International Standard 6951

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION•МЕЖДУНАРОДНАЯ ОРГАНИЗАЦИЯ ПО СТАНДАРТИЗАЦИИ•ORGANISATION INTERNATIONALE DE NORMALISATION

Information processing — Processor system bus interface (Eurobus A)

Traitement de l'information — Bus d'interfaces entre processeurs (Eurobus A)

First edition — 1986-12-15

UDC 681.3.01 **Ref. No. ISO 6951-1986 (E)**

Descriptors : data processing, information interchange, network interconnection, connecting equipment, specifications, instructions.

Price based on 54 pages

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work.

Draft International Standards adopted by the technical committees are circulated to the member bodies for approval before their acceptance as International Standards by the ISO Council. They are approved in accordance with ISO procedures requiring at least 75 % approval by the member bodies voting.

International Standard ISO 6951 was prepared by Technical Committee ISO/TC 97, *Information processing systems*.

Users should note that all International Standards undergo revision from time to time and that any reference made herein to any other International Standard implies its latest edition, unless otherwise stated.

Printed in Switzerland

Contents

This page intentionally left blank

Information processing — Processor system bus interface (Eurobus A)

0. Introduction

0.1 General. This standard specifies the set of signal lines that constitute the bus itself, and the interfacing of devices connected to the bus.

This standard specifies protocols for the allocation of bus time to devices wishing to make transfers and for the transfer of data between devices. The standard does not, however, specify priority rules, these being left to be formulated individually for each system.

This standard specifies a full set of signalling rules to be followed by the device responsible for bus allocation and by devices conducting transfers. Annex F gives illustrative examples of each of the possible types of transfer.

The set of electrical and signal timing requirements specified in clause **6** uniquely defines the interface that is Eurobus A. Certain mechanical requirements are specified in clause **6**, namely those that directly affect the electrical characteristics (e.g. the physical length of the bus, the spacing of device connectors on the bus, the pin pitch on connectors and the signal disposition on the connectors), but this standard does not further specify the mechanical implementation. An example of a possible mechanical implementation of Eurobus A is given in annex J.

Implementations of Eurobus A are possible with 8, 16, 24, 32, . . . -bit data widths and devices having different data widths can operate on the same bus. Logical implementation summaries for the first four of the possible data widths are given in annexes A to D. Annex E specifies the connector allocation.

The group of signal lines constituting an assembled Eurobus A provides the means for the transfer of binary digital information between up to 20 devices plugged into the backplane of a single equipment shelf. Devices share the bus on a time-division multiplex basis. The length of the backplane is limited to a maximum of 460 mm. The signal lines form an asynchronous unbalanced voltage interface capable of operating at transfer rates of up to $6{,}5 \times 10^6$ words or bytes per second.

0.2 Data width and addressing capability. The data/address width of any device using the bus is theoretically unconstrained. However, the asynchronous protocols and addressing facilities of Eurobus A permit devices of 8, 16, 24 and 32-bit data widths to share the same bus, and when the bus is so shared, the maximum data width is that of the widest device.

The full addressing capability of the bus enables devices to address any 8-bit byte of any word in a normal address space defined by both of the following.

(a) The addressing range determined by the number of data/address bits.

(b) A two-bit extension to the foregoing (a). The full two-bit extension is available on buses with non-shared width, but on shared-width buses the use of these bits is restricted.

In addition, any complete word can be addressed in a second address space of equal magnitude to the first, designated the pseudo address space.

0.3 Devices. Free choice is available to the system designer as to the devices connected to a Eurobus and the order in which they are connected. However, each bus needs to include both:

(a) an arbiter, the purpose of which is to control the time-division multiplexing of transfers on the bus;

(b) if communication with other buses is required, a bus link to each of the other buses.

Figure 1 shows an example of a bus with a number of typical devices including an arbiter and a bus link.

0.4 Bus allocation. Information is transferred between devices on a master-and-slave basis. A device bids for control of the bus by means of its starred Request line and becomes the master device for that transfer after the arbiter has allocated the bus to it. This standard specifies the protocols by which devices bid for use of the bus and by which the arbiter allocates the use of the bus to one of them. The standard does not, however, specify the algorithm used in making the selection, thus the system designer is given the choice of an allocation algorithm in order to optimize system performance.

The protocol whereby a master device may flag an interrupt to the arbiter is also specified, but the subsequent action by the arbiter is left to the system designer to define.

