

# MPC8260 GPCM Timing Diagram

## 1 Introduction

The following timing diagrams are generated based on the simulations:

- 60x bus access
- Local bus access
- Early termination by GTA
- Read-Modify-Write Cycle to GPCM
- ARTRY cycle

The general-purpose chip-select machine (GPCM) allows a glueless and flexible interface between the MPC8260, SRAM, EPROM, FEPRM, ROM device, and external peripherals.

## 2 60x Bus Access

60x bus access is partitioned to two sub-groups: single MPC8260 mode and external bus mode (60x compatible).

### 2.1 Single MPC8260 Bus Mode

The single MPC8260 bus mode is entered by clearing the HardResetConfiguration[EBM] bit during the configuration.

#### Contents

1. Introduction . . . . .	1
2. 60x Bus Access . . . . .	1
3. Local Bus Access . . . . .	11
4. Early Termination by GTA . . . . .	14
5. ARTRY Cycle . . . . .	19

Under this bus mode, the MPC8260 is the only bus master device in the system. The internal memory controller controls all devices on the external bus. All the 60x bus signals can not be used by the slave device because the addresses have memory timing, not address/data tenure timing.

All the timing diagrams are for 32-bit access, while port sizes are either 64 or 16. Other combinations of data size and port size are similar.

Assemble Language Example:

```
# Set SIUMCR[BCTLC] to control the BCTL0 polarity
```

```
addis r2, r0, 0x0000
```

```
ori r2, r2, 0x0000
```

```
addis r1, r0, 0x0f01
```

```
ori r1, r1, 0x0000
```

```
stw r2, 0x0000(r1)
```

```
# Set OR1 to 0xffff_0860
```

```
addis r2, r0, 0xffff
```

```
ori r2, r2, 0x0860
```

```
addis r1, r0, 0x0f01
```

```
ori r1, r1, 0x010c
```

```
stw r2, 0x0000(r1)
```

```
# Set BR1 = 0x0100_0001
```

```
addis r2, r0, 0x0100
```

```
ori r2, r2, 0x0001
```

```
addis r1, r0, 0x0f01
```

```
ori r1, r1, 0x0108
```

```
stw r2, 0x0000(r1)
```

```
# Set address @r2
```

```
addis r2, r0, 0x0100
```

```
ori r2, r2, 0x0000
```

```
# Read from 0x0100_0018 which is controlled by GPCM
```

```
lwz r3, 0x0018(r2)
```

```
# Write to 0x0100_0018 which is controlled by GPCM
```

```
stw r1, 0x0018(r2)
```

### Notes:

Basic steps to program GPCM:

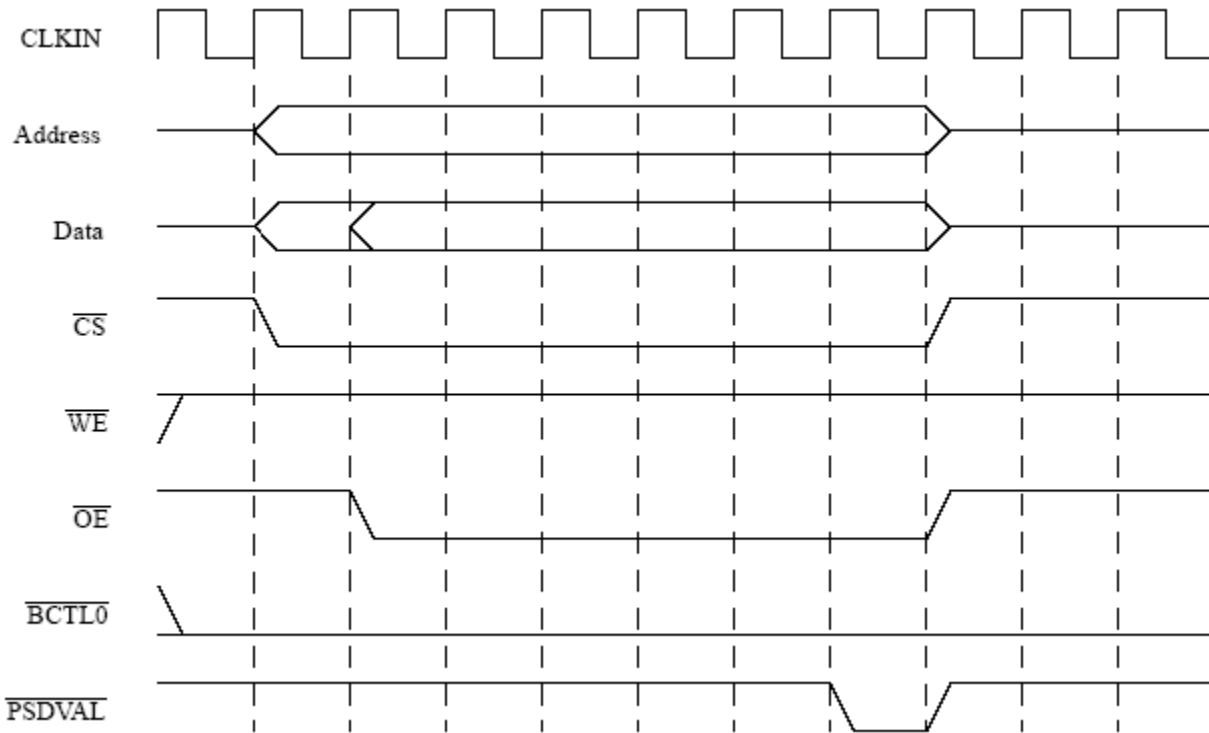
(1) For 60x bus, program SIUMCR[BCTLC] for BCTL0 polarity.

- (2) Program ORx for CS and WE timing<sup>(1)</sup>.
- (3) Program BRx[MS] to select GPCM and appropriate bus, BRx[PS] for port size, etc.

Reference:

(1) *MPC8260 PowerQUICC II User's Manual*, P10-53.

### 2.1.1 60x Access in Single MPC8260 Mode, Port Size=64, 32-Bit Read



**Figure 1. GPCM, 60x, Port Size=64, 32-Bit Read, 5 Cycle Wait**

- <sup>1</sup> When GPCM is used for peripheral device, CS is the strobe signal for both read and write. BCTL0 is used to indicate read/write direction. Peripherals drive data bus if CS is asserted.  
For memory device, data output is enabled by OE and valid after OE is asserted.
- <sup>2</sup> CS timing is controlled by programming ORx.

