



By Bob Napaa

INTRODUCTION

The IDT79R3051™ RISController™ family utilizes a high-performance computing core to achieve high performance across a variety of applications. Further, the amount of cache incorporated in the R3051 family allow these CPUs to achieve very high performance even with simple, low-speed low-cost memory subsystems.

The R3051 and the R3081™ RISController CPU families include a full R3000A core RISC processor, and thus are fully compatible with the standard MIPS processors. In order to provide high band-width to the CPU core, the families also incorporate relatively large instruction and data caches. The external memory interface from the R3051 family is very flexible and allows a wide variety of implementations depending on the price/performance goal of the application. The R3081 is upward compatible to the R3051 family with the same footprint and bus interface and the benefit of larger caches and a hardware floating-point coprocessor.

This paper will discuss the cost and performance impact of various trade-offs, and provide a concrete design of a DRAM memory subsystem around the R3051 and the R3081. This paper will specifically address the trade-offs between high-performance and low-cost memory systems, the impact of a

high-frequency system on the memory interface and the impact of systems which are intended to be field upgradeable.

DIFFERENT TYPES OF MEMORY

SRAM, DRAM and EPROM are today's industry standard for memory subsystems. EPROMs usually provide boot code in most systems and are much slower and more expensive than SRAMs or DRAMs. SRAMs are typically less dense and more expensive than DRAMs; however, they provide faster memory access time with a simpler interface and can be used in systems where performance (rather than cost) is the primary criterion. DRAMs are the most popular choice for main memory because of their position on the cost/performance curve and the densities in which they are available.

MEMORY SYSTEMS

Most of today's systems use one of two memory architectures: Non-Interleaved or Interleaved architectures. In this paper, a memory array is defined as the group of memory devices that produce a full width CPU data bus. For example a 16-bit data bus CPU requires 4 "x4" DRAMs to compose a memory array while a 32-bit data bus CPU requires 8 "x4" DRAMs to compose a memory array.

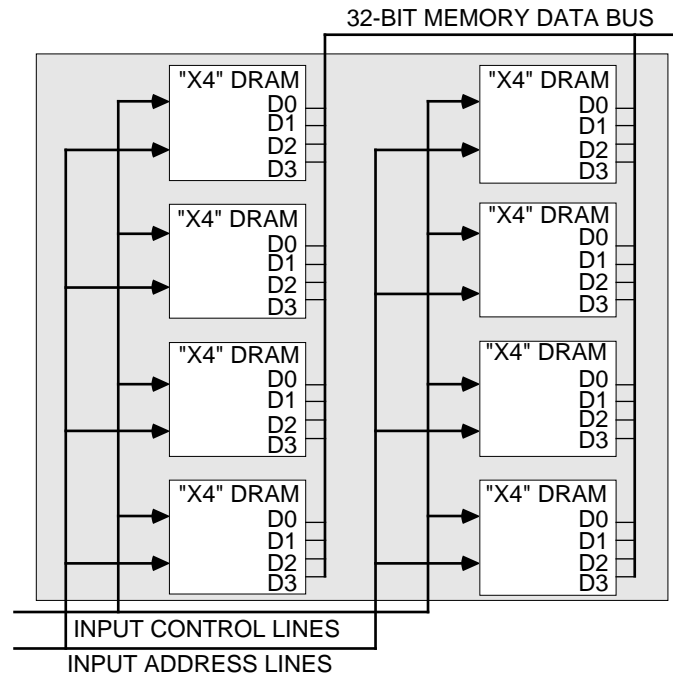
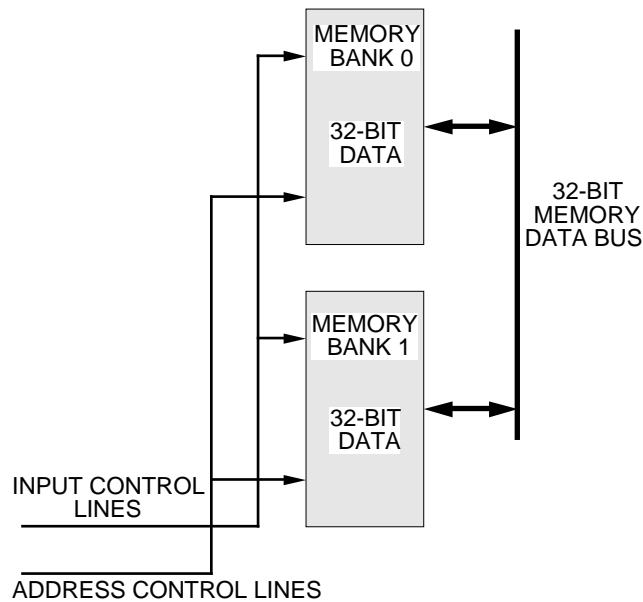


Figure 1a. Single-Bank Non-Interleaved System

The IDT logo is a registered trademark and RISController, IDT79R3051 and IDT79R3081 are trademarks of Integrated Device Technology, Inc. All others are trademarks of their respective companies.

