Immediately following T0, RAS for the memory bank containing the initial address is asserted. Following T0', the row addresses are released and the column addresses are as-

serted. At T1 the CAS strobe of this bank begins while RAS for the opposite bank is strobed. The request is acknowledged by the NSBMC290 during T2 and the (I,D)BACK signal is asserted prior to this time. The establishment phase is now complete and the processor address bus is now available for other (overlapping) transactions.

Both instruction and data reads operate in identical fashion during the active phase of burst access. During burst reads, data from the memory is not valid until some time (memory access delay) after the CAS strobe has been asserted. The RDY signal is delayed to reflect this requirement.

Write operations (which only occur during data accesses) require different cycle timing for DRDY. During the establishment phase of burst data writes, DRDY can be asserted earlier since the data to be written (to memory) is valid when the CAS strobe is asserted. By using the configuration bits to adjust both the RAS cycle time and the burst write speed, this type of cycle can be modified in several ways. These are detailed in Section 3.



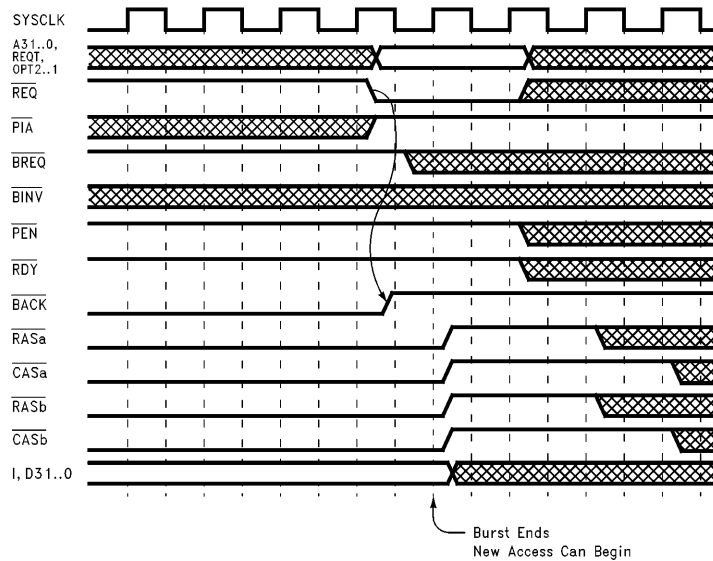
TL/V/11802-5

**FIGURE 5. Burst Mode Suspension**

### 2.3.2 Burst Suspension

When the processor de-asserts the burst request signal (IBREQ or DBREQ) during an active burst access then the NSBMC290 enters its suspended state. During this period, the memory control signals and address maintain the values held previous to the suspension and the ready signal is

deasserted (IRDY or DRDY). The suspended burst access is reactivated if the processor re-asserts the burst request (Figure 5). Designers using processors that predate Revision "C" should note the errata on this subject and implement the suggested change that forces a suspension to become a termination.



TL/V/11802-6

**FIGURE 6. Burst Suspended then Terminated by Master**

### 2.3.3 Burst Pause

When two cycle burst data write is selected at configuration, the NSBMC290 uses this state to extend the data hold period for slower memories. This type of bus transaction is implemented by deasserting DRDY for one clock cycle. As long as DRDY remains deasserted, the Am29000 considers the transaction incomplete and extends the cycle period. This signal is sampled by the processor at the rising edge of SYSCLK and is asserted/deasserted by the NSBMC290 to meet the required setup time at the processor.

### 2.3.4 Terminating a Burst Access

The processor terminates an active burst by asserting a new request. This can occur from either the active or the suspended phases. *Figure 6* shows a suspended burst access that is terminated when REQst is asserted. When termination occurs, the NSBMC290 immediately deasserts burst acknowledge (IBACK or DBACK).

## 3.0 MEMORY REQUIREMENTS

This section provides the calculations required to determine the DRAM speed requirements for a given memory configuration based on the NSBMC290 controller. The access speed required for system DRAM is a function of processor clock speed, processor data setup/hold times, bus buffer delays, NSBMC290 I/O delays and operating mode.

### 3.1 Memory Refresh

A RAS only refresh cycle, is performed at a rate determined by an internal refresh counter. When the counter decrements to its terminal count value, the NSBMC290 will start the refresh memory cycle as soon as an idle period is available. The refresh counter, however, does not pause for refresh memory cycle completion. It operates continuously and independently to guarantee the overall device refresh rate. The refresh address generator always maintains a full 11-bit refresh address.

An outstanding refresh request has priority over the initiation of a new access. If a pending refresh is delayed for more than 80% of the refresh period, because of a long burst access, the NSBMC290 will preempt the burst access by deasserting the acknowledge signal (I/DBACK). This forces an idle period and because of its arbitration priority, the refresh cycle will complete. The refresh row address is automatically incremented between refresh cycles.

The refresh period is derived from the system clock and the value programmed into configuration bits [13..8] as:

$$\frac{(\text{Programmed Value} + 1)}{(\text{System Clock Frequency})} - *16 (\mu\text{s per row})$$

The valid range for the programmed value is from 0 to 62. If the value 63 is used refresh cycle execution will be disabled.

### 3.2 Memory Performance Requirements

The frequency and performance of the system influence the selection of the DRAM device. The NSBMC290 only sup-

ports page mode devices. However, since these are the most prevalent types of devices on the market, this is not a constraint.

Particular attention must be paid to the page mode behavior of the memory devices. Most manufacturers offer both "page mode" and "high speed (or fast) page mode" memories. The characteristics of the fast page mode devices are generally those required for reliable operation at speeds of 20 MHz and above. The following sections present the critical performance requirements of memory devices as a function of system operating frequency.

### 3.2.1 RAS Pre-Charge Time

A minimum of two SYSCLK cycles are provided between successive RAS cycles. Therefore the required RAS pre-charge time is given by:

$$t_{RP} \leq 2(t_{SYS}) + t_{RHL} - t_{RLH}$$

Where:  $t_{RP}$  = DRAM RAS Pre-charge Time

$t_{SYS}$  = SYSCLK Cycle Time

$t_{RHL}$  = RAS High to Low Delay

$t_{RLH}$  = RAS Low to High Delay

Because  $t_{RHL} > t_{RLH}$  at all times, this requirement reduces to:  $t_{RP} \leq 2(t_{SYS})$

### 3.2.2 Access Time from RAS

It is possible to control the required RAS access time of the DRAM memory using the NSBMC290 Configuration register (Bit 17). The basic RAS access time is calculated as:

$$t_{RAC} \leq 3 * t_{SYS} - t_{RHL} - t_{BUF} - t_{SU}$$

Where:  $t_{RAC}$  = DRAM RAS Access Time

$t_{SYS}$  = SYSCLK Cycle Time

$t_{RHL}$  = RAS High to Low Delay

$t_{BUF}$  = Buffer Delay

$t_{SU}$  = Am29000 Data/Instruction Set Up Time

When configuration bit 17 is set to 0, an additional SYSCLK cycle is inserted into the RAS cycle and  $t_{RAC}$  is increased by an additional  $t_{SYS}$  period.