0.5 Bus transfers. In addition to specifying the protocols for the execution of Read cycles (in which the master addresses a device as slave and reads data from it) and Write cycles (in which the master transfers data to the addressed slave), this standard also specifies the protocol for a Vector cycle in which an address, without data, is transferred from master to slave.

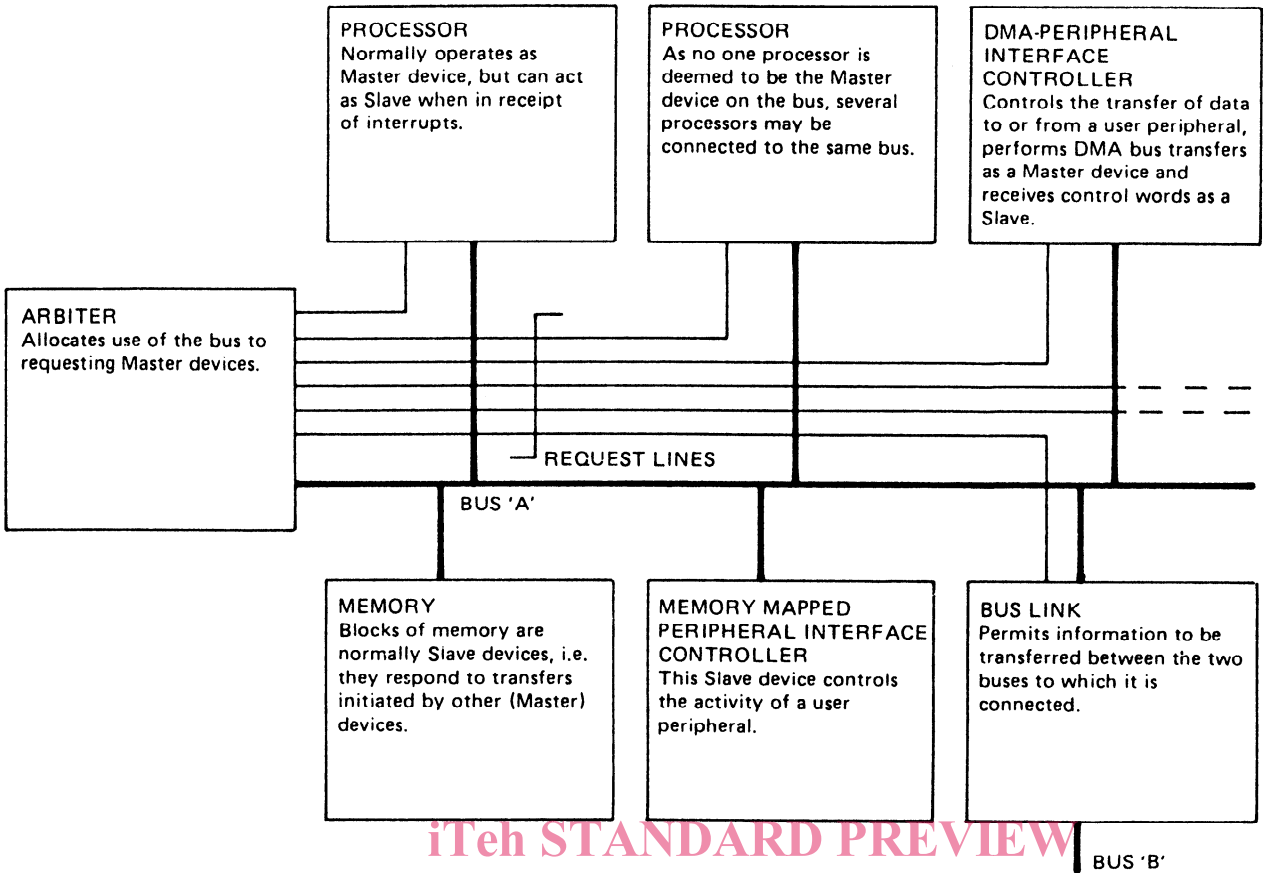

Figure 1. Eurobus with some typical devices

The bus allocation protocols permit a master to hold the bus for repeated use without the need to make a fresh bid for every transfer, while also giving the arbiter the ability to instruct any master to release the bus for reallocation. A master is also permitted to retain the bus for an indivisible sequence of cycles, such as a Read-Modify-Write sequence. An additional protocol is defined whereby the arbiter may abort a cycle that is deemed to have failed.