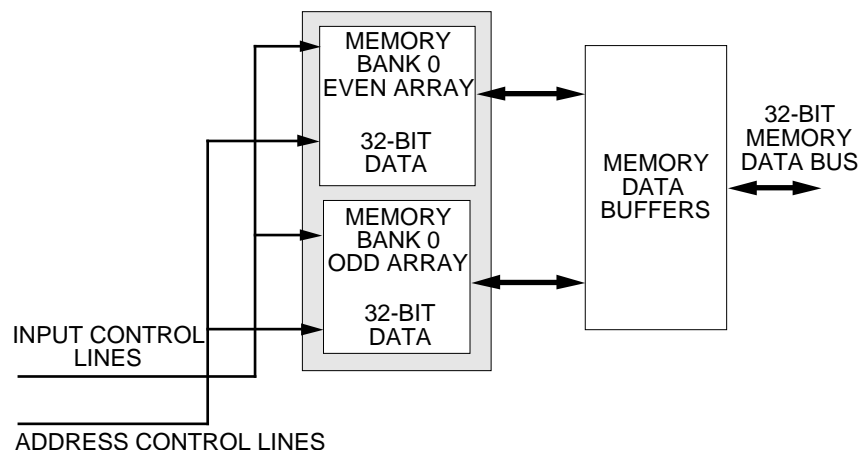


**Figure 1b. Two-Bank Non-Interleaved System**

In non-interleaved architectures, a memory bank consists of a single memory array with sequential addresses. Any read or write to a memory bank accesses a single location. Figure 1a illustrates the architecture of a single non-interleaved memory bank. Non-interleaved memory architectures are usually composed of multiple memory banks to satisfy the memory requirements of the system. In these topologies, the high order address lines select among the multiple memory banks and only one memory bank can be selected at a time. Figure 1b illustrates the architecture of a non-interleaved two banks memory system.

There are various types of interleaved architectures. The most popular one is the address interleaved. There are numerous variations of the address interleaved architectures. Mainly, 2-way address interleaved, 4-way address interleaved and so on. In a 2-way address interleaved architecture two

memory arrays are grouped together in parallel to form a Super memory bank. This Super memory bank thus has double the data bus width and double the memory density of a single non-interleaved bank, and consists then of an even array and an odd array. A memory controller must be able to select both arrays together or independently based on the type of access. The memory controller uses the low order address bit to select between the two arrays. It must be able to direct the data path from every memory array independently to the CPU through some data buffers. Figure 2 illustrates the architecture of a 2-way interleaved single Super memory bank system. In a 4-way address interleaved architectures four memory arrays are grouped together in parallel to form a Super memory bank. This Super memory bank consists thus of four quarters. The memory controller must be able to select these four arrays together or independently using the two low



**Figure 2. 2-Way Interleaved Single Super Memory Bank**

order address bits. It must be able to direct the data bus of every quarter independently to the CPU through some data buffers.

Address interleaved memory systems are thus inherently more expensive than non-interleaved architecture since they require a much more complex memory controller and wider data paths. The basic amount of memory banks in address interleaved architectures is a multiple of the basic memory bank in non-interleaved architectures; however, for systems with large amount of memory, the same memory banks could be configured as interleaved or non-interleaved. The major advantage of interleaved systems lie in block of data elements accesses from/to the CPU. Interleaved systems can double or quadruple the memory band-width and thus dramatically improve the performance when the CPU reads or writes 4, 8, 16, 32. . . data elements at a time. Interleaved systems do not offer any advantage for single independent read or write accesses. Interleaved architectures are usually used in systems where performance (rather than cost) is of importance. For embedded cost sensitive applications, non-interleaved is usually the architecture of choice.

## GENERAL DESCRIPTION OF THE DRAM SYSTEM AROUND THE R3051

The R3051 is designed around the R3000A MIPS RISC core and features a high level of integration with large on-chip instruction and data cache. It incorporates up to 8kB of instruction cache and 2kB of data cache. These relatively large caches achieve hit rates in excess of 90% and substantially contribute to the performance inherent in the R3051 family. The R3051 has also implemented on-chip a four-deep read and a four-deep write buffers that isolate the high frequency CPU core from the much slower external memory and modules. This high level of integration simplifies the interface between the R3051 and the external memory modules as is illustrated in Figure 3 and allows the use of low cost memory subsystems without penalizing the performance.

The R3051 family uses a double frequency input clock for its internal operation and provides a nominal frequency output clock for the external system. This output clock, SysClk, synchronizes the external memory subsystems to the CPU. Memory transactions from the R3051 use a single, time multiplexed 32-bit address and data bus and a simple set of

