Acorn Enhanced Expansion Card Specification

(formerly *Acorn expansion card specification*)

Published by Acorn Computers Technical Publications Department.

If you have any comments on this manual, please complete the form at the back of the manual and send it to the address given there.

Acorn supplies its products through an international distribution network. Your supplier is available to help resolve any queries you may have.

Within this publication, the term 'BBC' is used as an abbreviation for 'British Broadcasting Corporation'.

Published by Acorn Computers Limited
Part number 0472,200
Issue 5, July 1994

Contents

Acorn Enhanced Expansion Card

Introduction

This document outlines the expansion card interface implemented on current Acorn Archimedes, R-series, BBC A3000 computer systems, and the Acorn Risc PC. The majority of this document is also relevant to A4000 and the later A3000 series machines that can be fitted with internal expansion cards (known as mini expansion cards) – see *Appendix A*. For more information on the RISC OS software interface with expansion cards, and how to write loaders for them, see *Expansion cards and Extension ROMs* in Part 13 of the *RISC OS 3 Programmer's Reference Manual*, and *Part 18 – Expansion Card Support*, of the *RISC OS 3 Programmer's Reference Manual*, *Vol. 5* (version 3.5 supplement).

This document replaces the *Acorn expansion card specification* (Part No 0472,200 Issue 4).

The existing Acorn expansion bus is fully supported within the enhanced version. The Acorn enhanced expansion bus has several improvements on its predecessor; namely, the inclusion of the DMA Extended Bus Interface, or DEBI for short.

DEBI

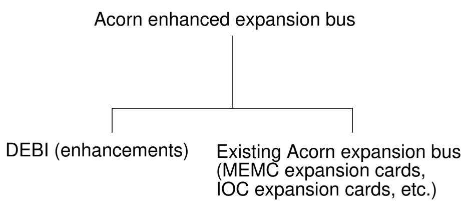
DEBI includes DMA (Direct Memory Access) interface support for expansion cards as well as an Extended Address Space Interface (EASI). In this documentation, DEBI is sometimes referred to as 'the enhancements'. A breakdown of the Acorn enhanced expansion bus is shown below.

The Acorn enhanced expansion bus has two parts, the structure of which is as follows:

1 DEBI (enhancements)

2 The existing Acorn Expansion bus (for backwards compatibility)

Figure 1: Breakdown of the Acorn enhanced expansion bus



DEBI is currently fully implemented on the Risc PC platform and may be supported in whole or part on future Acorn products.

DEBI is not supported on platforms before the Risc PC, and a concise table showing the different interfaces available on Acorn's range of machines is included in the section entitled *Acorn machine range I/O type inclusions* on page 2.

See *Appendix B* of this document for further detailed information relating to the DEBI.

Acorn expansion bus

The inclusion of the existing Acorn Expansion Bus interface provides support for all types of existing I/O hardware. These are known more specifically as IOC expansion cards, MEMC expansion cards and mini expansion cards.

Expansion cards for Archimedes computers were formerly known within Acorn as podules, and some relics of this nomenclature persist in software (and hence in some diagrams and software descriptions).

It is important to realise that future systems may have a different implementation of the I/O system, and in particular the addresses (and number) of expansion card locations may change. For this reason, and to ensure that any device may be plugged into any slot, all driver code for expansion cards **must** be relocatable. References to the direct expansion card addresses should never be used. It is up to the machine operating system, in conjunction with the expansion card ID, to determine the address at which an expansion card should be accessed.

Some models of computer, such as the A400/1 series, also support a co-processor card on expansion card slot 2. The co-processor interface is a superset of the expansion card interface. It is for Acorn use only, and is not described in this document.

It is assumed that the reader is familiar with the basic concepts of computer systems built around the ARM micro-processor unit (MPU) and its support devices, MEMC, VIDC, IOC and IOMD.

On the Risc PC platform, the IOC and MEMC type chips do not exist as separate devices. Instead, the I/O and memory functions are combined and performed by a single chip called the IOMD (Input Output Memory Device). Thus, references to IOMD in this document refer to the Risc PC platform, and references to IOC and MEMC refer to older platforms.

Detailed specifications of the original ARM chip set can be found in the Prentice-Hall publication: *Acorn RISC Machine Family Data Book*, ISBN 0-13-781618-9.

In this document logic active low signals are indicated by a bar over the signal name, e.g. $\overline{BL}$.

Physical dimensions

The expansion card printed circuit board (PCB) mechanics follow the single Eurocard format. On some platforms, double (233.4mm) width cards of standard length (160mm) may be used. All cards have a 25.4mm high (5 HP) metal back panel, for mounting externally accessible connectors.

Expansion cards for use inside the Risc PC must be of the single Eurocard dimension only – double Eurocards will not fit. All expansion cards fitted to the Risc PC should include an EMC gasket where required, fitted inside the rear panel. The EMC gasket should be as detailed in the section entitled *EMC design* on page 22 and as shown in Acorn Drawing No 0297,093.

Each expansion card can be fitted with a 64 or 96 way DIN41612 type connector for interface to the expansion card slot.

IOC and MEMC expansion cards will have a 64 way connector with rows a and c fully loaded. The new DEBI expansion cards must be fitted with a 96 way connector with rows a, b and c fully loaded. Pin out differences between DEBI and the existing Acorn expansion card are shown in more detail in the section entitled *The Acorn enhanced expansion bus backplane pin-out* on page 21.

A double-width expansion card, when viewed from above, with the metal back panel towards you, should have the DIN 41612 connector fitted in the left hand position.

Further details are given in the *Mechanical specification* on page 23.

Types of expansion card

All I/O is memory-mapped. The varying techniques of interfacing to the Acorn enhanced expansion bus give a wide variety of access speeds as well as a wide variety of data transfer widths and sizes. There are four different types of interfaces included. These are:

- The IOC expansion card type (the IOC generates all control signals to the expansion card).
- The MEMC expansion card type (the MEMC $\overline{\text{IOGT}}$ and $\overline{\text{IORQ}}$ signals are interfaced directly by the expansion card). **Note:** On the R260, A540 and A5000, MEMC's $\overline{\text{IORQ}}$ and $\overline{\text{IOGT}}$ signals are actually passed through synchronising logic before becoming the expansion bus $\overline{\text{IORQ}}$ and $\overline{\text{IOGT}}$ signals.
- The DMA controlled type (IOMD controlled interface – part of the DEBI spec).
- The Extended Address Space Interface (EASI) (IOMD controlled interface – part of the DEBI spec).

On the Risc PC, the IOC and MEMC devices have been replaced with a single device called the IOMD. All signals that once went between MEMC and IOC are now internal to IOMD on the Risc PC. The IOMD controlled interface creates all control signals so that they appear on the backplane as if MEMC and IOC still exist.

IOC expansion cards

For IOC expansion cards, IOC controls the I/O cycle by returning $\overline{\text{IOGT}}$ to MEMC. IOC offers four different access timings, selected by address value. All expansion cards must have either IOC expansion card logic, or EASI type interface capabilities, for the expansion card ID system.

MEMC expansion cards

With MEMC access cycles the expansion card works directly with the MEMC $\overline{\text{IORQ}}$ and $\overline{\text{IOGT}}$ signals. This means the expansion card designer can create optimised cycle timings for his or her application. System software should not access MEMC space unless it is certain that an expansion card is present to return $\overline{\text{IOGT}}$. If $\overline{\text{IOGT}}$ is not returned, the I/O system remains hung up waiting for it. While in the hung state the MPU clocks are stopped and only the video display process can continue. After 10 microseconds – for ARM2 – the MPU register state cannot be relied upon. The computer must be restarted with a system reset. The ARM610, however, employs static type registers, whose contents can theoretically be relied upon indefinitely, although a system reset will still be needed to restart it.

EASI expansion cards

The Extended Address Space Interface (EASI) is part of the DEBI specification. Transfer of data can be 32, 16 and 8 bits wide. The EASI bus supports two different cycle types – as described in *Appendix B: DMA Extended Bus Interface*. The EASI can be used for the transfer of the expansion card ID.

The title EASI expansion card should only be given to an expansion card which includes any form of the EASI, but **not** DMA. If an expansion card includes DMA support as well as an EASI capability it should be termed a DEBI expansion card.

Note: EASI expansion cards are not supported on platforms before the Risc PC.

DEBI expansion cards

The DMA Extended Bus Interface supports 32, 16 or 8 bit wide direct memory access. One of four cycle types is used to transfer data to and from the expansion bus. These cycle types vary in length and are named A, B, C and D, with A being the longest and D the shortest. The DEBI therefore provides a wide range of access cycle types to cover a range of peripheral speeds.

Expansion cards containing support for DMA should be termed 'DEBI' expansion cards.

Note: DEBI expansion cards are not supported on platforms before the Risc PC.

Acorn machine range I/O type inclusions

The table below provides a quick look-up detail of the different I/O interfaces used on the Acorn product range:

Table 1: I/O interfaces used on the Acorn product range

Machine	MEMC controlled	IOC controlled	Mini Expansion	DEBI	
				DMA	EASI
A305	●	●			
A310	●	●			
A420	●	●			
A440	●	●			
A440/1	●	●			

Table 1: I/O interfaces used on the Acorn product range

Machine	MEMC controlled	IOC controlled	Mini Expansion	DEBI	
				DMA	EASI
R140	●	●			
R260	●	●			
A540	●	●			
A3000	✻	✻	●		
A3010			●		
A3020			●		
A4000			●		
A5000	●	●			
Risc PC	●	●		●	●

Note: ✻ denotes an I/O interface which is resident outside of the machine, rather than inside.

System architecture

The number of expansion card slots varies with computer model:

- A305/310 have an optional 2-slot backplane.
- A440 and A400/1 series have a 4-slot backplane.
- A3000 has one internal mini expansion card and one external expansion card connector.
- A540 has a 4-slot backplane, one occupied by a SCSI card.
- A5000 has a 4-slot backplane, optional on some models.
- A3010, A3020 and A4000 have one mini expansion card connector.
- Risc PC has a 2 or 4-slot backplane. Third parties may supply 6- and 8-slot backplanes.

Expansion card access speed

IOC expansion cards

IOC expansion cards are mapped through the IOC, and may be accessed at one of four different cycle speeds, as determined by the address at which they are selected. The four cycle types are designated slow, medium, fast and synchronous. Their timings are detailed in the section entitled *IOC expansion cards* on page 10.

MEMC expansion cards

The cycle timing of a MEMC expansion card access must be controlled by the expansion card itself. A simple state machine clocked by the 8 MHz reference signal (REF8M) can be used to control these cycles. Refer to the section entitled *MEMC expansion cards* on page 18.

EASI expansion cards

EASI expansion cards are mapped through IOMD, and may be accessed at one of two different cycle speeds, as determined by the contents of the Expansion Card Timing Control Register (ECTCR) located in IOMD address space.

The two cycle types are designated A and C. Their timings are detailed in *Appendix B* in *Extended address space (EASI) timings* on page B-3.

DEBI expansion cards

DEBI expansion cards are controlled by registers inside IOMD. Each DMA channel has two buffers, which can be separately controlled, to transfer data from different areas of memory. These provide a ‘double buffering’ system for efficient data transfer. The DMA channel has a control and a status register. Each buffer is separately controlled by a pair of counter registers called the **current** and **end** registers. These two registers are used by IOMD to keep a track of the position of the DMA transfer within main memory.

The four cycle types available for each DMA channel are set by the DMA Cycle Timing Control Register (DMATCR). The registers mentioned in this section and all DMA timing information is covered in more detail in *Appendix B*.

Expansion card size

MEMC and IOC expansion cards

The expansion bus can be either 8 or 16 bits wide (or, on the Risc PC, 32 bits), allowing both byte and half-word access. Each expansion card slot has 4096 word addresses (for example, 03xx4000, 03xx4004, 03xx4008 etc.). The address range covered by each of the five different address modes is identical (synchronous, fast, medium, slow and MEMC). At each word address, because of the data width restriction, it is only possible to access either a byte or a half-word. Therefore, the size of an expansion card which is only byte-wide is 4KB, and the size of a half-word expansion card is 8KB. Some of the space will be used by the expansion card identification, the exact amount being chosen by the designer. Refer to *Expansion card identity* on page 6.

DEBI and EASI expansion cards

The EASI bus can be 8, 16 or 32 bits wide, allowing byte, half-word and word access. Each expansion card slot has 16 million byte wide addresses or alternatively 4 million word wide addresses (total 16 MB per EASI slot). It is important to note that when the ARM CPU does a byte read or write, the expansion card hardware must use the bottom 2 address lines to steer the byte to or from the correct position in the 32 bit word. The CPU can only read/write in either byte or word wide modes, and therefore half-word transfers to the expansion cards are treated as words. The EASI space used as word-wide is 16MB but for half-word wide it becomes 8MB. Some of the space can be used by the expansion card identification, the exact amount being chosen by the designer.

The DMA can use byte, half-word or word wide data transfers at four different transfer cycle speeds. The four cycle types/speeds which can be used are type A, B, C and D. The slowest cycle type is A and the quickest is D – cycle types B and C fall in-between. Data can be transferred to and from the expansion bus in chunks of between 1 byte and a maximum of 4KB. Each DMA channel has the capability of supporting two data buffers, each of which can be up to 4KB in size, so that data transfer speeds can be optimised by performing DMA transfers from one buffer whilst the programming of the other buffer takes place ready for the next block transfer. The use of two large buffers (4KB) for DMA transfers will achieve a close to continuous flow of data. A full description of how a DMA channel is programmed is beyond the scope of this document and can be found in the *RISC OS 3 Programmer's Reference Manual*, *Vol5* (version 3.5 supplement) part number 0497,551.

Data bus mapping

MEMC and IOC expansion cards

The I/O data bus (BD[0:15]) connects to the main system data bus (D[0:31]) via a set of bidirectional data latches.

The mapping of the BD[0:15] bus onto the D[0:31] bus is as follows:

- During a WRITE (i.e. MPU to peripheral), BD[0:15] is mapped to D[16:31]
- During a READ (i.e. peripheral to MPU), BD[0:15] is mapped to D[0:15].

DEBI and EASI expansion cards

During an EASI type access the I/O data bus (BD[0:31]) connects to the main system data bus (D[0:31]). The lower 16 bits D[0:15] of the bus are latched through IOMD, and the upper 16 bits D[16:31] are latched externally through a set of data latches. The mapping of the BD[0:31] bus onto the D[0:31] bus is as follows:

- During a WRITE (i.e. MPU to peripheral), BD[0:31] is mapped to D[0:31]
- During a READ (i.e. peripheral to MPU), BD[0:31] is mapped to D[0:31].

The data path for 32 bit DMA transfers is identical to the above except transfers are between main memory and the peripheral.