0.6 Interbus transfers. The protocols for Read, Write and Vector cycles permit a master on bus A, for example, wishing to effect a transfer with a slave on bus B, to address a bus linker on bus A as slave. The bus linker then bids for use of bus B as master and addresses the required slave on bus B. Should master devices on both buses attempt simultaneous transfers, the bus link cannot become master on either bus and a condition of deadly embrace ensues. The Eurobus protocols permit the embrace to be broken simply.

The protocols used by bus links to perform interbus addressing and data transfer are not within the scope of this standard.

0.7 Electrical requirements. The standard specifies the electrical and timing requirements that need to be obeyed by Eurobus A devices. Aspects covered within the electrical requirements include:

(a) the voltage levels of the active and quiescent logic states on the bus;

(b) the required characteristics of the termination networks;

(c) the required characteristics of the bus transmitters and receivers;

(d) the required characteristics of the spurs to be connected to the bus.

The specified set of electrical characteristics presupposes certain bus settling times for the transitions on the signal lines. Arising from these, certain timing constraints are specified. These constraints ensure that the relevant signal lines will have settled to the appropriate state before an associated control signal transition is issued.

0.8 Commercial and military versions. Two versions of Eurobus A are specified in this standard, a version for a commercial temperature range (0 °C to 70 °C) and a version for a military temperature range (−55 °C to 125 °C). Where the requirements are different they are separately specified for each version.

1. Scope and field of application

This International Standard specifies a processor system bus interface known as Eurobus A (referred to in the following text as "the bus") that is one of a family of interfaces for use in modular data acquisition, processing communication and control systems for military, industrial and other applications.

NOTE 1. More detailed information about the requirements specified in this International Standard, including the data width and addressing capability, devices connected to the bus, bus allocation, bus transfers, interbus transfers and electrical requirements, and background information are given in clause **0**.

NOTE 2. In this International Standard, upper case letters are used for the first letter of names of bus cycles.

NOTE 3. The titles of the publications referred to in this International Standard are listed in annex P.

2. Definitions

For the purposes of this International Standard the following definitions apply.

2.1 address. The location of a data word, or the value on the highway during the addressing phase of any Read, Write or Vector cycle.

2.2 arbiter. The device that performs the function of arbitration for the bus and is also responsible for servicing interrupts and for timing-out failed cycles and aborting them.

2.3 arbitration. The means whereby use of the bus is allocated to one of the bidding devices which then becomes the master.

2.4 backplane. The assembly of the bus with connectors into which spur cards may be plugged.

2.5 bidding device. A device that wishes to initiate a cycle or group of cycles on a bus and that requests use of the bus.

2.6 bus. The complete set of bus lines used by a particular implementation of Eurobus.

2.7 bus cycle. A closed group of signals on the bus that convey information between devices connected to it. This group consists of an addressing phase, in which the master places an address on the highway for recognition by a slave and, except in Vector cycles, a subsequent data transfer phase.

2.8 bus line. An electrical connection between two or more devices.

2.9 bus linker. A device that plugs into two or more buses thus providing a means whereby a master on one bus may transfer information to or from a slave on another bus.

2.10 bus voltage. The voltage on a bus line measured relative to the bus zero voltage reference.

2.11 byte. A contiguous group of 8 bits.

2.12 circuit card. A card on which various electronic components are mounted and that plugs into a Eurobus backplane as a spur.

2.13 data. The information held at, written to, or read from an address.

2.14 deadly embrace. The conditions when two interbus transfers, using the same bus linker, have been commenced and neither transfer can be completed.

2.15 device. A functional block, occupying one or more circuit cards, that communicates with other functional blocks by means of the bus or a subset of the bus.

2.16 extender panel. A circuit card that can be inserted between the bus and another circuit card to permit easy access to the latter while it is still connected to the bus.

2.17 Hold cycles. A sequence of cycles during which the master is not asked by the arbiter to release the bus for reallocation.

2.18 highway. Those bus lines used to convey data and addresses between devices on the Eurobus.

2.19 indivisible operation. A sequence of bus cycles for which the correct system function can only be guaranteed if no other bus cycles occur within that sequence, e.g. a Read-Modify-Write sequence.

2.20 interbus transfer. A transfer of information between devices that uses two or more buses and one or more bus linkers.