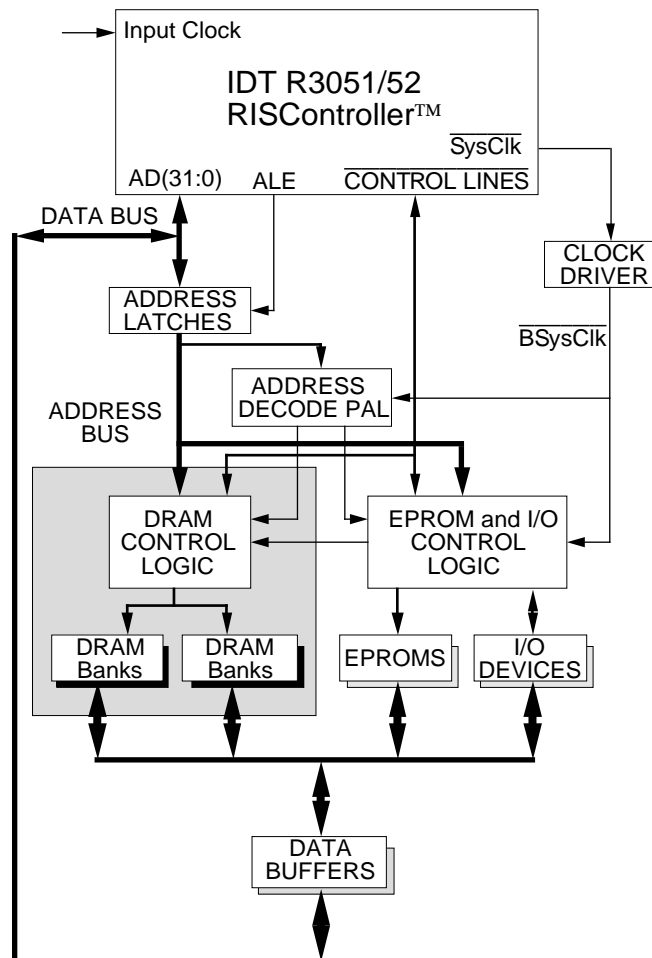


Figure 3. R3051 RISController Family-Based System

control signals. External logic then performs address demultiplexing and decoding, memory control, interface timing and data path control.

The system shown in Figure 3 is a 25MHz system with a 50MHz input clock. The R3051 interfaces to a DRAM system as the main memory, to an EPROM system and to various I/O devices and controllers. Address latches decouple the address bus from the data bus. Address decoders select among the various external modules. The output clock from the R3051 (Sysclk) is usually buffered to reduce the loading effect and to provide clock drive capability with minimum clock skew for the system.

The main DRAM memory system is based on 1 to 4 banks of non-interleaved DRAMs with 80ns of access time ( $t_{rac} = 80ns$ ). The DRAMs used are 256k x 4 to provide a maximum memory space of 4MB. The DRAM memory space occupies the lower 4MB of the physical memory space. Figure 4 illustrates the architecture of the main DRAM memory system. The DRAM memory space resides between addresses 0000\_0000 and 3FFF\_FFFF. Address bits A(21:20) select among the four banks while the Rd and Wr outputs from the R3051 differentiate between read and write accesses.

Each memory bank (32-bit array) of DRAM, which corresponds to 1MB when using 256k x 4 DRAMs, is individually controlled by a separate RAS signal. RAS0 controls DRAM bank 0, RAS1 controls DRAM bank 1,... Each bank of DRAM is also controlled by an individual WriteEnable signal. WriteEnable0 controls DRAM bank 0, WriteEnable1 controls

DRAM bank 1,... This architecture enables only a single DRAM bank for any DRAM read or a write access. The DRAM banks are arranged so that each bank represents a single, contiguous range of 1MB.

In an R3051 system, it is possible to perform a 32-bit read even when smaller data elements are requested. However on writes, it is important to enable only those bytes which are actually being written by the CPU. The R3051 bus interface provides four individual byte-enable signals to indicate which byte lanes are involved in a particular transfer. The DRAM subsystem encodes the byte-enable information from the R3051 into the CAS control signals of the DRAMs. In this encoding, CAS0 corresponds to byte lane 0, CAS1 corresponds to byte lane 1, etc. Each CAS signal is connected to the DRAM devices that correspond to the byte lane under its control in all four banks of the DRAM subsystem. That is to say that CAS0 is connected to the two DRAM devices that compose byte 0 in every DRAM bank.

Data buffers isolate the DRAM banks from the R3051 data bus to reduce the loading effect and to prevent contentions between the R3051 and the DRAMs. Note that this also alleviates concerns about the relatively slow tri-state times associated with DRAM devices. The data buffers selected are industry standard bidirectional transceivers (74FCT245). These data buffers actually isolate the data bus of the R3051 from all the external modules.

DRAM addresses are provided by multiplexing the latched R3051 address bus using the IDT FBT2827B memory drivers.