Byte accesses

To access bytewide expansion cards, byte instructions should be used. When a StoreByte instruction is executed, the ARM MPU will place the written byte on all four bytes of the word, and will therefore correctly place the desired value on the lowest byte of the I/O bus. A byte or word load may be used to read a bytewide expansion card into the lowest byte of an ARM register. For example:

```
  ...
  ...
LoadByte
  LDRB Rdata, [Raddress]
  ...
StoreByte
  STRB Rdata, [Raddress]
  ...
  ...
```

Half-word accesses

To access a 16-bit wide expansion card, word instructions should be used. When storing, the half-word must be placed on the upper 16 bits, D[16:31] for MEMC and IOC expansion cards. To maintain upwards compatibility with future machines, half-word stores should replicate the written data on the lower half-word, D[0:15]. When reading, the upper 16 bits are undefined.

For example:

```
  ...
  ...
LoadHalfWord
  LDR Rdata, [Raddress]
  MOV Rdata, Rdata, ASL#16
  MOV Rdata, Rdata, ASR#16
  ...
StoreHalfWord
  MOV Rdata, Rdata, ASL#16
  ORR Rdata, Rdata, Rdata, LSR#16
  STR Rdata, [Raddress]
  ...
  ...
```

Word accesses

To access a 32-bit (word-wide) expansion card, a word instruction should be used for both reads and writes to expansion cards.

For example:

```
  ...
  ...
LoadWord
  LDR Rdata, [Raddress]
  ...
  ...
StoreWord
  STR Rdata, [Raddress]
  ...
  ...
```

Expansion card interrupt handling

There are two interrupt lines on the expansion card bus, $\overline{\text{PIRQ}}$ and $\overline{\text{PFIQ}}$. Both lines are vectored through the IOC or IOMD and generate the ARM $\overline{\text{IRQ}}$ and $\overline{\text{FIQ}}$ signals respectively. $\overline{\text{PIRQ}}$ is the normal interrupt request line, and appears as bit 5 in the IOC or IOMD IRQ status B register (hex address 03200020). $\overline{\text{PFIQ}}$ is the fast interrupt request line, and appears as bit 6 in the IOC or IOMD FIQ status register (hex address 03200030). For further details on interrupt handling, refer to the IOC and/or the IOMD data sheets.

An expansion card generating an $\overline{\text{IRQ}}$ interrupt must drive the $\overline{\text{PIRQ}}$ line low. Both interrupt lines have a resistive pullup of 1k2Ω. In order that the MPU can determine which expansion card is generating the interrupt, an expansion card which is driving the $\overline{\text{PIRQ}}$ line low must also set its IRQ status bit high.

An expansion card generating a $\overline{\text{FIQ}}$ interrupt must drive the $\overline{\text{PFIQ}}$ line low. In order for the operating system to determine which expansion card is generating the interrupt, a card which is driving the $\overline{\text{PFIQ}}$ line low must also set its FIQ status bit high. (See *Expansion card identity low byte* on page 6.)

Some variants of the computer (Archimedes 400/1, 540, A5000 and R-Series) have extra logic on the backplane PCB, for expansion card interrupt management. The default/power on state of the logic leaves expansion card interrupts enabled, i.e. the logic can be ignored and the system will behave identically to the A300 and early A440 models. Two functions are added by the extra logic, a mask register and a status register. The logic is fitted to support RISC iX.

Expansion card interrupt mask register

This section only applies to those variants of the computer that have extra logic on the backplane PCB, for expansion card interrupt management (Archimedes 400/1, 540, A5000 and R-Series).

This register allows individual expansion card $\overline{\text{IRQ}}$ interrupts to be masked off, and provides a means of implementing an interrupt priority scheme for expansion cards.

Writing a ‘0’ to a bit in the expansion card interrupt mask register disables interrupts from the corresponding slot on the backplane.

Writing a ‘1’ to a bit in the expansion card interrupt mask register enables interrupts from the corresponding slot on the backplane.

The mechanism for identifying which slot is generating the interrupt is described in the section entitled *Expansion card interrupt status register* below.

The mechanism for clearing the interrupt from a particular slot will depend on the device installed in that slot.

The table below shows the correspondence between bit position and slot number.

BD7	BD6	BD5	BD4	BD3	BD2	BD1	BD0
RSVD	RSVD	RSVD	RSVD	slot 3	slot 2	slot 1	slot 0

(RSVD = Reserved)

Expansion card interrupt status register

This read-only register allows the processor to identify which expansion card is generating the interrupt without scanning the IRQ flag on each expansion card.

A logic one read in a bit position indicates that the slot is enabled and interrupting. The status register uses the same bit allocation as the mask register.

Layout and drive

The drive capability of the expansion card interface is limited, so expansion cards should offer the lowest practical load. Expansion card PCB track layout should minimise track length to the DIN edge connector, to avoid ringing or excessive capacitive loading of the interface signals.

Track lengths from the DIN41612 connector should always be less than 50mm with the REF8M tracking less than 25mm.

Signals driving the expansion card interface should include series damping resistors if possible, to reduce ringing and ground bounce problems in the card and the computer. About 68Ω is usually sufficient.

Data signals BD[0:31] should have a maximum logic low input current of -0.4mA, e.g. one LS TTL gate input load. The recommended drive capability for drivers of the data bus is 6mA (e.g. an HC series bus driver). The use of HC or AC family logic is recommended.

Address lines should have a maximum logic low input current of -1.2mA; again the use of HC or AC logic is recommended.

Control signals: the logic low input current of control signals, e.g. CLK2, CLK8 and REF8M should be less than -0.4mA (e.g. one LS TTL gate input load, input load = 20pF max). Again the use of HC or AC logic is recommended.

Open drain/open collector drivers should be able to sink at least 6mA (1k2Ω pull up, plus four LS TTL gate input loads) and still achieve a logic low voltage of less than 0.5V.

All output signals from the computer to the expansion card interface are TTL logic compatible. Expansion cards may drive the interface with TTL levels, but CMOS logic levels are recommended. The signals Ready, $\overline{\text{RST}}$, $\overline{\text{PIRQ}}$, $\overline{\text{PFIQ}}$, $\overline{\text{IOGT}}$ and $\overline{\text{BL}}$ are open drain/collector with a resistive pull-up.

In the case of the A3000 mini expansion card interface, it is recommended that the load on each signal does not exceed 3 HCT gates. It is also recommended that the drivers provided on the data bus be capable of driving at least 7 HCT and 3 TTL loads.

Heating

The power allocated for an external expansion card (for example, the A3000) should not be dissipated within the case even if no external card is fitted. This extra heating may create excessive temperatures within the case.

If an Archimedes is fitted with a 4-way backplane, a fan must be installed to improve the ventilation, if not already fitted.

Expansion card identity

An expansion card must identify itself to the host operating system. This is done with the expansion card identity (ECId). This consists of at least one byte (the low byte) of which bits 3 to 7 carry ECId information, and is usually followed by several more bytes. The ECId must appear at the base of the IOC synchronous address space, or, in the case of the Risc PC, can be located at the base of EASI space. The operating system will perform an IOC synchronous read of address 0 of each expansion card slot in turn.

The Risc PC however is slightly different. Because the expansion card ECId may be held in EASI space or IOC synchronous space, the operating system will perform a synchronous read followed by an EASI type A read from each of the possible slots. If a synchronous read is performed and the ECId byte read back by the operating system is valid, the EASI read is not performed and the operating system moves onto the next slot.

Note: Regardless of expansion card type used (IOC or EASI), the operating system always reads ECId data as bytes.

Expansion card identity space

The expansion card identity space starts at expansion card address 0 and extends into the expansion card space as required. The minimum ECId, which all expansion cards must support, is a single readable byte at address 0, called the ECId low byte. Most expansion cards will support an extended ECId which consists of eight bytes starting from address 0. The ECId (whether extended or not) must appear at the bottom of the expansion card space, from address zero upwards after reset. It does not however have to remain readable at all times, so it can be in a paged address space so long as the expansion card is set to page zero on reset. Refer to *Expansion card interrupts* on page 6 and *FIQ and IRQ status* on page 7. This has the effect that the ECId, including the expansion card present bit, is only valid from reset until the expansion card driver is installed.

If there isn't a valid ECId low byte in the IOC synchronous space then it will also be checked for at address 0 in the EASI space. The same is also true of an extended ECId; it may also exist in EASI space starting at address 0. Note that this does mean that location zero of IOC synchronous space will be read at startup time, so it is unwise to put read-sensitive hardware here even if the ECId is in EASI space.

Code space

In addition to the expansion card identity, which all expansion cards must support, an expansion card can have code or data in ROM. RISC OS usually uses this for expansion card driver code, which is downloaded into system memory, by the operating system, before it is used. Often this code will be in a paged address space. The manner in which this code is accessed is variable and so it is accessed via a loader. The format of the loader is defined for each operating system, and gives access to a paged address space. The loader must live in the expansion card space above the ECId after reset, the position and size of it being defined by a chunk directory entry. Note that the ECId and the loader may themselves be in the paged address space as long as they appear at address zero after reset. Refer to the section entitled *Chunk directory structure* on page 8.

The design of the Network Card is such that the hardware for addressing a paged ROM is fully defined. This also means a loader has been built in to RISC OS. There is no need to write or supply a loader.

When the ECId is in EASI space there is no need for a loader since the entire EASI space is directly addressable.

Expansion card identity low byte

The low byte of the ECId is as follows:

0,IRQ:	= 0 : = 1 :	not requesting IRQ requesting IRQ	see text
1,P:	= 0 :	expansion card is present	
2,FIQ:	= 0 : = 1 :	not requesting FIQ requesting FIQ	see text
3,ID0: 4,ID1: 5,ID2: 6,ID3:	= 0 : <> 0 :	extended ECId id field	
7,A:	= 0 : = 1 :	Acorn conformant expansion card non-conformant expansion card	

Expansion card presence

The host operating system has to know if there are any expansion cards present. Normally BD[1] is pulled high by a pullup. Reading the low byte of the ECId will therefore read a 1 on this bit if an expansion card is absent. All expansion cards must have bit 1 low in the low byte of the ECId.

Expansion card interrupts

An expansion card which is capable of generating a $\overline{\text{PIRQ}}$ or a $\overline{\text{PFIQ}}$ **must** carry a status bit for each of these interrupt sources. These two status bits must be in the low byte of the ECId, unless the expansion card contains an extended ECId, in which case the status bits may be relocated in the expansion card address space. An expansion card which is holding $\overline{\text{PIRQ}}$ low must set bit 0 high in the low byte of the ECId. An expansion card which is holding $\overline{\text{PFIQ}}$ low must set bit 2 high in the low byte of the ECId. In this way the operating system can quickly find which expansion card is generating the interrupt.

If the expansion card contains paged ROM, these status bits may be located elsewhere in the expansion card address space, in which case the two bits in the ECId low byte should be zeros. The location of these status bits must appear in the ROM space above the extended ECId.

If an expansion card is not capable of generating either a $\overline{\text{PIRQ}}$ or a $\overline{\text{PFIQ}}$, then bits 0 and 2 in the low byte of the ECId must be zero. If the interrupt status bits have been relocated, then the respective position mask should be set to zero. Refer to *Interrupt status pointers* on page 8.

ID field

There are four bits in the low byte of the ECId (BD[3:6]) which may be used for expansion card identification. These should only be used for the very simplest of expansion cards. Most expansion cards should implement the extended ECId which eliminates the possibility of expansion card IDs clashing. When an extended ECId is used, all four bits in the ID field of the low byte ECId must be zero.

Conformance bit

The most significant bit in the low byte of the ECId must be zero for expansion cards that conform to this Acorn specification. If this is not the case, then the ECId and chunk directory will not be recognised.

Identification extension

If the ID field of the low byte of the ECId is zero, then the ECId is extended. This means that the next seven bytes of the ECId will be read by the operating system. The extended ECId starts with the ECId low byte, either at address 0 in the IOC synchronous space or at address 0 in EASI space, and consists of eight bytes as defined below. If bit 0 of byte 1 is not set then the extended ECId is just eight bytes long. If bit 0 of byte 1 (CD) is set, then a chunk directory follows the interrupt status pointers.

7	6	5	4	3	2	1	0	
C[7]	C[6]	C[5]	C[4]	C[3]	C[2]	C[1]	C[0]	0x1C
M[15]	M[14]	M[13]	M[12]	M[11]	M[10]	M[9]	M[8]	0x18
M[7]	M[6]	M[5]	M[4]	M[3]	M[2]	M[1]	M[0]	0x14
P[15]	P[14]	P[13]	P[12]	P[11]	P[10]	P[9]	P[8]	0x10
P[7]	P[6]	P[5]	P[4]	P[3]	P[2]	P[1]	P[0]	0x0C
R	R	R	R	R	R	R	R	0x08
R	R	R	R	W[1]	W[0]	IS	CD	0x04
A	0	0	0	0	F	0	I	0x00

A	= 0 : Acorn conformant card = 1 : non-conformant card	
F	= 0 : not requesting FIQ = 1 : requesting FIQ	– see text
I	= 0 : not requesting IRQ = 1 : requesting IRQ	– see text
R	= 0 : mandatory at present = 1 : reserved for future use	
CD	= 0 : no chunk directory follows = 1 : chunk directory follows interrupt status pointers	
IS	= 0 : interrupt status appears in low byte ECId = 1 : interrupt status has been relocated	
W[1:0]	= 0 : 8 bit code follows after byte 15 of Id = 1 : 16 bit code follows after byte 15 of Id = 2 : 32 bit code follows after byte 15 of Id = 3 : reserved	
C[7:0] M[15:0] P[15:0]	= Reserved = Manufacturer = Product Type	

FIQ and IRQ status

If bit 1 in byte 1 of the extended ECId (IS) is not set, then the interrupt status bits have not been relocated within the expansion card address space. In this case the FIQ and IRQ status bits must appear as bits 2 and 0 respectively in the low byte of the ECId. Expansion cards which cannot generate interrupts must drive these bits to zero. If bit 0 in byte 1 of the extended ECId (CD) is not set either, then the interrupt status pointers do not need to be defined, as the operating system will not read them. If CD is set, then the interrupt status pointers should be defined to point to the respective bits in the ECId low byte.

If bit 1 of byte 1 of the extended ECId (IS) is set, then the interrupt status bits have been relocated within the expansion card space. In this case the interrupt status pointers must be defined as described in *Interrupt status pointers* below. Note that if both IRQ and FIQ sources are provided by an expansion card, then a separate status bit must exist for each type of interrupt source, though the two status bits may appear at the same address if convenient. Refer to *Interrupt status pointers* below.

Country code

Previously C[7:0] was the country code. This is no longer used; the UK code 00 should now be used instead.

Manufacturer's code

Every expansion card should have a manufacturer's code. The following are some examples of these:

Manufacturer	Code Value
Acorn UK	0
Olivetti	2
Watford	3
Computer Concepts	4
Wild Vision	9

Consult Acorn for the allocation of codes.

Product type code

Every expansion card type must have a unique number allocated to it. These are a few examples:

Product Type	Code Value
SCSI	2
Ethernet	3
RAM/ROM	5
BBC IO	6
MIDI	19

Consult Acorn for the allocation of codes.

