Application Note

AN1768/D
Rev. 0.1, 6/2003

Using Registered SDRAM DIMMs with the MPC106

Gary Milliorn and Doug McQuaid

RISC Applications

risc10@email.sps.mot.com

In this document, the term *MPC106* is used as an abbreviation for the phrase *MPC106 PCI bridge/memory controller*. Also, the term *60x* denotes a 32-bit microprocessor from the PowerPC™ architecture family that conforms to the bus interface of the PowerPC 603, PowerPC 604, or PowerPC 750 microprocessors. The 60x processors implement the PowerPC architecture as it is specified for 32-bit addressing, which provides 32-bit effective (logical) addresses, integer data types of 8, 16, and 32 bits, and floating-point data types of 32 and 64 bits (single-precision and double-precision).

This document contains the following topics:

To locate any published errata or updates for this document, refer to the website at http://www.mot.com/SPS/PowerPC/.

1.1 Introduction

The use of registered SDRAM modules (registered DIMMs) can eliminate many of the design challenges associated with the increased capacitive loads of larger memory arrays. The MPC106 PCI bridge/memory controller was not designed to directly control registered DIMMs. However, with some software and hardware modifications, the MPC106 can support these modules with minimal cost. This application note describes a method of using registered SDRAM DIMMs with the MPC106.

Throughout this application note the term “registered DIMM” is used to denote the use of a clocked buffer that preserves the SDRAM control signals between clock cycles. The *MPC106 PCI Bridge/Memory Controller User’s Manual* uses the term “registered” when referring to the use of registered buffers on the data bus only; it does not refer to registering the control signals.

The techniques developed in this application note can also be used in embedded systems where the memory (and control registers) are connected directly on the circuit board and no

DIMM socket is used. For brevity, the term “registered DIMM” will refer to all such equivalent memory architectures.

2 Background

Registered SDRAM modules are a slight variation of JEDEC-standard, unbuffered, 168-pin memory modules, in which a registered driver has been inserted between the DIMM pins and the following SDRAM control signals:

- $\overline{\text{SDRAS}}$
- $\overline{\text{SDCAS}}$
- $\overline{\text{WE}}$
- DQMB(7–0)
- $\overline{\text{CS}}$(3–0)
- CKE
- A(11–0)
- BA(1–0)

Inserting a register between the MPC106 and the SDRAM components results in the addition of a one-clock-cycle delay to every memory transaction. While this is ordinarily an undesirable effect, when driving large capacitive loads, the MPC106 might require an additional clock delay for the control signals to stabilize, assuming that they would be intelligible at all.

As shown in Figure 1, a conventional SDRAM DIMM presents a 120 pF (8 x 15 pF) load on the address signals (as well as all the other control signals). If a memory array uses four dual-bank DIMMs, the total load would be 480 pF. Since the propagation delay of a signal is proportional to the capacitive load, it becomes difficult to design high-speed memory arrays without some sort of compensation. In comparison, a registered SDRAM DIMM presents only a 15 pF load.

