V3 SEMICONDUCTOR
THE EMBEDDED CHIPSET COMPANY

Application Note:

Introduction to interfacing the M68K™, ColdFire™ and PowerQUICC™ CPUs to the PCI Bus

V292PBC interface chip from V3 makes it easy!

1. Objective

This application note describes how to interface 32-bit synchronous Motorola M680x0™, ColdFire and PowerQUICC microprocessors with the V292PBC (PBC) PCI bridge and V292BMC (BMC) DRAM controller. Target applications include PCI based adapter cards and M680X0 based embedded systems. Although the M680x0 is specifically mentioned here, other synchronous 32-bit Motorola processors such as MC68030™, M68060™, ColdFire and PowerPC™ have very similar buses and the same circuit can be used (with small modifications in some cases).

Throughout this document, references will be made to the operation of the V292PBC and V292BMC components. Basic familiarity with these devices is assumed. If you don't have the relevant data sheets and user manuals for them then please contact V3. You can also download them from the V3 Semiconductor web site. Contact information (including the location of the V3 web site) is located on the back of this document.

2. Overview

The V292PBC uses a generic synchronous bursting bus protocol that can also be adapted for the 32 bit Motorola processors including the M68K, ColdFire and PowerQUICC (MPC860). Both the Am29030/40 and 32 bit Motorola CPUs are synchronous processors capable of bursting 32-bit data up to 40MHz (160MB/sec). Adapting the V292PBC as a Motorola CPU interface involves conversion of the V292PBC signals to the Motorola QUICC or PowerQUICC style bus access interface.

There are two possible interface methods:

- Adapt the Motorola CPU to the V292PBC/V292BMC bus: In this scenario, the V3 protocol of the V292PBC/V292BMC will be used as the internal local bus protocol. This is accomplished by converting the Motorola signals into V292PBC/V292BMC equivalents. Since the V292PBC/V292BMC protocol provides a long bust capability, this arrangement facilitates high speed movement of data directly to/from local memory and the PCI bus.

- Adapt the V292PBC to the Motorola bus: In this scenario, the common local bus will be Motorola based and the V292PBC signals will be converted to Motorola protocol.

This method is desirable when the local bus peripherals are designed specifically for the Motorola bus. There are several important performance drawbacks to this method that should be considered. First, the Motorola QUICC or PowerQUICC bus is not capable of performing long bursts which is essential for high throughput PCI designs. The second problem is that the burst protocol of the Motorola bus is not compatible with the PCI protocol. The PCI bus indicates burst length only at the end of the burst as opposed to the M680x0 protocol which demands that transfer size be asserted at the beginning of the access. This poses no problem when converting the Motorola CPU as a master into signals for the V292PBC as a slave. However, optimal conversion in the other direction is not possible and performance must be sacrificed.

2.1 M680x0/V292PBC/V292BMC PROTOCOL COMPARISON

This section will describe the main protocol differences and similarities between the M680x0 bus and V292PBC/V292BMC bus.

Main Protocol Similarities

- Synchronous buses that generate and sample all signals on the rising edge of the clock.
- $\overline{\text{TIP}}$ (M680x0) acts like a $\overline{\text{REQ}}$ (PBC/BMC) to indicate a cycle is in progress.
- $\overline{\text{TA}}$ (M680x0) acts like a $\overline{\text{RDY}}$ (PBC/BMC) to indicate that data is ready on that clock.
- R/$\overline{\text{W}}$ (M680x0) acts like a R/$\overline{\text{W}}$ (PBC/BMC) Read/Write indication.
- The functional operation of the address and data buses are the same.

Main Protocol Differences

- M680x0 uses A1:0 and SIZ1:0 instead of the byte enables used by the PBC/BMC.
- M680x0 uses SIZ1:0 to indicate burst length at the beginning of a cycle. PBC/BMC determine burst size with the $\overline{\text{BURST}}$ signal.
- Bursts in the M680x0 protocol are modulo 4 words while both V292PBC and PCI bursts are sequential.
- Bus arbitration on the M680x0 uses 3 signals ($\overline{\text{BR}}$, $\overline{\text{BG}}$ and $\overline{\text{BB}}$) and not the 2 signals used by the V292PBC. The $\overline{\text{BB}}$ signal (Bus Busy) is asserted in the same way that $\overline{\text{TIP}}$ (Transfer In Progress) is asserted and is not really needed for anything in this application.