2.21 interrupt. A flag passed to the arbiter by a device in order to initiate a predetermined system-dependent function.

2.22 master. The device that initiates the transfer in question.

2.23 normal address space. An addressing space whose size is determined by the number of lines in the highway and that is addressable as words or bytes.

2.24 protocol. The signalling rules used to convey information or commands between devices connected to the bus.

2.25 pseudo address space. A second, independent addressing space whose size is determined by the number of lines in the highway and that is addressable as words only.

2.26 Read cycle. A bus cycle in which the master obtains a word or byte from the slave.

2.27 reset. The operation whereby each device connected to the bus is put into a predetermined initial condition.

2.28 Retain cycle. A bus cycle at the end of which the master keeps control of the bus in order to complete an indivisible operation.

2.29 settling time. The time taken for a bus line to settle unambiguously into its new logical state from its previous state.

2.30 shelf. The physical structure that supports the backplane and the cards that plug into it.

2.31 skew. On the assumption that two logical transitions are launched simultaneously on two bus lines, the time difference between the receipt of those transitions at a given pair of receivers on a card connected to the bus at the point in question.

2.32 slave. The device that responds to the address placed on the bus by the master for the cycle in question.

2.33 spur. Device connected to the bus at some point between the two ends of the bus.

2.34 state (of a bus line). One of two conditions of a bus line, namely active or quiescent.

2.35 Vector cycle. A bus cycle in which the purpose is to pass an address from a master to a slave and in which no data transfer takes place.

2.36 word. A group of bits whose number corresponds to the maximum data width that can be conveyed over the bus in a single transfer.

2.37 0 V. The signal return path and, as such, the reference for all voltage measurements.

NOTE. 0 V is not a safety earth. Where a safety earth is referred to in this standard, it is specifically identified.

2.38 Write cycle. A bus cycle in which a master writes a word or byte to a slave.

3. Designation of a particular Eurobus

Each member of the Eurobus A family shall be designated using the following format.

Eurobus	address width	/A	qualifying information

The designation entries shall be determined as follows:

(a) address width = 10 or 18 or 26 or 34 . . . etc., as appropriate (see note 1);

(b) A designates the electrical characteristics specified in clause **6**;

(c) qualifying information is additional text stating the version (see **0.8**) and, optionally, text to enable users to identify a particular mechanical implementation.

NOTE 1. The address width is the number of highway bits, i.e. the number of data bits plus two.

NOTE 2. It is recommended that sufficient qualifying information should be provided to enable users and potential users to identify a particular mechanical implementation of Eurobus.

NOTE 3. For example, an 18-bit address implementation of Eurobus A (omitting optional qualifying information) is designated 'Eurobus 18/A commercial'.

4. Compliance

4.1 Full compliance of devices. Devices that are said, without qualification, to comply with this International Standard, shall comply with:

(a) the logical requirements of clause **5**;

(b) the electrical requirements of clause **6**;

(c) the requirements for connector allocation (see **4.4**).

4.2 Logical compliance

4.2.1 Devices said to be logically compliant shall comply with the protocol requirements of clause **5** or with an appropriate subset of those requirements.

NOTE. For example, a slave-only device need not be capable of acting as a bidding device.

4.2.2 In an implementation said to be logically compliant, *either:*

(a) the bus lines shall be used only for the purposes specified in this International Standard; *or*

(b) if one or more of the bus lines is used for purposes not so specified:

(1) normal signals on the bus between devices that operate according to the specified protocols shall not cause any malfunction in the implementation concerned;

(2) signals generated within that implementation shall not cause malfunction in devices, connected to the bus, that operate according to the protocols specified in this International Standard.

4.3 Electrical compliance. Devices said to be electrically compliant shall *either:*

(a) comply with **6.1** and **6.2**, relating to the electrical requirements of devices; *or*

(b) when incorporated into a bus, not prevent that bus from complying with **6.3**, relating to the electrical requirements of buses.

If a device is logically compliant (see **4.2**) and is electrically compliant in accordance with item (b) (i.e. not in accordance with item (a)), all descriptions of the device that refer to this International Standard shall include an explicit statement of the limitations imposed on a system into which the device may be incorporated.

4.4 Mechanical compliance. Connector allocations for data widths of 8, 16, 24 and 32 bits shall be as specified in annex E.