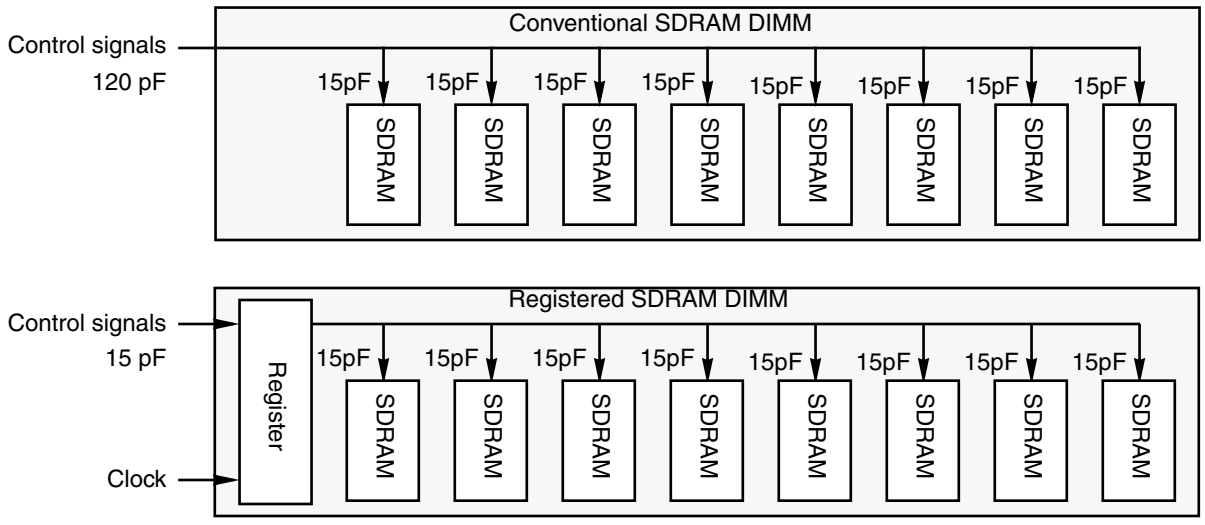

Figure 1. SDRAM DIMM Differences

Clearly, registering the control signals alleviates the difficulty of capacitive loading, and allows more DIMMs to be added to a system than was previously permitted. Registered SDRAM modules have other advantages as well, such as supplying an on-chip, low-delay PLL that eliminates clock loading and skew

problems. Since these features do not directly affect using registered DIMMs with the MPC106, they are not discussed in this application note.

Since registered DIMMs were developed after the MPC106 was designed, the MPC106 does not have all the necessary controls to directly support registered DIMMs. From an examination of the many waveforms and register settings in the *MPC106 PCI Bridge/Memory Controller User's Manual*, it might appear that the MPC106 has the programmabilty to allow a register to be added to the control signals with no additional hardware support. Unfortunately, this is not the case.

The solution presented in this application note uses the software programmability of the MPC106 together with external hardware to add a one-clock-cycle delay to the $\overline{\text{BCTL0}}$ signal (but not the $\overline{\text{BCTL1}}$ signal).

3 Read Cycles

The programmable parameter MCCR3[RDLAT] controls when the MPC106 asserts $\overline{\text{TA}}$ for read operations. At the time $\overline{\text{TA}}$ is asserted, the SDRAM must have completed a read transaction and data must be available on the 60x bus. The RDLAT parameter is similar to the CAS latency (C_L) parameter of any SDRAM, with the exception that the timing reference for RDLAT is when data is valid on the 60x bus rather than when data is valid on the SDRAM data outputs. Most MPC106-based designs use data bus buffers between the memory array and the 60x bus to eliminate loading (especially with large arrays of memory), so RDLAT is typically set to either $C_L + 0$ for flow-through transceivers or $C_L + 1$ for registered transceivers. The $C_L + 1$ setting compensates for the one-clock cycle delay caused by registered data bus buffers.

The upper portion of Figure 2 shows a timing diagram of an MPC106 SDRAM burst read transaction when using registered data bus buffers. The one clock cycle delay of the SDRAM-to-CPU data path shows the effect of the registered data bus buffers, and the reason that the $\overline{\text{TA}}$ signal must be delayed one clock cycle.

The lower half of Figure 2 shows the effect of using a registered DIMM. Since the control signals pass through a register on the registered DIMM, the SDRAM devices see delayed versions of the $\overline{\text{SDRAS}}$ and $\overline{\text{SDCAS}}$ signals from the MPC106. These delayed signals are shown in Figure 2 as $\overline{\text{SDRAS\_DLY}}$ and $\overline{\text{SDCAS\_DLY}}$, respectively. The delayed control signals cause the SDRAM data outputs to be delayed by one clock cycle. However, $\overline{\text{TA}}$ is still asserted to the processor at the time shown in the upper portion of Figure 2.