Conversion between protocols requires a relatively small and inexpensive programmable logic device. However, the approach chosen can drastically affect the overall performance of the system. The main factor affecting performance is the burst length. This alone will make the V292PBC based local bus approach the best choice although both approaches will be detailed in the next sections.

2.2 MPC860(PowerQUICC) Protocol Comparison

The bulk of this application note will deals specifically with the M680x0 bus interface. However, the bulk of the design will also work with other members of the Motorola 32 bit CPU product families such as the MPC860. The main differences are listed below.

Changes to accommodate the MPC860/PowerQUICC Processors

- The M680x0 uses SIZ1:0 to indicate a burst where the MPC860 uses $\overline{\text{BURST}}$ for this same purpose. Note that $\overline{\text{BURST}}$ is also a V292PBC/V292BMC signal. The following code will use $\overline{\text{BURST}}$ in the V292PBC sense. Thus the $\overline{\text{BURST}}$ signal from the MPC860 is not usable by the V292PBC and V292BMC.

2.3 ColdFire Protocol Comparison

The ColdFire bus is also similar to the M680x0 bus interface. The main differences are listed below.

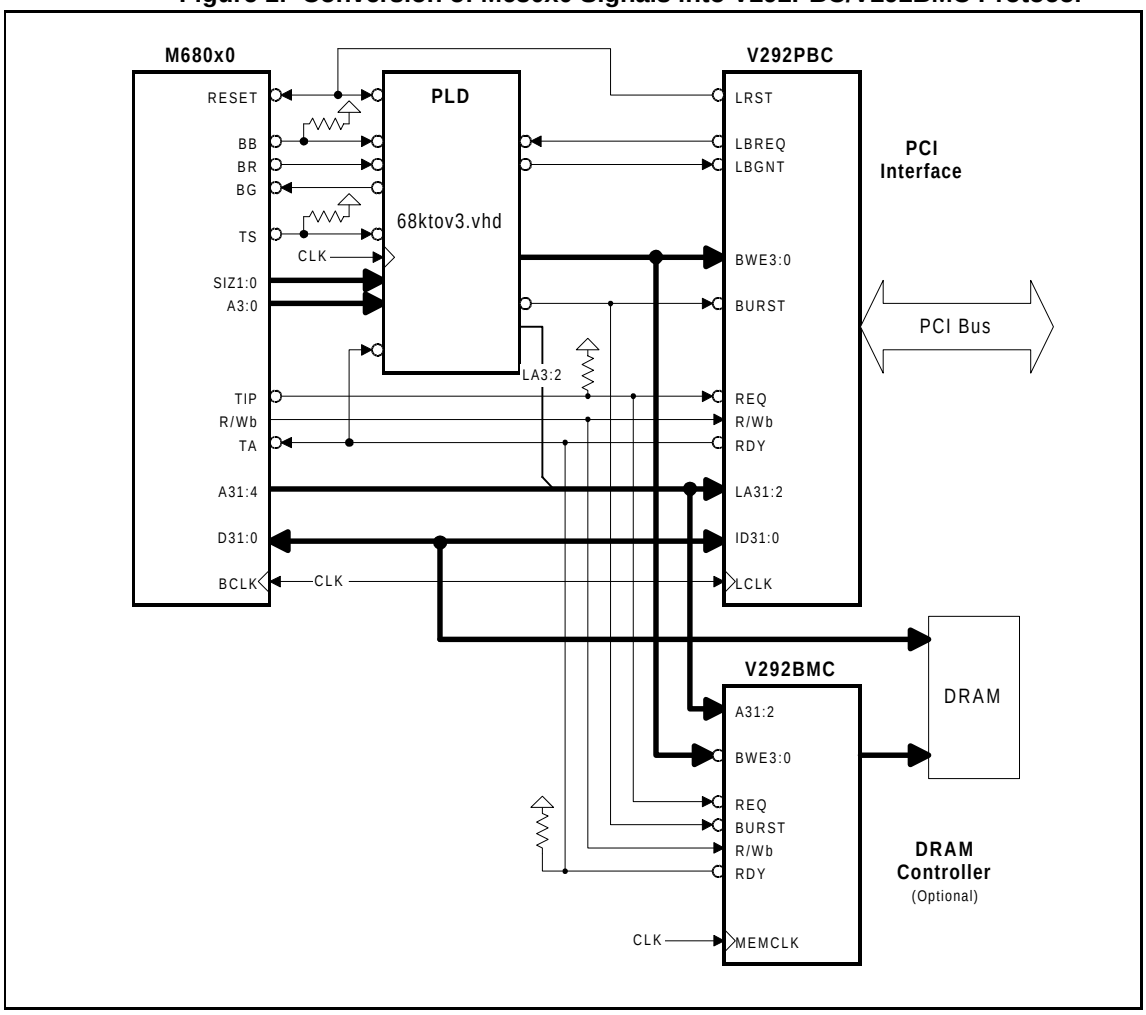
Changes to accommodate the ColdFire Processors

