AN2106/D
Rev. 0.1, 6/2003

PowerPC™ MPX Bus
Implementation Differences

This document details differences in the implementation of the PowerPC™ MPX Bus with the MPC7400, MPC7410, and the MPC7450 microprocessors.

The MPX Bus is a high-performance bus specification with separate address and data buses, each with its own set of arbitration and control signals. This allows for the decoupling of the data tenure from the address tenure of a transaction and provides for a wide range of system bus implementations, including:

- non-pipelined bus operation
- pipelined bus operation
- split transaction operation

1 I/O Voltage Level

Table 1 specifies the MPX Bus input/output voltage levels for the MPC7400/MPC7410 and the MPC7450.

Table 1. MPX Bus I/O Voltage Configuration

Pin	Connection	MPC7400	MPC7410	MPC7450
BVSEL	HRESET [1, 2]	2.5V I/O	2.5V I/O	2.5V I/O
	GND	1.8V I/O	1.8V I/O	1.8V I/O
	OVDD	3.3V I/O (ipu) [3]	2.5V I/O (ipu)	2.5V I/O (ipu)
	¬HRESET	3.3V I/O	not supported	not supported

[1] Configurations are determined by sampling the respective input pin before and after hreset negation.

[2] Connecting an input to HRESET provides the "low during hreset, high after hreset" encoding.

[3] ipu = internal pull-up; default if unconnected.

While the interfaces to the off-chip L2 on the MPC7400/MPC7410 and the off-chip L3 on the MPC7450 are not part of the MPX Bus spec, their i/o voltage configurations are included in Table 2 and Table 3 for convenience.

freescale™
semiconductor

Freescale Semiconductor, Inc.

Table 2. L2 I/O Voltage Configuration (MPC7400/MPC7410 only)

Pin	Connection	MPC7400	MPC7410
L2VSEL	$\overline{\text{HRESET}}$	2.5V L2 I/O	2.5V L2 I/O
	GND	1.8V L2 I/O	1.8V L2 I/O
	OVDD	.3V L2 I/O (ipu)	2.5V L2 I/O (ipu)
	^$\overline{\text{HRESET}}$	3.3V L2 I/O	not supported

Table 3. L3 I/O Voltage Configuration (MPC7450 only)

Pin	Connection	MPC7450
L3VSEL	$\overline{\text{HRESET}}$	2.5V L3 I/O
	GND	1.8V L3 I/O
	OVDD	2.5V L3 I/O (ipu)
	^$\overline{\text{HRESET}}$	not supported

2 Signal differences

Table 4 describes signal differences between the MPC7400/MPC7410 and the MPC7450.

Table 4. Signal Differences

Description	MPC7400/MPC7410	MPC7450
MPX Bus mode with address bus drive mode	$\overline{\text{EMODE}}$ = GND	$\overline{\text{BMODE0}}$ = GND $\overline{\text{BMODE1}}$ = VDD
MPX Bus mode without address bus drive mode	$\overline{\text{EMODE}}$ = $\overline{\text{HRESET}}$	$\overline{\text{BMODE0}}$ = HRESET $\overline{\text{BMODE1}}$ = VDD
Address bus	A[0:31]	A[0:35]
Address parity	AP[0:3]	AP[0:4]
Cache inhibited	$\overline{\text{CI}}$ (input/output) [1]	$\overline{\text{CI}}$ (output only)
Write through	$\overline{\text{WT}}$ (input/output) [2]	$\overline{\text{WT}}$ (output only)
Data transaction index	DTI[0:2]	DTI[0:3]
Reservation	$\overline{\text{RSRV}}$	not supported
Address monitor (optional)	AMON[0]	not supported
Data monitor (optional)	DMON[0]	not supported

[1] weak pull-up recommended

[2] weak pull-up recommended

3 Extended Addressing

3.1 MPC7400/MPC7410

The MPC7400/MPC7410 supports a 32-bit address bus with four bits of odd parity.