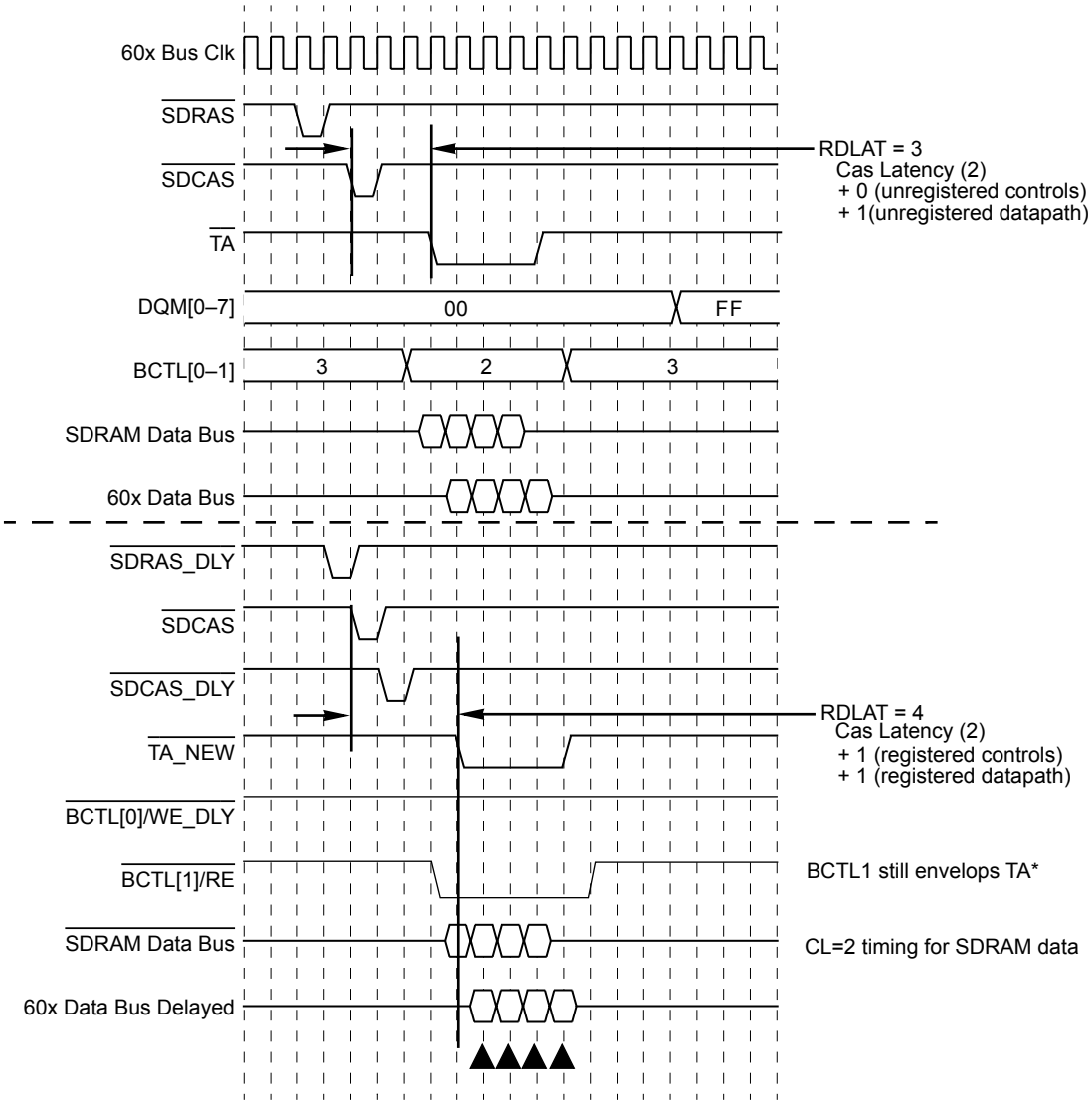

Figure 2. SDRAM Burst Read Timing

The RDLAT parameter of the MPC106 must be adjusted to delay the assertion of $\overline{TA}$ to the processor by an additional clock cycle to match the delayed valid data out of the registered DIMM. The new formula used to calculate RDLAT is:

```
RDLAT = SDRAM_CAS_Latency
        + 1 for registered data bus buffers
        + 1 for a registered DIMM
```

Note that since one cycle must be reserved for the control signals to propagate through the registered DIMM, and the maximum setting for RDLAT in the MPC106 is five clocks, the CAS latency of the SDRAM devices on the registered DIMM must be four or less. Early documentation for the MPC106 stated the maximum setting for RDLAT was four, but the maximum was increased to five beginning with Rev. 4.0 of the MPC106.