5. Protocols for Eurobus A

5.1 Preliminary

5.1.1 *General.* The set of signals constituting Eurobus A shall be as specified in **5.2**. The protocols, for use of those signals for the allocation of the bus to potential users and thereafter for effecting transfers on the bus, shall be as specified in **5.3** and **5.4**.

NOTE 1. Annex F gives illustrative examples of the operation of the bus protocols in accordance with these requirements.

NOTE 2. Any transfer using the Eurobus protocols generally involves three devices:

(a) the arbiter which:

(1) grants use of the bus to a bidding device that then becomes the bus master; *or*

(2) allows an existing master to continue using the bus;

(b) the master device that initiates the transfer by addressing another device;

(c) the device that, having recognized the address, accepts the transfer and so becomes the bus slave.

5.1.2 *Basic bus cycles.* There shall be three basic types of bus cycle (see table 1) as follows.

(a) Read cycle in which data is read from a slave device by a master device.

(b) Write cycle in which data is written to a slave device by a master device.

(c) Vector cycle in which an address is transferred from the master device to the slave device.

NOTE. The address used by the master device to identify one of a set of locations recognized by a slave device is the only information transferred over the bus in a Vector cycle.

5.1.3 *Bus cycle variants.* The number of possible variants of each basic type shall be two, as follows:

(a) Read and Hold;

(b) Read and Retain;

(c) Write and Hold;

(d) Write and Retain;

(e) Vector and Hold;

(f) Vector and Retain.

NOTE 1. The main purpose of a Hold cycle is to allow devices that have a requirement for numerous bus cycles, e.g. processors, to access the bus repeatedly without having to bid for each individual cycle. Because the use of such a cycle can delay the reallocation of the bus to another device, such cycles are only recommended where there is a high probability that the device concerned will make use of the next bus cycle.

The main purpose of Retain cycles is to enable indivisible operations to be performed, e.g. Read-Modify-Write.

NOTE 2. The differences between these variants and the basic cycles concern the time at which the bus is released by the device for reallocation by the arbiter.

NOTE 3. The bus allocation protocol is designed so that:

(a) the minimum avoidable delay is experienced when allocating an idle bus;

(b) wherever possible, bus allocation is overlapped with bus cycles in order to reduce delays;

(c) a device that requires numerous bus cycles, such as a processor, does not necessarily have to make a fresh request for each cycle;

(d) a device can perform indivisible cycles (this provides the facility necessary, for example, for the construction of a Ready-Modify-Write cycle from a Read cycle followed by a Write cycle);

(e) a device can, as an alternative to performing a Read, Write or Vector cycle, signal an Interrupt to the arbiter which is then responsible for servicing the Interrupt.

5.2 Signalling

5.2.1 *Use of bus lines.* All communication between the arbiter, devices acting as master and devices acting as slave shall be over the Eurobus protocol lines as specified in table 1. The usage of these lines shall be such that any device, or several devices simultaneously, can cause a line to be active. If no device has caused a line to be active, that line shall be quiescent.

5.2.2 *Bit numbering.* The bit number, (N), of the most significant data and address line shall be given by:

$$N = 8p - 1$$

where

p is any positive integer.

The bit number (M) of the most significant byte address line shall be given by:

M = the largest positive integer that is *less than* $\log_2 [(N + 1)/8]$

NOTE. For example, in order to address one of two bytes in a 16-bit data word:

$N = 15$ and

$M < \log_2 16/18$, i.e. $M < 1$

$\therefore M = 0$, i.e. only one byte address line is required.

5.2.3 *Byte mode address selection.* The current bus master shall specify, by coding the Byte Working and Byte Address lines, as given in table 2, the selection of either full-word working, or byte working. If full-word working is selected, the coding shall further specify the selection of pseudo or normal address space. If byte-working is selected, the coding shall further specify which byte is addressed and pseudo address space shall then be unavailable.

5.3 Address recognition protocol

5.3.1 *Data width.* The full addressing capability of the highway, $2(N+3)$ bits provided by $\overline{AdM(1)}$, $\overline{AdM(0)}$, $\overline{H(N)}$ to $\overline{H(0)}$, is available on a bus having only devices all of equal data widths. The use of this capability shall be by the method given in table 3 for a data width of 8 bits.

For buses that include devices having unequal data widths, the method of address allocation shall be as specified in annex G.

NOTE. The functioning of the bus is not dependent on this method.