## 2.1.2 60x Access in Single MPC8260 Mode, Port Size=64, 32-Bit Write

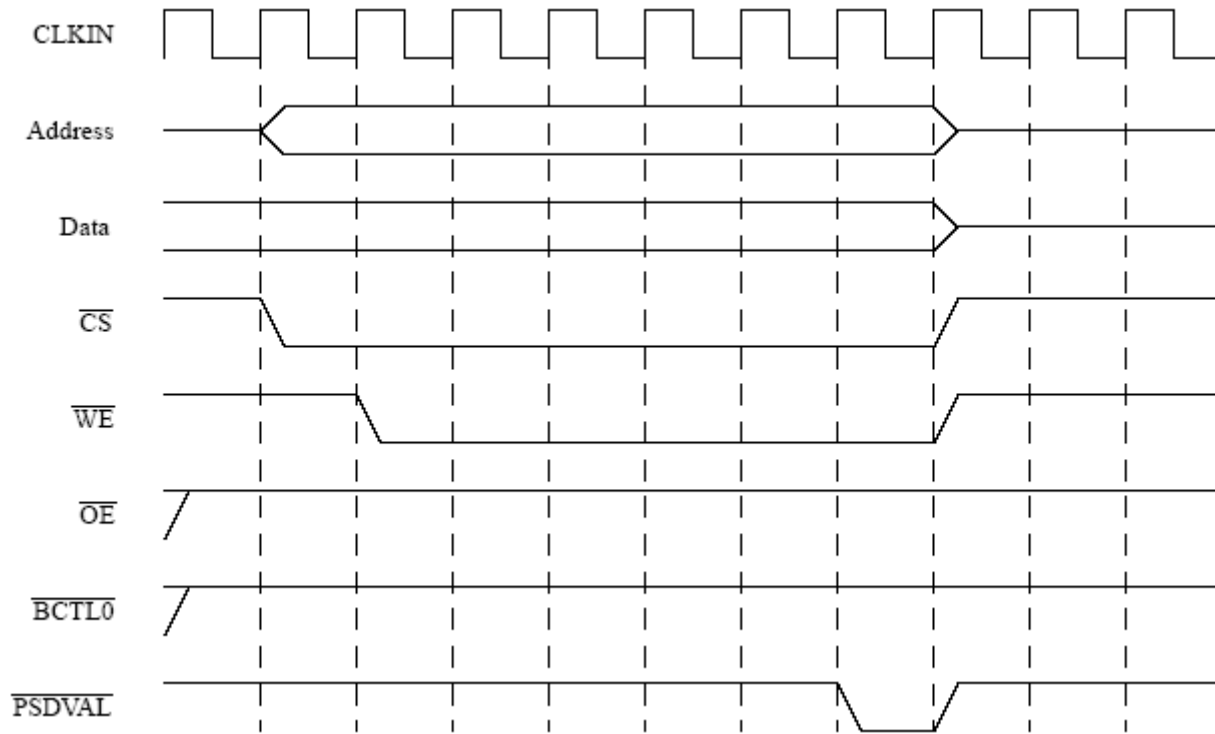
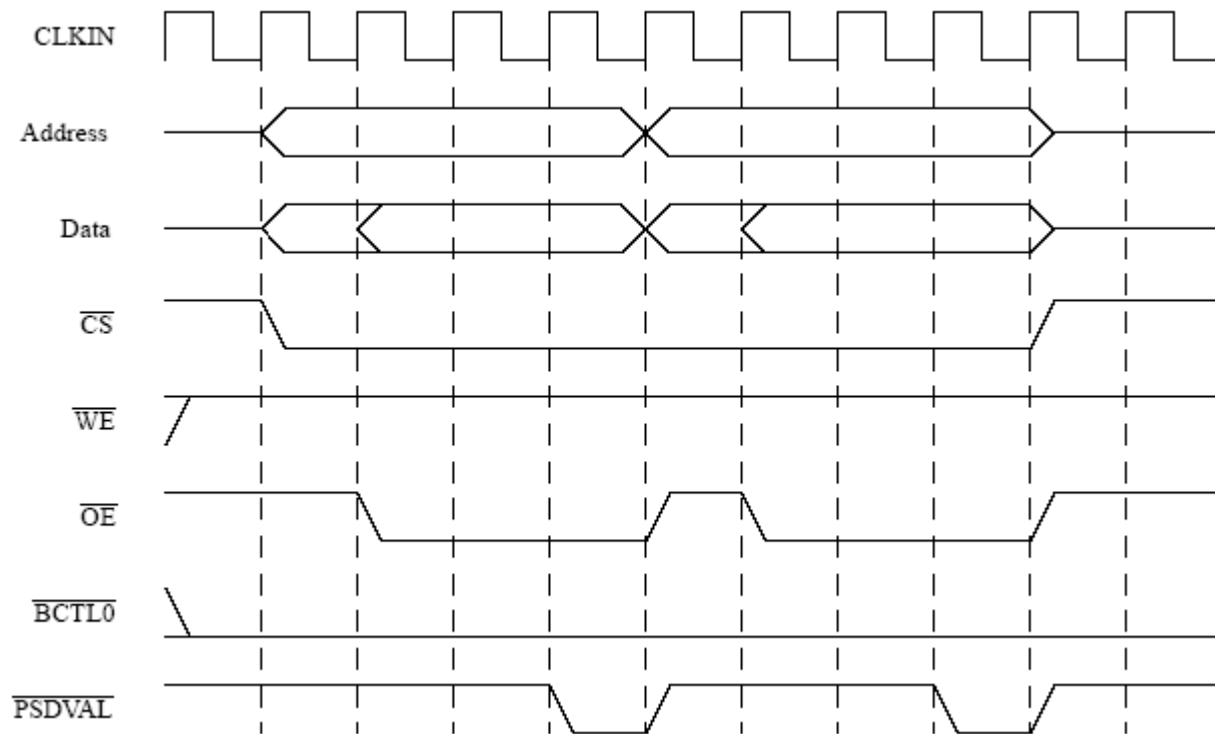


Figure 2. GPCM, 60x, Port Size=64, 32-Bit Write, 5 Cycle Wait

- <sup>1</sup> Data bus might be valid several cycles earlier than CS is asserted.
- <sup>2</sup> CS, WE timing is controlled by programming ORx.

### 2.1.3 60x Access in Single MPC8260 Mode, Port Size=16, 32-Bit Read



**Figure 3. GPCM, 60x, Port Size=16, 32-Bit Read, 2 Cycle Wait**

- <sup>1</sup> Use the same program except BRx[PS] = 10 to set the port size = 16 bits. Due to port size limitation, the 32-bit access is split to 2 beat 16-bit access.
- <sup>2</sup> When GPCM is used for peripheral device, CS is strobe for both read and write. BCTL0 is used to indicate read/write direction. Peripherals drive data bus if CS is asserted. For memory device, data output is enabled by OE and valid after OE is asserted.
- <sup>3</sup> CS timing is controlled by programming ORx.

## 2.1.4 60x Access in Single MPC8260 Mode, Port Size=16, 32-Bit Write

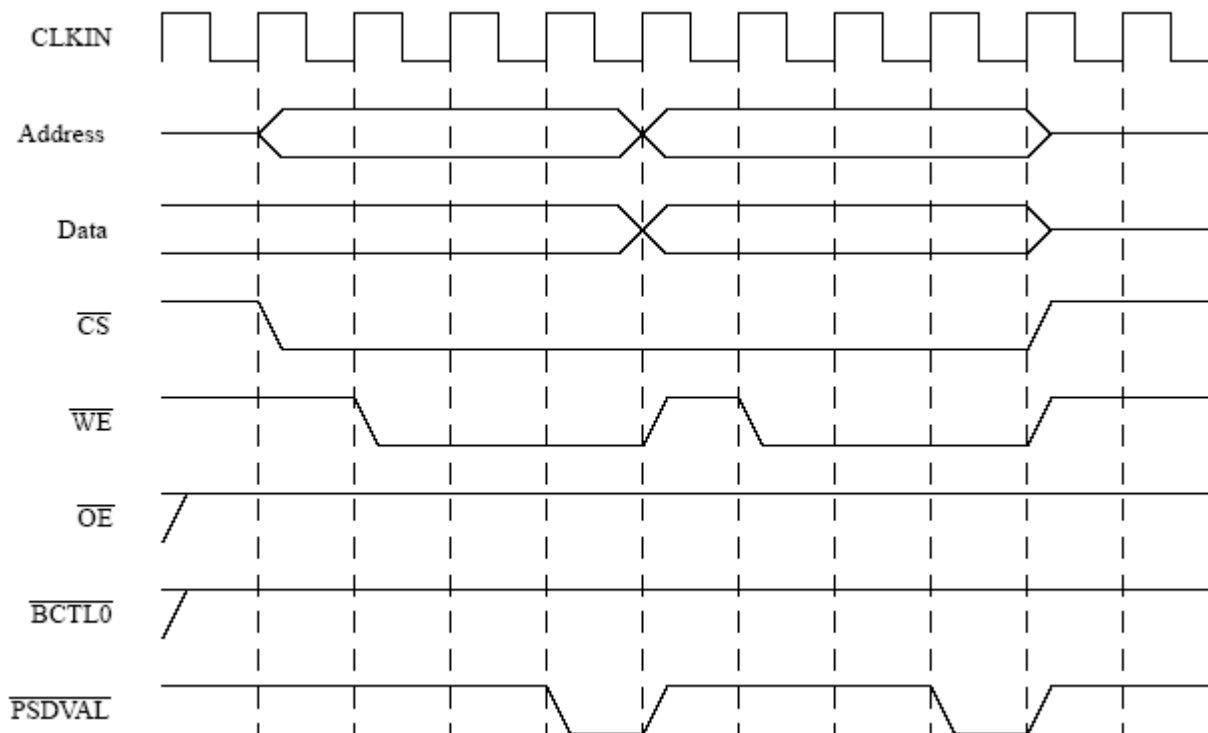


Figure 4. GPCM, 60x, Port Size=16, 32-Bit Write, 2 Cycle Wait

- <sup>1</sup> Use the same program except BRx[PS] = 10 to set the port size = 16 bits. Due to port size limitation, the 32-bit access is split to 2 beat 16-bit access.
- <sup>2</sup> When GPCM is used for peripheral device, CS is strobe for both read and write. BCTL0 is used to indicate read/write direction. Peripherals drive data bus if CS is asserted. For memory device, data output is enabled by OE and valid after OE is asserted.
- <sup>3</sup> CS, WE timing is controlled by programming ORx.

## 2.2 External Bus Mode (60x Compatible)

The 60x-compatible bus mode is entered by setting the HardResetConfiguration[EBM] bit during the configuration. Under this bus mode, the lower bits of the 60x address to the memory is controlled by baddr[27:31]. Note that baddr[29:31] is muxed with other signals. The muxing is controlled by SIUMCR[L2CPC]. baddr[29:31] function is chosen when L2CPC bits of configuration word are set to 10 during power up or programming 10 directly to SIUMCR after the configuration.

Assemble Language Example:

Same as GPCM single MPC8260 mode.

### NOTE

For external bus mode, all the 60x bus signals are added to better illustrate the 60x bus nature of the access.