Once the adjustment has be made to RDLAT, the MPC106 asserts $\overline{TA}$ at the appropriate time, shown as $\overline{TA\_NEW}$ in Figure 2. Thus, registered DIMM read cycles can be handled entirely by reprogramming MCCR3[RDLAT]. with no external hardware support required. The signals $\overline{BCTL0}$ and $\overline{BCTL1}$, shown in the lower half of Figure 2, have been separated to show the effect of the required hardware modification (described in Section 4.1, “Buffer Control Modifications”) to the $\overline{BCTL0}$ signal (write-enable, active-low). The $\overline{BCTL1}$ signal (read-enable, active low) retains its standard timing and definition.

4 Write Cycles

While read cycles from registered DIMMs are fairly easy to accommodate using the RDLAT parameter, write cycles are slightly more difficult because the MPC106 provides no equivalent WRLAT parameter to control write timing. Instead, the MPC106 always asserts $\overline{TA}$ for writes to SDRAM at one of two fixed points:

- If MCCR4[WCBUF] = 1, $\overline{TA}$ is asserted one clock cycle before the SDRAM Write command ($\overline{CS}n$, $\overline{SDCAS}$, and $\overline{WE}$ asserted) is issued.
- If MCCR4[WCBUF] = 0, $\overline{TA}$ is asserted in the same clock cycle as the SDRAM Write command ($\overline{CS}n$, $\overline{SDCAS}$, and $\overline{WE}$ asserted).

This allows the data from the processor to begin flowing into the SDRAM with the least amount of latency. In effect, the WCBUF parameter functions as a simple, but limited, write-latency parameter.

Normally, when using registered data bus buffers, the WCBUF parameter is set. However, as shown in Figure 3, when a registered DIMM is used with WCBUF = 1, the SDRAMs receive the Write command before data begins streaming in. This causes burst transfers to lose the first cycle of data, and subsequent data cycles are stored at incorrect locations in memory. For single-beat writes, the DQM masking signals arrive one clock cycle late, again causing the write data to be stored to the wrong address. Based on these symptoms, it appears that some means of delaying $\overline{TA}$, or the data stream, is required.