PowerPC™ MPX Bus Implementation Differences

3.2 MPC7450

The MPC7450 supports both a 32-bit addressing mode and a 36-bit extended addressing mode. The MPC7450 can be configured to support extended addressing by setting the XAEN bit of HID0 (bit 14).

When extended physical addressing is disabled, the MPC7450 will drive the four most significant bits to zeroes. Note that the four most significant bits are still sampled and should be actively pulled to zero if they are not being used in a system. For compatibility between the MPC7400/MPC7410 and the MPC7450 when extended physical addressing is disabled, do the following:

- connect MPC7450 A[0:3] to b'0000'
- connect MPC7450 A[4:35] to MPC7400/MPC7410 A[0:31]
- connect MPC7450 AP[0] to b'1'
- connect MPC7450 AP[1:4] to MPC7400/MPC7410 AP[0:3]

When extended physical addressing is enabled, the MPC7450 will drive a 36-bit physical address with five bits of odd parity.

- AP[0] contains odd parity for A[0:3].
- AP[1] contains odd parity for A[4:11].
- AP[2] contains odd parity for A[12:19].
- AP[3] contains odd parity for A[20:27].
- AP[4] contains odd parity for A[28:35].

The boot address/reset vector is the same as on previous chips (0x0FFF00100) because extended addressing is only enabled when translation is enabled and the reset vector is mapped in real mode.

4 Bus Request

4.1 MPC7400/MPC7410

The MPC7400/MPC7410 may issue $\overline{BR}$ for a non-window-of-opportunity transaction and not be able to accept immediately a qualified bus grant if the limit of outstanding address tenures has been reached. This situation resolves itself if forward progress can be made on the system data bus. It may be necessary to re-arbitrate the $\overline{BG}$ every cycle to allow for a higher priority request when the current $\overline{BR}$ was not made in the window of opportunity.

5 Address Transfer Attributes

Differences between the MPC7400/MPC7410 and the MPC7450 with respect to the setting of the WIM bits are as follows:

- $\overline{WT}$. On all three processors, this bit reflects the write-through status for a transaction as determined by the MMU address translation. It is also asserted for burst writes due to **dcbf** (Flush) and **dcbst** (Clean) instructions, and for snoop pushes; it is negated for **ecowx** transactions.

  The MPC7450 also uses the $\overline{WT}$ signal during read transactions to indicate the transaction is an instruction fetch ($\overline{WT}$ = 1) or not an instruction fetch ($\overline{WT}$ = 0).

- $\overline{CI}$. On all three processors, this bit reflects the cache-inhibited status for a transaction as determined by the MMU address translation unless the L1 cache is disabled. It is always asserted for **eciwx/ecowx** bus transactions independent of the address translation.

- $\overline{GBL}$. On all three processors, this bit indicates that the transaction is global and should be snooped by other masters (output) or must be snooped by the processor (input). The $\overline{GBL}$ bit is asserted for **sync**, **tlbsync**, **tlbie**, and **eieio** instructions. The $\overline{GBL}$ bit is always deasserted for castouts, snoop pushes, and **eciwx/ecowx** instructions.

  The MPC7400/MPC7410 also asserts $\overline{GBL}$ for Kill transactions due to **dcba** and **dcbz** instructions that reach the MPX Bus.

6 Instruction Fetch Differentiation

6.1 MPC7400/MPC7410

The MPC7400/MPC7410 can be configured to differentiate instruction fetches from data fetches by setting the IFTT bit (bit 23) of HID0. The TT code for all D-side reads will be changed from Read (TT=01010) to Read Atomic (TT=11010). I-side reads will continue to be identified as Read (TT=01010).

6.2 MPC7450

The MPC7450 uses the WT signal during read transactions to indicate that the transaction is an instruction fetch (WT = 1) or not an instruction fetch (WT = 0).