### 3.2.3 CAS Pre-Charge Time

The CAS pre-charge time during page mode access is one of the factors that determines the page mode cycle time. The maximum permissible value is given as:

$$t_{CP} \leq \frac{1}{2}(t_{SYS}) + t_{CHL} - t_{CHL}$$

Where:  $t_{CP}$  = DRAM CAS Pre-charge Time

$t_{SYS}$  = SYSCLK Cycle Time

$t_{CHL}$  = CAS High to Low Delay

$t_{CLH}$  = CAS Low to High Delay

Because  $t_{CHL} > t_{CLH}$  at all times, this requirement reduces to:  $t_{CP} \leq \frac{1}{2}(T_{SYS})$

### 3.2.4 Access Time from CAS

The simple or pipelined access time from CAS is 1 clock cycle less than the RAS access time. Thus the required CAS access time is given as:

$$t_{CAC} \leq 2 * t_{SYS} - t_{CHL} - t_{BUF} - t_{SU}$$

Where:  $t_{CAC}$  = DRAM Page Mode CAS Access Time

$t_{SYS}$  = SYSCLK Cycle Time

$t_{CHL}$  = CAS High to Low Delay

$t_{BUF}$  = Buffer Delay

$t_{SU}$  = Am29000 Data/Instruction Set up Time

However for burst mode operation the CAS access time decreased to:

$$t_{CAC} \leq 1.5 * t_{SYS} - t_{CHL} - t_{BUF} - t_{SU}$$

Where:  $t_{CHL}$  is taken from the burst mode timing parameter

### 3.3 Write Cycle Restrictions

During a burst write cycle the data hold time of the DRAM must be considered. Typically the DRAM requires a non-zero data hold time following the assertion of the CAS strobe for an early write cycle. Two alternatives exist in meeting the hold time specification:

1. Configure the NSBMC290 or two cycle write, or
2. use registers or latches on the data into the DRAM.

The method chosen will depend on desired cost/performance ratio for a given application. These methods are further described in Section 4.2, Subsections 1 thru 5.

## 4.0 SYSTEM INTERCONNECT

This section describes the connection of the NSBMC290 to both the host Am29000 processor, and the DRAM memory devices. It describes some of the possible buffer strategies for interconnecting the Am29000 data and instruction buses to the memory array inputs and outputs.

### 4.1 Signal Description

The NSBMC290 signals can be subdivided into two distinct categories: processor local channel interface and memory interface signals. A complete listing of these signals and the device pin-outs is presented in the NSBMC290 data sheet as well as Appendices 3 and 4 of this document. What follows is a brief functional description of the major signal groups.

#### 4.1.1 Local Channel Interface

The signals in this group are assigned the same names as their counterparts on the Am29000. They are designed to be connected directly to the Am29000 and the use of multiple (up to 4) NSBMC290 in a system is fully supported. The outputs on the reply signals (I/DBACK, I/DRDY, PEN) have been designed so that they may be simply wire "OR"ed together and connected **directly** to the processor interface. The outputs are TRI-STATE not open collector and require only a nominal pull-up to  $V_{CC}$  of approximately 10 k $\Omega$ .

**Note:** Devices such as the Am29027 Arithmetic Accelerator drive these signals from a non-TRI-STATE output. If they are used in the circuit, it will be necessary to first AND the wire "OR"ed signal from the NSBMC290s with the signal from this device (using a 74F08 for example) and connect the output to the Am29000.

### 4.1.2 Memory Interface

The memory block controlled by the NSBMC290 is organized as two banks of 32 bits each. Parity is not directly supported by the controller, and if parity is implemented the parity error signal should drive the (I,D)ERR on the processor only during access to the associated memory block if an error is detected.

The memory interface outputs support high current drivers that will drive loads to 320 pF per bank. Given a nominal input load of 7 pF per device, and a trace capacitance of 18.5 pF/ft<sup>†</sup>, 36 memory devices per bank (byte parity included) are easily supported. These outputs however must be externally matched to the input impedance of the DRAM memory array. A passive serial or parallel terminating network is all that is required.

### 4.1.3 Buffer Control Signals

The transfer of Instructions and Data from the memory subsystem to the Local Channel occurs through buffers controlled by the NSBMC290. Of the six signals provided for this purpose, four operate in multiple modes; the remaining two (DBLEA,B) have fixed interpretation.

These two signals provide high true transparent latch enable controls for use during data transfers from the Am29000 to memory. They are designed to operate with 74F373 style transparent latches to provide additional data hold time during write operations at high SYSCLK speeds or with slow memories at lower speeds.

The functions performed by the remaining four signals change according to programmed mode. The signal names, as they appear on the logic symbol, reflect the functions performed in operational mode 0. Table I shows the configurable control signals and their assigned names in the various operating modes.

TABLE I. Mode Dependent Buffer Control Signals

| Mode 0 (Default) | Mode 1 | Mode 2   | Mode 3   |
|------------------|--------|----------|----------|
| *DBTxA           | *DBCeA | *DBTx    | *DBCe    |
| *DBTxB           | *DBCeB | BankB/*A | BankB/*A |
| *IBTxA           | *IBTxA | *IBTx    | *IBTx    |
| *IBTxB           | *IBTxB | *IBTx    | *IBTx    |

Signals prefixed by \* are active low.

Modes 0 and 1 are primarily designed for use with bit wide memories. Since these memories do not have output enables, separate buffers for each bank are required. Modes 2 and 3 are designed to take advantage of the output enables on nibble wide memories.

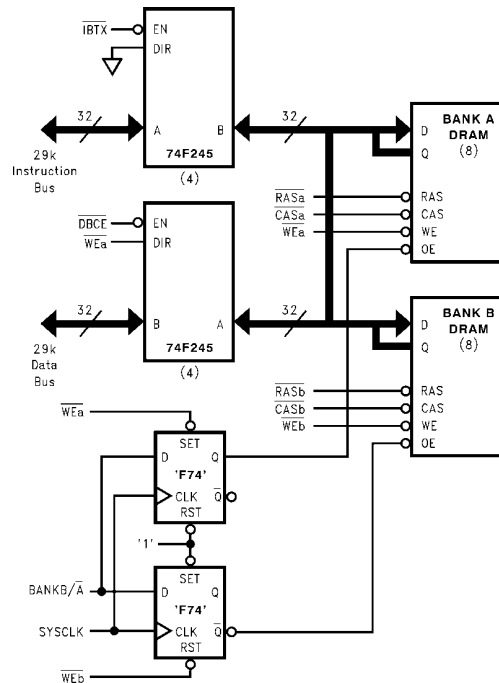
Modes 1 and 3 can be used with buffers which have a chip enable and a direction control. The direction controls are connected to the memory write enables, and the chip enables are connected to the appropriate chip enable signal (\*IBTx(A,B), \*DBCe(A,B)). The write enable controls apply only to data transfers since, by definition, write operations do not occur from the instruction bus.

When devices with select and direction controls are used, care should be taken to connect the I/O ports to the bus so that the low true write enables signals (\*MWe(A,B)) set the data transfer direction from the processor into the memories (e.g., for 74F245's connect port A to the memories).

<sup>†</sup>Ref: "Am29000 Streamlined Instruction Processor Memory Design Handbook".



For modes 2 and 3 the NSBMC290 generates a signal called BANKB/\*A. The high level of this signal indicates that BANK B must be enabled starting at the next rising edge of SYSCLK. Conversely, the low level indicates that BANK A must be enabled starting at the next rising edge of SYSCLK. In order for the memory chip enables for each bank to operate with correct timing, the bank select signal BANKB/\*A delay from SYSCLK must be kept to a minimum. Figure 7 shows how this signal is synchronized to achieve proper control.



**FIGURE 7. NSBMC290 Memory System Using 4 DRAM Devices and 74F245 Buffers**

#### 4.1.3.1 Control Signal Interpretation—Mode 0

This mode is designed to be used with bit wide memories and buffers which have separate output enables for each direction (e.g., Am29861). The bus transmit enable controls for each bank (\*I/DBTx(A,B)) are asserted low for read operations from memory, while the memory bank write enables (\*MWeA/B) are asserted low for write operations. The buffer signals that enable processor to memory transfers must be connected to the memory bank write enables, while those that permit memory to processor transfers are connected to the appropriate transmit enables \*I/DBTx(A,B).