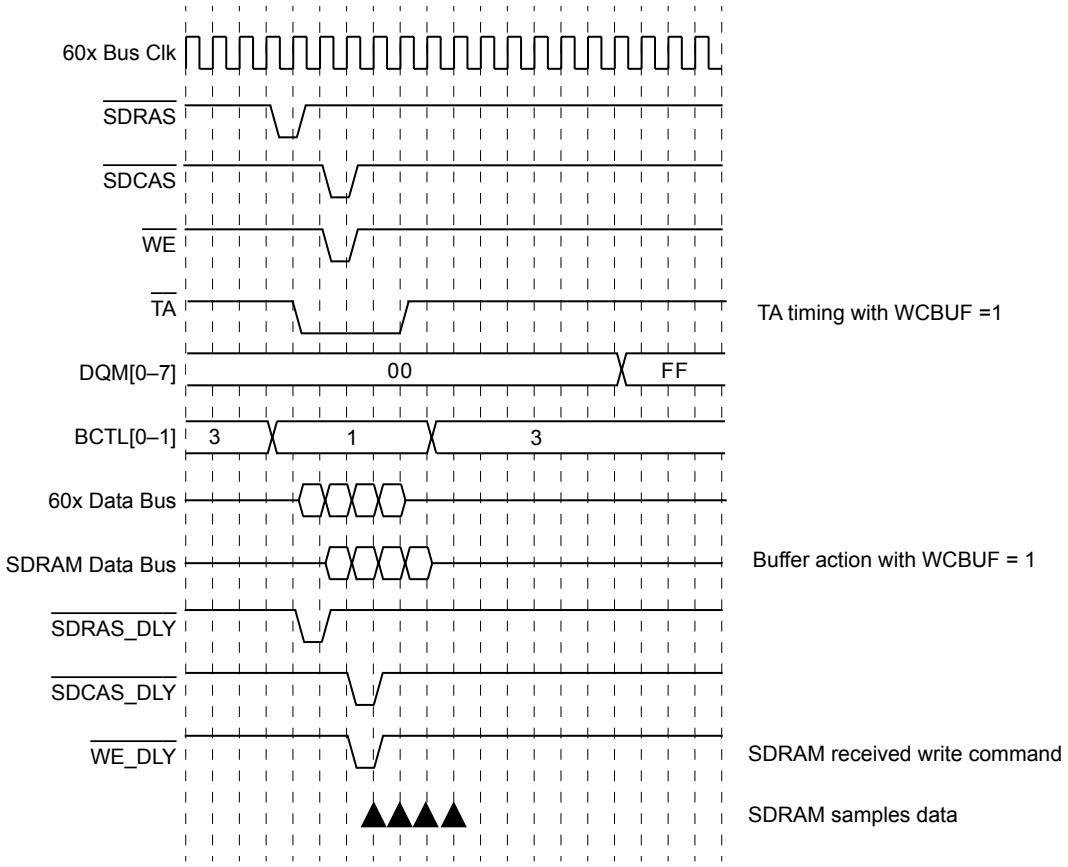

Figure 3. SDRAM Burst Write Timing with WCBUF = 1

While it might be possible to use external hardware to delay the $\overline{TA}$ signal to the processor, it would be difficult, as $\overline{TA}$ is bidirectional and the MPC106 does not drive the $\overline{DBB}$ signal. Determining when the MPC106 owns the data bus, versus some other master, requires substantial external logic, which must track bus ownership and either delay $\overline{TA}$ when the MPC106 is the target, or allow $\overline{TA}$ to be driven without delay when the MPC106 is the master. So, while delaying $\overline{TA}$ is theoretically possible, it is beyond the scope of this application note.

Instead of using the cumbersome logic to delay $\overline{TA}$, we can rely on the fact that if the MPC106 is configured for WCBUF = 0, then $\overline{TA}$ is asserted in the same clock cycle as the SDRAM Write command. Note that the MPC106 is programmed for WCBUF = 0, even though there are registered buffers in the data path. In this implementation, the MPC106 asserts $\overline{TA}$ to the processor and issues the SDRAM Write command in the same clock cycle. There is a one-clock-cycle delay through the registered data bus buffers before data is valid at the memory data inputs. With registered DIMMs, there is a one-clock-cycle delay through the control-signal register before the control signals are valid at the memory device control inputs. Thus, the control signals and the data appear at the SDRAM device inputs in the same clock cycle, as required. This is shown in Figure 4. Unfortunately, this work-around has an undesirable side-effect on the timing for the buffer control signal, $\overline{BCTL0}$, that must be compensated for using hardware.