Interrupt status pointers

If bit 1 of byte 1 of the extended ECId (IS) is set, then the address of the FIQ and IRQ status bits must be provided in the eight bytes which follow the extended ECId, even if the two status bits are at address 0. There are two sets of four byte numbers as detailed below, each consisting of a three byte address field and a one byte position mask field. The position mask defines which bit within the status byte refers to the status bit. It should consist of a single one and seven zeros. The other bits within the status byte may be 'don't cares'. The status byte should signal that the expansion card is interrupting by setting the appropriate bit to 1. If the expansion card does not provide one or other of these interrupt sources, then the respective position mask should consist of eight zeros.

When the ECId is in EASI space, the addresses are still relative to the normal expansion card space. The interrupt status byte(s) may not be in EASI space.

The 24-bit address field allows for an absolute byte address with an offset from hex 03240000 to be defined. Hence the cycle speed to access the interrupt status byte(s) can be included in the address (encoded by bits 19 and 20). Bits 14 and 15 should be zero.

	Hex address
	xxxxxx40
IRQ Status Bit Address (24 bits)	
	xxxxxx34
IRQ Status Bit Position mask	
	xxxxxx30
FIQ Status Bit Address (24 bits)	
	xxxxxx24
FIQ Status Bit Position mask	
	xxxxxx20

Chunk directory structure

If bit 0 of byte 1 of the extended ECId (CD) is set, then bit 1 of byte 1 of the extended ECId (IS) must be set and hence the addresses must be present.

Note that these eight bytes are always assumed to be bytewide. Only the lowest byte in each word should be used. After byte 15 (hex address 40 upwards), wider words may be used, according to the setting of W[1] and W[0] in the extended ID. See the *Identification extension* on page 7.

If bit 0 of byte 1 of the extended ECId (CD) is set, then following the interrupt status pointers is a directory of chunks of data and/or code stored in the ROM. The lengths and types of these chunks and the manner in which they are loaded is variable, so after the eight bytes of interrupt status pointers there follow a number of entries in the chunk directory.

Note that from here on the definition is in terms of bytes. If the expansion card supports a 16 bit or 32 bit wide interface then the driver code must take this into account.

The chunk directory entries are eight bytes long and all follow the same format. There may be any number of these entries. This list of entries is terminated by a block of four bytes of zeros.

	n + 8
Start Address 4 Bytes (32 bits)	
	n + 4
Size in bytes 3 Bytes (24 bits)	
	n + 1
Operating System Identity Byte	
	n

One of these blocks may need to be the code loader. It contains the code to load bytes from the (optionally paged) ROM into main memory, and as such is capable of updating the page register as required. There may be more than one loader present, to cater for different operating systems. All the loader code must be accessible after reset. After the loader is transferred to main memory, all further chunks are transferred via the loader. The chunks are again referenced by the chunk directory as above, starting at virtual address zero. Note that after the loader has been loaded, the main expansion card ID area may be mapped out. An example of a typical use of the chunk directory is shown in *Figure 4*. The shaded areas refer to chunks which are transferred via the loader.

Since RISC OS uses its built-in loader to access any ROM on the Risc PC Network Card also, there only needs be one chunk directory.

Since there is no need for a loader when the ECId and chunk directory are in EASI space there only needs be one chunk directory.

Operating system identity byte

The operating system identity byte is the first byte of the chunk directory entry, and determines the type of data which appears in the chunk to which the chunk directory entry refers.

OS[3]	= 0	reserved		
OS[3]	= 1	mandatory at present		
OS[2:0]	= 0	Acorn Operating System #0 (RISC OS)		
		D[3:0]	= 0	loader
			= 1	BBC ROM
			= 2-15	OS dependent
	= 1	Acorn Operating System #1		
		D[3:0]	= 0	loader
			= 1-15	reserved
	= 2	Acorn Operating System #2		
		D[3:0]	= 0	loader
			= 1-2	reserved
			= 3	Helios
			= 4-15	reserved
	= 3-5	reserved		
		D[3:0]	= 0-15	reserved
	= 6	manufacturer defined		
		D[3:0]	= 0-15	manufacturer specific
	= 7	device data		
		D[3:0]	= 0 link	(for 0, the object pointed to is another directory)
			= 1	serial number
			= 2	date of manufacture
			= 3	modification status
			= 4	place of manufacture
			= 5	description
			= 6	part number (for 1-6, the data in the pointed-to location contains the ASCII string of the information.)
			=7	Ethernet ID
			=8	Hardware revision
			=9	ROM CRC
			=10-15	reserved

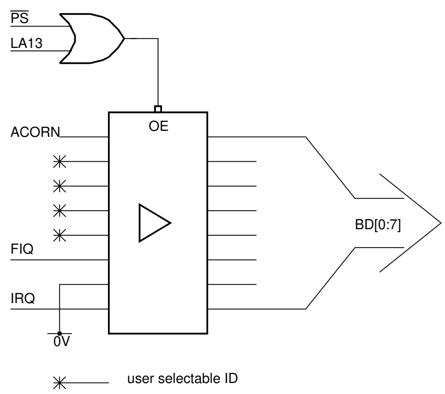
Examples of use

The previous sections explained the system of expansion card identification. You do not need to use all of these features on all expansion cards, and the implementation depends on the needs and complexity of the expansion card in question. All expansion cards must implement at least the simplest form of expansion card identification. Synchronous cycles are used by the operating system to read and write any locations within the ECId space (to simplify the design of synchronous expansion cards).

Non-extended expansion card identity

This is the simplest possible expansion card identity mechanism. It may be used for temporary expansion cards or where expansion cards are used in a localised, closed environment. It should not be used for expansion cards for general sale. Non-extended ECIds do not need PR/$\overline{W}$ factored into their enable, as the operating system will only read the ECId space. An example is shown in *Figure 2* below.

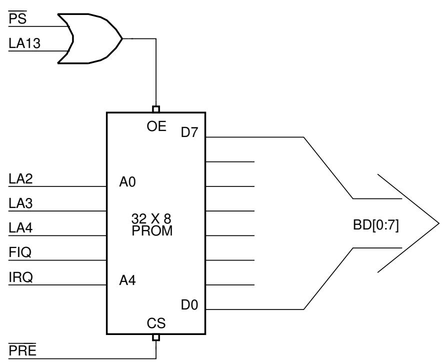
Figure 2: Non-extended ECId



Extended expansion card identity

The next simplest case which most expansion cards should implement as a minimum is the case of an extended ECId but no code in ROM. This can be achieved by a 32 x 8 bit PROM. An example is in *Figure 3* below.

Figure 3: Extended ECId



Extended expansion card identity with paged ROM

When the expansion card includes driver code in ROM, there are several possibilities for implementing the ECId. One example showing an EPROM with a paging register is shown in Figure 5: *Extended ECId with paged ROM* on page 11. Simplifications can be made where there is only one page, or where a larger EPROM allows the inclusion of the low byte of the ECId. (FIQ and/or IRQ can be factored into the address space as in the previous example).

IOC expansion cards

IOC expansion cards are controlled by the IOC. These expansion cards may be accessed by one of four types of cycle, designated slow, medium, fast, and synchronous. The cycles are mapped at different addresses.

Once the cycle has started, MEMC may de-assert IORQ (and hence IOC will de-assert RBE) in order to carry out memory refresh or DMA operations. This is indicated by the shaded area in Figure 6: *IOC driving an expansion card read cycle* on page 11. If, when the IOC has finished the cycle, the MEMC has not reasserted IORQ then the I/O cycle will be stretched until the MEMC is ready to complete the cycle. This does not, however, alter the cycle the expansion card sees, because the cycle is in effect finished before the stretching takes place. In the case of a write the WBE and PS have already been de-asserted. In the case of a read the data from the expansion card has already been latched into the data buffers by BL.

Figure 6 shows how the IOC generates a fast expansion card read: the IOC generates the expansion card read, write and select strobes, and also controls the IOGT and BL signals. The expansion card read, write and select signals are timed with respect to the signal CLK8 and **no** relationship between REF8M and CLK8 can be assumed.

Figure 7: *Stretched expansion card read cycle* on page 12 shows the same cycle, but this time the MEMC has de-asserted IORQ at the time when the IOC is about to finish the cycle. Accordingly, the cycle is stretched by one REF8M (the shaded region), but note that the access to the expansion card has not been stretched.

Figure 4: Chunk directory structure

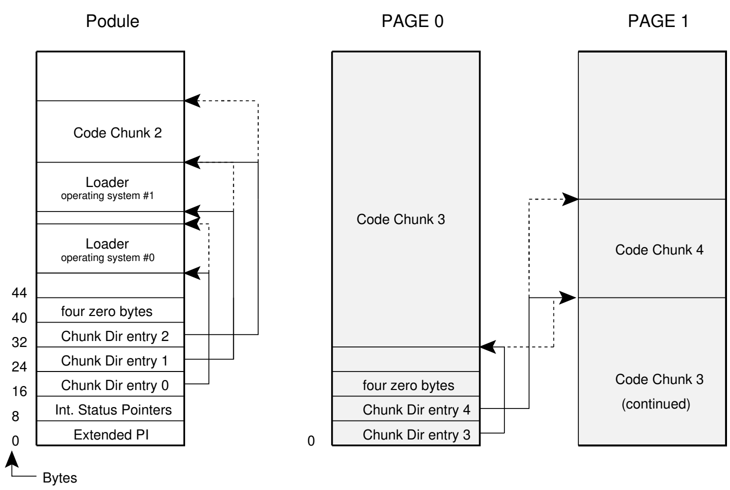


Figure 5: Extended ECId with paged ROM

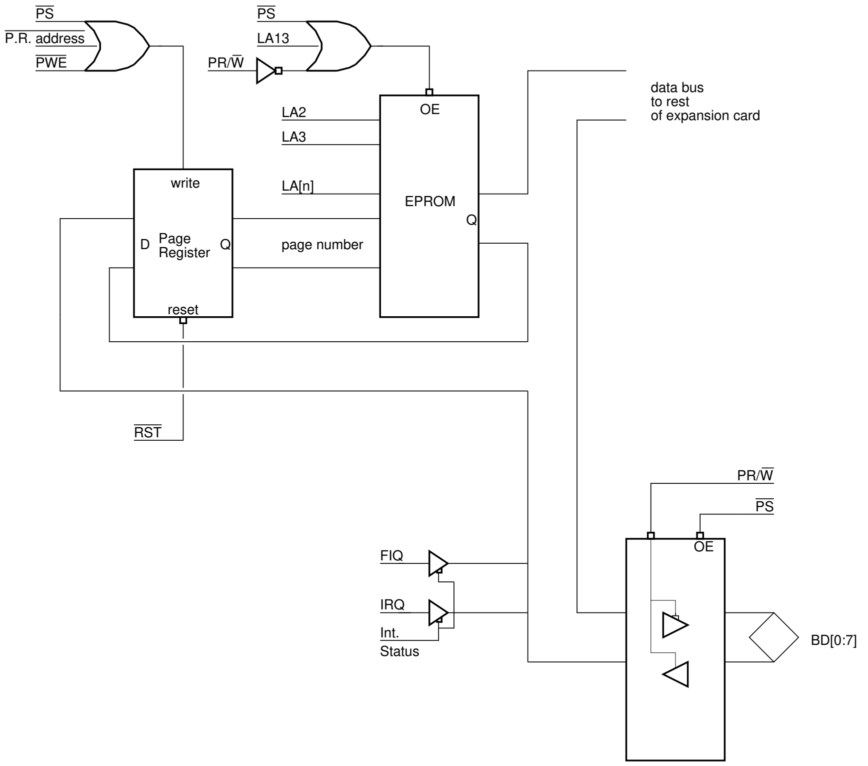


Figure 6: IOC driving an expansion card read cycle

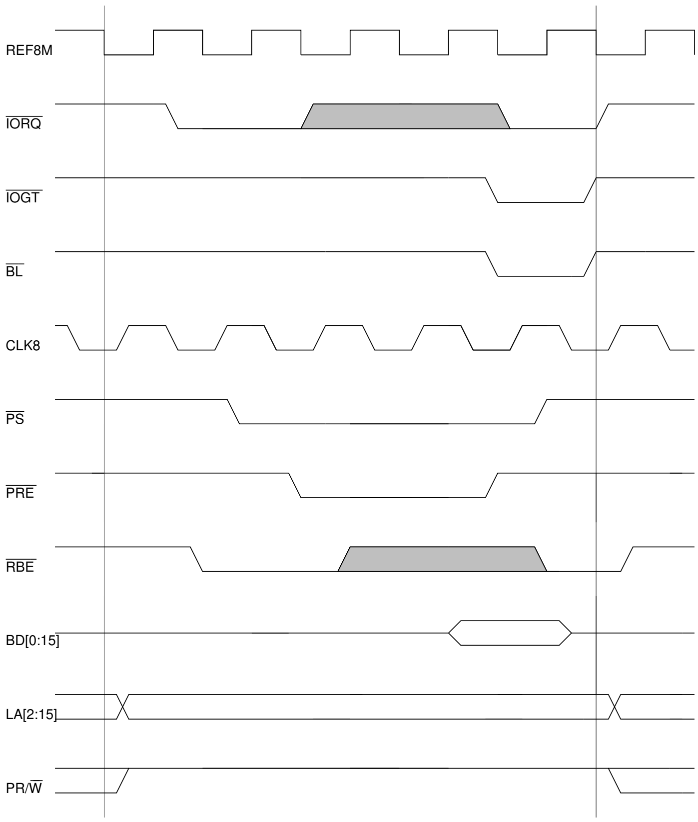
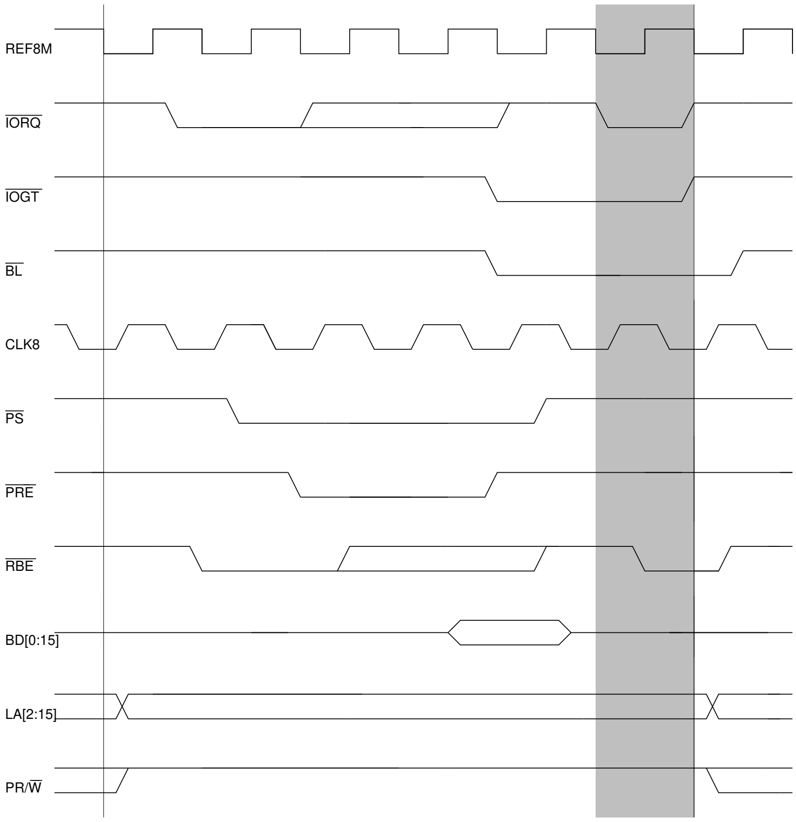


Figure 7: Stretched expansion card read cycle



Expansion card accesses

The following diagrams detail the four possible types of IOC expansion card access (slow, medium, fast, and synchronous). In each diagram REF8M is shown, but this is only for reference. The phase relationship of REF8M and CLK8 is NOT guaranteed.