#### 4.1.3.2 Control Signal Interpretation—Mode 1

When configured in this mode, the NSBMC290 asserts the separate data bus chip enable controls for each bank during both read and write operations. The signals \*MWe(A,B) are asserted low only during write operations and determine the direction of data transfers. This configuration mode is appropriate for the direct control of 74F245 style devices that have a single enable and a direction control.

#### 4.1.3.3 Control Signal Interpretation—Mode 2

This configuration supports nibble wide memories and Am29861 style buffers or bit wide memories with Am29C983 four port buffers. The NSBMC290 generates a single transmit enable for Instruction reads and also a single enable for data accesses. The transmit enables are asserted low for read operations to either bank. The memory bank write enables \*MWe(A,B) are asserted low for write operations to either bank and are used as in Mode 0. The BANKB/\*A signal is used to select which bank is to be next accessed for read operations.

#### 4.1.3.4 Control Signal Interpretation—Mode 3

If this mode is selected, a single chip enable is generated for data access to both banks. It is asserted low for both read and write operations to either bank. The BANKB/\*A signal is used to select which bank is read. Both memory bank write enables are asserted low for write operations to either bank.

### 4.2 Bus Buffer Interconnect Strategies

The size, and cost of the memory system design is a function of the types of DRAM chosen and the desired system performance. The sections following expand some of the possible configurations and illustrate them with appropriate diagrams.

#### 4.2.1 Nibble Wide Memories with '245 Style Buffers

If nibble wide DRAM devices are used in conjunction with 74F245 octal buffers, it is possible to design a small, low cost system. If 256k by 4-bit (1 Mbit) devices are used, a memory size of 2 MB results. Because the NSBMC290 accommodates devices up to 16 MB (4M by 4-Bit) in size, the system memory size may be easily expanded.

Figure 7 shows the connection of the control signals to the data/instruction bus buffers. Buffer mode 3 is required for the NSBMC290 at startup. In order to guarantee sufficient data hold time during data burst writes to memory, two SYSCLK cycles are required. The NSBMC290 should be initialized with configuration bit 18 set to 0 and buffer mode 3.

The output enables of the DRAM are driven by signals derived from the BANKB/\*A signal. The write enable strobes from the NSBMC290 are used to disable the output enable during write cycles.

#### 4.2.2 Bit Wide Memories with '245 Style Buffers

Output enables are typically not available with bit wide DRAMs. It is thus necessary to individually buffer each of the two memory bank as illustrated in *Figure 8*. The NSBMC290 must be configured for buffer mode 1 with two

cycle burst write. This configuration is the most simple and inexpensive when using single bit DRAM devices.

The disadvantage of this Strategy is that a large number of buffers are required. Printed circuit routing problems may require additional signal layers or larger PCB area.

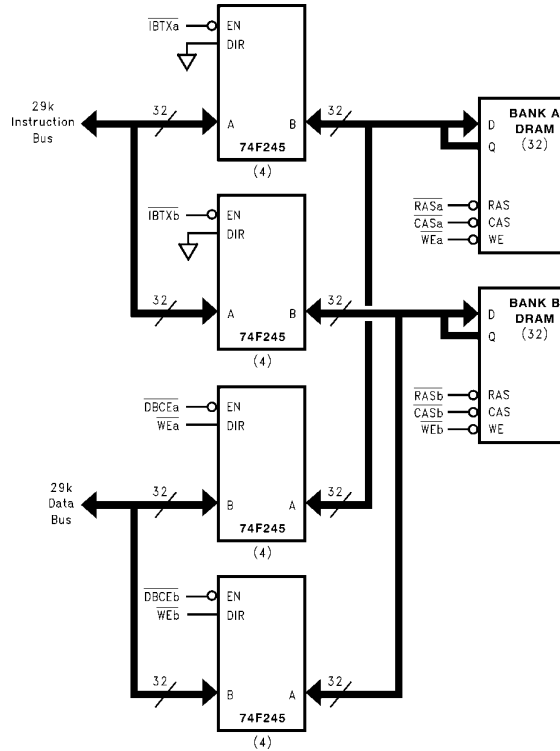


FIGURE 8. NSBMC290 Memory System Using by 1 DRAM Devices and 74F245 Buffers

TL/V/11802-8

#### 4.2.3 Enhanced Performance Using Nibble Wide Devices

If single cycle data burst writes are desired, it is necessary to latch the data from the processor. This is accomplished by substituting 74F646 registered buffers in the data path to replace the 74F245s used in *Figure 8*. In *Figure 10* the 74F646 buffers are connected so that the transparent (un-registered) path is from the memory array to the processor, and the clocked (registered) path is from the processor to the memory array. In this way "245" style response occurs during read cycles, and "374" style response occurs during write cycles. The NSBMC290 should be configured to buffer mode 3 in configuration register bits 14, 15.

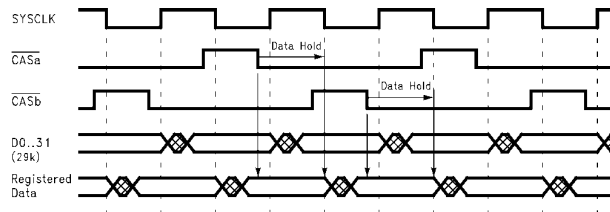
The clock for the registered path is the complement of SYSCLK. The Am29000 asserts data following the rising edge of SYSCLK. This data is then latched on the falling edge of SYSCLK. Simultaneously CAS is asserted for the bank addressed in that cycle. *Figure 9* illustrates the write cycle timing achieved using 74F646 buffers. The data hold time is calculated as:

$$t_{DH} = t_{SYS} - t_{CHL}$$

Where:  $t_{DH}$  = Data Hold Time to DRAM

$t_{SYS}$  = SYSCLK Cycle Time

$t_{CHL}$  = CAS High to Low Delay



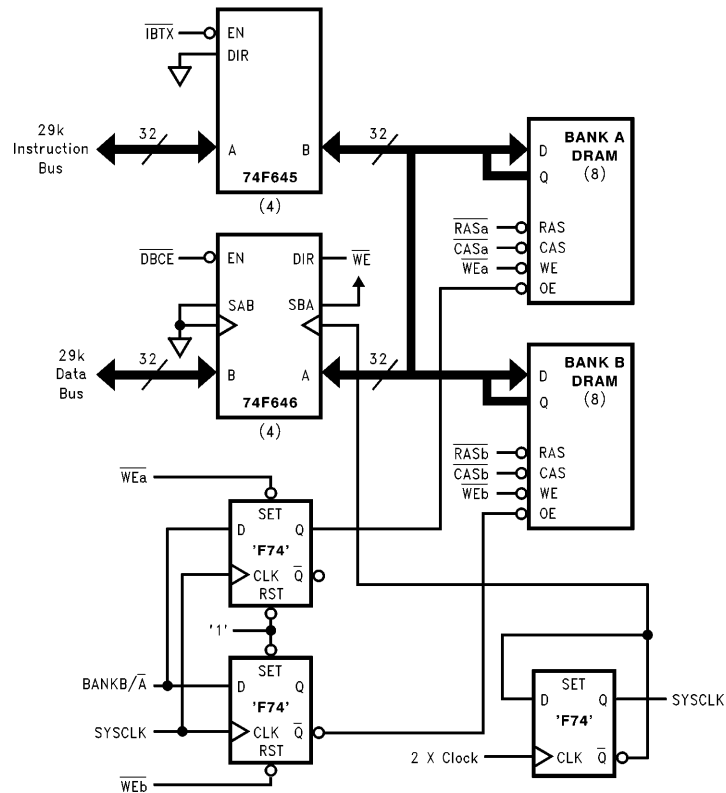
TL/V/11802-9

FIGURE 9. Burst Write Cycle Timing Using 74F646 to Register the Data from the Processor

In order to avoid problems, the memory data should be demultiplexed and transparent latches (such as 74F543) should be used to replace the registered latches and the DBLE(A,B) signals used to control the buffer latch enables. This method will guarantee correct operation as the DBLE(A,B) signals are not asserted until after the falling edge of SYSCLK Setup time violations are thus avoided. The use of transparent latches requires that the data buses for the A and B memory banks be separated. Consequently, this buffer method is best suited for use with bit wide memories although nibble devices will work as well.