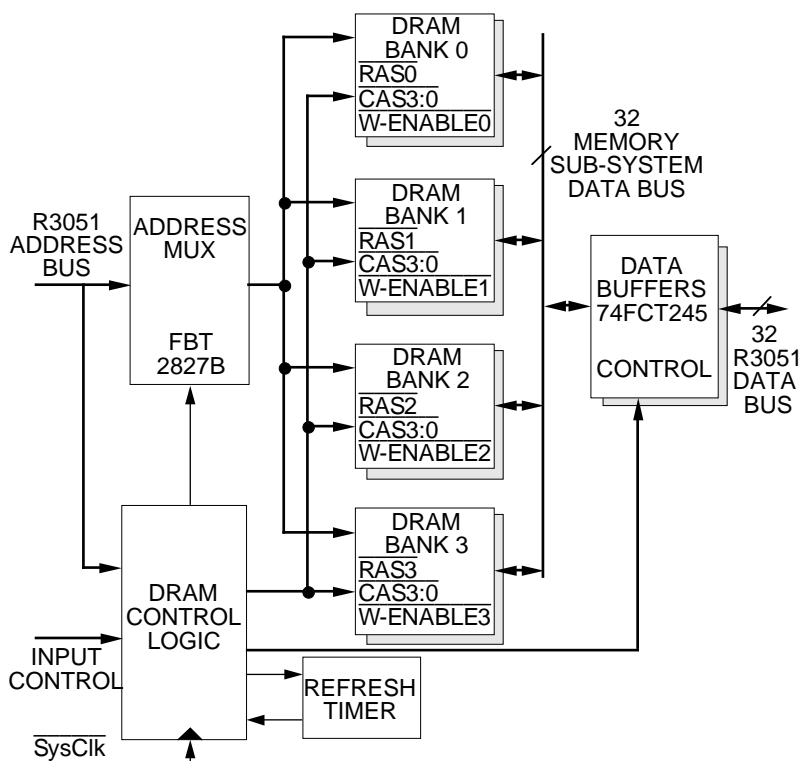


Figure 4. DRAM Memory Subsystem Architecture

This device type was selected based on its ability to drive large capacitive loading, such as found when driving 32 DRAM devices. A single FBT output has a series resistance incorporated in the output driver and is capable of driving all four banks of the DRAM subsystem. To minimize the signal skew among the DRAM devices, the address lines and the control lines to the DRAMs must use the "star" or the "fork" topology on the PCB board. In this method, all the loads on a given signal are lumped at the far end of the PCB trace. Series termination is also well suited to drive lumped (or forked) CMOS loads (like DRAMs) at the end of a PCB trace. The series termination minimizes overshoots and undershoots at the receiving end and does not add any power dissipation to the system.

Every DRAM cell consists of a MOS cell and a capacitor which encodes logic 1 and 0 in its charge. The capacitors in the DRAM cells tend to lose their charges with time through leakage. This is why DRAMs require to be refreshed at a regular time interval. The refresh mechanism is internal to the DRAMs where bits (cells) are rewritten with the same value to keep the capacitors charged. This refresh mechanism is enabled by the input control signals to the DRAM devices through the RAS and the CAS signals. In this design a refresh timer requests the refreshing of the DRAMs every 9.6µs. This refresh timer can be driven by the Sysclk from the R3051 or from an independent oscillator. The 9.6µs refresh interval chosen is more frequent than is actually required by the DRAMs. The use of this value simplified the control logic associated with page mode write. DRAMs require that RAS be maintained low no longer than 10µs; by choosing a refresh value smaller than this maximum time, the system is assured that maximum RAS low time will not be violated.

### DRAM STATE MACHINE DESIGN

For the system described in this paper, a simple state machine performs the major aspects of DRAM control. The state machine uses a simple four-bit counter (C(3:0)) to dictate the timing for the DRAM control and CPU response, and is sequenced using SysClk. There are nine major states to the state machine as is illustrated in Figure 5. These states are dictated by the type of transfer requested and the state the DRAM control logic was left in by the prior transfer.

The DRAM control logic uses the Reset pulse to reset its internal states and to synchronize its operation to the R3051. During the RESET state, it also performs one refresh cycle before entering the IDLE state. In the IDLE state, the DRAM control logic arbitrates between a refresh cycle and a bus access. A DRAM bus access is started whenever the DRAM-Chip-Select and the Rd or the Wr signals are asserted. A refresh request is detected using the REF\_REQ (Refresh\_Request) pulse from the refresh timer. The DRAM controller supports 4 types of CPU bus accesses: "quad-word read", "Single-word read", "Single-word write" and "Page-word write". After a "Single-word write" or a "Page-word write" access, the DRAM control logic enters the IDLE RAS ASSERTED state which is an IDLE state with the RAS signals kept asserted. The RAS signals need to be precharged upon exiting this state.

#### Reset Cycle

A reset cycle is initiated by the assertion of the Reset signal. This is a hardware reset which initializes the control logic to the correct IDLE state. After the Reset signal is de-asserted, one DRAM refresh cycle is initiated. Most DRAMs require at least 8 refresh cycles for proper initialization. This DRAM