5.3.2 *Recognition of address modifier bits.* Any slave device operating on a bus that it is sharing with other devices having different data width(s) from itself, shall in all instances recognize one coding and one coding only on the Address Modifier lines.

The specific codes recognized by devices of a particular data width shall, in any system, be allocated according to the following rules.

(a) The codes listed in table 4 under the heading first block (columns 3 and 4) shall first be allocated.

(b) If further codes are required, the codes listed under the headings second block, third block and fourth block (columns 5 to 10) shall be allocated in that order.

Such codes shall be allocated by data width in order from the smallest widths to the largest width, and any code that has then been allocated shall not be available for allocation to another data width as follows.

(1) If the second, third or fourth block is already allocated for recognition by 8-bit devices, 16-bit devices shall be allocated the next higher available block;

(2) If the second, third or fourth block is already allocated for recognition by 8-bit or 16-bit devices, 24-bit devices shall be allocated the next higher available block.

(3) If the second, third or fourth block is already allocated for recognition by 8-bit, 16-bit or 24-bit devices, 32-bit devices shall be allocated the next higher available block.

(c) Any unused blocks shall be available for extending the address range of any of the devices.

5.4 Eurobus A protocol rules

5.4.1 *Preliminary.* The rules for the use of the bus lines specified and named in table 1 shall be as specified in **5.4.2** to **5.4.6**, as follows:

(a) for devices bidding for, and the arbiter granting, use of the bus; **5.4.2**, rules A1 to A12;

(b) for a device acting as master selecting and communicating with a device acting as slave; **5.4.3**, rules M1 to M11;

(c) for a device acting as slave responding to the master; **5.4.4**, rules S1 to S6;

(d) for the arbiter aborting a bus cycle; **5.4.5**, rules C1 to C4;

(e) for a device, being a bus linker, requesting deallocation of the bus; **5.4.6**, rules D1 to D2.

NOTE. Within this point-numbered text the rules and conditions have been arranged and identified, in order to aid understanding, by use of a code of upper case letters, lower case letters and small roman numerals. Each rule is designated by a letter and number (e.g. 'A 9'). Within each rule the alternative ('or') conditions are distinguished by lower case letters while simultaneous ('and') conditions are distinguished by small roman numerals.

5.4.2 *Rules for bus allocation*

Rule A1. When device n requires use of the bus it shall bid for the bus by holding $\overline{Rq(n)}$ active. It shall do this if and only if:

(i) it is not locked (rules A7 and A12); and

(ii) $\overline{BusGr}$ is quiescent; and

(iii) $\overline{Rs}$ is quiescent.

Rule A2. The arbiter shall allocate the bus to one of the bidding devices by also holding the appropriate $\overline{Rq(n)}$ line active. It shall do this if and only if:

(i) $\overline{Rq(n)}$ is already active; and

(ii) the arbiter is not already holding a $\overline{Rq}$ line active; and

(iii) $\overline{BusGr}$ is quiescent; and

(iv) $\overline{BusAcq}$ is quiescent; and

(v) $\overline{It}$ is quiescent; and

(vi) $\overline{Bus\ Deallocate}$ is quiescent; and

(vii) $\overline{CcAbort}$ is quiescent; and

(viii) $\overline{Rs}$ is quiescent.