Figure 8: Slow cycle read

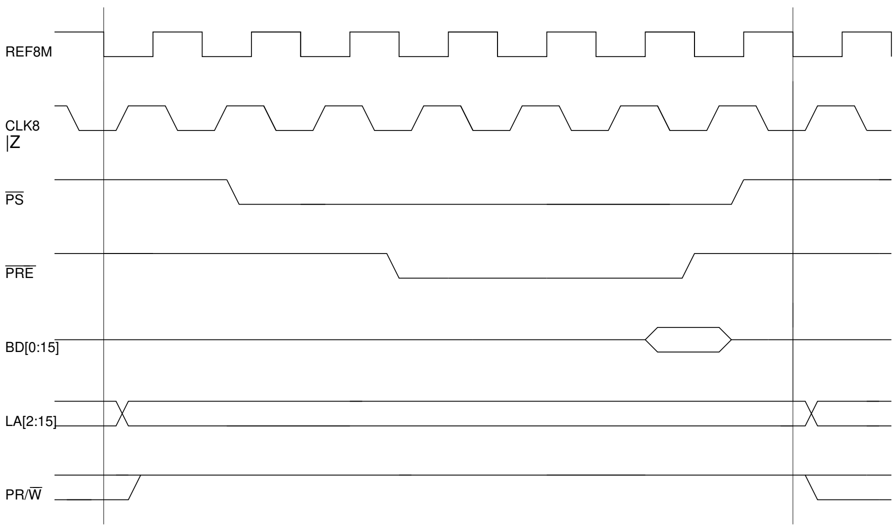


Figure 9: Slow cycle write

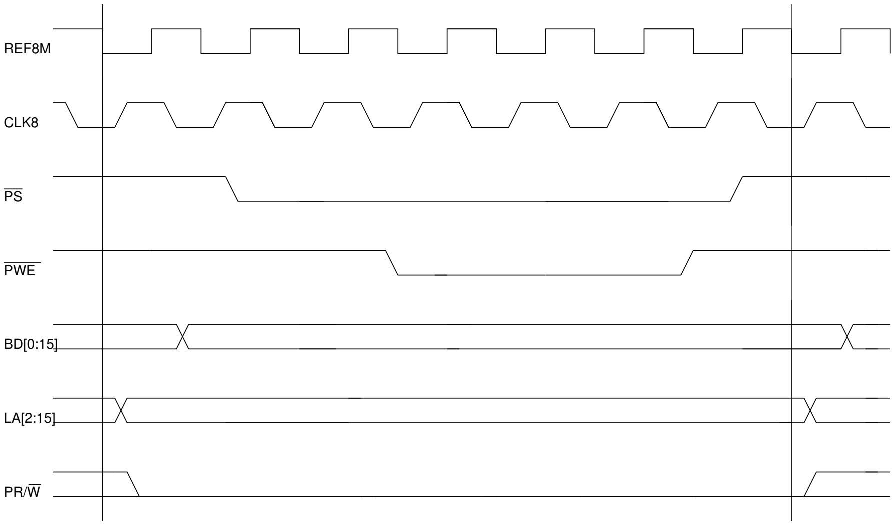


Figure 10: Medium cycle read

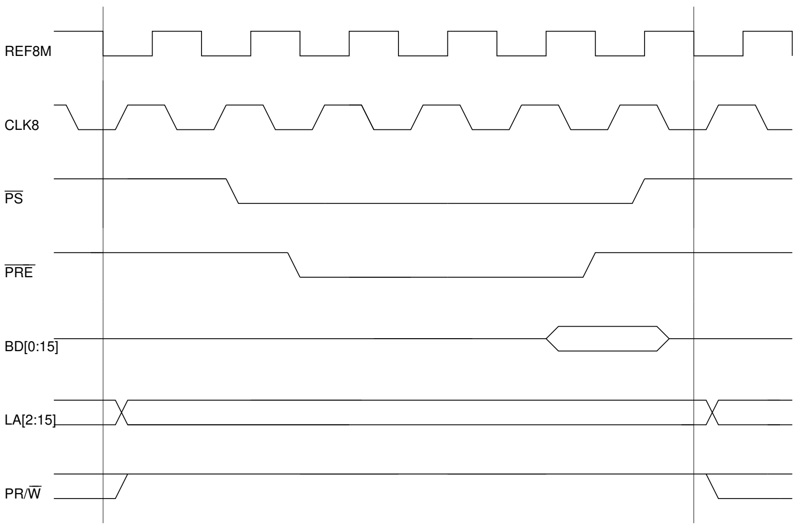


Figure 11: Medium cycle write

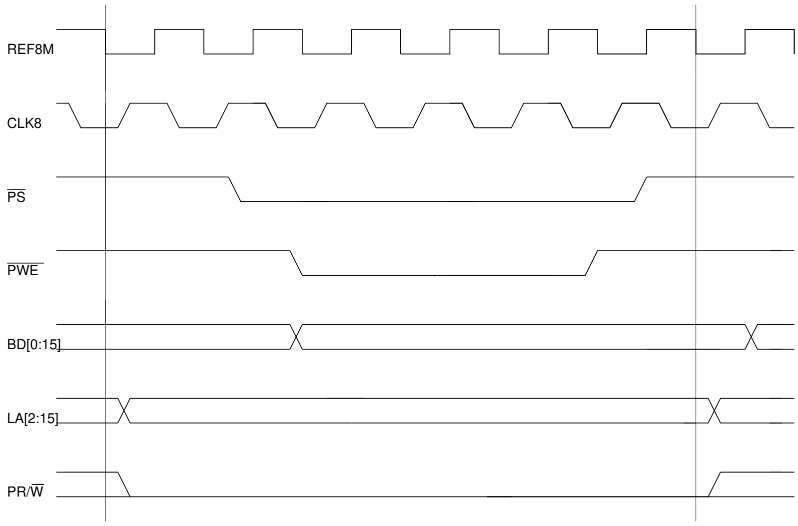


Figure 12: Fast cycle read

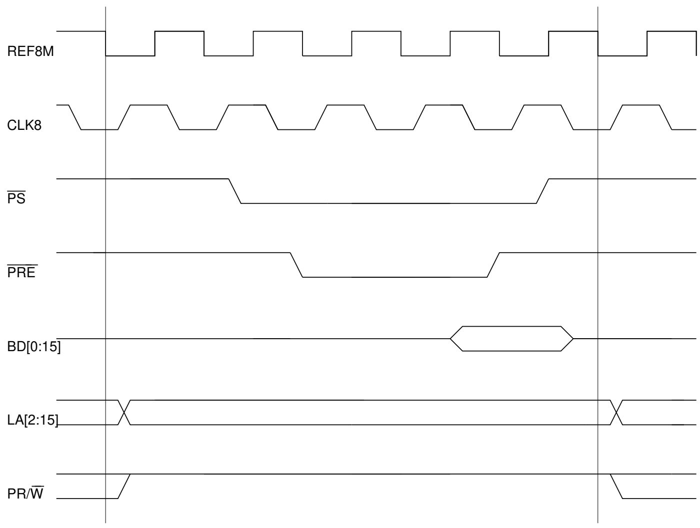


Figure 13: Fast cycle write

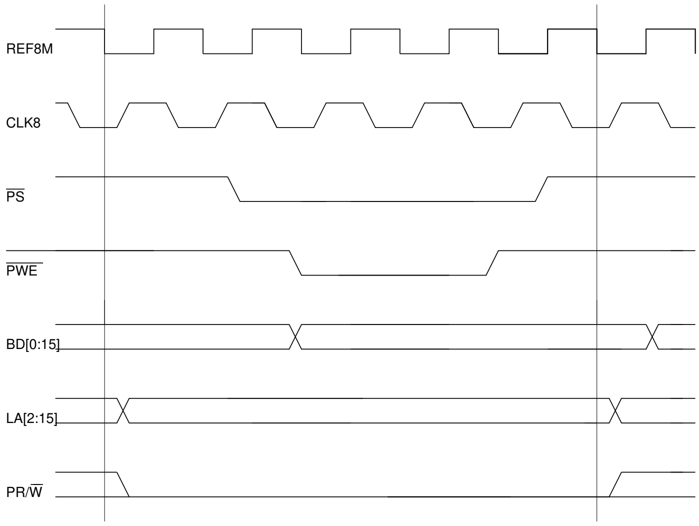
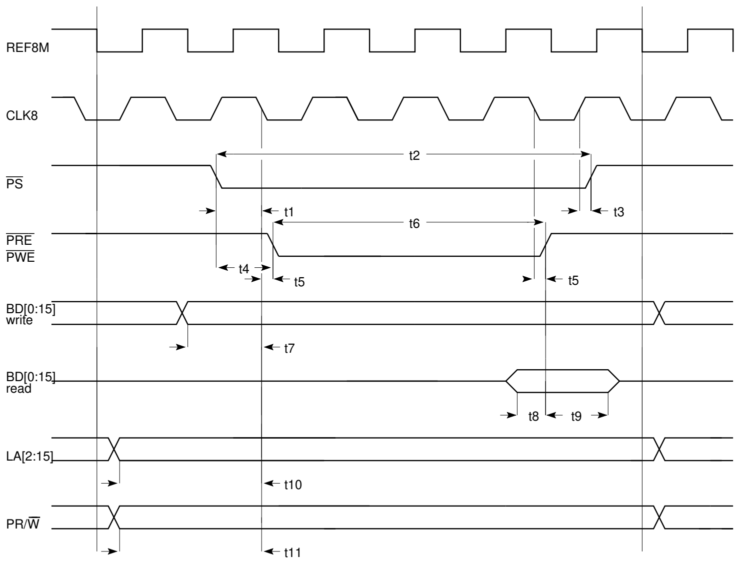


Figure 14: General timing for slow, medium, and fast cycle types



Sym	Parameter	Min	(ns) Nom	Max
t1	$\overline{PS}$ setup to CLK8	40		120
t2	$\overline{PS}$ width TYPE slow		625	
t2	$\overline{PS}$ width TYPE med		500	
t2	$\overline{PS}$ width TYPE fast		375	
t3	$\overline{PS}$ hold from CLK8	0		50
t4	$\overline{PS}$ to $\overline{PRE}$ or $\overline{PWE}$ TYPE slow		187	
t4	$\overline{PS}$ to $\overline{PRE}$ or PWE TYPE med/fast		62	
t5	$\overline{PRE}$ or $\overline{PWE}$ delay from CLK8	0		15
t6	$\overline{PRE}$ or $\overline{PWE}$ width TYPE slow/med		375	
t6	PRE or $\overline{PWE}$ width TYPE fast		250	
t7	write data setup to $\overline{CLK8}$	100		
t8	read data setup to $\overline{PRE}$	20		
t9	read data hold from $\overline{PRE}$	15		
t10	address setup to CLK8	150		
t11	PR/$\overline{W}$ setup to CLK8	140		