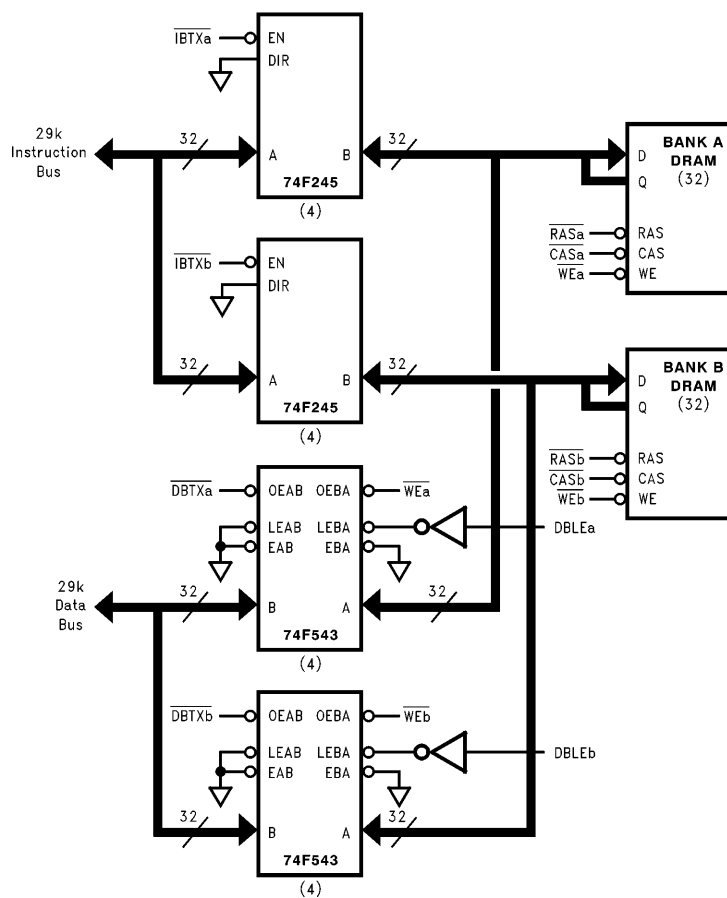
Since bit wide DRAMs lack the fast output enable control found on the nibble wide DRAMs, the data outputs of the two banks must be separated. The buffer control signals from the NSBMC290 require that write operations to memory be overlapped during burst access so that while a write cycle is being completed to one bank of memory, data for the other bank is being latched and written. A transparent latch is used to hold data from the previous cycle while that from the current cycle is being saved for the other bank.

The 74F543 buffer contains two sets of latches that allow for bi-directional latched or transparent data flow. During read operations, the transparent mode is used, during write operations, data is latched. *Figure 11* illustrates the required interconnect for a “mode 0” circuit. Because the latch controls are inverted, it is necessary to invert the control signals from the NSBMC290.



**FIGURE 10. NSBMC290 Memory System with Improved Burst Write Performance Using Nibble Wide DRAM Devices and 74F646 Buffers**

TL/V/11802-10



TL/V/11802-11

**FIGURE 11. NSBMC290 Memory System Using 1X DRAM Devices and 74F543 Buffers for Single Cycle Burst Read and Write Operation**

#### 4.2.5 Using Bit Wide Devices with the Am29C983

The Am29C983 4 port bus exchange buffer has several distinct advantages in a memory system design. The principle advantage is that only four packages are required. This will significantly reduce PCB congestion and lower the board space requirement. Another advantage of the Am29C983 is the 9-bit data path size. The additional bit can be used to transmit byte parity in systems where parity is desired.

Finally, they also provide transparent latch capabilities that enable single cycle operation without violation of memory setup and hold times. In our view these devices offer the most efficient method of obtaining high memory density, low device count, and full operational speed.

In Figure 12 the Am29C983s are configured to operate as non-latching buffers for read operations and as transparent latches during data writes. The latch enable signals indicate when the A/B port output data is to be latched while the write enable signals determine when these outputs are enabled. The BANKB/\*A signal, synchronized to SYSCLK, selects between the A/B ports.

#### 4.2.6 Expanding Memory Size

When using nibble wide memories, it may be desirable to add a second block of memory to NSBMC290. Using 1 me-

gabit (256k by 4) devices the base memory size is 2 MB. If 4 MB is required, then a second NSBMC290 should be used so that split Instruction and data architecture can be implemented. With two NSBMC290s in place, further memory expansion can be accomplished as in Figure 13 using buffer configuration "mode 3".

In Figure 13 the output enables of the four banks of DRAM are controlled with a PAL device (Listing 1). Address line A21 is latched and used to select one of the memory blocks. As a result the NSBMC290 should be initialized to a larger block size. With 256k by 4 memories, a block size of 8 MB should be selected instead of 2 MB. As a result the 4 MB block of memory will be redundantly mapped twice in the 8 MB address space. Note that the write enables are tied together for each block compared to the NSBMC290 which has separate write enables. However the NSBMC290 write enables WEa and WEb are logically identical. Two outputs are generated only for the purpose of distributing the load capacitance.

It is not possible to drive more than one block of bit wide devices per NSBMC290 since the load capacitance would be prohibitive (only 32 devices per bank are allowed).

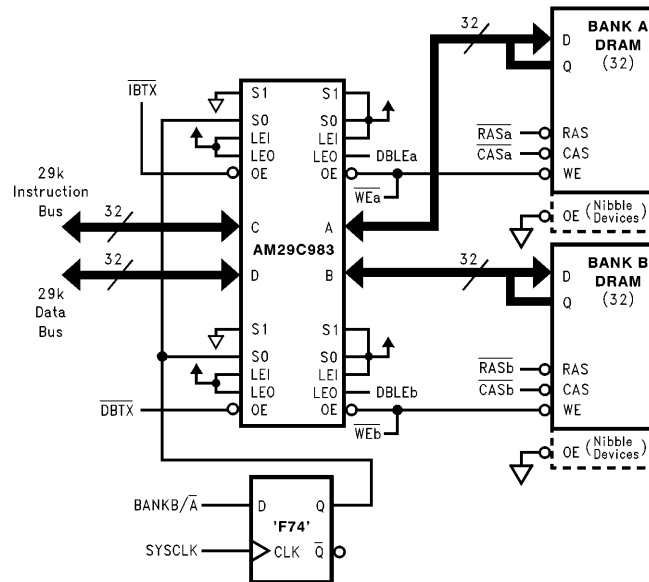
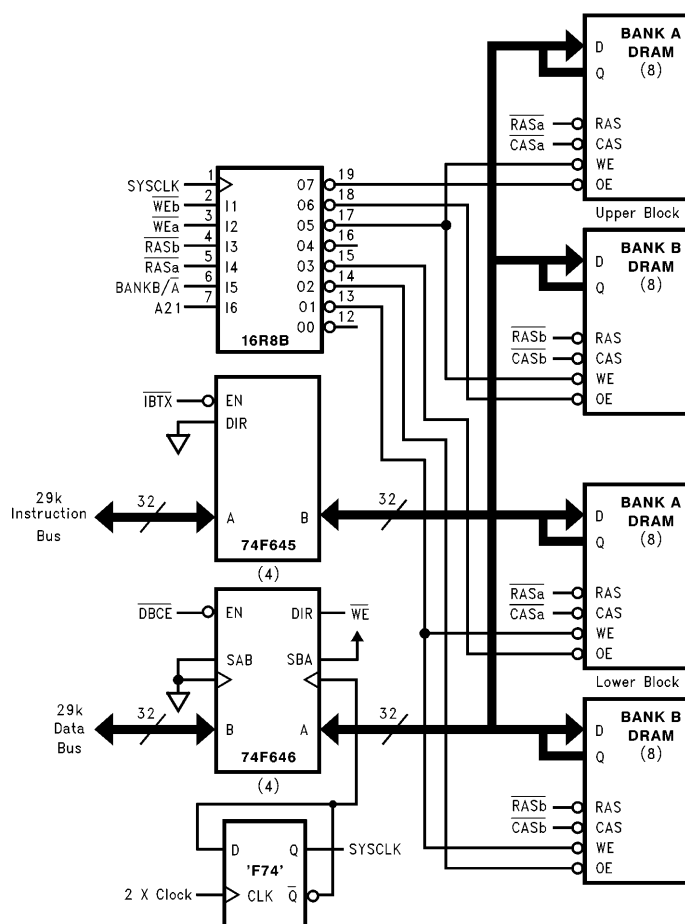