## 2.2.1 60x Bus Access in 60x-Compatible Mode, Port Size = 64, 32-Bit Read

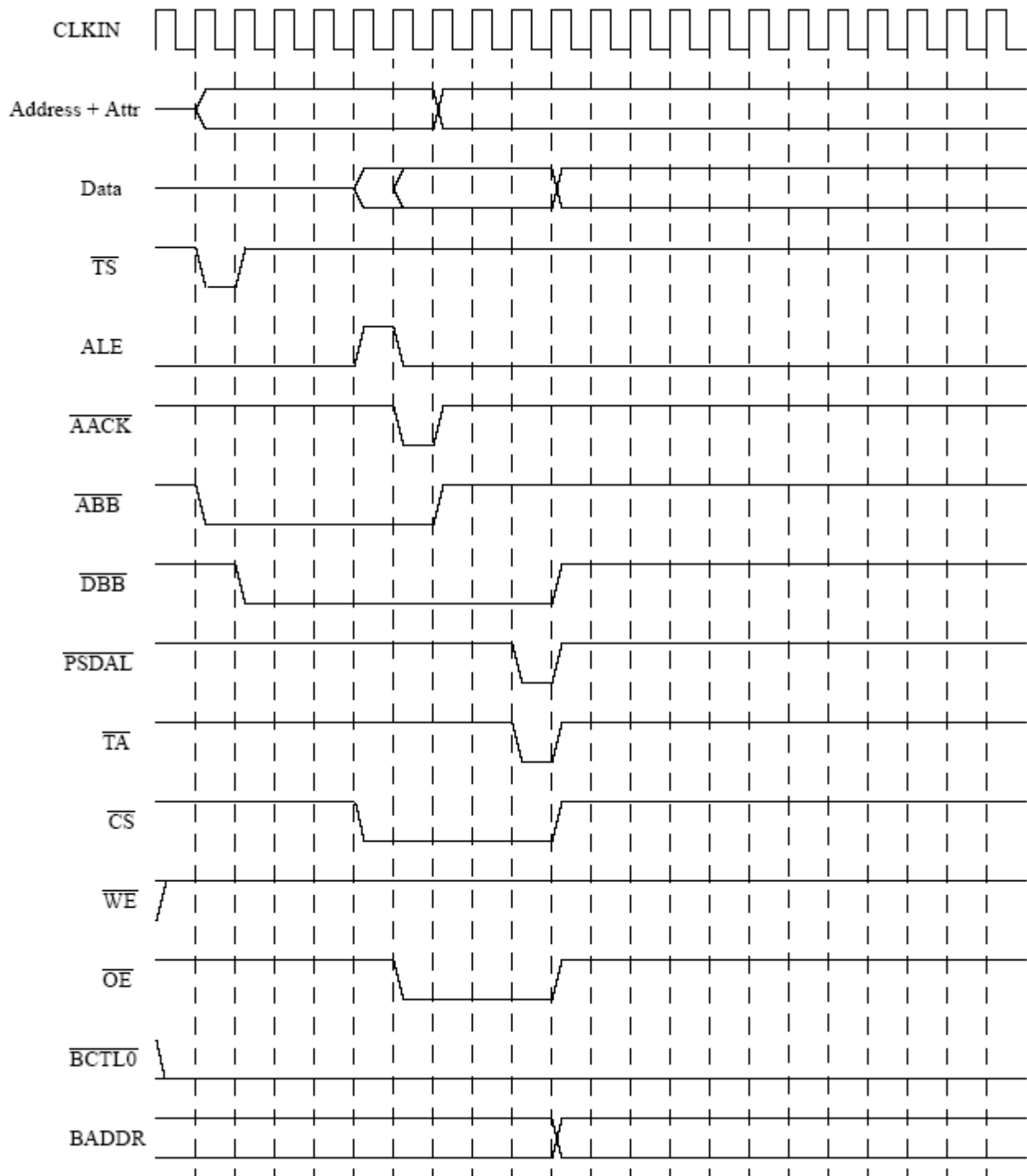


Figure 5. GPCM, 60x, Port Size=64, 32-Bit Read, 3 Cycle Wait

<sup>1</sup> In 60x-compatible mode, ALE should be used to latch the address bus.