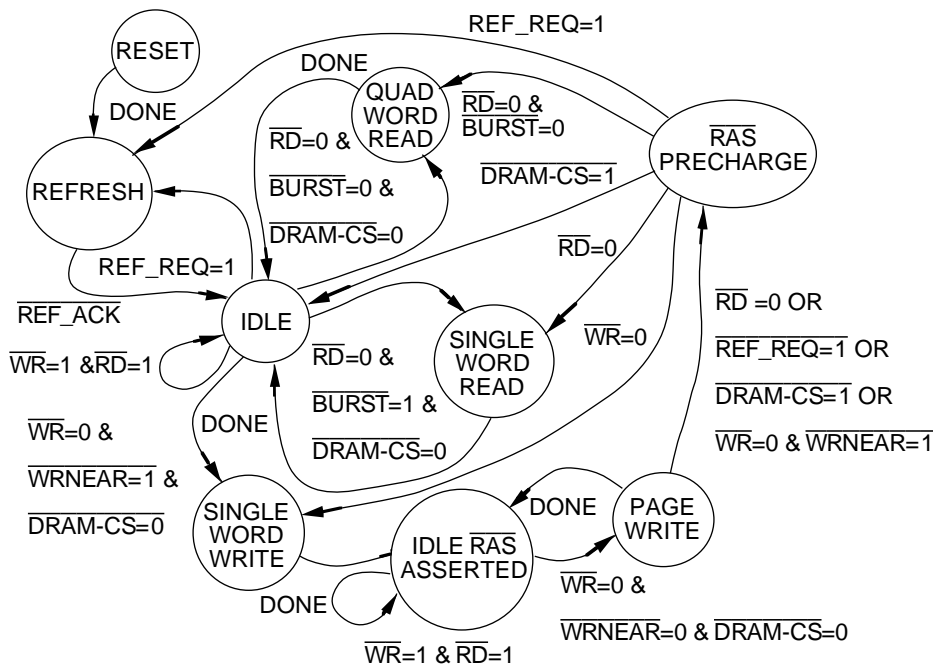


Figure 5. DRAM Control State Machine

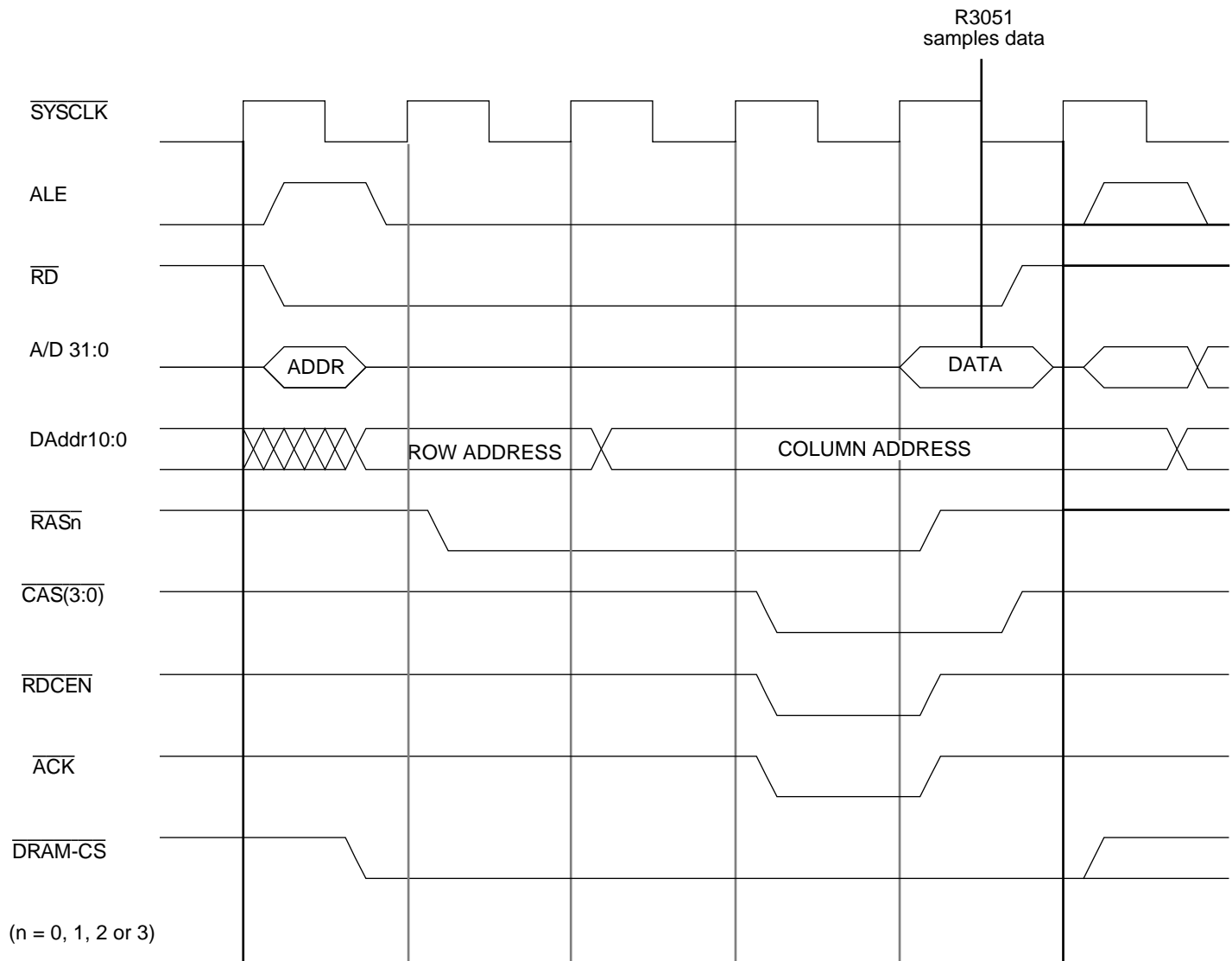


Figure 6. Single-Word Read Access Timing

control logic provides only one refresh cycle at reset time. It is the responsibility of the software to ensure that no DRAM access is made prior to the elapsing of 8 refresh periods. This can be insured by normal operation of the boot PROM; however, software could “spin-lock” for a predetermined number of loops to insure that sufficient time has elapsed.