7 Resume Transaction Type

The MPC7450 defines an additional transaction type called Resume (TT=0x07). The MPC7450 can not generate the Resume transaction type in MPX Bus mode but it can snoop the Resume transaction type. Providing this transaction type to a MPC7450 could lead to boundedly undefined results.

8 Address Tenure Termination

8.1 MPC7400/MPC7410

The MPC7400/MPC7410 requires a minimum of two processor cycles to process a snoop and generate a response after latching $\overline{TS}$ and associated transfer attributes.

8.2 MPC7450

The MPC7450 requires a minimum of five processor cycles to process a snoop and generate a response after latching $\overline{TS}$ and associated transfer attributes. As a result, if the system bus is running faster than one-fifth the processor frequency, the system must extend the address tenure of all transactions that will be snooped by a MPC7450 by delaying assertion of $\overline{AACK}$. For Core:Bus frequency multiples of 2:1 and 2.5:1, $\overline{AACK}$ must be delayed a minimum of two bus cycles. For Core:Bus frequency multiples of 3:1, 3.5:1, 4:1, and 4.5:1, $\overline{AACK}$ must be delayed a minimum of one bus cycle.

9 Enveloped Transactions

9.1 MPC7400/MPC7410

The MPC7400/MPC7410 supports enveloped transactions where $\overline{AACK}$ is delayed long enough so that the entire data tenure is contained within the address tenure.

9.1.1 MPC7450

The MPC7450 does not support enveloped transactions.

10 Earliest Assertion of $\overline{TA}$

10.1 MPC7400/MPC7410

In an MPC7400/MPC7410 system, the system chipset logic must ensure the first (or only) assertion of $\overline{TA}$ for a data transfer does not occur sooner than the first cycle of the snoop response window (two cycles after $\overline{TS}$). This guarantees the relationship between $\overline{TA}$ and $\overline{ARTRY}$ so that, in case of an address retry, the BIU discards the data before it can be forwarded internally to the cache and load/store unit. Typically, the external memory controller also detects the $\overline{ARTRY}$ address tenure and aborts the read or write operation in progress. If this $\overline{TA}/\overline{ARTRY}$ relationship is not met, the MPC7400/MPC7410 may enter an undefined state.

10.2 MPC7450

In an MPC7450 system, the system chipset logic must ensure the first (or only) assertion of $\overline{TA}$ for a data transfer does not occur sooner than the last cycle of the snoop response window (cycle after $\overline{AACK}$). This guarantees the relationship between $\overline{TA}$ and $\overline{ARTRY}$ so that, in case of an address retry, the BIU discards the data before it can be forwarded internally to the cache and load/store unit. Typically, the external memory controller also detects the $\overline{ARTRY}$ address tenure and aborts the read or write operation in progress. If this $\overline{TA}/\overline{ARTRY}$ relationship is not met, the MPC7450 may enter an undefined state.

Freescale Semiconductor, Inc.

Note that when $\overline{AACK}$ is not delayed, the snoop response window is a single cycle, and the second cycle after $\overline{TS}$ and the cycle after $\overline{AACK}$ are coincident.

11 Address Retry

In the MPC7450, a retry response (assertion of the $\overline{ARTRY}$ signal) indicates the address tenure should be rerun at a later time. There should be no assumptions about internal effects that the retried operation had on the MPC7450.

Some assertions of $\overline{ARTRY}$ by the MPC7450 indicate that the snooping processor requires access to the bus to eliminate the retry condition but is unable to use the window of opportunity to do so (for example, for TLBSYNC). The system must perform fair arbitration to allow all retrying processors to clear such stall conditions. Continuing to give arbitration priority to the last master after an $\overline{ARTRY}$ may cause a deadlock condition.