FIGURE 12. Memory System using bit wide DRAMs and Am29C983 buffers. Configuration mode 2, single cycle write.

TL/V/11802-12



TL/V/11802-13

**FIGURE 13. Memory System with Single Cycle Burst Write Performance Using 2 Blocks of Nibble Wide DRAM Devices**

## 5.0 SOFTWARE DESIGN

The ease with which the NSBMC290 may be integrated into a system design has been illustrated in the foregoing sections. The memory sub-systems described support the Am29000 with between 2 MB and 32 MB of memory (depending on the storage devices selected) managed by a single NSBMC290. All examples shown may use 64k, 256k, 1M or 4M nibble or bit wide devices.

### 5.1 General Considerations

Since the NSBMC290 manages both instruction and data accesses to a memory block, minimally one is required. However, the interleaving of instruction and data accesses to a single memory block can result in an increased number of instruction burst restarts compared to designs that separate instruction and data memory blocks. The performance

degradation that results is not severe, but, if maximum performance is absolutely required, the use of two or more NSBMC290s is an effective solution.

In minimal systems of the type shown, performance degradation due to access type interleave can be virtually eliminated by careful software design. Software design, as it relates to the behavior of systems using RISC type processors with near 1 cycle per instruction performance rates, is governed by slightly different parameters from those operative in CISC designs where everything is equally slow.

There are two major architectural constructs that should be taken advantage of in designing software for the Am29000. The first is the large internal register file and its local/global management structure, the second is the branch target cache. These features relate respectively to program data organization and subroutine usage.

```

Name          4meg;
Partno        Control for 4 meg of DRAM;
Date          89/07/17;
Revision      0.0;
Designer
Company
Assembly
Location      U1;

/*****
/* Control logic for 1 - V29BMC controlling 4 meg of 256K*4 DRAM */
*****/
/* Allowable Target Device Types: 16V8, 16R8 */
*****/

/** Inputs **/
Pin 1  = sysclk      ; /* CPU clock */
Pin 2  = !weB        ; /* From V29BMC */
Pin 3  = !weA        ; /*
Pin 4  = !rasB       ; /*
Pin 5  = !rasA       ; /*
Pin 6  = bankB       ; /*
Pin 7  = A21         ; /* From Am29000 */

/** Outputs **/
Pin 19 = !oehB       ; /* OE high memory block B bank */
Pin 18 = !oehA       ; /* OE high memory block A bank */
Pin 17 = !weh        ; /* WE high memory block */
Pin 16 = !high       ; /* High memory block selected */
Pin 15 = !oelB       ; /* OE low memory block B bank */
Pin 14 = !oelA       ; /* OE low memory block A bank */
Pin 13 = !wel        ; /* WE low memory block */
Pin 12 = !low        ; /* Low memory block selected */

/** Declarations and Intermediate Variable Definitions **/

/** Logic Equations **/
high.d = !(rasA # rasB) & A21
        # (rasA # rasB) & high;

low.d  = !(rasA # rasB) & !A21
        # (rasA # rasB) & low;

oelA.d = !weA & !bankB & low;
oelB.d = !weA & bankB & low;
oehA.d = !weA & !bankB & high;
oehB.d = !weA & bankB & high;

wel.d  = (weA # weB) & low;
weh.d  = (weA # weB) & high;

```

**FIGURE 14. Logic Equations of PAL Device Used when the NSBMC290 Controls 2 Blocks of 256\*4 DRAM Devices Resulting in 4 MB Memory System**

TL/V/11802-14

If performance is to be optimized, single external data accesses should be eliminated where possible and replaced by large scale register loads and stores ("spills and fills"). In order to do this, call tree analysis and subroutine flow tracing should be performed with a view to maximizing data locality of reference. While an optimizing compiler can go a long way in maximizing local variable placement, array access, and loop constructs, the responsibility of global or external data placement is the province of the programmer. By supporting burst and pipelined access to data memory, the NSBMC290 promotes the use of locality of reference augmented by register "spill/fill" as a means of enhancing program performance.

The fact that the branch target cache stores the first four instructions at an address jump target and that the NSBMC290 requires five cycles to restart a new burst access, allows the programmer to issue repetitive subroutine call/returns and to implement loops with single cycle penalty. For repetitive subroutine invocation the real performance related issue is data organization, not instruction burst restart. By passing arguments through registers and taking advantage of the global/local register support of the register file manager, throughput can be maintained at near theoretical rates.

## 5.2 Dynamic separation of Instruction and Data spaces

Systems that employ multiple NSBMC290s enjoy the benefit of being able to perform instruction/data partitioning dynamically through the translation look-aside buffer. As long

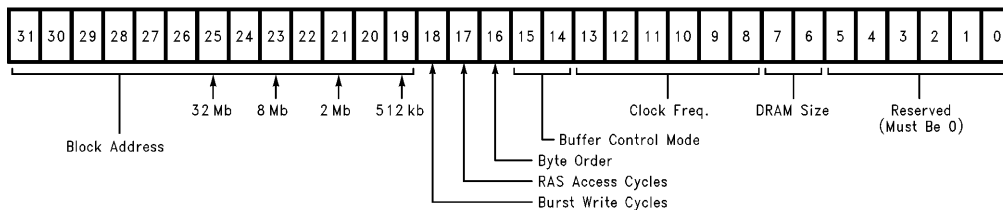
as the page allocation algorithm avoids assigning an active code segment to the same memory block as a simultaneously active data segment, performance is not affected. This rule in conjunction with standard best fit and/or least recently used algorithms makes task and page table management simple and fast.

## 5.3 Software Configuration of the NSBMC290

Configuration of the NSBMC290 occurs on the first 32-bit instruction or data read from an address space other than system ROM after RESET. The example presented assumes that RESET is connected to SYSTEM RESET. This configuration is not mandatory and external logic may be used to separately control RESET to the NSBMC290. All configuration bits are set to zero following a reset.

The values placed in the fields comprising bits 6 thru 18 are completely determined by the hardware implementation. The address assignment is a software option. It is important to remember that the start address of a block is on a boundary that is an integer multiple of the block size.