- The MCF5xxx ColdFire family is comprised of both multiplexed (e.g. MCF5102) and demultiplexed (e.g. MCF5206) processors. If the V292PBC is used for multiplexed versions, then the address out of the ColdFire should be latched. In this case, it may be more convenient to use the V961PBC for PCI interfacing since it provides a multiplexed bus. The interface is a little different but of similar complexity.
- ColdFire uses 2 "data acknowledge" inputs DA1:0 instead of a single $\overline{\text{TA}}$. DA1:0 to the ColdFire will both be driven low when $\overline{\text{RDY}}$ from the V292PBC is returned. This indicates a 32 bit transfer to the ColdFire.
- The Address Acknowledge input of the ColdFire CPU should be driven low so that no wait states are used on the address

3. Interconnection

3.0.1 Using a V292PBC and V292BMC

Figure 1: Conversion of M680x0 Signals into V292PBC/V292BMC Protocol



3.0.1.1 PLD Utilization (Altera EPM7032)

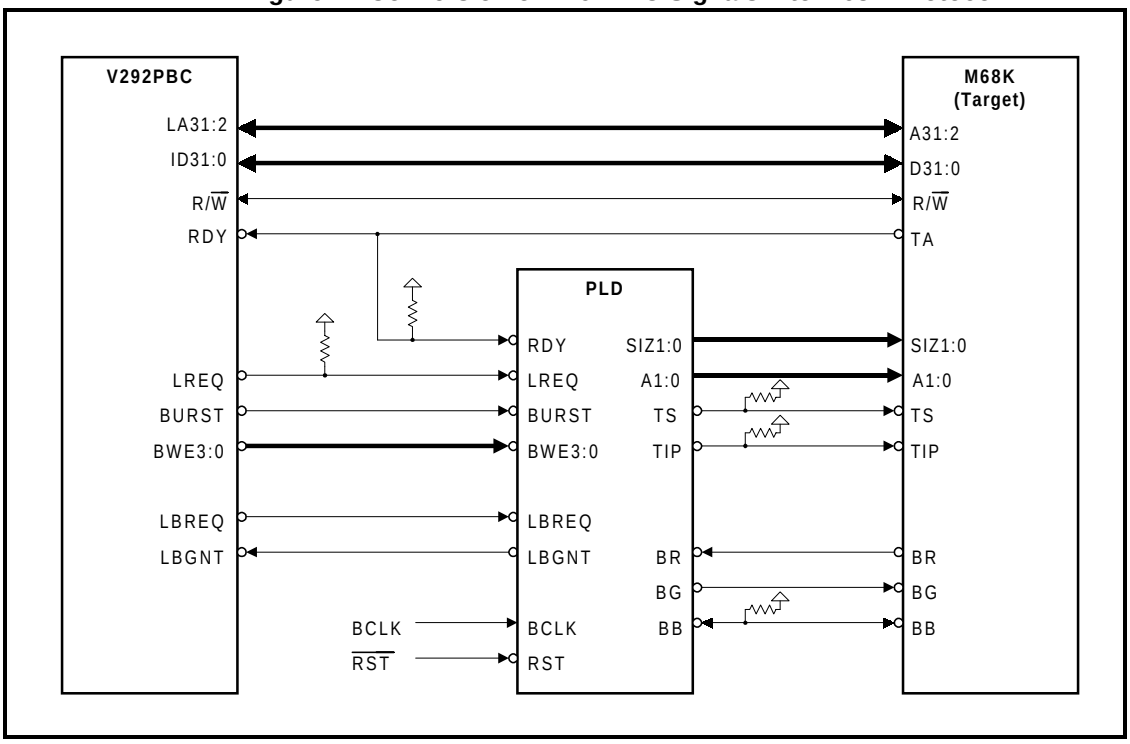
```
** DEVICE SUMMARY **
Chip/                      Input   Output   Bidir         Shareable
POF       Device           Pins    Pins     Pins     LCs  Expanders  % Utilized
m68ktov3  EPM7032LC44-6     14       3        7      12      6           37 %
User Pins:                  13       2        7

                                ~        B  B
             S                  P        W  W
             I                  I        E  E
             Z     A            N        B  B
             _  T  _  V  G  G  0  C  G  _  _
             1  A  0  C  N  N  0  L  N  3  0
             _  B  _  C  D  D  1  K  D  _  _
           ----------------------------------_
         /   6  5  4  3  2  1 44 43 42 41 40   |
  SIZ_0_ |  7                                  39 | BWEB_1_
  RESETB |  8                                  38 | BURSTB
  LBREQB |  9                                  37 | LA_2_
     GND | 10                                  36 | LA_3_
     TSB | 11                                  35 | VCC
    A_1_ | 12         EPM7032LC44-6            34 | RESERVED
    A_2_ | 13                                  33 | RESERVED
    A_3_ | 14                                  32 | ~PIN002
     VCC | 15                                  31 | BGB
     BBB | 16                                  30 | GND
     BRB | 17                                  29 | LBGNTB
         |_  18 19 20 21 22 23 24 25 26 27 28  _|
           ----------------------------------
             R  R  R  R  G  V  R  R  R  R  B
             E  E  E  E  N  C  E  E  E  E  W
             S  S  S  S  D  C  S  S  S  S  E
             E  E  E  E        E  E  E  E  B
             R  R  R  R        R  R  R  R  _
             V  V  V  V        V  V  V  V  2
             E  E  E  E        E  E  E  E  _
             D  D  D  D        D  D  D  D
```