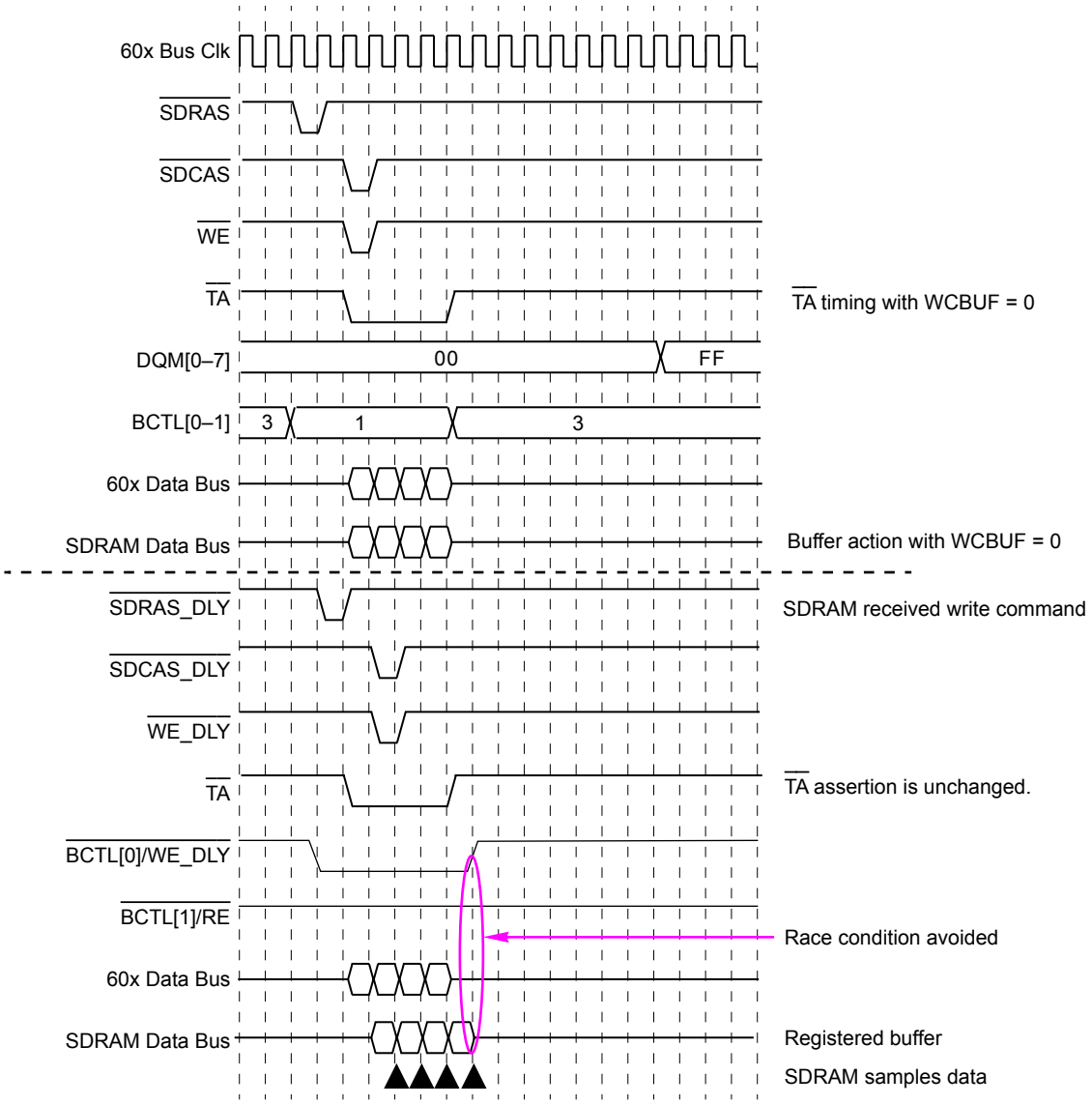

Figure 4. SDRAM Burst Write Timing with WCBUF = 0

4.1 Buffer Control Modifications

When MCCR2[BUF_MODE] = 0, the $\overline{\text{BCTL0}}$ signal controls data clocking into the registered data bus buffer during write operations. It is unacceptable to rely upon latching data into the buffer while the $\overline{\text{BCTL0}}$ signal transitions to a new state. Under normal conditions, the transition occurs during the “fifth-beat” of a four-beat burst, and so is ignored. However, using the MPC106 with WCBUF = 0 causes the $\overline{\text{BCTL0}}$ signal to transition during the fourth data beat.

It might be possible to use delay lines or trace routing to insure that $\overline{\text{BCTL0}}$ is significantly delayed. However, this would require expensive hardware delay lines or extensive board traces. The simplest solution is to use external logic to insert a one-clock delay on the $\overline{\text{BCTL0}}$ signal between the MPC106 and the registered data bus buffer. This ensures that the registered buffer accepts clock inputs during the last beat of a burst. With the addition of this delay, the data to the SDRAM is sampled as indicated in the lower portion of Figure 4.

The $\overline{\text{BCTL}}$[0–1] signals are driven well in advance of the actual memory cycle, so the propagation delay of the delaying device is not usually a critical factor, as long as the delayed $\overline{\text{BCTL0}}$ is available and stable on the next clock edge. Depending on the bus speed, the logic may even be incorporated into a spare PAL or FPGA, if available. Note that most registered DIMMs are 3.3V or less, so appropriate low-voltage logic must be used.

Figure 5 shows the needed hardware change. In this case a simple 74LVT74 D flip-flop is used to insert the one-clock delay.