For example, if two NSBMC290s are used, one of which supports an 8 MB memory block, the other a 2 MB block, the 8 MB block will reside on a 8 MB boundary, and the 2 MB block on a 2 MB boundary. If linear addressing is required, the 8 MB block is assigned to start at 0 and the 2 MB block at 800000(Hex). Alternately, the translation look aside buffer may be programmed to linearize the organization of these address spaces.



TL/V/11802-15



## 6.0 APPENDIX 1—SAMPLE CONFIGURATUION PROGRAM

```

/* File bmc.h -- Define configuration of all BMC's in system */
/*
 * BMC0- configured for:
 *      1M X 1 DRAMs at 100 ns. access time,
 *      Block Address: 0, Memory Block Size: 8M bytes
 *      Byte Order: Little Endian, Buffer Mode: 245 Style (Enable & Direction)
 *      System Clock Freq.: 20 MHz., Mode: 1 cycle burst read and write
 * BMC1 - configured for:
 *      256K X 4 DRAMs at 120 ns. access time,
 *      Block Address: 800000#h, Memory Block Size: 2M bytes
 *      Byte Order: Little Endian, Buffer Mode: 245 Style (Enable & Direction)
 *      System Clock Freq.: 20 MHz., Mode: 1 cycle burst read, 2 cycle burst write
 * BMC2 - configured for:
 *      256K X 4 DRAMs at 100 ns. access time,
 *      Block Address: A00000#h, Memory Block Size: 2M bytes
 *      Byte Order: Big Endian, Buffer Mode: 245 Style (Enable & Direction)
 *      System Clock Freq.: 20 MHz., Mode: 1 cycle burst read and write
 * BMC3 - configured for:
 *      1M X 1 DRAMs at 100 ns. access time,
 *      Block Address: C00000#h, Memory Block Size: 8M bytes
 *      Byte Order: Little Endian, Buffer Mode: AM29C983 Style Buffers (LE & Dir)
 *      System Clock Freq.: 20 MHz., Mode: 1 cycle burst read and write
 *
 *      BITS 0-5      Reserved *** MUST BE 0 ***
 *      BITS 6,7      DRAM ADDRESS SIZE
 *      BITS 8-13     Clock Frequency
 *      BITS 14,15    Buffer Control Style
 *      BITS 16       Byte Order
 *      BITS 17       RAS Access Cycles
 *      BITS 18       Burst Write Speed
 *      BITS 19-31    Block Address Start
 */
RAM_ADDR0      .equ      0#h      ; Start Address of Memory controlled by BMC0
RAM_ADDR1      .equ      0800000#h ; Start Address of Memory controlled by BMC1
RAM_ADDR2      .equ      0A00000#h ; Start Address of Memory controlled by BMC2
RAM_ADDR3      .equ      1000000#h ; Start Address of Memory controlled by BMC3

BANK0_SIZE     .equ      0800000#h ; Size of Memory block controlled by BMC0
BANK1_SIZE     .equ      0200000#h ; Size of Memory block controlled by BMC1
BANK2_SIZE     .equ      0200000#h ; Size of Memory block controlled by BMC2
BANK3_SIZE     .equ      0800000#h ; Size of Memory block controlled by BMC3

BMC0_CONFIG    .equ      (RAM_ADDR0 | 1100001001110000000#b) ; Configuration for BMC0
BMC1_CONFIG    .equ      (RAM_ADDR1 | 0000001001101000000#b) ; Configuration for BMC1
BMC2_CONFIG    .equ      (RAM_ADDR2 | 1100101001101000000#b) ; Configuration for BMC2
BMC3_CONFIG    .equ      (RAM_ADDR3 | 1101001001110000000#b) ; Configuration for BMC3

/* File bmcinit.s -- Define memory segments controlled by V29BMC's */

        .title      "Memory Block Address Definitions"

/*
 * The following segments define the memory addresses controlled by the BMC's.
 */
        .include "bmc.h"

bmc0seg      .seg      bss,absolute BMC0_CONFIG
              .use      bmc0seg
              .block    BANK0_SIZE
bmc1seg      .seg      bss,absolute BMC1_CONFIG
              .use      bmc1seg
              .block    BANK1_SIZE
bmc2seg      .seg      bss,absolute BMC2_CONFIG
              .use      bmc2seg
              .block    BANK2_SIZE
bmc3seg      .seg      bss,absolute BMC3_CONFIG
              .use      bmc3seg
              .block    BANK3_SIZE
              .end

main:
/*
 * Configure Memory -
 * Must be done before first RAM access.
 */
        const      gr50, BMC0_CONFIG ; set up BMC0
        consth     gr50, BMC0_CONFIG >> 16
        const      gr51, BMC1_CONFIG ; set up BMC1
        consth     gr51, BMC1_CONFIG >> 16
        const      gr52, BMC2_CONFIG ; set up BMC2
        consth     gr52, BMC2_CONFIG >> 16
        const      gr53, BMC3_CONFIG ; set up BMC3
        consth     gr53, BMC3_CONFIG >> 16

        load       no, gr55, gr50 ; load configuration into BMC0
        load       no, gr55, gr51 ; load configuration into BMC1
        load       no, gr55, gr52 ; load configuration into BMC2
        load       no, gr55, gr53 ; load configuration into BMC3

/*
 * System Startup Code Follows
 */
        .
        .
        .
        .end

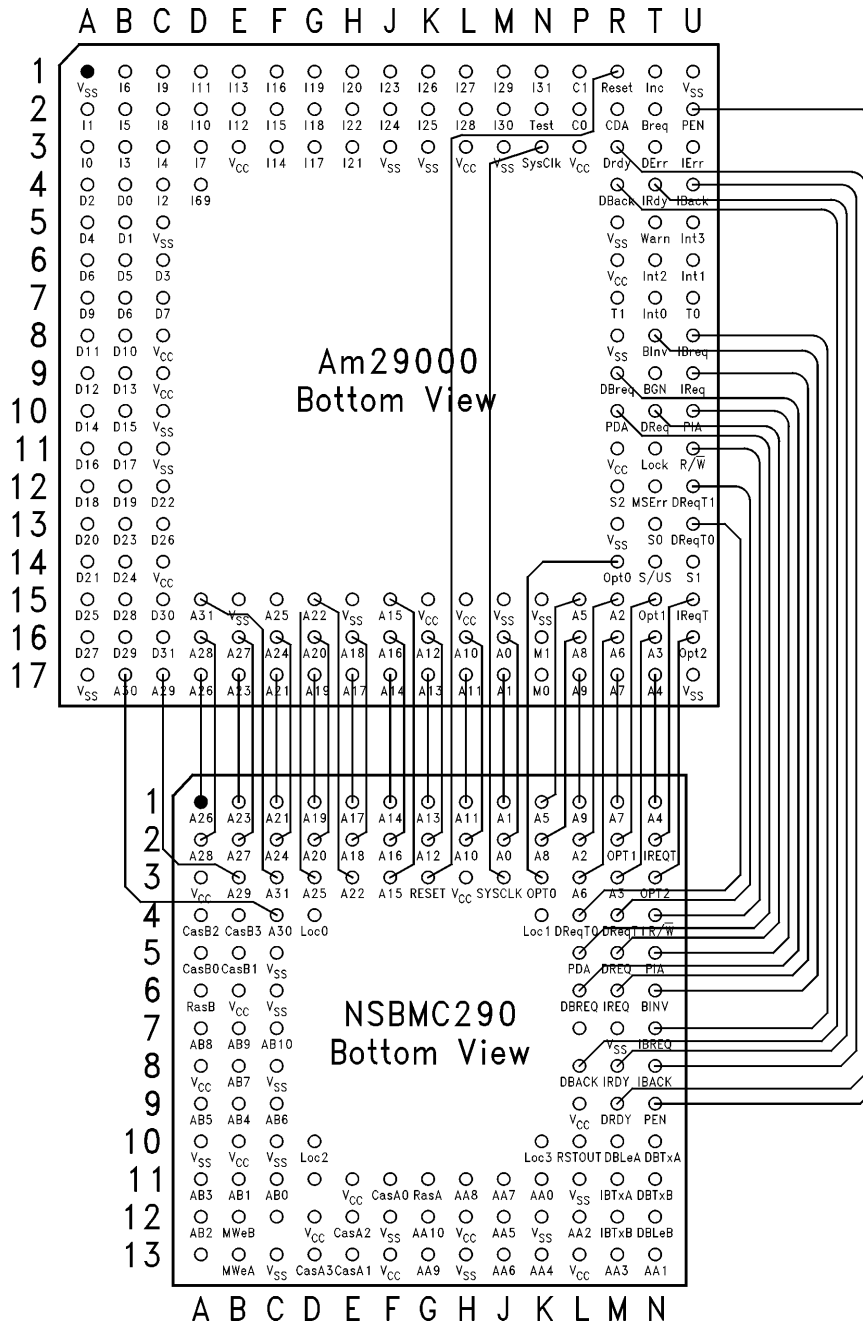
```