### Refresh Cycle

A refresh cycle is initiated every time a REF\_REQ pulse from the refresh timer is detected. The refresh timer issues a REF\_REQ pulse every 9.6 $\mu$ s. The DRAM control logic responds with a refresh acknowledge (REF-ACK) signal which locks the refresh timer until the refresh is serviced. The refresh interval has been set to 9.6 $\mu$ s which is shorter than the maximum 15.5 $\mu$ s refresh period that most DRAM require. The 9.6 $\mu$ s refresh period ensures that for an IDLE RAS ASSERTED state, where the RAS signals can be left asserted for long time periods, the maximum RAS pulse width of 10 $\mu$ s is not violated.

In the DRAM control logic, a refresh request has the highest priority over any other CPU requests. However, if a CPU bus requested is being serviced at the time the refresh is requested, the refresh cycle will be delayed until the end of the current bus cycle. The inverse is also true when bus requested are being delayed until the end of a refresh cycle. In this design, only the RAS-before-CAS refresh method is implemented.

### Idle State

The Idle state is when the state machine is not performing any bus access or a refresh access but is constantly monitoring the bus for any access request. All the signals are deasserted and the operation of the 4-bit counter is halted.

### Single-Word Read Cycle

There are two types of read transactions from the R3051: quad-word reads and single-word reads. A single-word read access is initiated by the R3051 by asserting the Rd signal.

The DRAM control logic responds by providing the R3051 with a single data element (32-bit word). Both the Ack and the RdCEn signals are used to terminate the single-word read access. In the system described in this paper, the Ack and the RdCEn signals are returned to the R3051 after 4 clock cycles, as illustrated in Figure 6.

### Quad-Word Read Cycle

Quad-word reads from the R3051 occur only in response to internal cache misses. All instruction cache misses are processed as quad-word reads while data cache misses may be processed as either quad-word reads or single-word reads. The R3051 indicates quad-word read accesses by asserting both the Rd and the Burst signals. In the quad-word read access, address lines Addr(3:2) from the R3051 act as a two-bit counter to provide the address of 4 consecutive words, always starting on a word boundary.

The DRAM control logic handles quad-word read accesses using the Throttled Block Refill mode of the R3051. In a throttled read, RdCEn controls the data rate of the memory back to the CPU (latches the data into the on-chip read buffer). The Ack input is not provided back to the processor until the read transfer has sufficiently progressed such that the last word of the transfer is clocked into the on-chip read buffer (using RdCEn) one clock cycle before the processor core requires it.

In this non-interleaved system, the first word read of a quad-word read access takes the same time as a single read while the 3 subsequent words are read into the on-chip read buffer at the rate of 1 word every two clock cycles. The RdCEn is asserted for every word being read to latch the data into the R3051 read buffer. The Ack is asserted between the second

and the third-word read. This ensures that for 4 subsequent falling edges of Sysclk the on-chip read buffer can provide data to the R3000A core at the rate of a word every clock cycle. Figure 7 illustrates the timing involved in quad-word read accesses.

Quad-word read accesses use the page-mode characteristics of the DRAM to obtain subsequent data word at a higher data rate. In this access, the RAS signal is kept asserted while the CAS signals are toggled 4 times to produce 4 data words.

### Single-Word Write cycle

Unlike instruction fetches and data loads, which are usually satisfied by the on-chip caches, all write activity to the caches is seen at the bus interface of the R3051 as single write transactions. The R3051 indicates a single-word write access by asserting the Wr signal. The DRAM control logic enables the writing of the CPU word or partial word into the DRAMs and returns the Ack signal to terminate the write access. The Ack signal is returned to the R3051 after 3 clock cycles, as illustrated in Figure 8.

The DRAM memory system takes advantage of the WrNear signal from the R3051 by defaulting to the case that any single write to the DRAM subsystem will be followed by another write with the same upper 22 address bits. Based on this information the RAS signal must be kept asserted after every write access to enter the page mode of the DRAMs. The end of a single-word access is then different from a single read access in that the RAS signal is kept asserted.