Figure 15: Synchronous cycle read

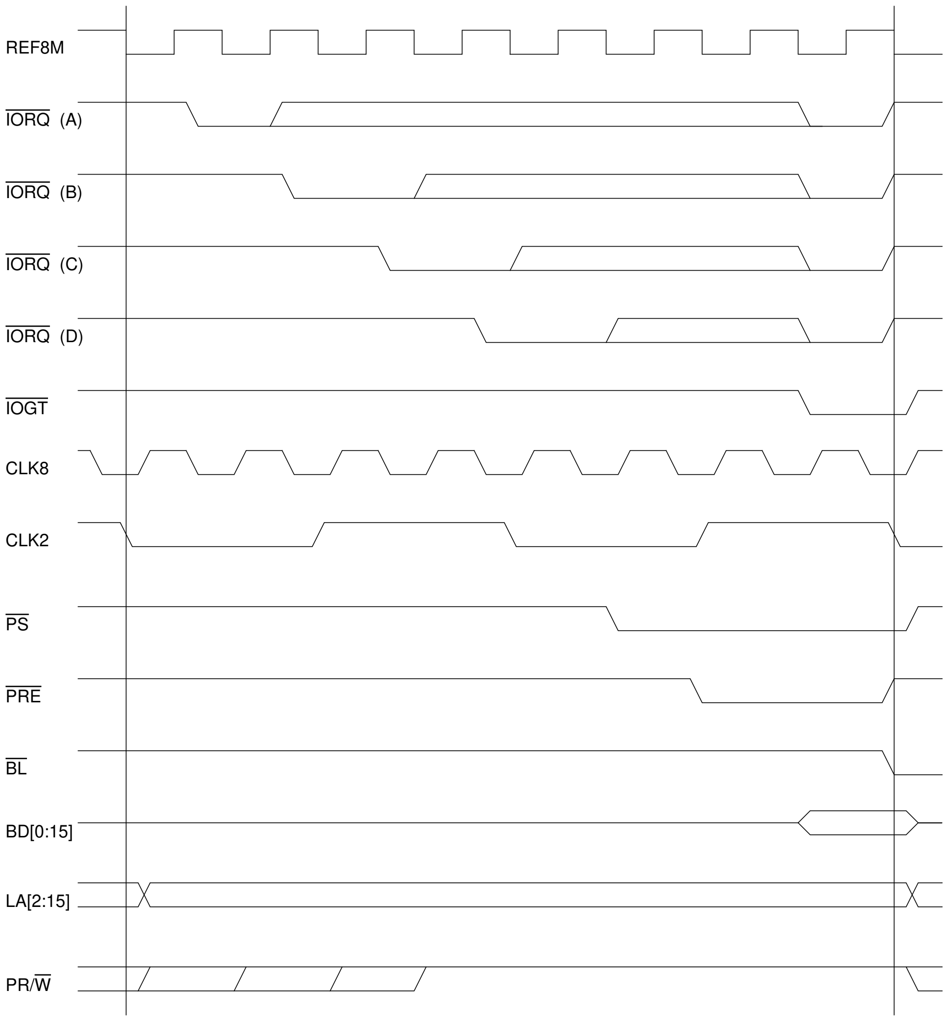


Figure 16: Synchronous cycle write

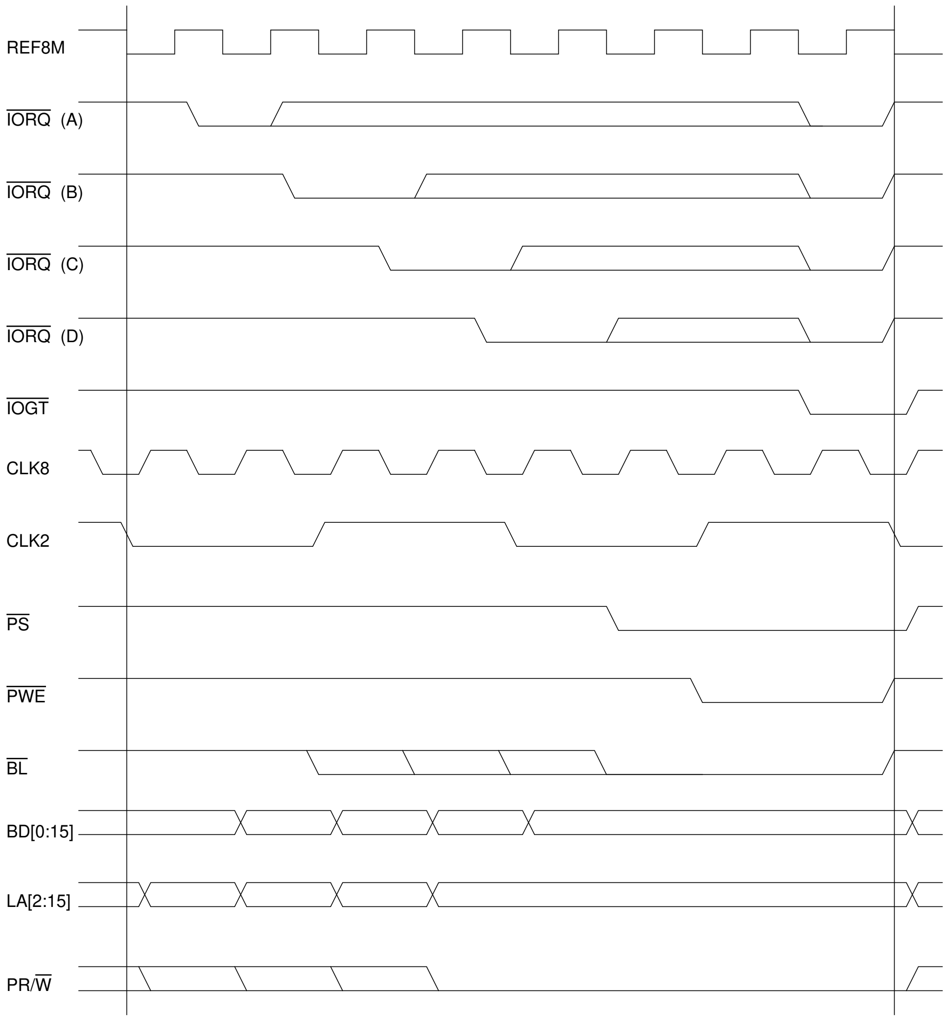
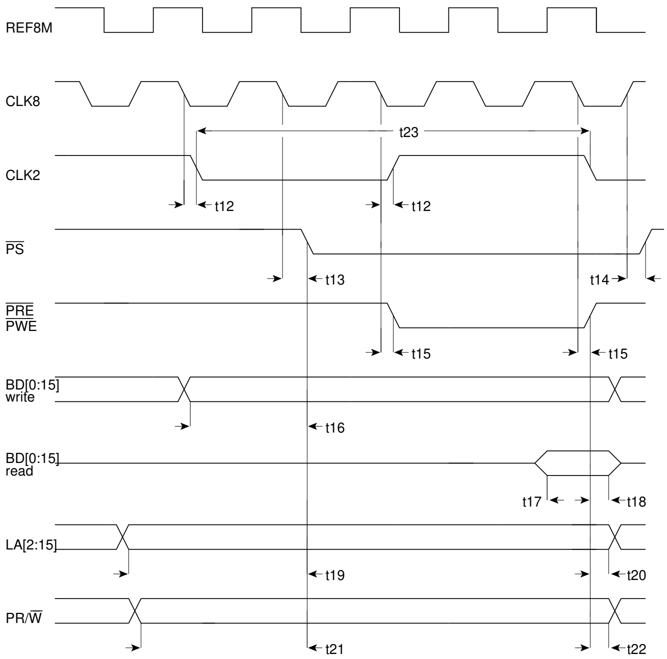


Figure 17: Timings for synchronous cycle types



Sym	Parameter	Min	Max
t12	CLK2 delay from CLK8	0	15
t13	$\overline{PS}$ delay from CLK8	5	50
t14	$\overline{PS}$ hold from CLK8	10	
t15	$\overline{PRE}$ or $\overline{PWE}$ delay from CLK8	0	15
t16	write data setup to $\overline{PS}$	100	
t16a	write data hold from $\overline{PWE}$ or CLK2	20	
t17	read data setup to $\overline{PRE}$ or CLK2	50	
t18	read data hold from $\overline{PRE}$ or CLK2	15	
t19	address setup to $\overline{PS}$	150	
t20	add. hold from $\overline{PRE}$ or $\overline{PWE}$ or CLK210		
t21	PR/$\overline{W}$ setup to $\overline{PS}$ 140		
t22	PR/$\overline{W}$ hold from $\overline{PRE}$ or $\overline{PWE}$ or CLK2	10	
t23	cycle time square wave	500	

MEMC expansion cards

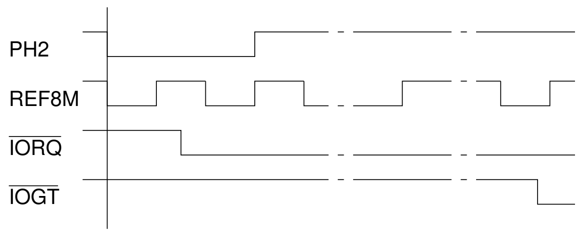
MEMC expansion cards are not controlled by IOC, although they share the same interface, so they have to time their own cycles, with their own I/O control logic. The interface has two control lines, $\overline{IORQ}$ (driven by MEMC) and $\overline{IOGT}$ (driven by the expansion card or IOC). $\overline{IOGT}$ is an open drain signal allowing multiple devices to drive this signal. MEMC expansion cards are decoded by LA[21] low, and IOC is decoded by LA[21] high. But even when IOC is not selected, it continues to control the external buffer enables, $\overline{RBE}$ and $\overline{WBE}$. The latching of the buffer must however be controlled by the expansion card which is controlling the cycle. This is done by pulling $\overline{BL}$ low. $\overline{BL}$ is an open drain signal.

I/O controller interface

I/O controllers use a handshaking system to synchronise I/O peripherals with the system data bus. The interface is timed with respect to the REF8M clock, and cycles may be produced in multiples of 8 MHz clock ticks. When the processor accesses the I/O controller address space (while MEMC is in supervisor mode), MEMC starts an I/O cycle by driving $\overline{IORQ}$ low and holding the processor clocks (stretching the processor cycle when PH2 is high). The I/O controller signals when it is ready to end the I/O cycle by driving $\overline{IOGT}$, low. The I/O cycle ends when both $\overline{IORQ}$ and $\overline{IOGT}$ are seen low on the rising edge of REF8M. Then MEMC drives $\overline{IORQ}$ high and releases the processor clocks. The I/O controller de-asserts $\overline{IOGT}$ which goes high on the next falling edge of REF8M.

A MEMC I/O cycle is shown in *Figure 18* below. The cycle starts with $\overline{\text{IORQ}}$ being taken low. There follows a number of 8 MHz clock ticks until the I/O controller is in a position to complete the cycle. The $\overline{\text{IOGT}}$ line is taken low, and both MEMC and the I/O controller see $\overline{\text{IORQ}}$ and $\overline{\text{IOGT}}$ low on the rising edge of REF8M, so the I/O cycle terminates on the next falling edge of REF8M.

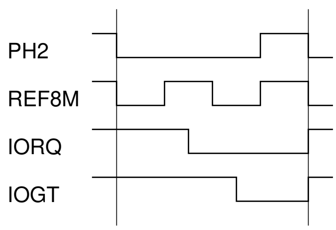
Figure 18: I/O cycle



Some I/O cycles may only take 250ns as shown in *Figure 19* below. To give the I/O controller adequate time to recognise such operations, MEMC produces the first $\overline{\text{IORQ}}$ early in the I/O cycle.

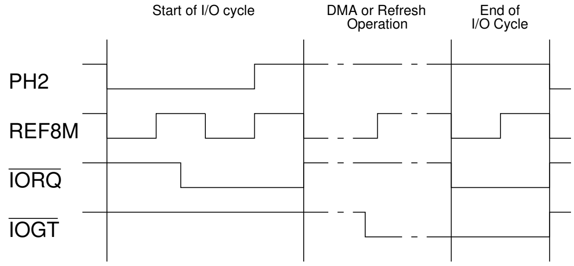
The extension of $\overline{\text{IORQ}}$ only happens at the start of an I/O cycle; if the $\overline{\text{IORQ}}$ signal is removed during a DMA or refresh operation, it will be reasserted when REF8M goes low.

Figure 19: Fast I/O cycle



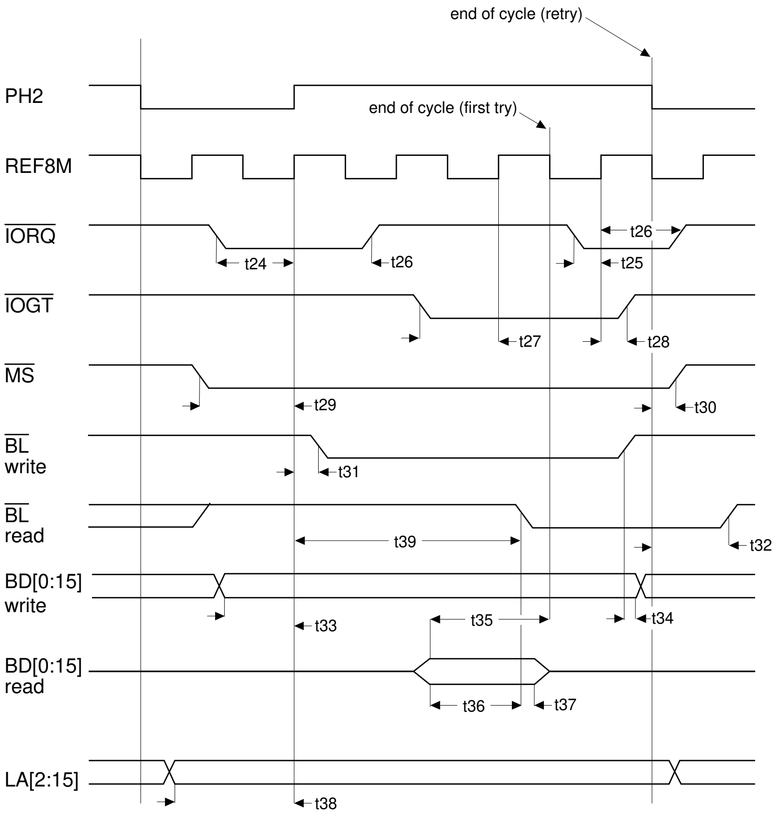
I/O cycles may be interrupted by DMA and refresh operations, as shown in *Figure 20* below. If a DMA or refresh operation is pending, the $\overline{\text{IORQ}}$ signal is driven high when REF8M next goes low. The DMA/refresh operation may then begin. When the operation completes, the I/O cycle is resumed by setting $\overline{\text{IORQ}}$ low (provided no more DMA or refresh operations are pending). The DBE line is always driven low by MEMC during DMA/refresh operations to disable the processor data bus drivers. Hence the I/O cycle is stretched, and the write data would become invalid during the cycle. The data must therefore be latched into the data bus buffers by the I/O controller during the first $\overline{\text{IORQ}}$ low period, and be held until the I/O cycle has completed. This is done by the I/O controller driving $\overline{\text{BL}}$ low for this period. The maximum time for which an I/O cycle may be interrupted in this way is 1875ns (i.e. 15 REF8M cycles).

Figure 20: I/O cycle interrupted by a DMA or refresh operation



Note: Care must be taken not to address a non-existent I/O controller, as MEMC will hold the processor clocks indefinitely until a low is seen on the $\overline{\text{IOGT}}$ line, or RESET is set high.

Figure 21: MEMC expansion card timing



Sym	Parameter	Min	Max
t24	$\overline{IORQ}$ setup (first attempt)	70	115
t25	$\overline{IORQ}$ setup (retries)	50	75
t26	$\overline{IORQ}$ hold	50	
t27	$\overline{IOGT}$ setup	25	120
t28	$\overline{IOGT}$ hold	20	100
t29	$\overline{MS}$ setup to REF8M[3]	110	
t30	$\overline{MS}$ hold	5	
t31	$\overline{BL}$ delay write	0	65
t32	$\overline{BL}$ hold read	10	100
t33	BD[0:15] setup write	85[1]	
t34	BD[0:15] hold from $\overline{BL}$ write	5	
t35	BD[0:15] setup to REF8M read	50[2]	
t36	BD[0:15] setup to $\overline{BL}$ read	20	
t37	BD[0:15] hold from $\overline{BL}$ read	15	
t38	LA[2:15], PR/$\overline{W}$ setup to REF8M	140	
t39	$\overline{BL}$ Read delay	60	

[1] with $\overline{BL}$ high i.e. buffers transparent.

[2] setup to the earliest possible end of cycle.

[3] the first rising edge of REF8M after $\overline{IORQ}$ goes low.

MEMC expansion card timing

A typical cycle with timing parameters is shown in Figure 21: *MEMC expansion card timing* on page 20. The sequence consists of a MEMC expansion card access, a DRAM refresh by MEMC, with the end of the expansion card access delayed by one REF8M cycle.

The Acorn enhanced expansion bus backplane pin-out

The Acorn enhanced expansion bus pin-out is based around the existing two row pin-out of the basic Acorn expansion bus. A third row as been added and several pins which were previously reserved have been used to create the Acorn enhanced expansion bus pin-out. The enhanced pin-out includes an extension on the existing I/O data and address buses, and also includes the extra control signals needed for the DEBI.

The Acorn enhanced expansion bus backplane pin-out is shown below. The signals shown in bold show the enhancements made to the existing Acorn expansion bus.