## 2.2.2 60x Bus Access in 60x-Compatible Mode, Port Size = 64, 32-Bit Write

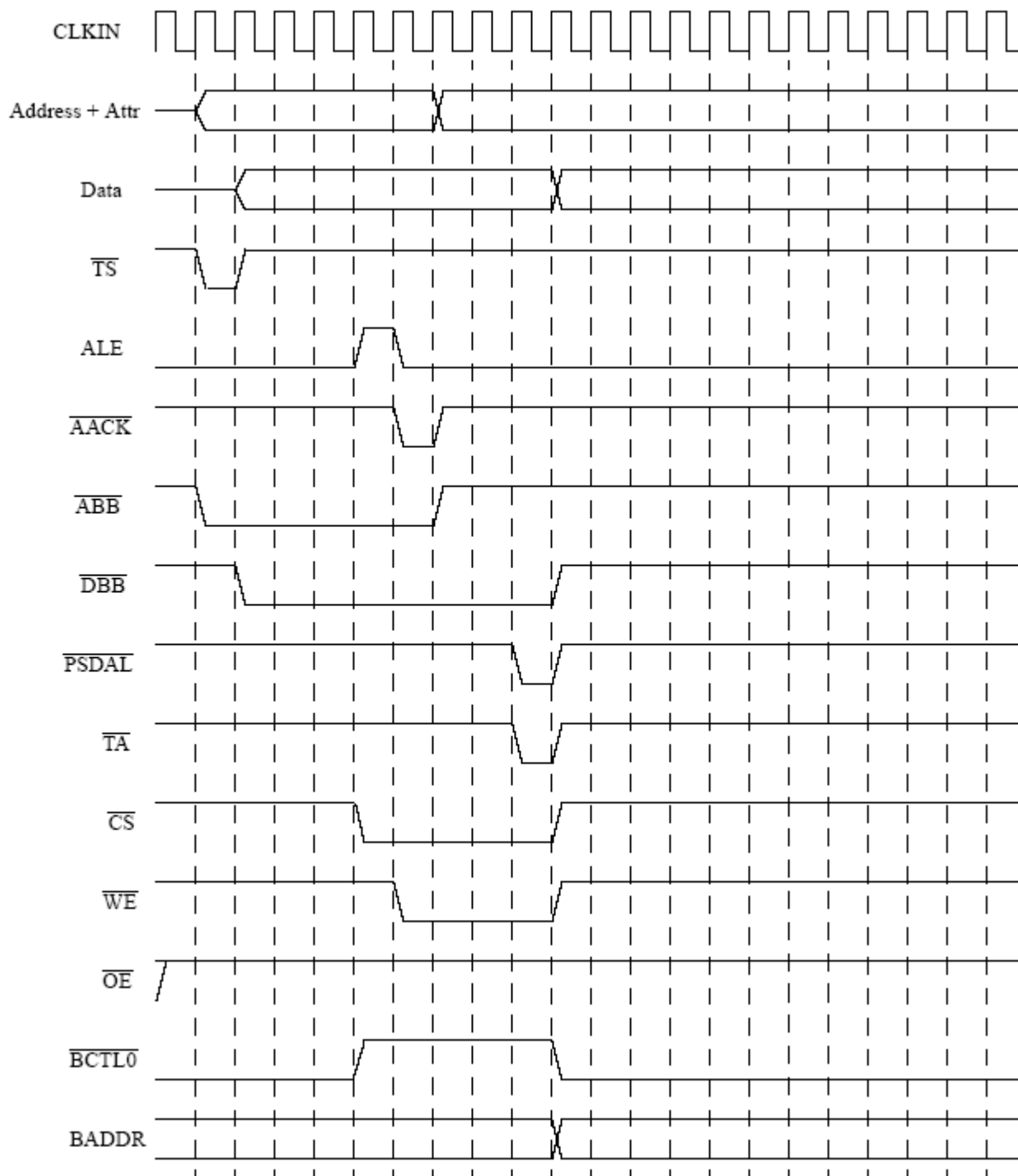


Figure 6. GPCM, 60x, Port Size=64, 32-Bit Write, 3 Cycle Wait



### 2.2.3 60x Bus Access in 60x-Compatible Mode, Port Size = 16, 32-Bit Read

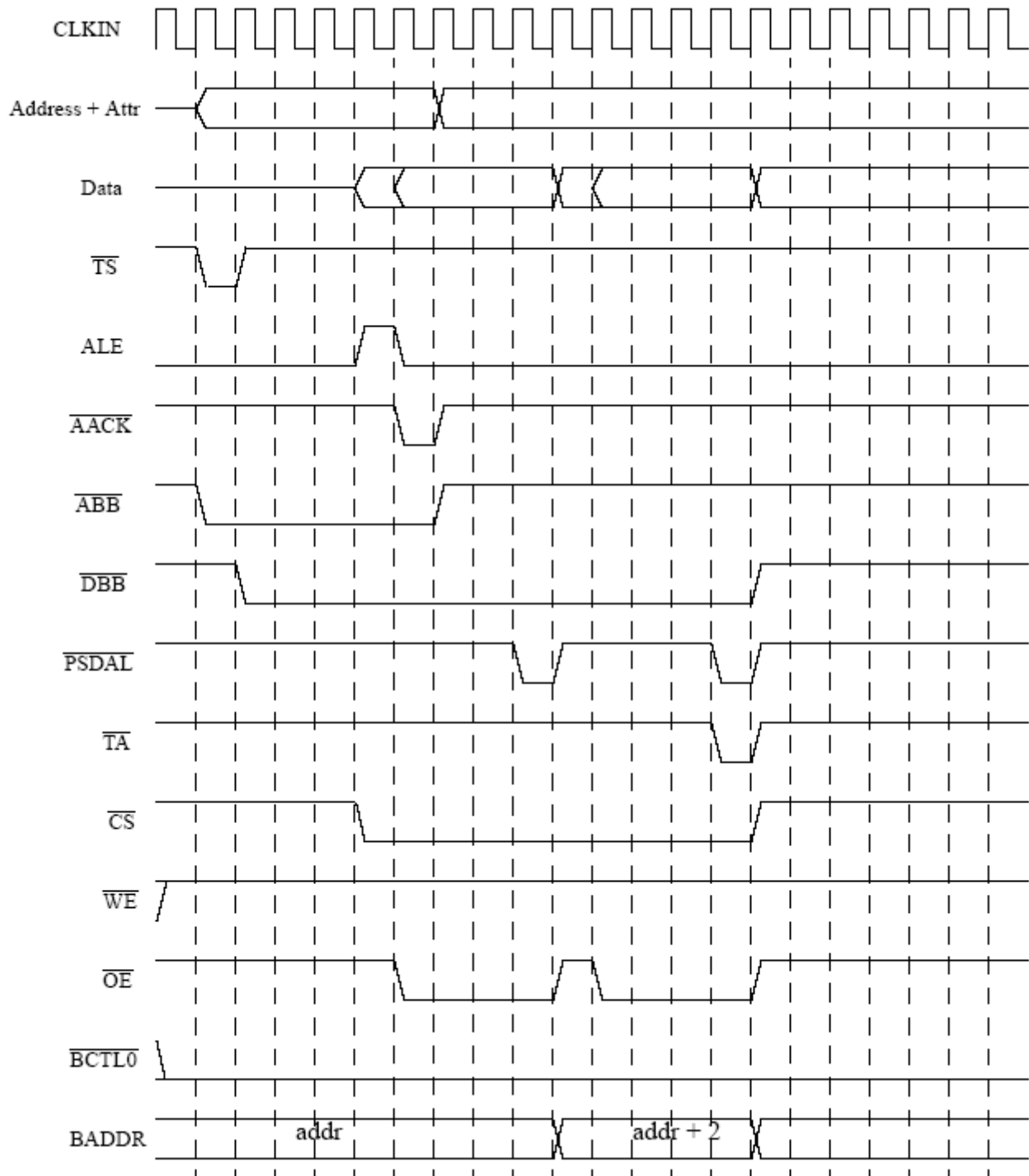


Figure 7. GPCM, 60x, Port Size=16, 32-Bit Read, 3 Cycle Wait

### 2.2.4 60x Bus Access in 60x-Compatible Mode, Port Size = 16, 32-Bit Write

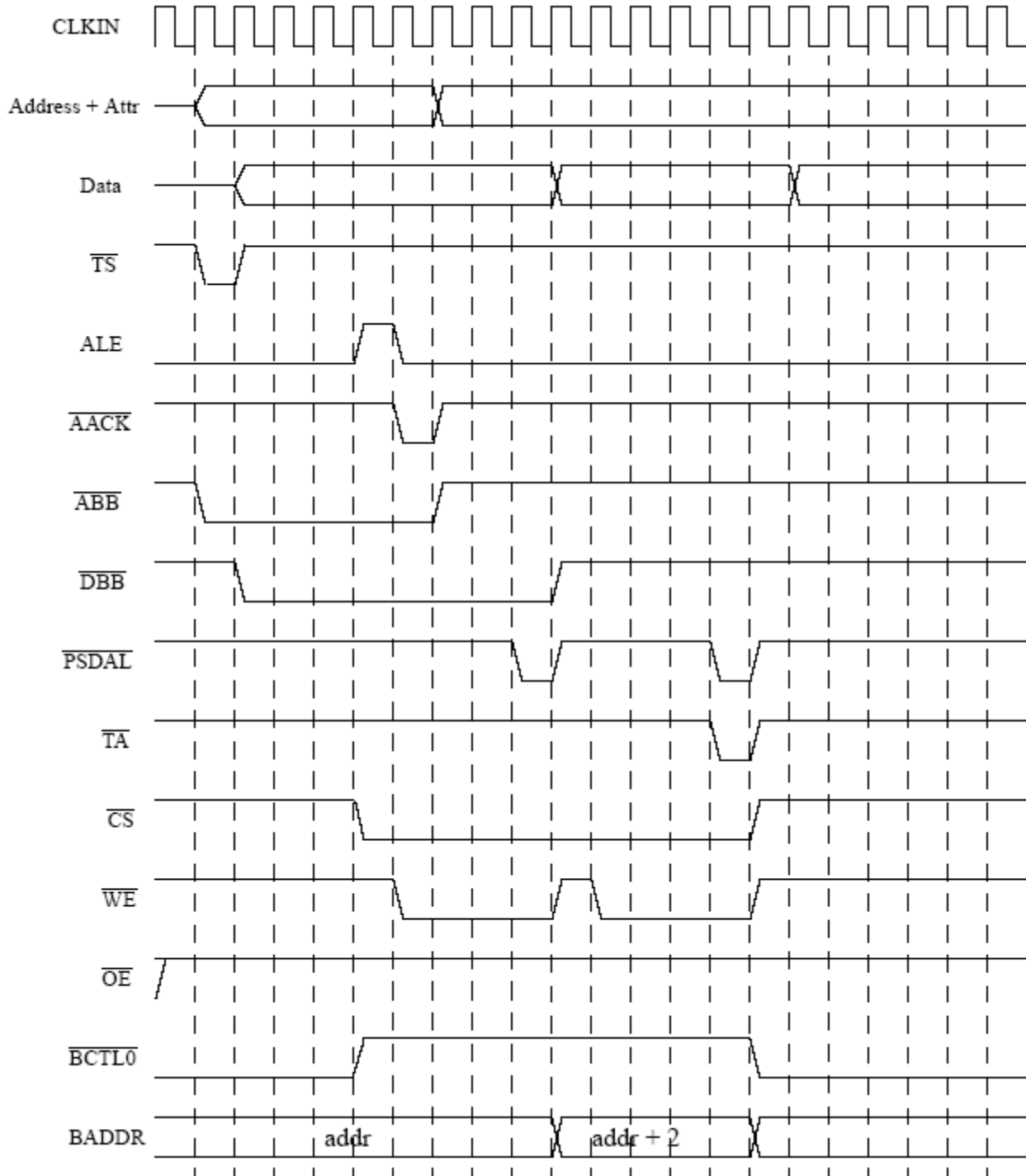


Figure 8. GPCM, 60x, Port Size=16, 32-Bit Write, 3 Cycle Wait

### 3 Local Bus Access

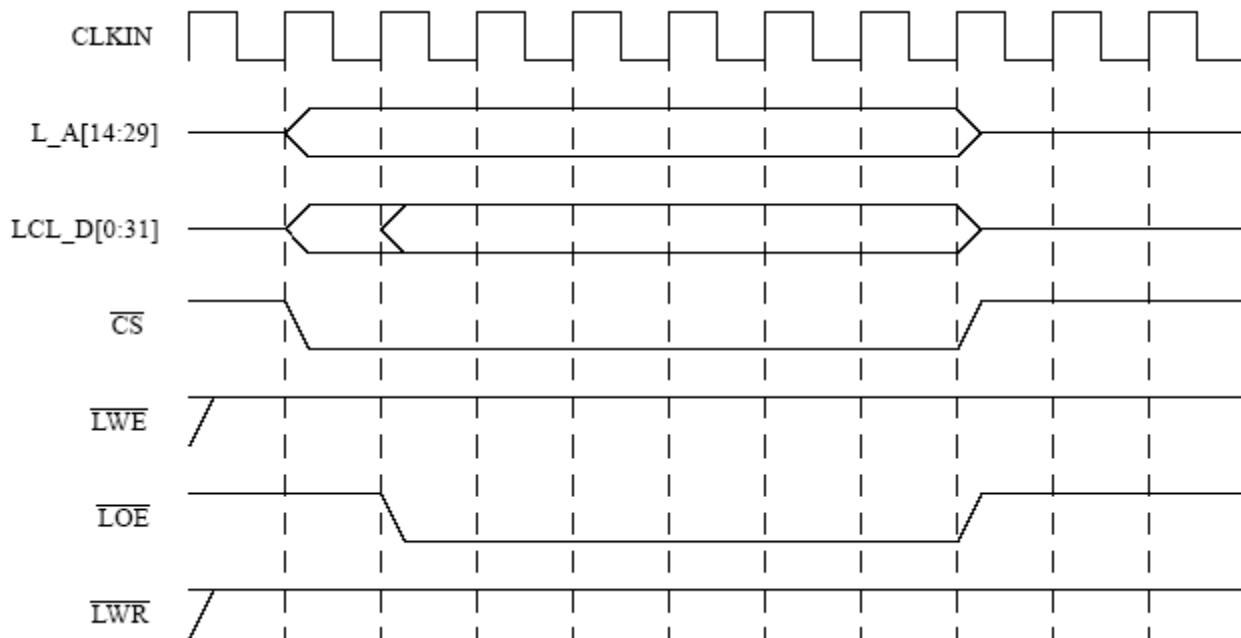
Local bus access functions the same for both Single MPC8260 mode and 60x-compatible mode.