### Idle RAS Asserted State

At the end of a write access the DRAM control logic enters this idle state where a RAS signal is kept asserted while the

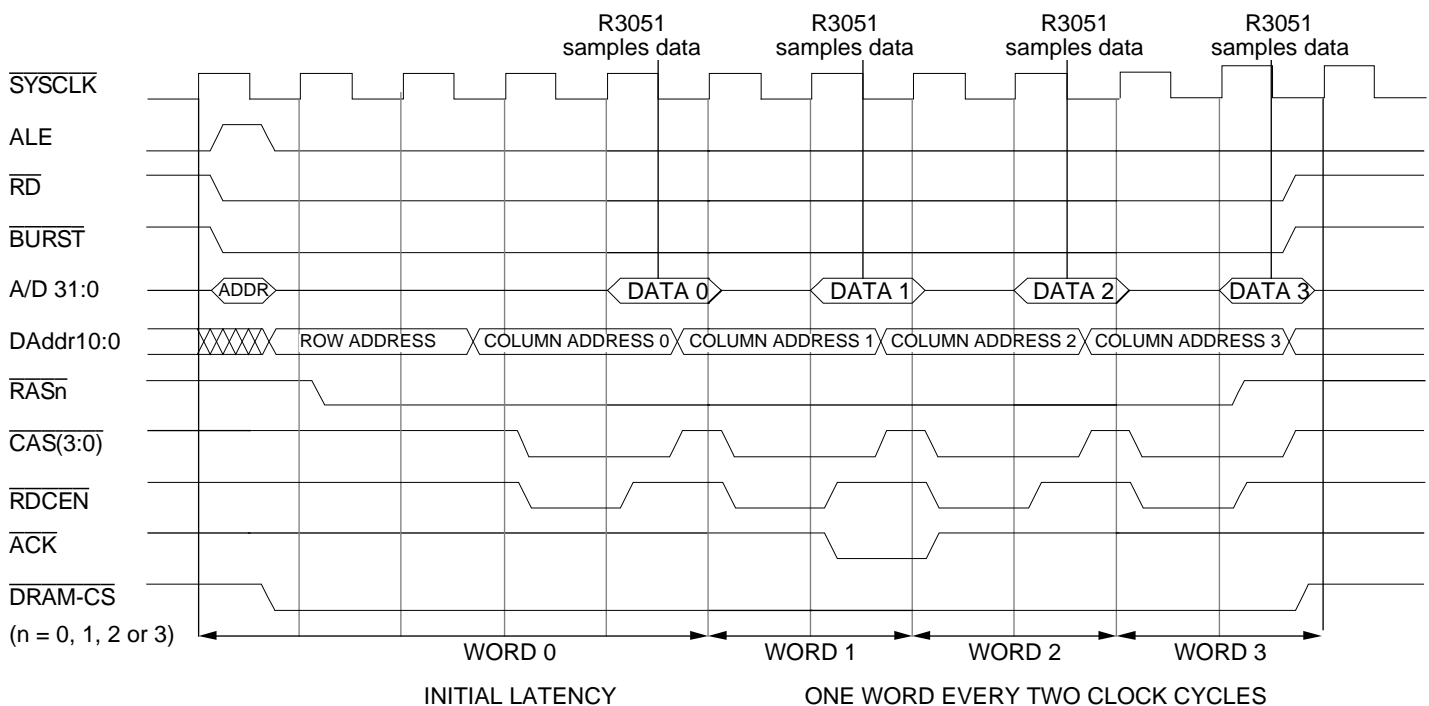


Figure 7. Quad-Word Read Access Timing

state machine awaits a subsequent transaction. If the next access is a local write ( $\overline{WrNear}$  from the R3051 is asserted) the DRAM control logic enters the page write mode. If a different access type occurs, the state machine exits this state.

### Page Write Cycle

A page write cycle is a single write access from the R3051 following a previous single write access with the same upper 22 address bits. The R3051 indicates a page write access by asserting the  $\overline{Wr}$  and the  $\overline{WrNear}$  signals.

The timing for a page write access is very similar to a single-write access but shorter since the RAS signal has been kept asserted from the previous write cycle. The Ack is returned back to the R3051 after 2 clock cycles. Figure 8 illustrates the timing for a page write access.

### Precharge RAS

Any access, except a page write access, following an Idle RAS Asserted state needs to have the RAS signal precharged (driven to a level HIGH) before the access is responded to.

## PERFORMANCE

The performance of the different types of R3051 bus accesses to the DRAM memory subsystem is usually measured by the number of clock cycles it takes to send the Ack

back to the R3051. This time is computed from the beginning of the external access. The performance of the DRAM system can be summarized as follows:

- single read: 4 clock cycles
- block refill: 7 clock cycles
- first write: 3 clock cycles
- page write: 2 clock cycles.

This is a relatively high performance for a low-cost and easy-to-implement DRAM memory subsystem. The performance of the system can be improved by using more elaborate DRAM memory controller and/or more complex memory architectures such as address interleaving. Such systems should be able to achieve optimum performance.

## FIELD UPGRADEABILITY

Many of today's systems are designed to allow for future fields upgrades of the base memory system to more memory banks and/or deeper DRAM devices. The ability to offer a base configuration (at a lower selling price) with upgrade capabilities is often a selling feature of the end product.