Table 2: Acorn enhanced expansion bus backplane pin-out

Pin	a	b	c	Description
1	0v	**0v**	0v	ground
2	LA[15]	**nBW**	-5v[1]	processor BYTE/WORD
3	LA[14]	**LA[23]**	0v	ground
4	LA[13]	**LA[22]**	0v	ground
5	LA[12]	**LA[21]**	**Ready**	ready stretch
6	LA[11]	**LA[20]**	MS[2]	MEMC exp card select
7	LA[10]	**LA[19]**	**DRQ[3]**	DMA request
8	LA[9]	**LA[18]**	**DACK[3]**	DMA acknowledge
9	LA[8]	**LA[17]**	Reserved	
10	LA[7]	**LA[16]**	Reserved	
11	LA[6]	**LA[1]**	**TC[3]**	terminal count
12	LA[5]	**LA[0]**	RST	reset (see note)
13	LA[4]	**0v**	PR/W	read / not write
14	LA[3]	**BD[31]**	IOWR	write strobe
15	LA[2]	**BD[30]**	IORD	read strobe
16	BD[15]	**BD[29]**	PIRQ	normal interrupt
17	BD[14]	**BD[28]**	PFIQ	fast interrupt
18	BD[13]	**BD[27]**	**EAS[4]**	EASI space strobe
19	BD[12]	**BD[26]**	I2Cclk	I^2C serial bus clock
20	BD[11]	**BD[25]**	I2Cdat	I^2C serial bus data
21	BD[10]	**BD[24]**	Reserved	
22	BD[9]	**0v**	PS[5]	IOC expansion card select
23	BD[8]	**BD[23]**	IOGT	MEMC card handshake
24	BD[7]	**BD[22]**	IORQ	MEMC card request
25	BD[6]	**BD[21]**	BL	I/O data latch control
26	BD[5]	**BD[20]**	0v	supply
27	BD[4]	**BD[19]**	CLK2	2 MHz synchronous clock
28	BD[3]	**BD[18]**	CLK8	8 MHz synchronous clock
29	BD[2]	**BD[17]**	REF8M	8 MHz reference clock
30	BD[1]	**BD[16]**	+5v	supply
31	BD[0]	**0v**	**CLK16**	16MHz system clock
32	+5v	**+5v**	+12v[1]	supply

1. The A3000 is 5 volt only. Pins 2c and 32c are not connected.
2. On the A3000 External expansion card bus connector, this signal is MS[0].
3. Not all machines with DMA will support all slots. For instance, the Risc PC 600 only supports slots 0 and 1.
4. Expansion cards designed to be used in platforms prior to the Risc PC must leave pin C18 as no-connect. Expansion cards designed for use on both Risc PC and non-Risc PC machines must possess the ability of making C18 no-connect by the use of a jumper or link etc.
5. On the A3000 External expansion card bus connector this signal is PS[0].

To support interface cards of the existing Acorn expansion bus type (i.e. DEBI not included) only the signals on rows a and c are required, therefore the use of a 64-way DIN 41612ac connector is sufficient to connect to the expansion bus. If the interface designer chooses to use the DEBI then they must interface to rows a, b and c of the backplane, i.e. they must use a DIN 41612abc type connector.

The reserved pins must be left unused, as in some early machines these pins carry co-processor signals.

I^2C bus

This bus is primarily intended for Acorn's use and it should only be used within the computer.

- The maximum recommended load per expansion card is 20pF.
- The speed and timings may vary between computer models and operating systems.
- This bus is not a full implementation of the I^2C specification (e.g. it does not support other bus masters).
- This bus may not be incorporated in future machines.

Reset

The RST signal is the system reset signal. It is driven low at power-on, or by a user reset. It is an open-collector signal, and expansion cards may drive it, to generate a system reset. The pulse width should be at least 50ms.

Power consumption

It is strongly recommended that expansion cards work from the 5 volt rail only. Acorn expects future models to continue the trend, started with A3000, towards single supply computers.

The maximum current drain allowed from the computer +5 volt supply is 1.0A per expansion card slot, for external expansion cards. See *Appendix A* for details of internal expansion cards.

Each expansion card may draw a maximum of 10mA from the –5V rail and 250mA from the +12V rail. These voltages are not available on the A3000 and may not be available on new computers.

Double width cards may use twice the above current values when used with Archimedes.

Safety

The current industry wide IT safety standard is IEC950 whose European ‘harmonised’ version is EN60950 / BS7002, but your particular application may also be within the scope of other additional standards.

The main requirements are that the equipment provide protection against:

- the spread of fire
- hazardous voltages or energy.

The spread of fire

The Underwriters Laboratory (UL) of the USA have devised several standard ways of testing plastic materials for their flammability properties. The UL94 test procedure is used within IEC950 to specify the required flame retardant level for materials and components. The ratings start at 94v-0 down to 94v-1, 94v-2 and finally 94HB.

Confirmation of a UL test pass will be the issue of a ‘yellow card’, a copy of which can be obtained from your supplier, for either the plastic material itself or the component.

IEC950 specifies that PCBs will have a minimum flame-retardancy rating of UL94v-1 and that any components mounted on the PCB meets the lower standard of UL94v-2.

The choice and layout of components should also seek to prevent the spread of fire across the PCB and within the computer.

The computer external enclosure forms a fire barrier and as such its material must meet the higher standard of UL94v-1. As the expansion card rear panel will be part of the fire enclosure the panel itself must meet this standard; metal is acceptable.

If you fit a large plastic connector into this rear panel the connector will also have to be UL94v-1.

Hazardous voltages or energy

The computer PSU is designed to provide only SELV (Safety Extra Low Voltage) to the computer. This means that the PCB supply voltages have two independent means of protection against hazardous voltages, thus ensuring that even under single fault conditions the voltage on the computer PCB and interfaces will be safe. Within IEC950 hazardous voltages are those greater than 42.4V peak or 60V DC.

Physical access by a user to any hazardous voltages must be prevented by a physical barrier in which no hole is greater than 5 mm in diameter, and no slot is wider than 1mm, regardless of its length. For full details, see Section 4.3.16 of IEC950.

Expansion cards **must** be designed in such a way that they do not introduce either a reduction in fire protection or of voltage isolation into the host computer. Expansion card designers should obtain a copy of the standard and if necessary seek further clarification by consulting a reputable test facility such as BSI.

With the EEC Directive on General Product Liability manufacturers and importers within the Community are subject to strict liability. This existing law removes the plaintiff's requirement to prove that a product was ‘defective’ when seeking damages. The possession of an EN certification may not be an adequate defence and therefore manufacturers need to be aware that there may be safety aspects of their product which are not covered by the standards.

Safety testing: Electricity at work regulations, 1989

U.K. legislation now requires that portable electrical equipment be tested regularly, usually annually, for safety. Be aware that this testing may be carried out by a variety of people of varying technical competence and experience.

Class 1, earthed, equipment will be tested by the application of a high current, low voltage, source of 4-25 Amps, between any exposed metalwork and the earth pin in the mains plug. The presence of your expansion card in a computer will make it liable for testing.

You should therefore consider whether to provide information in your documentation to either the user or the dealer on how your card should be tested.

Important Note: Within some Risc PCs, the EMC coating provides a functional earth to the Eurocard style expansion cards. The EMC gasket on the rear panel of the card contacts the coating at a number of points. These contact points are **not** capable of carrying the protective earth test current of 25 Amps. If a card is subjected to this high current the coating at the contact points will burn away and you will lose the functional / EMC earthing of the expansion card rear panel. This will not reduce the safety of the equipment but will compromise the EMC performance, and possibly the functionality, of the card. You should therefore ensure that any expansion card documentation mentions this point.

EMC design

Acorn's computers are designed to meet current EMC requirements. Expansion cards should be designed such that the existing EMC performance of the computer is not reduced. The following points should be borne in mind.

- To provide the Risc PC with good EMC performance all expansion cards should be fitted with an EMC gasket. The blank version of the EMC gasket is shown on Acorn drawing Number 0297,093. The gasket is fitted on the inside of the back panel. To guarantee correct EMC performance the Risc PC should always contain a blanking gasket in each unused expansion card slot.

- All external connectors should be of a robust, recognised EMC design.
- All external connectors should have a comprehensive low impedance bond to the rear panel.
- The rear panel must be conductive and fitted to the host computer frame via low impedance fixings.
- Any painted finish on the rear panel must not cause the expansion card to be isolated from the computer's earthing system.
- When fitted, there should not be any continuous slot longer than 20mm around the expansion card rear panel.
- Any external cables must have an EMC performance that does not compromise the host computer system.
- If external cables are required for the card, but not supplied, full details of the recommended cable, connectors and construction methods of the leads should be specified.
- A four-layer PCB is preferred together with a good layout.
- Connection of the card 0V line and the rear panel is not generally recommended. However, on some PCB designs this may be found to give an improvement in static immunity.
- Be aware that the rear panel may be subject to high current earth continuity testing; see the *Safety testing: Electricity at work regulations, 1989* on page 22.
- The EMC performance of the card must not deteriorate as a result of insertion and removal during the life of the product.

Mechanical specification

Expansion cards should be built to the specifications given in drawing number 0276,099, noting that the thickness of the back panel is 1.6mm. This drawing and drawing number 0276,204 give the relevant dimensions of single- and double-width expansion cards, and their mounting brackets respectively. Both drawings are attached. Note that not all machines support both single- and double-width cards.

Mini expansion cards are described in drawing number 0280,080, attached.

Note that solder tails should be no longer than 2mm.

Blanking panels

Single-width expansion cards not targeted at the Risc PC should be accompanied by a single-width blanking panel and T-piece, to blank off the aperture otherwise left when the full-width Acorn blanking panel is removed to fit the card. Drawings of the T-piece (0276,036) and the blanking panel (0276,035) are attached.

To minimise costs to Risc PC owners, it may be advisable to provide a blanking panel and T-piece as a separate installation kit.

Paint specification

The cream paint colour for back panels on machines prior to the Risc PC is RAL 1013C. For the Risc PC, the paint colour is PMS Cool Grey 2. Expansion cards which are designed to fit machines of all types can be finished in nickel plate, and need not be painted.

Archimedes 300 series and 440 computers and R140 workstations

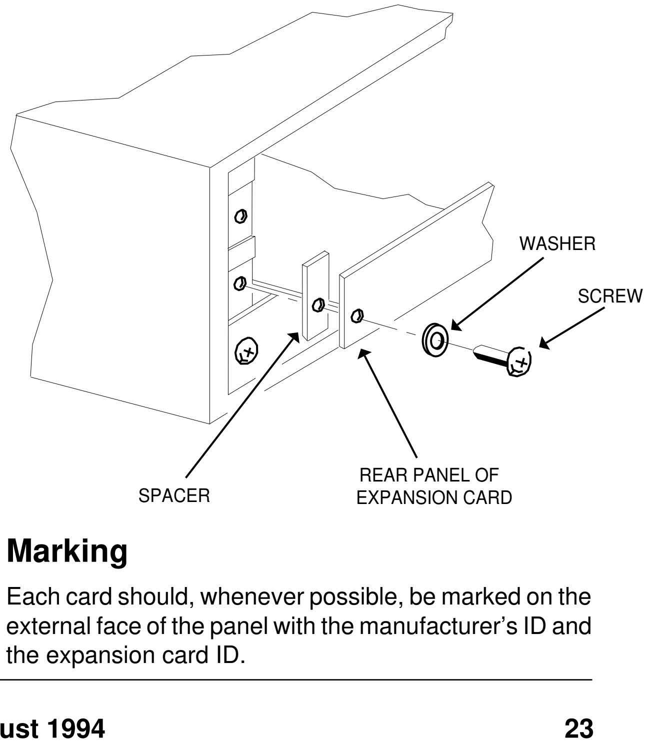
Note that the expansion card slot in Archimedes 300 series, 440 and some R140 machines is 2.5mm shorter than it should be (from backplane to rear panel) for a standard Eurocard. This was corrected for Archimedes 400/1 series, R260 and later computers. Expansion cards which may be used by these earlier computers should therefore be supplied with two spacers (see the attached drawing, number 0276,227, for details of the spacer). This will allow the expansion card to be seated securely, without causing the backplane to lean excessively towards the front of the machine.

The original screws supplied with the computer will be long enough to accommodate the spacers, provided that the expansion card backplate thickness does not exceed 1.6mm. If your card has a thicker rear panel, you may need to supply longer fixing screws.

Fitting

The diagram below shows how the spacers are fitted to earlier computers:

Figure 22: Fitting spacers to earlier models



Marking

Each card should, whenever possible, be marked on the external face of the panel with the manufacturer's ID and the expansion card ID.

Appendix A: Mini Expansion Card Specification

Introduction

This appendix describes some differences to the text in the *Acorn Enhanced Expansion Card Specification* that apply to mini expansion cards.

The mini expansion card is the name given to the single internal expansion card upgrades which fit in the A4000 and A3000 series computers. The mini expansion card interface plugs into four SIL Molex sockets (two 17-way connectors and two 5-way connectors) towards the back of the main PCB. It has a metal back panel as provision for external plug and socket connections.

Some information given in the *Acorn Enhanced Expansion Card Specification* is repeated here in the interests of clarity.

Types of mini expansion card

There are two types of mini expansion card: IOC access and MEMC access. All mini expansion cards decode at Podule slot 1.

IOC access mini expansion cards

These use IOC to generate the $\overline{\text{IOGT}}$ signal required by MEMC. IOC allows four variations of access timing to I/O, selected by address value. These different timings are slow, medium, fast and synchronous. They are defined in the *Acorn Enhanced Expansion Card Specification*, though the following signals are **not** supplied to the mini expansion card interface:

- CLK8 (not supplied, though signals shown are relative to it)
- $\overline{\text{RBE}}$
- BD[8:15]
- LA[14:15] (the podule slot decodes).

See Figure 14: *General timing for slow, medium, and fast cycle types* on page 15 of the *Acorn Enhanced Expansion Card Specification*, and note that other figures refer to these signals, which should be ignored for mini expansion cards. Also note comments in *MEMC mini expansion card timing* below, regarding access timings.

Note that a synchronous read of the byte at base address 0 of the mini expansion card is performed to establish the expansion card identity.

MEMC access mini expansion cards

These work directly with MEMC $\overline{\text{IORQ}}$ and $\overline{\text{IOGT}}$ signals, and can therefore optimise the cycle time of the mini expansion card application. The signals mentioned in the previous section are obviously not supplied.

Warning: If the MEMC I/O space is accessed when no MEMC mini expansion card is present, the system will hang waiting for $\overline{\text{IOGT}}$ to be returned, and will need to be reset.

Note: MEMC mini expansion cards must also have some IOC access logic to allow synchronous read of the expansion card identity byte, initial ROM data, etc.

Notes on expansion card specification

System architecture

All expansion cards except MEMC podule 1 and IOC podule 1 are irrelevant with respect to mini expansion cards.

I/O space memory map

Only the addresses reserved in the table (on page 5 of the *Acorn Enhanced Expansion Card Specification*) for expansion card 1 apply to the mini expansion card.

Data bus mapping

As only BD[0:7] are available for mini expansion cards, the I/O data bus BD[0:7] connects to the main system data bus D[0:31] via bi-directional latches as follows:

- During WRITE to peripheral, BD[0:7] is mapped to D[16:23].
- During READ from peripheral, BD[0:7] is mapped to D[0:7].