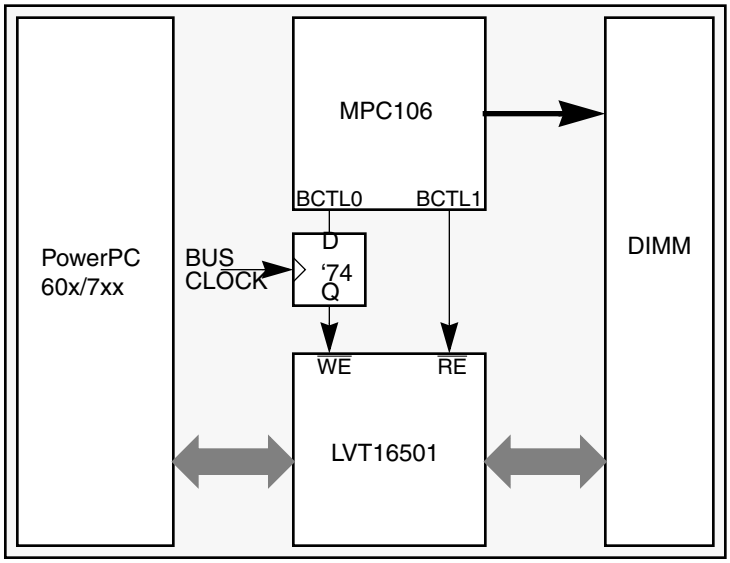

Figure 5. MPC106 Registered DIMM Architecture

The MPC106 buffer control signals $\overline{\text{BCTL}}$[0–1] can be configured to any of the definitions found in Figure 1.

Table 1. Available $\overline{\text{BCTL}}$[0–1] Configurations

501_MODE	BUF_MODE	BCTL0	BCTL1	Example
0	0	WE	RE	SN74ALVCH16501
0	1	W/$\overline{\text{R}}$	OE	—
1	0	WE	RE	SN74ALVCH162601
1	1	R/$\overline{\text{W}}$	OE	—

However, since the $\overline{\text{BCTL0}}$ delay is for write cycles only, then the registered data bus buffers must support separate read and write signals (MCCR2[BUF_MODE] = 0). As shown in Figure 1, only the first and third configurations can be supported; these are compatible with the readily available 16501 and 16601 series of registered buffers, respectively.

5 Example

Figure 6 shows an example of the connections of a large (256 MB or greater) memory array using registered DIMMs and the MPC106. Only essential memory connections are shown. For power, clocking, PCI and other connections refer to the various documents (user's manuals, hardware reference manuals, application notes, and reference platform schematics) available from the Freescale website: http://www.mot.com/SPS/PowerPC.