TL/V/11802-16

## 7.0 APPENDIX 2—NSBMC290 TO Am29000 INTERCONNECT (PGA)

Relationship between signal pins of NSBMC290 and Am29000. The Diagram shows a possible Printed Circuit Board routing strategy.

Fine line design rules (two signals between pins) are assumed. The use of multiple NSBMC290s require only that the connection bus be extended. No intermediate logic is required.



TL/V/11802-17

### 8.0 APPENDIX 3—PIN LIST SORTED BY SIGNAL NAME (PGA)

All signals marked as Reserved should be connected to V<sub>CC</sub> through individual 10k resistors so as to maintain compatibility with future releases of the NSBMC290.

| Pin Number | Signal Name |
|------------|-------------|
| J2         | A0          |
| J1         | A1          |
| L2         | A2          |
| M3         | A3          |
| N1         | A4          |
| K1         | A5          |
| L3         | A6          |
| M1         | A7          |
| K2         | A8          |
| L1         | A9          |
| H2         | A10         |
| H1         | A11         |
| G2         | A12         |
| G1         | A13         |
| F1         | A14         |
| F3         | A15         |
| F2         | A16         |
| E1         | A17         |
| E2         | A18         |
| D1         | A19         |
| D2         | A20         |
| C1         | A21         |
| E3         | A22         |
| B1         | A23         |
| C2         | A24         |
| D3         | A25         |
| A1         | A26         |
| B2         | A27         |
| A2         | A28         |
| B3         | A29         |
| C4         | A30         |
| C3         | A31         |
| K11        | AA0         |
| N13        | AA1         |
| L12        | AA2         |
| M13        | AA3         |
| K13        | AA4         |
| J12        | AA5         |
| J13        | AA6         |
| J11        | AA7         |

| Pin Number | Signal Name |
|------------|-------------|
| H11        | AA8         |
| G13        | AA9         |
| G12        | AA10        |
| C11        | AB0         |
| B11        | AB1         |
| A12        | AB2         |
| A11        | AB3         |
| B9         | AB4         |
| A9         | AB5         |
| C9         | AB6         |
| B8         | AB7         |
| A7         | AB8         |
| B7         | AB9         |
| C7         | AB10        |
| N6         | BINV        |
| F11        | CASA0       |
| E13        | CASA1       |
| E12        | CASA2       |
| D13        | CASA3       |
| A5         | CASB0       |
| B5         | CASB1       |
| A4         | CASB2       |
| B4         | CASB3       |
| L8         | DBACK       |
| M10        | DBLEA       |
| N12        | DBLEB       |
| L6         | DBREQ       |
| N10        | DBTXA       |
| N11        | DBTXB       |
| M9         | DRDY        |
| M5         | DREQ        |
| L4         | DREQT0      |
| M4         | DREQTI      |
| N8         | IBACK       |
| N7         | IBREQ       |
| M11        | IBTXA       |
| M12        | IBTXB       |
| M8         | IRDY        |
| M6         | IREQ        |
| N2         | IREQT       |

| Pin Number | Signal Name     |
|------------|-----------------|
| B13        | MWEA            |
| B12        | MWEB            |
| K3         | OPT0            |
| M2         | OPT1            |
| N3         | OPT2            |
| L5         | PDA             |
| N9         | PEN             |
| N5         | PIA             |
| N4         | R/*W            |
| G11        | RASA            |
| A6         | RASB            |
| A13        | Reserved        |
| C12        | Reserved        |
| D11        | Reserved        |
| L7         | Reserved        |
| G3         | RESET           |
| L10        | RSTOUT          |
| J3         | SYSCCLK         |
| A3         | V <sub>CC</sub> |
| A8         | V <sub>CC</sub> |
| B6         | V <sub>CC</sub> |
| B10        | V <sub>CC</sub> |
| D12        | V <sub>CC</sub> |
| E11        | V <sub>CC</sub> |
| F13        | V <sub>CC</sub> |
| H3         | V <sub>CC</sub> |
| H12        | V <sub>CC</sub> |
| L9         | V <sub>CC</sub> |
| H13        | V <sub>CC</sub> |
| A10        | V <sub>SS</sub> |
| C5         | V <sub>SS</sub> |
| C6         | V <sub>SS</sub> |
| C8         | V <sub>SS</sub> |
| C10        | V <sub>SS</sub> |
| C13        | V <sub>SS</sub> |
| F12        | V <sub>SS</sub> |
| H13        | V <sub>SS</sub> |
| K12        | V <sub>SS</sub> |
| L11        | V <sub>SS</sub> |
| M7         | V <sub>SS</sub> |

#### 9.0 APPENDIX 4—PINS LIST SORTED BY PIN NUMBER (PGA)

All signals marked as Reserved should be connected to  $V_{CC}$  through individual 10k resistors so as to maintain compatibility with future releases of the NSBMC290.