MEMC mini expansion card timing

Note that the timings quoted in the tables of the *Acorn Enhanced Expansion Card Specification* are relative to the indicated vertical timing marks, but the drawn signal positions are not necessarily correct with reference to the REF8M clock signal.

Signals

The following signals are defined in the *Acorn Enhanced Expansion Card Specification* (note that pin 1 is at the lefthand end of the plugs as viewed in *Figure 1* below):

Plug position	A	B	C	D
Pin				
1	0V	0V	C0	+5V
2	+5V	REF8M	C1	$\overline{PWE}$
3	$\overline{PRE}$	$\overline{PFIQ}$	$\overline{B1}$	$\overline{PS1}$
4	PR/nW	$\overline{MS1}$	$\overline{IORQ}$	CLK2
5	LA4	+5V	$\overline{IOGT}$	LA2
6	LA5			LA3
7	LA6			BD0
8	LA7			BD1
9	0V			BD2
10	LA8			BD3
11	LA9			BD4
12	LA10			BD5
13	LA11			BD6
14	LA12			BD7
15	LA13			$\overline{RST}$
16	$\overline{PIRQ}$			0V
17	0V			+5V

Power specification

The following table gives the maximum power requirements for the mini expansion card.

	Inside machine	External
Mini expansion card	100mA	500mA
Maximum power dissipation	500mW	2.5W

Note: The external 2.5W allocation must **not** be dissipated inside the case as the additional heating may cause a reduction in reliability.

Mechanical specification

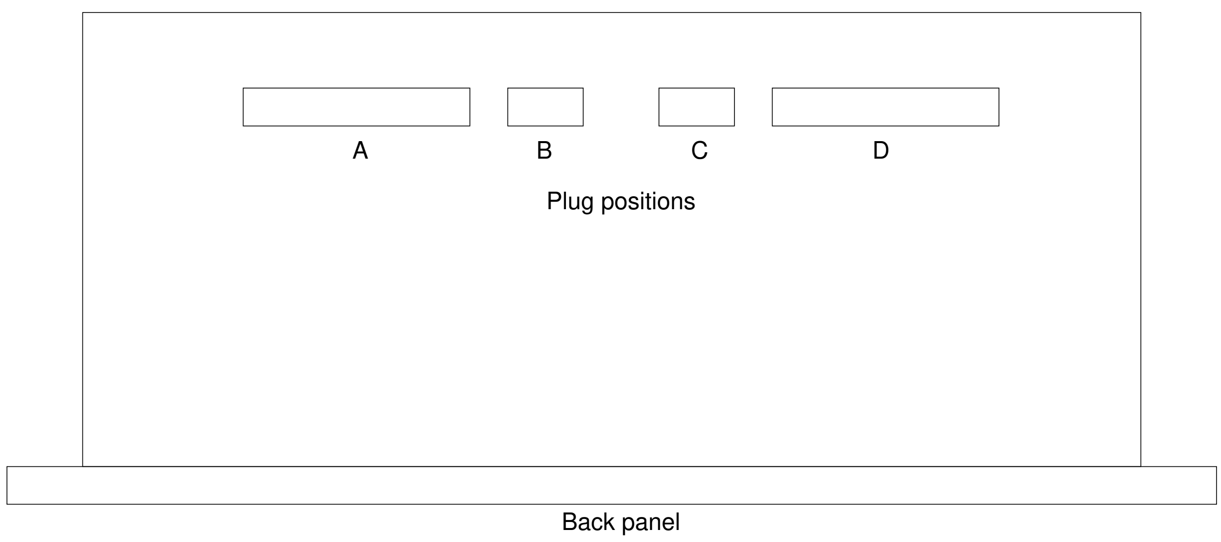
The physical dimensions and component clearance dimensions of the mini expansion card are specified in the drawing called the Mini Expansion Upgrade, (0280,080). An EMC gasket to drawing 0294,081 is also required on A3010 and A3020 machines to maintain EMC performance when a mini expansion card is fitted.

Additional points

Each card should, whenever possible, be marked on the external face of the panel with the manufacturer's ID and the expansion card ID.

Access to these machines is covered by the *Welcome Guide*. Instructions must be supplied to detail how the mini expansion card and label are fitted, thus allowing users to upgrade these machines themselves, and to test that the installation has been done correctly.

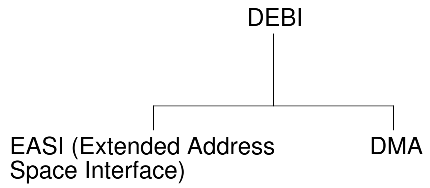
Figure 1: Top view of mini expansion card



Appendix B: DMA Extended Bus Interface

Introduction

This appendix describes the DMA Extended Bus Interface (DEBI). DEBI is presently fully supported on the first two expansion slots of the Risc PC 600, although future machines may support all or only part of the interface. The DEBI bus includes two types of interface, DMA and the new Extended Address Space Interface (EASI). Any expansion card which includes a DMA interface should be termed a 'DEBI' card, and any expansion card which supports EASI but not DMA should be termed an 'EASI' card.



DMA

The Direct Memory Access (DMA) transfer can be byte, half-word or word wide and appear on the expansion card as one of four different transfer cycle types. The four cycle types/speeds which can be used are types A, B, C and D. The slowest of these is A and the quickest is D. As mentioned in the section entitled *DEBI and EASI expansion cards* on page 3 of the *Acorn Enhanced Expansion Card Specification*, the data buffer size of the DMA channel means that data can be transferred in continuous blocks of between 1 byte and 4KB. Each DMA channel has two such data buffers, both up to 4KB in size, so data transfer speeds can be optimised by performing DMA transfers from one buffer whilst programming the other buffer ready for the next block transfer. The use of two buffers for large DMA transfers will achieve a close to continuous flow of data in 4KB page sizes.

Timing information for all DMA cycle types is shown on page B-4 of this appendix. A full description of how the DMA channel is programmed is beyond the scope of this document and can be found in the *RISC OS 3 Programmer's Reference Manual*, Vol. 5 (version 3.5 supplement) part number 0497,551.

As suggested above, expansion cards containing any DMA capabilities should be termed 'DEBI' expansion cards. Expansion cards containing both DMA and EASI capabilities should also be termed 'DEBI' expansion cards.

EASI

EASI accesses can be byte, half-word and word wide transfers and may be of two different access cycle speeds. The two cycle types/speeds are type A and type C, type C being the shorter, and therefore quicker, cycle of the two.

Each expression slot supports up to 16MB of extended address space which can be addressed as byte or word wide, hence the use of half word data transfers to EASI space will only utilize 50% of the area.

The two access cycle types can be stretched by the expansion card to match slower peripherals. To stretch an EASI transfer cycle the expansion card must assert the Ready line soon after the transfer cycle begins and keep the line asserted until the expansion card is ready to complete the cycle – see *Extended address space (EASI) timings* on page B-3 for detailed timing.

As already mentioned, expansion cards containing an EASI type interface with no DMA capabilities should be termed 'EASI' expansion cards.

Major features of DEBI

The major features of the DEBI are as follows:

- 32 bit data bus allowing byte, half-word and word wide transfers
- An additional 16MB of address space per expansion slot
- DMA is available on selected slots
- A wide range of access cycle times
- 6MB/second transfer rate
- Intel-compatible control signals

It is anticipated that all DEBI and EASI expansion cards will be of the single Eurocard form factor. This enables card guides to be provided in the computer so making it easier for the user to install cards. Single Eurocard width also improves the EMC qualities of the computer and improves the integrity of the fire enclosure.

Description of signals

The following table shows the complete pin-out of the Acorn enhanced expansion bus. The DEBI specific signals are shown in bold and described in this section.

Table 1: Acorn enhanced expansion bus pin-out

Pin	A	B	C
1	0v	**0v**	0v
2	LA[15]	**nBW**	-5v
3	LA[14]	**LA[23]**	0v
4	LA[13]	**LA[22]**	0v
5	LA[12]	**LA[21]**	**Ready**
6	LA[11]	**LA[20]**	$\overline{\text{MS}}$
7	LA[10]	**LA[19]**	**DRQ**[1]
8	LA[9]	**LA[18]**	**$\overline{\textbf{DACK}}$**[1]
9	LA[8]	**LA[17]**	Reserved
10	LA[7]	**LA[16]**	Reserved
11	LA[6]	**LA[1]**	**TC**[1]
12	LA[5]	**LA[0]**	$\overline{\text{RST}}$
13	LA[4]	**0v**	PR/$\overline{\text{W}}$
14	LA[3]	**BD[31]**	$\overline{\text{IOWR}}$
15	LA[2]	**BD[30]**	$\overline{\text{IORD}}$
16	BD[15]	**BD[29]**	$\overline{\text{PIRQ}}$
17	BD[14]	**BD[28]**	$\overline{\text{PFIQ}}$
18	BD[13]	**BD[27]**	**$\overline{\textbf{EAS}}$**
19	BD[12]	**BD[26]**	I2Cclk
20	BD[11]	**BD[25]**	I2Cdat
21	BD[10]	**BD[24]**	Reserved
22	BD[9]	**0v**	$\overline{\text{PS}}$
23	BD[8]	**BD[23]**	$\overline{\text{IOGT}}$
24	BD[7]	**BD[22]**	$\overline{\text{IORQ}}$
25	BD[6]	**BD[21]**	$\overline{\text{BL}}$
26	BD[5]	**BD[20]**	0v
27	BD[4]	**BD[19]**	CLK2
28	BD[3]	**BD[18]**	CLK8
29	BD[2]	**BD[17]**	REF8M
30	BD[1]	**BD[16]**	+5v
31	BD[0]	**0v**	**CLK16**
32	+5v	**+5v**	+12v

1. The Risc PC only supports DMA on slots 0 and 1 of the backplane

The functional descriptions of the signals are as follows:

Table 2: Functional description of signals

Signal	Type	Description
LA [0..23]	type O	Latched version of the main system address bus.
BD[0..31]	type I/O	Buffered version of the main system data bus, see note.
nBW	type O	When low indicates byte wide access; when high indicates word or half word access. Not used during DMA transfers.
Ready	type I	Used to stretch EASI access cycles. Active low.
DRQ	type I	DMA request. Active high.
$\overline{\textbf{DACK}}$	type O	DMA acknowledge. Active low.
TC	type O	Terminal count. Indicates last cycle of DMA transfer is taking place. Active high.
PR/$\overline{\textbf{W}}$	type O	When high, this signal indicates that the cycle taking place is a read cycle, when low it indicates that a write cycle is taking place.
$\overline{\textbf{IOWR}}$	type O	Write strobe for I/O space access cycles. Active low.
$\overline{\textbf{IORD}}$	type O	Read strobe for I/O space access cycles. Active low.
$\overline{\textbf{EASI}}$	type O	EASI address space select signal. Active low.
CLK16	type O	16MHz system clock.

Note: On the Risc PC, the upper sixteen data lines BD[31:16] are driving at all times except:

- During DMA reads
- During reads from EASI space

Consequently, any expansion card with data bus buffers connected to the upper sixteen data bits should not enable them for output except during extended address space and DMA reads.

DMA control

A full explanation of the use of the DMA control registers is beyond the scope of this document. Please refer to the *RISC OS 3 Programmer's Reference Manual* for further details of the DMA channel control registers and DMA data transfers.

Signal timings

Extended address space (EASI) timings

The Extended Address Space Interface (EASI) access timings for cycle types A and C are as follows:

Table 3: EASI access timings

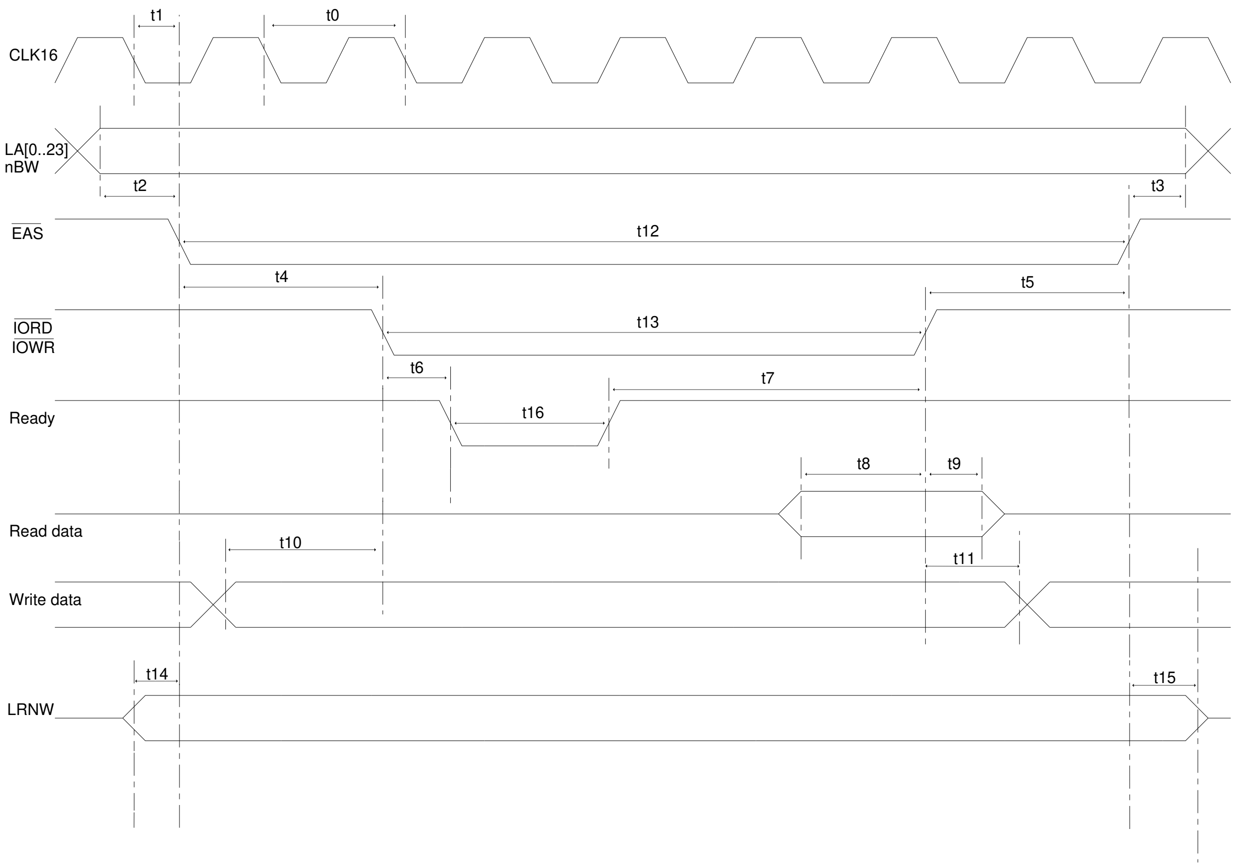
Sym.	Description	Cycles	Min	Typ	Max	Units
t0	CLK16 cycle time	all		62.5		ns
t1	Clock falling edge to $\overline{\text{EAS}}$ low	all	0		20	ns
t2	Address and nBW setup to $\overline{\text{EAS}}$ low	all	10			ns
t3	Address and nBW hold from $\overline{\text{EAS}}$ high	all	10			ns
t4	$\overline{\text{EAS}}$ low to $\overline{\text{IORD}}$ or $\overline{\text{IOWR}}$	A	80			ns
		C	15			ns
t5	$\overline{\text{IORD}}$ or $\overline{\text{IOWR}}$ high to $\overline{\text{EAS}}$ high	A	80			ns
		C	15			ns
t6	$\overline{\text{IORD}}$ or $\overline{\text{IOWR}}$ low to Ready low	A			50	ns
		C			50	ns
t7	Ready high to $\overline{\text{IORD}}$ or $\overline{\text{IOWR}}$ high	A	0		265	ns
		C	0		140	ns
t8	Read data setup to $\overline{\text{IORD}}$ high	all	20			ns

Table 3: EASI access timings (Continued)

Sym.	Description	Cycles	Min	Typ	Max	Units
t9	Read data hold from $\overline{\text{IORD}}$ high	A	10		90	ns
		C	10		90	ns
t10	Write data setup to $\overline{\text{IOWR}}$ low	all	10			ns
t11	Write data hold from $\overline{\text{IOWR}}$ high	all	25			ns
t12	$\overline{\text{EAS}}$ select width	A	427			ns
		C	175			ns
t13	$\overline{\text{IORD}}$ and $\overline{\text{IOWR}}$ strobe width	A	240			ns
		C	115			ns
t14	LRNW active to $\overline{\text{EASI}}$ low	all	10			ns
t15	$\overline{\text{EASI}}$ high to LRNW in-active	all	10			ns
t16	Ready Strobe width	all	0ns		2µS	

On future machines the DEBI may not use CLK16 as the master clock. It is not therefore recommended that CLK16 is used. The edge relationship of CLK16 to the other interface signals cannot be relied upon for simple state machine design. It is recommended that expansion cards contain their own 'on-board' clock to synchronise the interface signals.