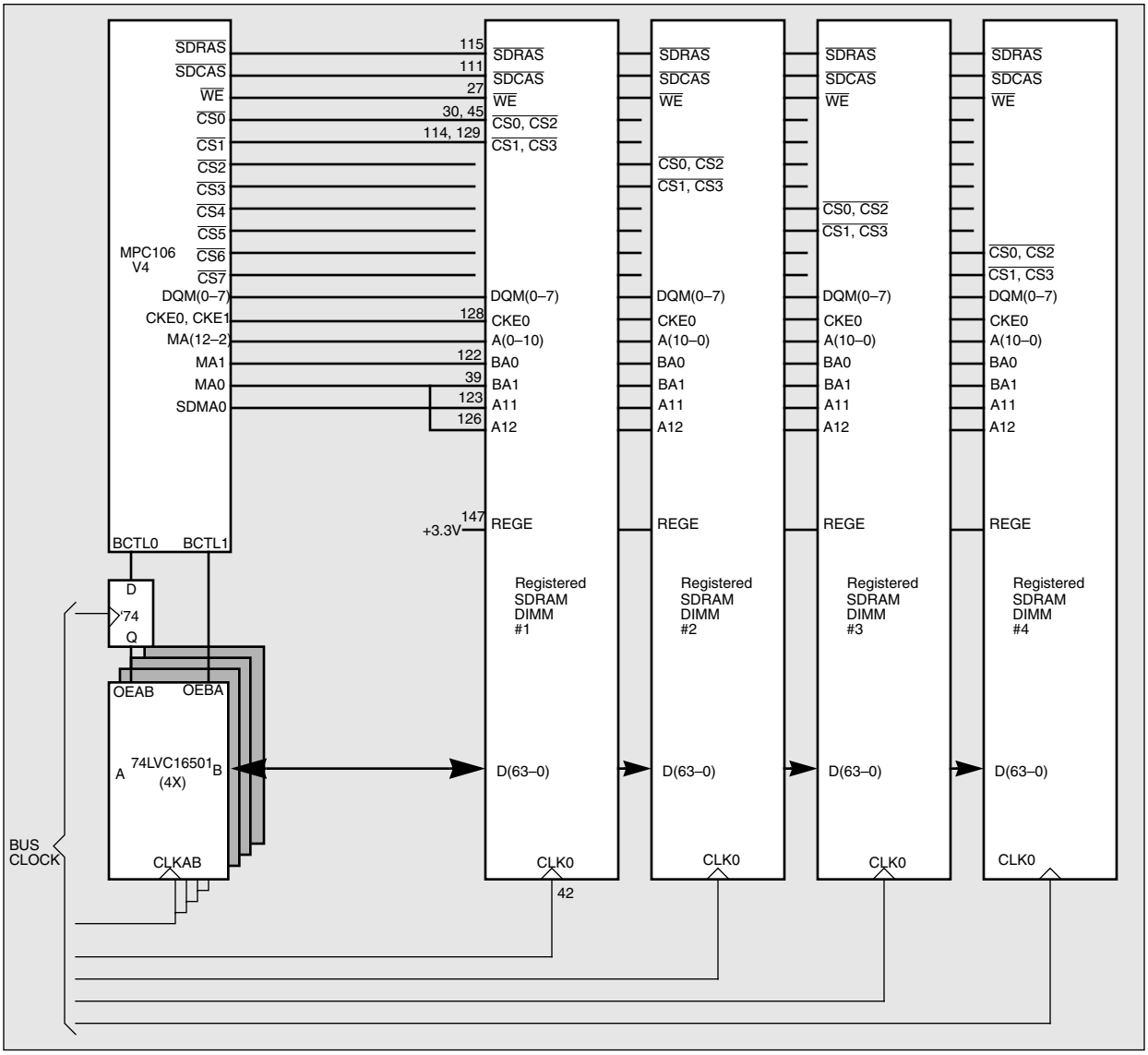

Figure 6. Example System

6 Summary

Although the MPC106 was not designed to directly control registered DIMMs, by implementing the following software and hardware modifications, the MPC106 can be made to support registered SDRAM modules:

- Use registered data-bus buffers (for example, 16501, 16601, or 16952 type buffers).
- Use external hardware to insert a one-clock-cycle delay on the $\overline{\text{BCTL0}}$ signal between the MPC106 and the data bus buffers. Leave $\overline{\text{BCTL1}}$ unchanged.
- Clear MCCR4[WCBUF]
- Clear MCCR2[BUF_MODE] to establish separate read and write controls.
- Increase the setting for MCCR4[RDLAT] by one clock cycle.

With these modifications, the MPC106 can support registered DIMMs.

7 Revision History

Table 2 shows the revision history of this document.

Table 2. Revision History

Revision Number	Changes
0.0	Initial release
0.1	Nontechnical reformatting

THIS PAGE INTENTIONALLY LEFT BLANK

How to Reach Us:

Home Page:
www.freescale.com

E-mail:
support@freescale.com

USA/Europe or Locations Not Listed:
Freescale Semiconductor
Technical Information Center, CH370
1300 N. Alma School Road
Chandler, Arizona 85224
+1-800-521-6274 or +1-480-768-2130
support@freescale.com

Europe, Middle East, and Africa:
Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
support@freescale.com

Japan:
Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku,
Tokyo 153-0064
Japan
0120 191014 or +81 3 5437 9125
support.japan@freescale.com

Asia/Pacific:
Freescale Semiconductor Hong Kong Ltd.
Technical Information Center
2 Dai King Street
Tai Po Industrial Estate
Tai Po, N.T., Hong Kong
+800 2666 8080
support.asia@freescale.com

For Literature Requests Only:
Freescale Semiconductor Literature Distribution Center
P.O. Box 5405
Denver, Colorado 80217
1-800-441-2447 or 303-675-2140
Fax: 303-675-2150
LDCForFreescaleSemiconductor@hibbertgroup.com

freescale™
semiconductor

AN1768/D