| Pin Number | Signal Name | Pin Number | Signal Name | Pin Number | Signal Name |
|------------|-------------|------------|-------------|------------|-------------|
| A1         | A26         | D2         | A20         | K13        | AA4         |
| A2         | A28         | D3         | A25         | L1         | A9          |
| A3         | $V_{CC}$    | D11        | Reserved    | L2         | A2          |
| A4         | CASB2       | D12        | $V_{CC}$    | L3         | A6          |
| A5         | CASB0       | D13        | CASA3       | L4         | DREQT0      |
| A6         | RASB        | E1         | A17         | L5         | PDA         |
| A7         | AB8         | E2         | A18         | L6         | DBREQ       |
| A8         | $V_{CC}$    | E3         | A22         | L7         | Reserved    |
| A9         | AB5         | E11        | $V_{CC}$    | L8         | DBACK       |
| A10        | $V_{SS}$    | E12        | CASA2       | L9         | $V_{CC}$    |
| A11        | AB3         | E13        | CASA1       | L10        | RSTOUT      |
| A12        | AB2         | F1         | A14         | L11        | $V_{SS}$    |
| A13        | Reserved    | F2         | A16         | L12        | AA2         |
| B1         | A23         | F3         | A15         | L13        | $V_{CC}$    |
| B2         | A27         | F11        | CASA0       | M1         | A7          |
| B3         | A29         | F12        | $V_{SS}$    | M2         | OPT1        |
| B4         | CASB3       | F13        | $V_{CC}$    | M3         | A3          |
| B5         | CASB1       | G1         | A13         | M4         | DREQT1      |
| B6         | $V_{CC}$    | G2         | A12         | M5         | DREQ        |
| B7         | AB9         | G3         | RESET       | M6         | IREO        |
| B8         | AB7         | G11        | RASA        | M7         | $V_{SS}$    |
| B9         | AB4         | G12        | AA10        | M8         | IRDY        |
| B10        | $V_{CC}$    | G13        | AA9         | M9         | DRDY        |
| B11        | AB1         | H1         | A11         | M10        | DBLEA       |
| B12        | MWEB        | H2         | A10         | M11        | IBTXA       |
| B13        | MWEA        | H3         | $V_{CC}$    | M12        | IBTXB       |
| C1         | A21         | H11        | AA8         | M13        | AA3         |
| C2         | A24         | H12        | $V_{CC}$    | N1         | A4          |
| C3         | A31         | H13        | $V_{SS}$    | N2         | IREQT       |
| C4         | A30         | J1         | A1          | N3         | OPT2        |
| C5         | $V_{SS}$    | J2         | A0          | N4         | R/*W        |
| C6         | $V_{SS}$    | J3         | SYSCLK      | N5         | PIA         |
| C7         | AB10        | J11        | AA7         | N6         | BINV        |
| C8         | $V_{SS}$    | J12        | AA5         | N7         | IBREO       |
| C9         | AB6         | J13        | AA6         | N8         | IBACK       |
| C10        | $V_{SS}$    | K1         | A5          | N9         | PEN         |
| C11        | AB0         | K2         | A8          | N10        | DBTXA       |
| C12        | Reserved    | K3         | OPT0        | N11        | DBTXB       |
| C13        | $V_{SS}$    | K11        | AA0         | N12        | DBLEB       |
| D1         | M9          | K12        | $V_{SS}$    | N13        | AA1         |

10.0 APPENDIX 5—QUAD FLATPAK PIN-OUT

NSBMC290  
(Top)

Pin #1  
IDENT

Pin 1: IDENT

Pin 2: RESET

Pin 3: A11

Pin 4: A10

Pin 5: VCC

Pin 6: A0

Pin 7: A5

Pin 8: A8

Pin 9: A9

Pin 10: A7

Pin 11: A2

Pin 12: A4

Pin 13: OPT1

Pin 14: SYSCLK

Pin 15: OPT0

Pin 16: A3

Pin 17: A6

Pin 18: DREQ0

Pin 19: A3

Pin 20: A6

Pin 21: IREQT

Pin 22: OPT2

Pin 23: OPT1

Pin 24: DREQT1

Pin 25: R/W

Pin 26: DREQ

Pin 27: PIA

Pin 28: PDA

Pin 29: IREQ

Pin 30: BINV

Pin 31: DBREQ

Pin 32: IBREQ

Pin 33: VSS

Pin 34: reserved

Pin 35: IBACK

Pin 36: IRDY

Pin 37: DBACK

Pin 38: PEN

Pin 39: DRDY

Pin 40: DBTXA

Pin 41: DBLEA

Pin 42: DBTXB

Pin 43: DBLEB

Pin 44: IBTXA

Pin 45: IBTXB

Pin 46: RSTOUT

Pin 47: VCC

Pin 48: VSS

Pin 49: 49

Pin 50: 50

Pin 51: AA0

Pin 52: AA1

Pin 53: AA2

Pin 54: AA3

Pin 55: VCC

Pin 56: VSS

Pin 57: AA4

Pin 58: AA5

Pin 59: AA6

Pin 60: AA7

Pin 61: VCC

Pin 62: VSS

Pin 63: AA8

Pin 64: AA9

Pin 65: AA10

Pin 66: PAA

Pin 67: VSS

Pin 68: VSS

Pin 69: VSS

Pin 70: VSS

Pin 71: CASA0

Pin 72: CASA1

Pin 73: CASA2

Pin 74: CASA3

Pin 75: VCC

Pin 76: VSS

Pin 77: MWEA

Pin 78: reserved

Pin 79: MWEB

Pin 80: VCC

Pin 81: VSS

Pin 82: reserved

Pin 83: reserved

Pin 84: 84

Pin 85: 85

Pin 86: 86

Pin 87: 87

Pin 88: AB0

Pin 89: AB1

Pin 90: AB2

Pin 91: AB3

Pin 92: VCC

Pin 93: VSS

Pin 94: AB4

Pin 95: AB5

Pin 96: AB6

Pin 97: AB7

Pin 98: VCC

Pin 99: VSS

Pin 100: AB8

Pin 101: AB9

Pin 102: RASB

Pin 103: VCC

Pin 104: VSS

Pin 105: CASB0

Pin 106: CASB1

Pin 107: CASB2

Pin 108: CASB3

Pin 109: VCC

Pin 110: VSS

Pin 111: A28

Pin 112: A29

Pin 113: A27

Pin 114: A30

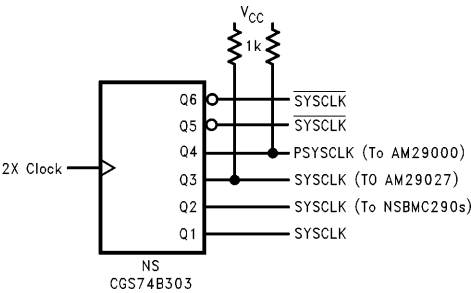
Pin 115: VSS

Pin 116: A31

Pin 117: 117

11.0 APPENDIX 6—RECOMMENDED SYSTEM CLOCK GENERATOR

Due to the high loading factor of the sysclk input of the Am29000 (90 pF), the following clock generator circuit should be used to minimize skew and distortion.



TL/V/11802-19

LIFE SUPPORT POLICY

NATIONAL'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF THE PRESIDENT OF NATIONAL SEMICONDUCTOR CORPORATION. As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and whose failure to perform, when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury to the user.
2. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.



**National Semiconductor Corporation**  
2900 Semiconductor Drive  
P.O. Box 58090  
Santa Clara, CA 95052-8090  
Tel: (408) 272-9959  
TWX: (910) 339-9240

**National Semiconductor GmbH**  
Livny-Gargan-Str. 10  
D-82256 Fürstenfeldbruck  
Germany  
Tel: (81-41) 35-0  
Telex: 527849  
Fax: (81-41) 35-1

**National Semiconductor Japan Ltd.**  
Sumitomo Chemical  
Engineering Center  
Bldg. 7F  
1-7-1, Nakase, Mihama-Ku  
Chiba-City,  
Chiba Prefecture 261  
Tel: (043) 299-2300  
Fax: (043) 299-2500

**National Semiconductor Hong Kong Ltd.**  
13th Floor, Straight Block,  
Ocean Centre, 5 Canton Rd.  
Tsimshatsui, Kowloon  
Hong Kong  
Tel: (852) 2737-1600  
Fax: (852) 2736-9960

**National Semicondutores Do Brazil Ltda.**  
Rue Deputado Lacorda Franco  
120-3A  
Sao Paulo-SP  
Brazil 05418-000  
Tel: (55-11) 212-5066  
Telex: 391-1131931 NSBR BR  
Fax: (55-11) 212-1181

**National Semiconductor (Australia) Pty. Ltd.**  
Building 16  
Business Park Drive  
Monash Business Park  
Nottingham, Melbourne  
Victoria 3168 Australia  
Tel: (3) 558-9999  
Fax: (3) 558-9998

National does not assume any responsibility for use of any circuitry described, no circuit patent licenses are implied and National reserves the right at any time without notice to change said circuitry and specifications.