Local bus has its own signals:

- LOE for read output enable
- LWE for write enable
- LWR for read/write direction: 0-write; 1-read.

Local address pins are muxed with PCI signals. Local bus function of those pins is chosen by setting `HardResetConfiguration[L2CPC]` to 00 during configuration or programming them to 00 after configuration.

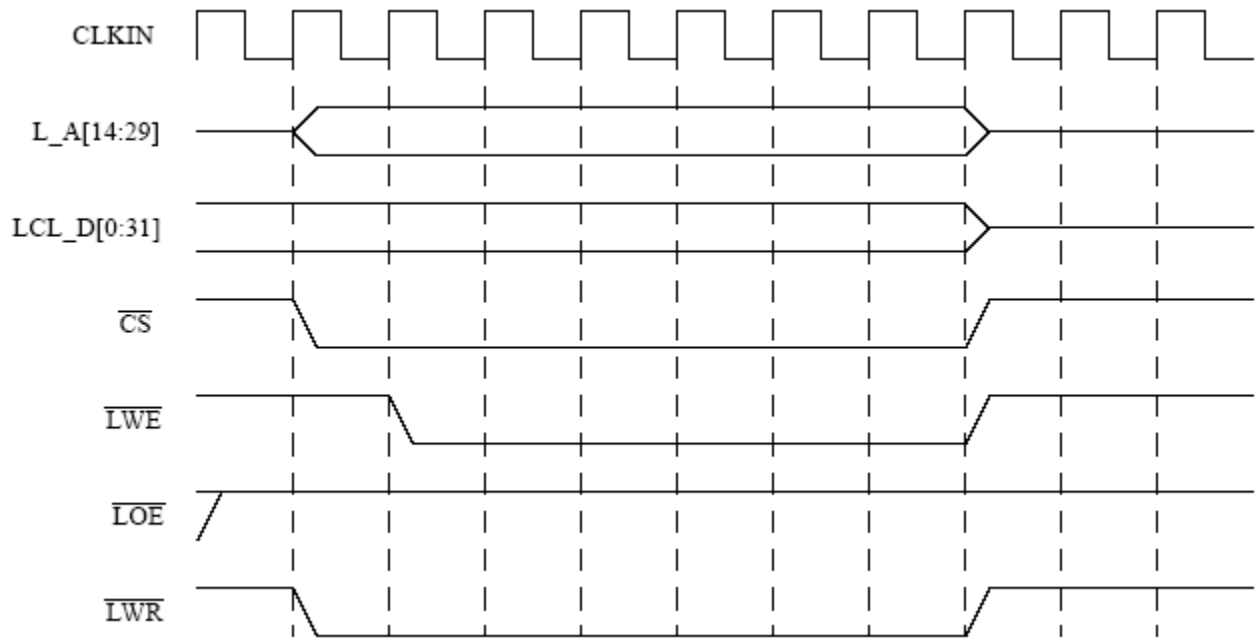
#### 3.1 Local Bus Access, Port Size=32, 32-Bit Read



**Figure 9. GPCM, Local Bus, Port Size=32, 32-Bit Read, 5 Cycle Wait**

- <sup>1</sup> When GPCM is used for peripheral device, CS is strobe for both read and write. Data bus is valid after CS is asserted. For memory device, data bus will be valid after LOE is asserted.
- <sup>2</sup> CS timing is controlled by programming ORx.

### 3.2 Local Bus Access, Port Size=32, 32-Bit Write



**Figure 10. GPCM, Local Bus, Port Size=32, 32-Bit Write, 5 Cycle Wait**

- <sup>1</sup> Data bus is valid several cycles earlier than CS is asserted.
- <sup>2</sup> CS, LWE timing is controlled by programming ORx.

### 3.3 Local Bus Access, Port Size=16, 32-Bit Read

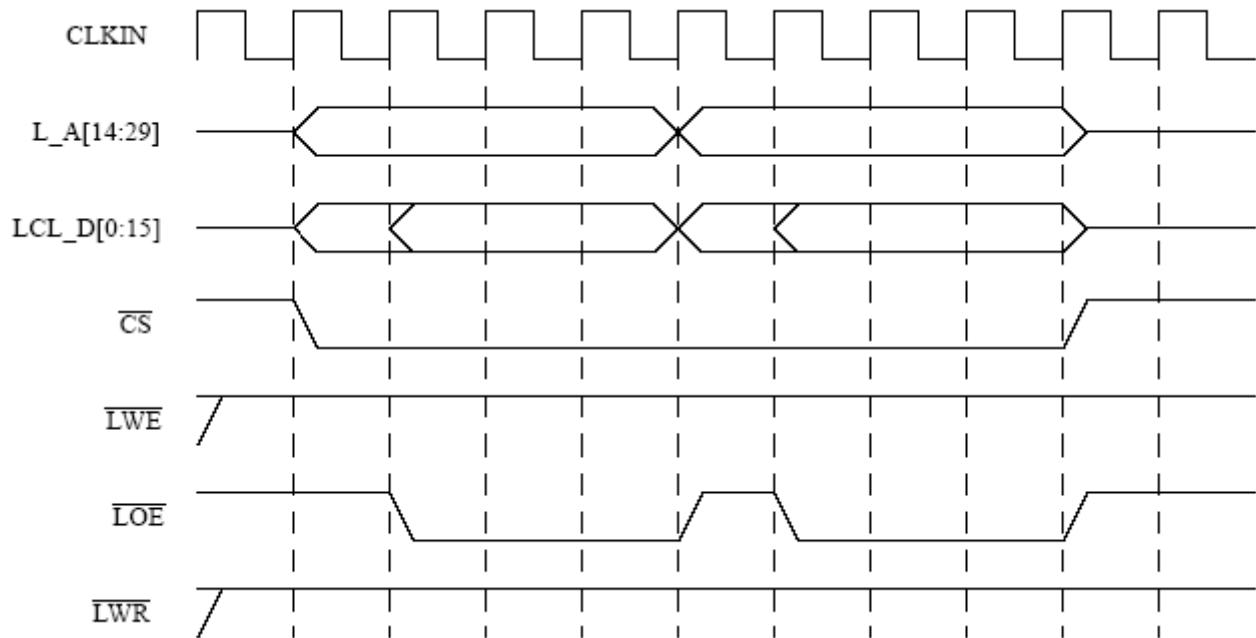


Figure 11. GPCM, Local Bus, Port Size=16, 32-Bit Read, 2 Cycle Wait

- <sup>1</sup> Use the same program except BRx[PS] = 10 to set the port size = 16 bits. Due to port size limitation, the 32-bit access is split to 2 beat 16-bit access.
- <sup>2</sup> When GPCM is used for peripheral device, CS is strobe for both read and write. LWE is used to indicate read/write direction. Data bus is valid after CS is asserted. For memory device, data output is enabled by OE and valid after OE is asserted.
- <sup>3</sup> CS timing is controlled by programming ORx.