12 Window of Opportunity

12.1 MPC7400/MPC7410

If the MPC7400/MPC7410 has a snoop copyback to perform and can perform it, the MPC7400/MPC7410 asserts $\overline{BR}$ in the window of opportunity. However, in some cases, the processor may not be able to accept a qualified bus grant until many clocks after the window of opportunity. This is true if the processor is providing data from the off-chip L2, if the internal data transaction queue is full, or if the processor owns a cache line because the address tenure of a bus transaction has been performed but does not yet have possession of the data because the data tenure of the bus transaction has not been performed.

In some circumstances, forward progress on the data bus must be maintained to allow the address bus to make forward progress. Therefore, to perform a snoop copyback, if the system bus arbiter asserts $\overline{BG}$ to the processor during the cycle after the window of opportunity (such as the third cycle after $\overline{AACK}$), it should keep $\overline{BG}$ asserted until it recognizes the negation of $\overline{BR}$ or the assertion of $\overline{TS}$ indicating that the processor has started the snoop copyback transaction. During those waiting clocks, the processor will keep $\overline{BR}$ asserted to keep the bus arbiter informed.

12.2 MPC7450

If the MPC7450 has a snoop copyback to perform and can perform it, the MPC7450 asserts $\overline{BR}$ in the window of opportunity. However, in some cases, the processor may not be able to accept a qualified bus grant until several clocks after the window of opportunity. Therefore, to perform a snoop copyback, if the system bus arbiter asserts $\overline{BG}$ during the window of opportunity, it must keep $\overline{BG}$ asserted until it recognizes the negation of $\overline{BR}$ or the assertion of $\overline{TS}$ indicating that the processor has started the snoop copyback transaction. During those waiting clocks, the processor keeps $\overline{BR}$ asserted to keep the bus arbiter informed. Depending on the system operation, failure to do this may cause the processor to miss the window of opportunity and possibly block its internal snoop copyback queue when a new snoop transaction is captured. At this point the arbiter must use fair arbitration to ensure that the snoop queue has an opportunity to empty to the address bus with a qualified $\overline{BG}$.

13 HIT and Data Snarfing

13.1 MPC7400/MPC7410

The MPC7400/MPC7410 can be configured to support exclusive or shared intervention. A MPC7400/MPC7410 system optimizes shared and exclusive intervention by holding the $\overline{\text{HIT}}$ signal asserted for a second cycle after the snoop response window to indicate to the system that the data being supplied through intervention is not modified and therefore does not need to be forwarded to memory ("snarfed").

13.2 MPC7450

The MPC7450 does not perform exclusive or shared intervention and the system is responsible for knowing when to snarf. The system must snarf for interventions due to a Read, RCLAIM, Clean, or Flush transaction but does not need to for those caused by a RWITM transaction.

14 DTI and Pipeline Depth

14.1 MPC7400/MPC7410

The MPC7400/MPC7410 supports a 3-bit data transaction index, with a maximum value of DTI[0:2] = b'101'. The processor can queue six transactions before receiving a qualified $\overline{\text{DBG}}$ for any transaction. Note that once a qualified $\overline{\text{DBG}}$ has been received, one more transaction can be queued.

14.2 MPC7450

The MPC7450 supports a 4-bit DTI index, with a maximum value of DTI[0:3] = b'1111'. The processor can queue 16 transactions before receiving a qualified $\overline{\text{DBG}}$ for any transaction. Note that once a qualified $\overline{\text{DBG}}$ has been received, one more transaction can be queued.

For compatibility between the MPC7400/MPC7410 and the MPC7450 when using DTI, do the following:

*   connect the MPC7450 DTI[0] to b'0'
*   connect the MPC7450 DTI[1:3] to the MPC7400/MPC7410 DTI[0:2]

15 Revision History

Table 5. Revision History

Revision Number	Changes
0.0	Initial release
0.1	Nontechnical reformatting

NXP

Freescale Semiconductor, Inc.

freescale ™
semiconductor

AN2106/D

For More Information On This Product,
Go to: www.freescale.com