The system software should then run diagnostics at boot time to determine the maximum size of the available memory. Typical strategies for such diagnostics include writing distinct values into a given location within each bank, and then reading the data back to see if any of the writes did not occur properly

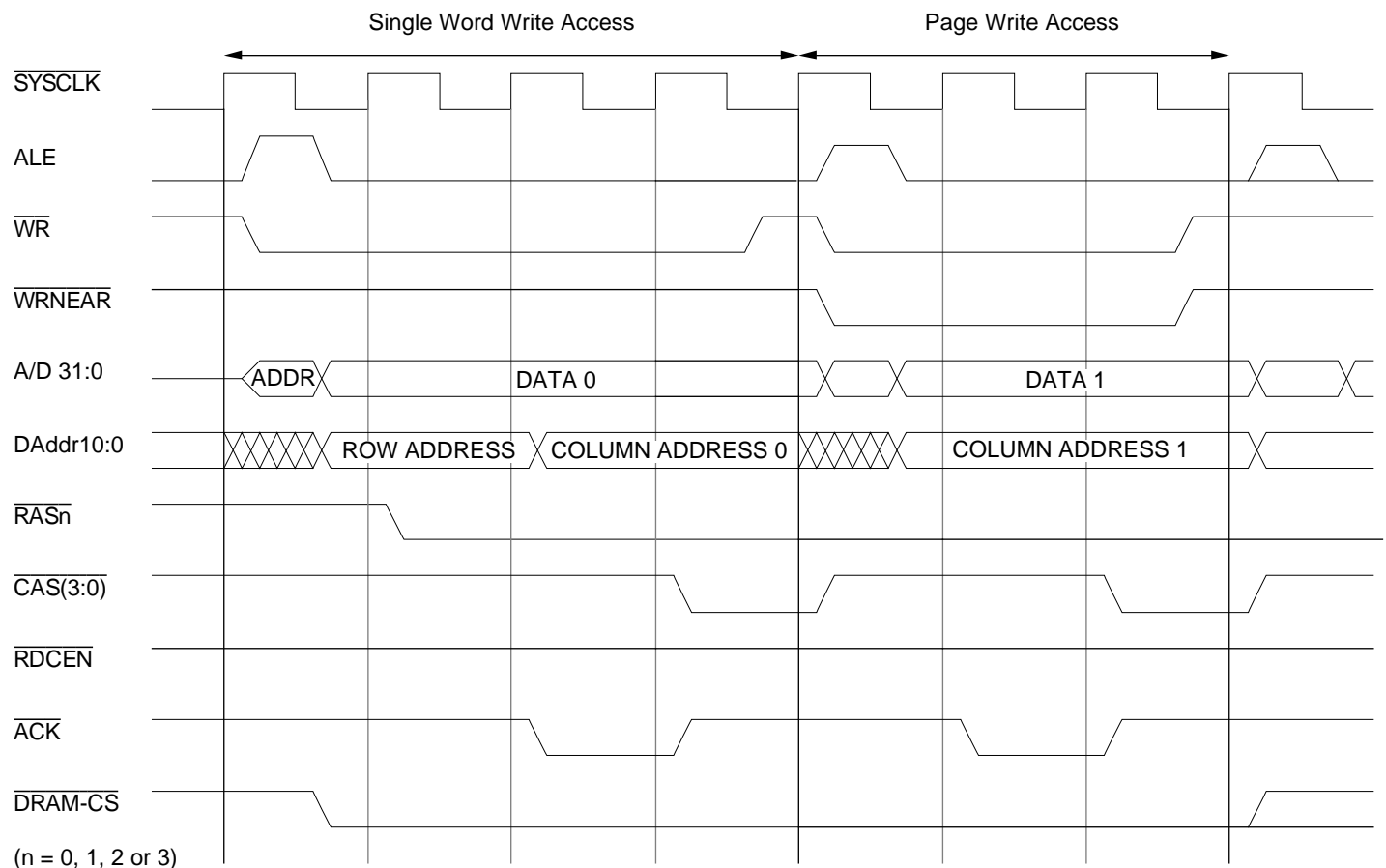


Figure 8. Single-Word Write Access Timing and Page Write Access Timing



or altered data previously written. Non-interleaved or interleaved memory architectures should be transparent to the system software.

The system hardware should make provision for extra memory banks or deeper memory devices by routing all the necessary signals to unused pins or sockets of future upgrade memory. The system hardware should try to minimize the use of jumpers to make the system much more user friendly.

In the system described in this paper, the user can upgrade to deeper memory by replacing the 256k x 4 DRAMs with deeper 1MB x 4 DRAMs to obtain a maximum memory space of 16MB. It is also possible to replace the R3051 with the R3081 to increase the performance of the system since they both have the same footprint. The R3081 with its on-chip FPA will have a great impact on the performance of floating-point intensive applications; a further benefit is the larger on-chip caches of the R3081.

## CONCLUSION

The R3051 and the R3081 RISController families bus interface was designed to allow memory systems of differing complexity and performance to be implemented. Even a relatively simple DRAM system, as the one described here, offers very high performance. With simple modifications, this approach is applicable to higher frequencies (33 and 40MHz) and to interleaved memory systems yielding even higher performance. The R3081 can also be used for existing R3051 designs to improve the floating-point performance and the overall system throughput with no modifications of the external hardware.

## REFERENCES

- AN-50: "Series Termination" Application Note, by Suren Kodical, 1990/91 IDT Logic Data Book.