### 3.4 Local Bus Access, Port Size=16, 32-Bit Write

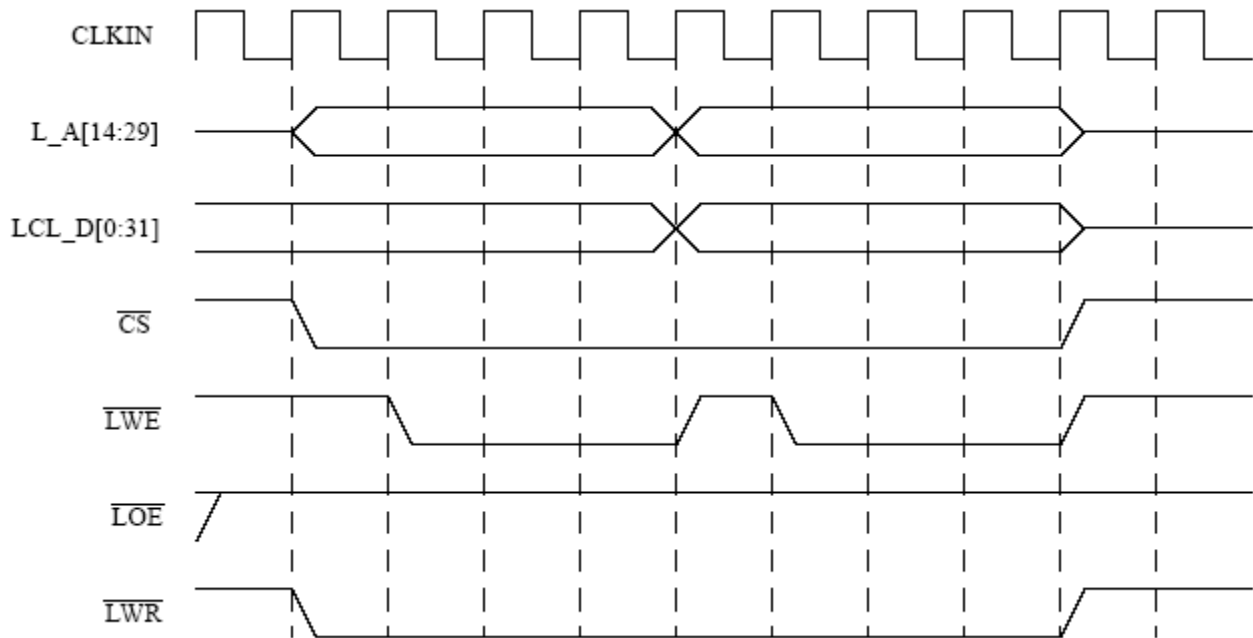


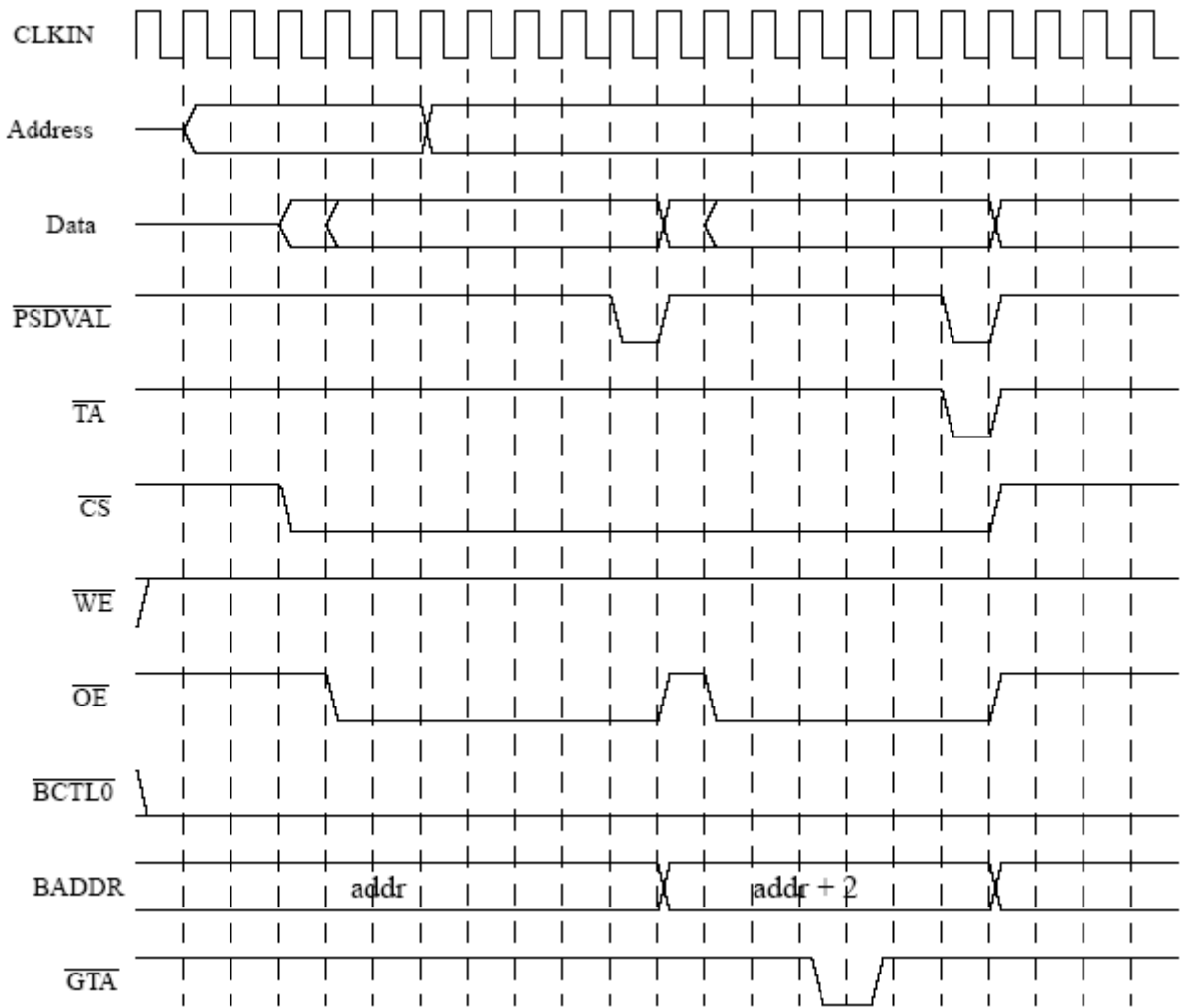
Figure 12. GPCM, Local Bus, Port Size=16, 32-Bit Write, 2 Cycle Wait

- <sup>1</sup> Use the same program except BR1[PS]=10 to set the port size=16 bits. Due to port size limitation, the 32-bit access is split to 2 beat 16-bit access.
- <sup>2</sup> When GPCM is used for peripheral device, CS is strobe for both read and write. LWE is used to indicate read/write direction. Data bus is valid after CS is asserted.  
For memory device, data output is enabled by OE and valid after OE is asserted.
- <sup>3</sup> CS, LWE timing is controlled by programming ORx.

## 4 Early Termination by GTA

If ORx[SETA]=0, the PSDVAL is generated by the memory controller. But if GTA is asserted at least two clocks earlier before the wait state counter has expired, the cycle will terminate as a result of GTA assertion. Local bus access is terminated by LGTA.

## 4.1 GPCM 60x Bus Access Terminated by GTA



**Figure 13. GPCM, 60x Bus in 60x Compatible Mode, Port Size=16, 32-bit Read, 6 Cycle Wait**

- <sup>1</sup> Due to port size limitation, one 32-bit access is split to two 16-bit access. The first access is terminated normally by internal memory controller with 8 bus cycle wait.  
The second access is terminated earlier as a result of GTA assertion.

## 4.2 GPCM Local Bus Access Terminated by LGTA

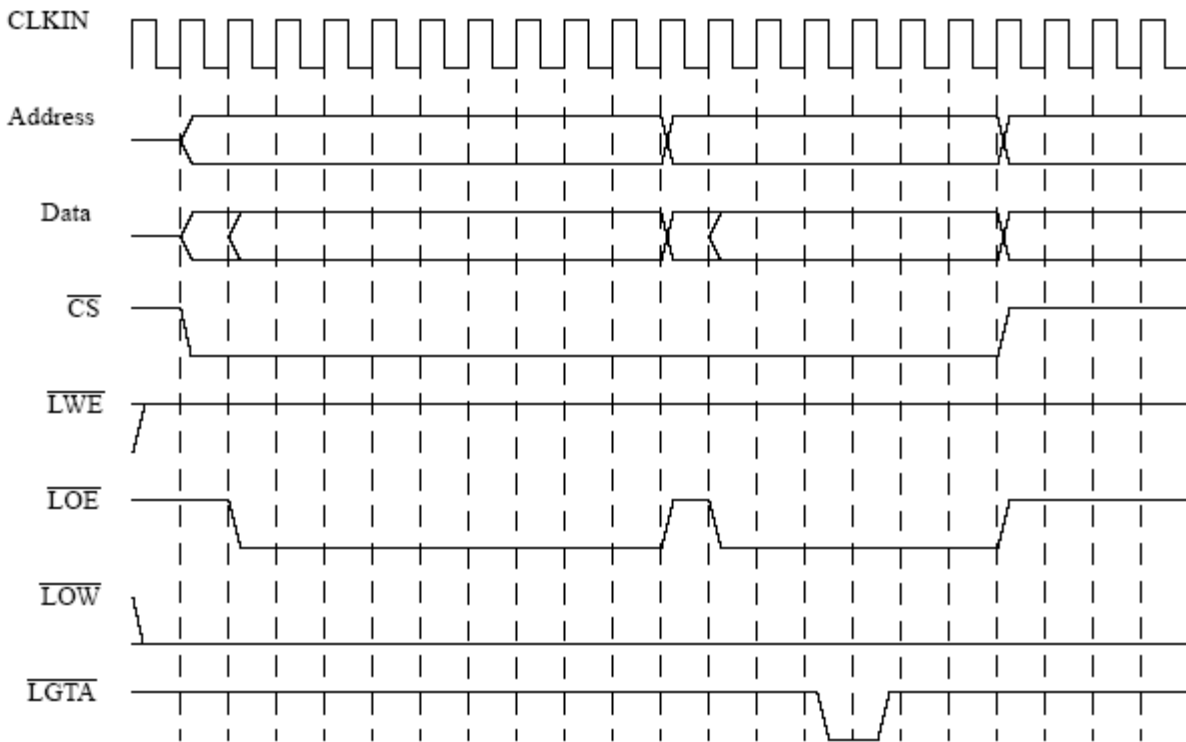


Figure 14. GPCM, Local Bus, Port Size=16, 32-Bit Read, 8 Cycle Wait

<sup>1</sup> Due to port size limitation, one 32-bit access is split to two 16-bit access. The first access is terminated normally by internal memory controller with 8 bus cycle wait. The second access is terminated earlier as a result of LGTA assertion.