Figure 1: Extended address space (EASI) timings



DMA access timings

The DMA access timings for Cycle types A, B, C, and D are as follows:

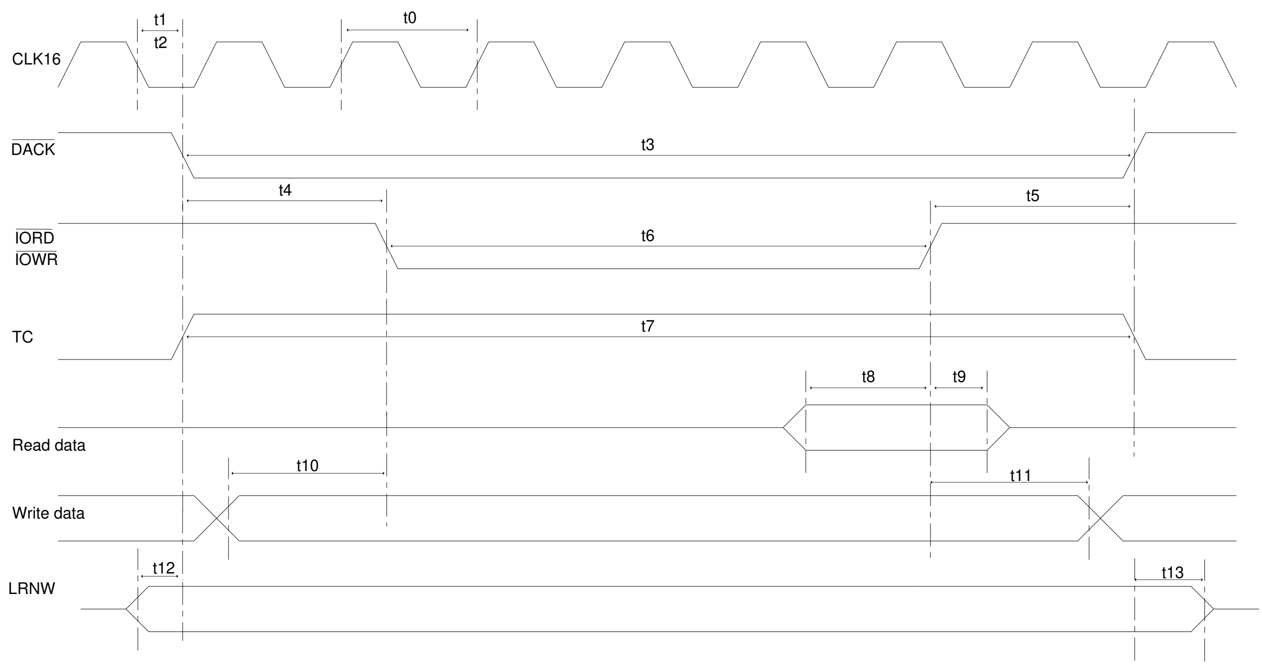
Table 4: DMA access timings

Sym.	Description	Cycles	Min	Typ	Max	Units
t0	CLK16 clock cycle	all		62.5		ns
t1	Clock low to $\overline{\text{DACK}}$ low	all	0		20	ns
t2	Clock low to TC high	all	0		20	ns
t3	$\overline{\text{DACK}}$ strobe width	A	427			ns
		B	302			ns
		C	175			ns
		D	115			ns
t4	$\overline{\text{DACK}}$ low to $\overline{\text{IORD}}$ or $\overline{\text{IOWR}}$ low	A, B	80			ns
		C, D	15			ns
t5	$\overline{\text{IORD}}$ or $\overline{\text{IOWR}}$ high to $\overline{\text{DACK}}$ high	A	80			ns
		B, C, D	15			ns
t6	$\overline{\text{IORD}}$ or $\overline{\text{IOWR}}$ strobe width	A	240			ns
		B	175			ns
		C	115			ns
		D	52			ns
t7	TC strobe width	A	427			ns
		B	302			ns
		C	175			ns
		D	115			ns
t8	Read data setup to $\overline{\text{IORD}}$ high	all	35			ns
t9	Read data hold from $\overline{\text{IORD}}$ high	all	10			ns
t10	Write data setup to $\overline{\text{IOWR}}$ low	all	10			ns
t11	Write data hold from $\overline{\text{IOWR}}$ high	all	25			ns
t12	LRNW active to $\overline{\text{DACK}}$ low	all	10			ns
t13	LRNW in-active from $\overline{\text{DACK}}$ high	all	10			ns

On future machines the DEBI may not use CLK16 as the master clock. It is not therefore recommended that CLK16 is used. The edge relationship of CLK16 to the other interface signals cannot be relied upon for simple state machine design. It is recommended that expansion cards contain their own 'on-board' clock to synchronise the interface signals.

Note: The LRNW timings of t12 and t13 may not be supported on future platforms.

Figure 2: DMA access timings



DRQ timings

Table 5: DRQ timings

Sym.	Description	Min	Typ	Max	Units
t1	$\overline{\text{DACK}}$ low from CLK16 low	0		20	ns
t2	DRQ low to $\overline{\text{DACK}}$ high	30			ns
t3	$\overline{\text{DACK}}$ low to DRQ de-assert	5		See below[1]	ns

1. The maximum figure will vary depending on cycle type, i.e. width of DACK. The only requirement for a maximum figure is that DRQ is de-asserted a minimum of 15nS before DACK rises (see t2).

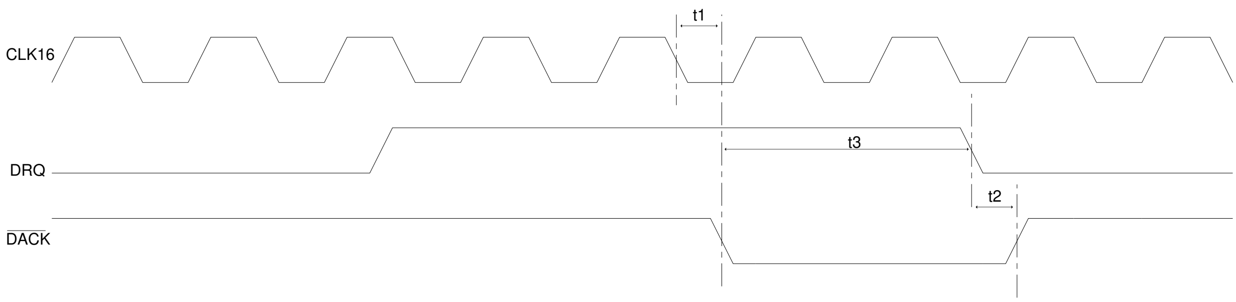
Design notes

It is allowable to keep DRQ asserted if a peripheral wants to perform multiple DMA cycles. The removal of DRQ at the end of a multiple transfer must still adhere to t2 and t3.

Once asserted DRQ should never be de-asserted before a responding $\overline{\text{DACK}}$ low as occurred (see t3).

If DRQ is de-asserted during $\overline{\text{DACK}}$ low, it should only be re-asserted a minimum of 40ns after $\overline{\text{DACK}}$ goes high.

Figure 3: DRQ timings

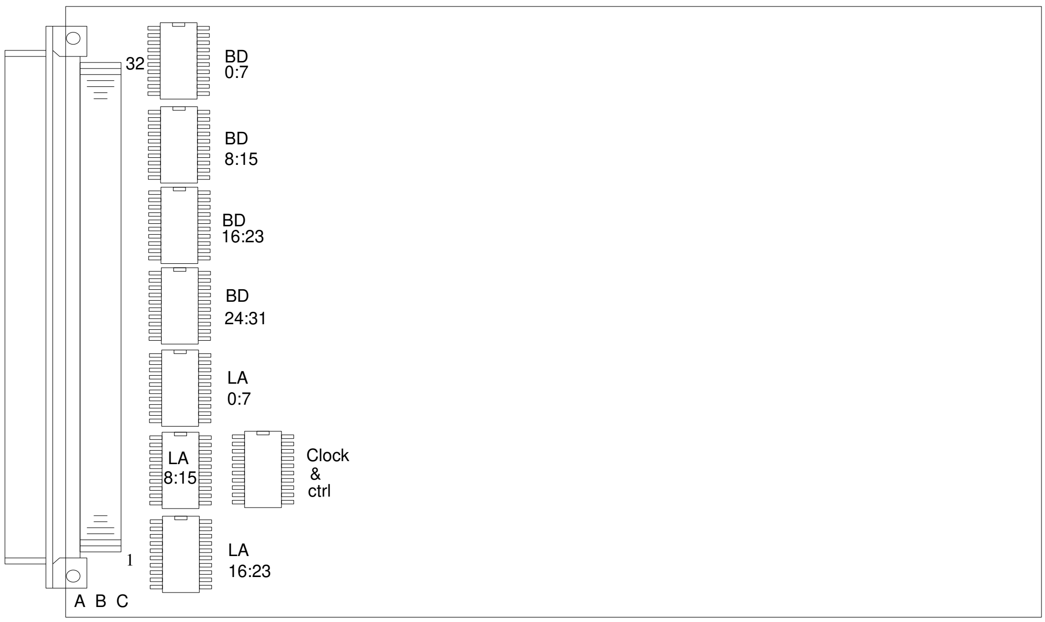


Electrical characteristics

Table 6: Electrical characteristics

Parameter	Symbol	Min	Typ	Max	Units
All signals					
Low-level input voltage	VIL		0.8	V	
Vcc=4.75v					
High-level input voltage	VIH	2.0		V	
Low-level output voltage	VOL		0.15	0.26	V
I0L = -4.0mA					
High-level output voltage	VOH	3.5		V	
IOH=Max Vcc=4.75					
High-level output current	IOH		-400	uA	
V0 = 3.0v Vcc=4.75v					
Power available per slot					
Output voltage	Vcc	4.75	5.0	5.25	V
Output current	Icc		1.0	A	
Output voltage	V(+12)	11.4	12.0	12.6 V	
Output current	Icc	250	mA		
Output voltage	V(-5)	-4.75	-5.0	-5.25	V
Output current	Icc		-10	mA	

Figure 4: Recommended buffer layout



Design considerations

It is recommended that all bus signals on an expansion card that are to be used on multiple inputs or devices far from the connector are buffered, with the buffer devices being physically close to the signal connector.

Signals should not be presented with a capacitive load greater than 20pF; this is equivalent to the connector (5pF), 6cm of 0.2mm copper track (5pF), and 10pF input capacitance of the buffer device (e.g. 74HC245 for data lines). Unidirectional buffers will have lower input capacitance (typically 3.5pF). Tracks should always be kept as short as possible.

This gives worst case rise and fall times for all data bus signals of 12nS. Other signals will be equal or better. *Figure 4* shows a recommended layout for buffer devices on an expansion card. In practice it is unlikely that all 32 data bits and all 24 address lines would be used. Additional space savings can be achieved by using surface mount devices.

The direction of data bus buffers should be controlled by LRNW and enabled by the combination of the required select strobes and the nBW signal.

Clock signals may be buffered with AC logic where their timing in relation to other bus signals is critical to the proper operation of the design.

Power consumption limitations must be adhered to and consideration must be given to potential noise generated by many outputs switching simultaneously. If all 32 data lines are to be driven, extra smoothing of the +5v rail may be required, or a system of staggered buffer enabling added. As a matter of course all power rails in use should be smoothed with 47uF electrolytic capacitors, in addition to the 100nF decoupling provided close to each digital integrated circuit device.

Four-layer PCBs will provide cleaner power supply rails and make the task of keeping track lengths within recommended limits easier. However, it is permissible to use two-layer PCBs, provided that power tracks are the optimum thickness to keep impedance to a minimum, and component density is low.

Risc PC expansion bus backplane

The motherboard of the Risc PC provides a 132-way edge connector for use with plug-in backplane add-ons. Electrically, the Risc PC I/O interface supports backplanes with up to 8 slots. All decoding needed to interface the relevant select signals per slot is provided on the plug-in backplane.

Firmware support

Full support is provided for expansion card driver software located in the extended address area. This means that an expansion card utilising the extended address space need not have ROM located in IOC space and as a consequence no page latch is required.

A ROM located in the extended address space may be byte, half-word or word wide. Driver firmware must be transferred to DRAM before being executed; it cannot be executed in place.

Reader's Comment Form

Acorn Enhanced Expansion Card Specification (Issue 5)

We would greatly appreciate your comments about this Manual, which will be taken into account for the next issue:

Did you find the information you wanted?

Do you like the way the information is presented?

General comments:

If there is not enough room for your comments, please continue overleaf

What do you use this manual for?

☐ **General interest** ☐ **H/W development** ☐ **S/W development** ☐ **Other (please specify)** ....

Cut out (or photocopy) and post to:

Dept RC, Technical Publications
Acorn Computers Limited
Acorn House
Vision Park
Histon
Cambridge CB4 4AE
England

Your name and address:

This information will only be used to get in touch with you in case we wish to explore your comments further