The interface between the M68K and V292PBC is shown in Figure 2.

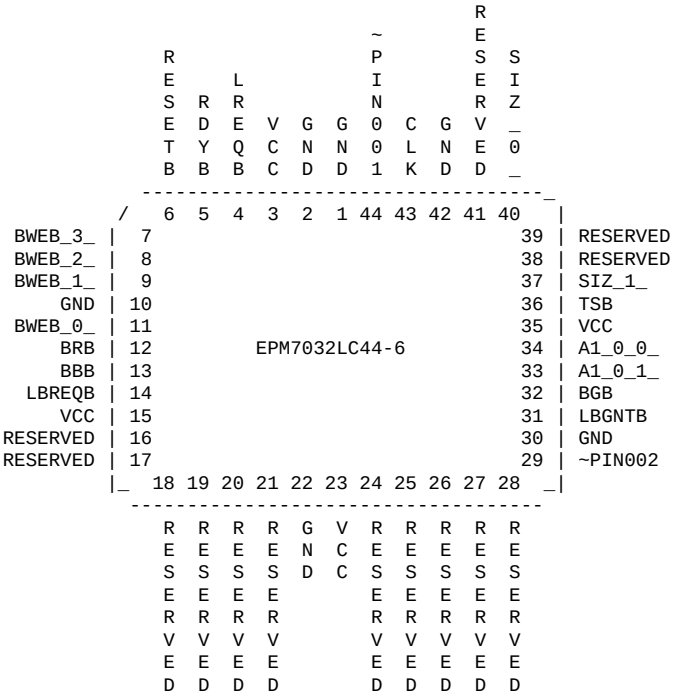
3.0.2 Using only a V292PBC

Figure 2: Conversion of V292PBC Signals into M68K Protocol



3.0.2.1 PLD Utilization (Altera EPM7032)

```
** DEVICE SUMMARY **
Chip/                          Input   Output   Bidir         Shareable
POF       Device               Pins    Pins     Pins     LCs  Expanders  % Utilized
v3tom68k  EPM7032LC44-6         12       3        5      11      0           34 %
User Pins:                      11       2        5
```



4. PLD Design

To obtain detailed design information and PLD source code, please contact V3 Semiconductor at:

EMAIL: v3help@vcubed.com

URL: http://www.v3semiconductor.com/

Applications Engineering: (416) 497-8884

Sales and Marketing: 1-800-488-8410 or (408) 988-1050

V3 Semiconductor
2348 Walsh Ave., Suite G,
Santa Clara, California,
USA 95051

5. Performance

The performance of the various alternatives have been calculated in the following tables. "Real world" performance will depend on a number of factors:

Bursting: The best performance will depend on the ability to burst long data transfers with no wait states. Since the M68K processors are limited to 4 word bursts, it is better to use the DMA controller on the V292PBC to transfer large volumes of data. It is capable of sustaining up to 1KB bursts. The DRAM controller built into some integrated M68K devices will yield very poor results due to the lack of burst support. For this reason, a controller such as the V292BMC is highly recommended since it supports long zero-wait bursting.

Data Flow: It is better to "push" (write) data rather than "pulling" (reading). Pushing takes advantage of the large 256 byte write FIFOs on the V292PBC. Writes can also be posted so that the final target device doesn't have to respond in order for the originating master to start data transfer. Reading non-consecutive locations will require a new address to be transferred to the target before any progress can be made with data movement.

Sequential Reads: If a pushing data flow cannot be accommodated then reading should be done as sequential bursts. This will take advantage of the read prefetching buffers. When possible use the dual apertures to the best advantage since they each have their own prefetch buffers.

System Bottlenecks: If non-bursting or slow target devices are used then obviously performance is affected. Also, other traffic on the PCI or local bus will allow less time for V292PBC transfers. PCI system main memory can also be a bottleneck since the V292PBC must compete with the host CPU to get access to this shared resource. Host CPU intensive applications that are cache bound will allow improved PCI data movement since the CPU doesn't need access to main memory very often.

Table 1: Performance as a PCI Master

Transaction Type	Number of PCLKs	PCLK=33MHz Rate (MB/s)
Single Posted Writes	3	44.0
Burst Write of 4 words	6	88.0
Burst Write of 8 words	10	105.6
Burst Write of 16 words	18	117.3
Burst Write of 256 words	258	131.0

Table 2: Performance as a PCI Target

Transaction Type	Number of PCLKs	PCLK=33MHz Rate (MB/s)
Single Posted Writes	3	44.0
Burst Write of 4 words	5	105.6
Burst Write of 8 words	9	117.3
Burst Write of 16 words	17	124.2
Burst Write of 256 words	257	131.5

Table 3: Performance with V292PBC as Master, M68K as Target

	Using V292PBC Bus Mode			Using 68K Bus Mode		
Transaction Type	Number of LCLKs	LCLK=33MHz Rate (MB/s)	LCLK=40MHz Rate (MB/s)	Number of LCLKs	LCLK=33MHz Rate (MB/s)	LCLK=40MHz Rate (MB/s)
Single Posted Writes	2	66.0	80.0	2	66.0	80.0
Burst Write of 4 words	5	105.6	128.0	8	66.0	80.0
Burst Write of 8 words	9	117.3	142.2	16	66.0	80.0
Burst Write of 16 words	17	124.2	150.6	32	66.0	80.0
Burst Write of 256 words	257	131.5	159.4	512	66.0	80.0

Table 4: Performance with M68K as Master, V292PBC as Target Cycles

	Using V292PBC Bus Mode			Using 68K Bus Mode		
Transaction Type	Number of LCLKs	LCLK=33MHz Rate (MB/s)	LCLK=40MHz Rate (MB/s)	Number of LCLKs	LCLK=33MHz Rate (MB/s)	LCLK=40MHz Rate (MB/s)
Single Posted Writes	3	44.0	53.3	3	44.0	53.3
Burst Write of 4 words	6	88.0	106.7	12	88.0	106.7
Burst Write of 8 words	10	105.6	128.0	24	105.6	128.0
Burst Write of 16 words	18	117.3	142.2			
Burst Write of 256 words	258	131.0	158.8			

6. Conclusion

The V292PBC from V3 provides a high performance PCI solution for various members of the Motorola processor family. When used together with the V292BMC burst DRAM controller, the limited burst capabilities of the M68K derivatives are easily overcome. Only a small (44 pin) low cost (<$2) PLD is required in addition to the highly integrated V292PBC and optional V292BMC.