## 4.3 Read-Modify-Write Cycle

If the GPCM is programmed to do read-modify-write parity checking or ECC correction and checking, every write access to memory that is less than the port size will cause read-modify-write cycle automatically.



## 4.4 Read-Modify-Write Cycle on 60x Bus in 60x-Compatible Mode

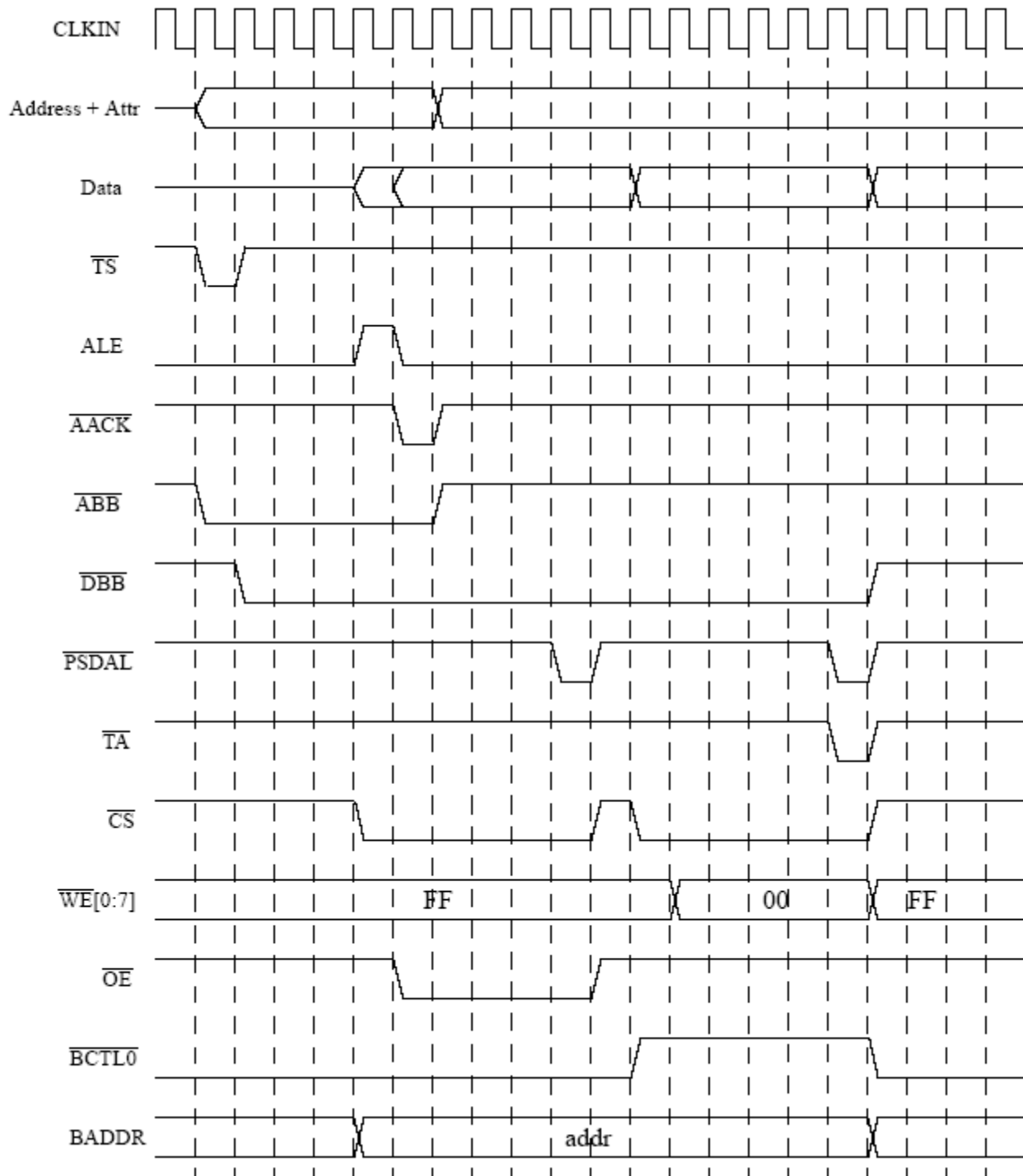


Figure 15. GPCM, 60x, Port Size=64, 32-Bit Write, 4 Cycle Wait

## 4.5 Read-Modify-Write Cycle on 60x Bus in Single MPC8260 Mode

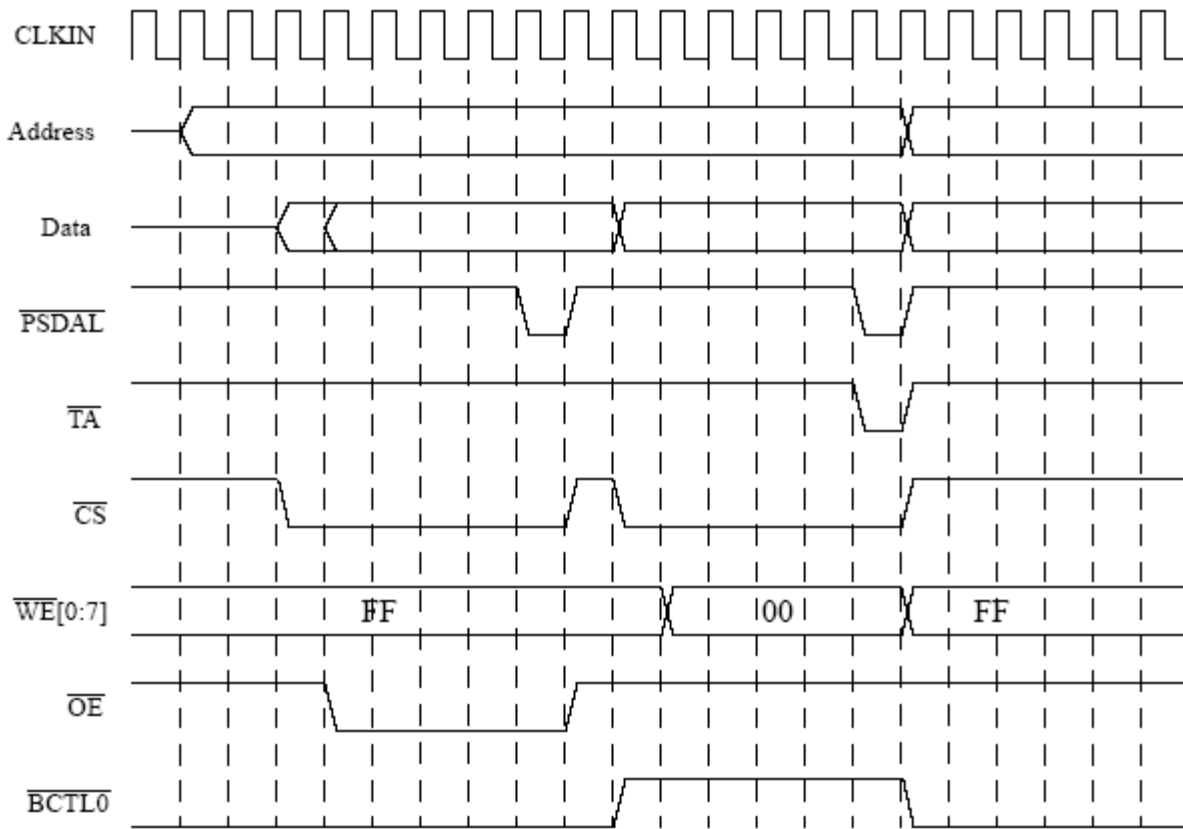


Figure 16. GPCM, 60x, Port Size=64, 32-Bit Write, 4 Cycle Wait

<sup>1</sup> The RMW cycle is triggered by 32-bit write to port with size of 64.

## 4.6 Read-Modify-Write Cycle on Local Bus

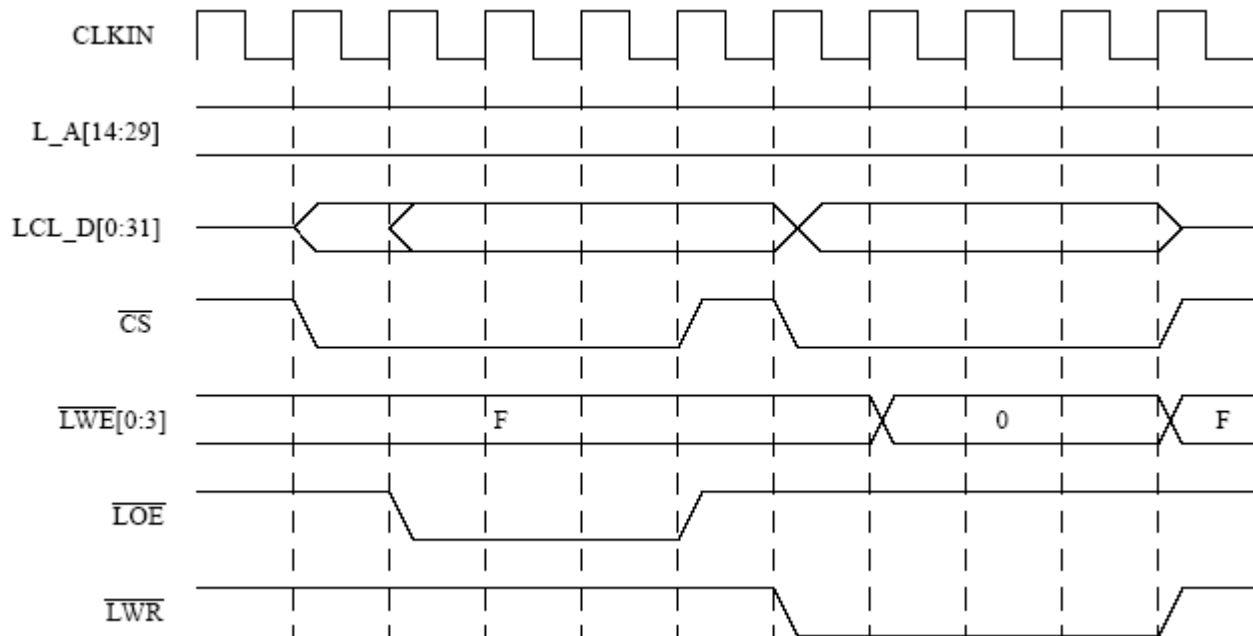


Figure 17. GPCM, Local Bus, Port Size=32, 32-Bit Write, 2 Cycle Wait

<sup>1</sup> The BRx[DECC] is set to 10 (RMW parity checking). ECC can only work for 64-bit port, so it can't be used for local bus. The RMW cycle is triggered by 32-bit write to port with size of 32. It is interesting to note that for local bus, if port size=32, any size write including 32-bit will trigger the RMW cycle.

## 5 ARTRY Cycle

In 60x-compatible mode, the address transfer can be terminated with the requirement to retry if ARTRY is asserted during the address tenure and through the cycle following AACK. The assertion causes the entire transaction (address and data tenure) to be rerun.

## 5.1 ARTRY Cycle in 60x-Compatible Mode, Port Size=32, 32-Bit Write

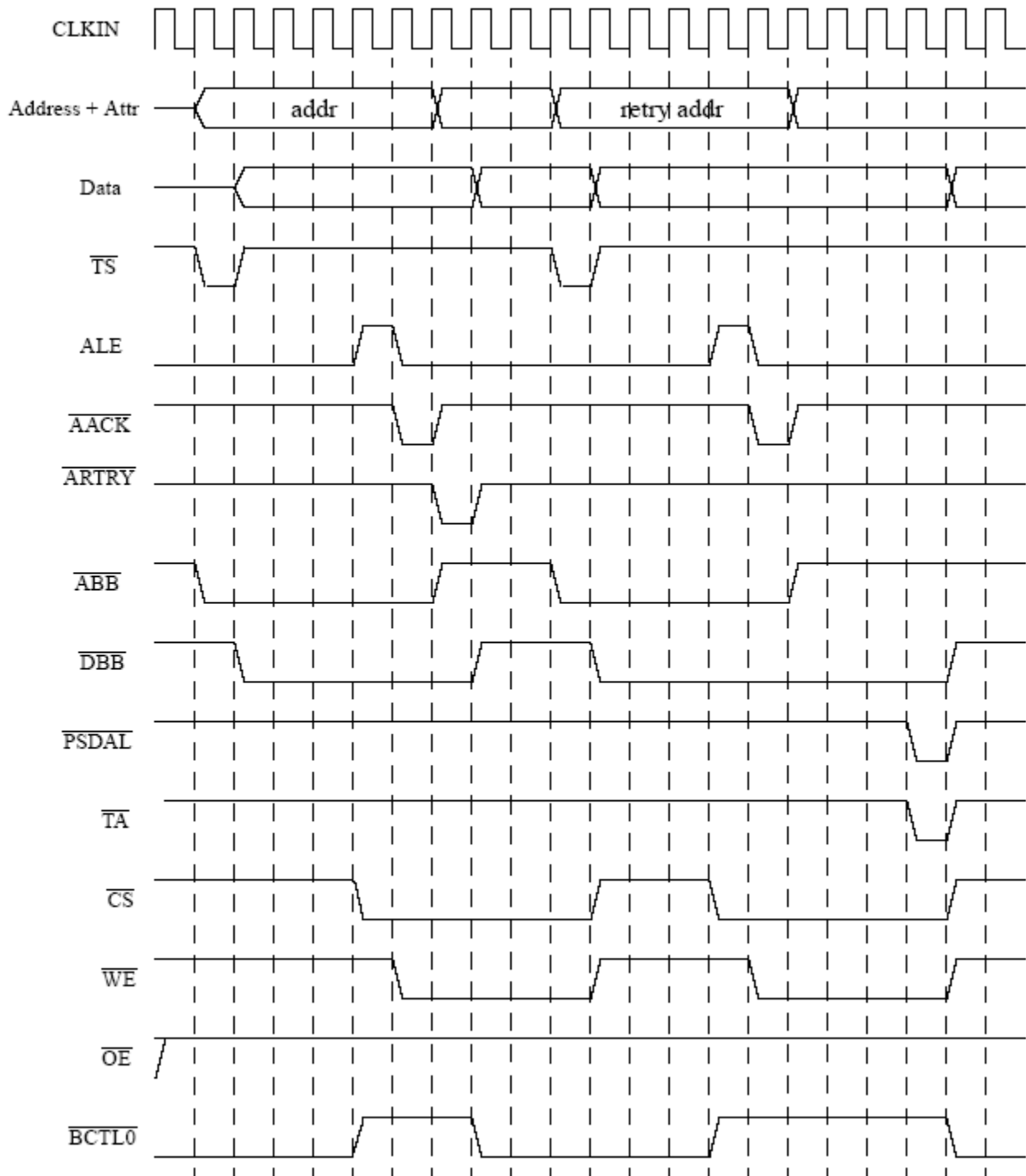


Figure 18. GPCM, 60x, Port Size=32, 32-Bit Write

**THIS PAGE INTENTIONALLY LEFT BLANK**

**THIS PAGE INTENTIONALLY LEFT BLANK**

**THIS PAGE INTENTIONALLY LEFT BLANK**

### **How to Reach Us:**

#### **Home Page:**

www.freescale.com

#### **email:**

support@freescale.com

#### **USA/Europe or Locations Not Listed:**

Freescale Semiconductor  
Technical Information Center, CH370  
1300 N. Alma School Road  
Chandler, Arizona 85224  
(800) 521-6274  
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  
+81 2666 8080  
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  
(800) 441-2447  
303-675-2140  
Fax: 303-675-2150  
LDCForFreescaleSemiconductor  
@hibbertgroup.com

Information in this document is provided solely to enable system and software implementers to use Freescale Semiconductor products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits or integrated circuits based on the information in this document.

Freescale Semiconductor reserves the right to make changes without further notice to any products herein. Freescale Semiconductor makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale Semiconductor assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters which may be provided in Freescale Semiconductor data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including "Typicals" must be validated for each customer application by customer's technical experts. Freescale Semiconductor does not convey any license under its patent rights nor the rights of others. Freescale Semiconductor products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Freescale Semiconductor product could create a situation where personal injury or death may occur. Should Buyer purchase or use Freescale Semiconductor products for any such unintended or unauthorized application, Buyer shall indemnify and hold Freescale Semiconductor and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Freescale Semiconductor was negligent regarding the design or manufacture of the part.

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. The described product is a PowerPC microprocessor core. The PowerPC name is a trademark of IBM Corp. and is used under license. All other product or service names are the property of their respective owners.

© Freescale Semiconductor, Inc., 2004, 2005.

Document Number: AN2176  
Rev. 2  
11/2005