Table 1. Eurobus A protocol lines

Signal name (abbreviations)	Number of lines	Requirements
Highway lines		
Data/Address ($\overline{H(0)}$ to $\overline{H(N)}$)	$N+1$	Time division multiplexed bi-directional data and address lines. $\overline{H(0)}$ shall be associated with the least significant bit. The number of the most significant bit, $\overline{H(N)}$, shall be determined as specified in **5.2.2**
Address Modifier Bits (0), (1) ($\overline{AdM(0)}$, $\overline{AdM(1)}$)	2	Address Modifier lines. These are available and shall be used when it is required to increase the address range beyond that definable by $\overline{H(0)}$ to $\overline{H(N)}$, and also for selection between devices of different data width that share the same bus (see **5.3**)
Byte mode/address space selection lines		
Byte Working ($\overline{BytWk}$)	1	These lines shall be used to qualify the address on the highway in terms of the byte mode/address space selection coding (see **5.2.3**). The number of the most significant Byte Address bit, (M), shall be determined as specified in **5.2.2** If $N=7$ the Byte Working line shall remain quiescent
Byte Address Bits (0) to ($\overline{BytAd(0)}$ to $\overline{BytAd(M)}$)	$M+1$	
Bus allocation protocol lines		
Request(n) ($\overline{Rq(n)}$)	1 per potential bus master	One Request line shall be star-connected to the arbiter from each device that requires to be able to bid for bus allocation. Request(n) line shall be activated by device(n) to signal its bid for bus allocation. The arbiter shall activate the Request line, in conjunction with the Bus Granted line, to allocate use of the bus to device(n)
Bus Granted ($\overline{BusGr}$)	1	This line shall be activated by the arbiter at the same time as it activates the Request line of a particular device in order to affect a new bus allocation. The device in question shall then be the current bus master for the next transfer This line shall also be activated by the arbiter without activating a Request line to deallocate the current bus master (see **5.4.5**)
Bus Acquired ($\overline{BusAcq}$)	1	This line shall be activated by the current bus master to signify that the bus grant has been accepted. The line shall be held active by a slave if that slave is about to activate the Bus Deallocate line
Interrupt ($\overline{It}$)	1	This line shall be activated by the current bus master to indicate to the arbiter that its request is for an Interrupt cycle
Transfer control handshake lines		
Cycle Begin ($\overline{CcBn}$)	1	This line shall be activated by the current master device to indicate that: (a) there is a stable address on the highway and byte mode/address space selection lines; (b) the Cycle Finish line (which at this point of the transfer indicates Read/Vector cycle or Write cycle) is stable
Cycle Response ($\overline{CcRes}$)	1	This line shall be activated by a device to indicate that it has recognized the address on the highway and has become the current bus slave. On detecting that this line has become active the master shall remove the address from the highway
Cycle Finished ($\overline{CcFin}$)	1	This bi-directional line shall be activated: (a) by the current master: (1) in a Read cycle to indicate that the address has been removed; (2) in a Write cycle to identify the cycle as a Write cycle and, when released, to indicate that Write data has been removed; (b) by the current slave during the transmission of Read data

Table 1 (*concluded*)

Signal name (abbreviations)	Number of lines	Requirements
Cycle Abort ($\overline{\text{CcAbort}}$)	1	This line shall be activated by the arbiter to terminate an invalid bus cycle
Reset ($\overline{\text{Rs}}$)	1	Reset line. This line shall be connected to all devices on the bus. When the line is activated by any device that is permitted to do so, it shall cause a general bus reset operation
Bus Deallocation ($\overline{\text{BusDeallocate}}$)	1	This line shall be activated by a bus link device in order to indicate to an arbiter *either:* (a) that the bus link requires a deadly embrace to be broken; *or* (b) that the arbiter is required to ask the bus master to release the bus for reallocation. The state of the Cycle Response line shall be used to specify which of the indications is valid at the time the Bus Deallocate line is activated

Table 2. Coding of byte mode/address space selection lines

$\overline{\text{BytWk}}$	$\overline{\text{BytAd}(M)}$	$\overline{\text{BytAd}(M-1)}$	...	$\overline{\text{BytAd}(0)}$	Selection
Q	Q	X	X	X	The whole word (N+1 bits) in normal address space
Q	Acv	X	X	X	The whole word (N+1 bits) in pseudo address space
Acv	Q	Q	Q	Q	Least significant byte (byte 0)
Acv	Q	Q	Q	Acv	Byte 1
.	.	.	.	.	in normal address space
.	.	.	.	.	
Acv	Acv	Acv	Acv	Q	Byte (2(M+1) − 2)
Acv	Acv	Acv	Acv	Acv	Byte (2(M+1) − 1)

NOTE 1. In table 2 and later tables the abbreviation 'Acv' signifies the active state, and 'Q' signifies the quiescent state. Symbol 'X' signifies that either state may exist.

NOTE 2. For values of M and N see 5.2.2.

Table 3. Address recognition protocol (N = 7)

$\overline{\text{AdMd}(1)}$	$\overline{\text{AdMd}(0)}$	Addresses	Data
Acv	Acv	Fourth block for 8-bit devices	$\overline{\text{H}(7)}$ … $\overline{\text{H}(0)}$
Acv	Q	Third block for 8-bit devices	$\overline{\text{H}(7)}$ … $\overline{\text{H}(0)}$
Q	Acv	Second block for 8-bit devices	$\overline{\text{H}(7)}$ … $\overline{\text{H}(0)}$
Q	Q	First block for 8-bit devices	$\overline{\text{H}(7)}$ … $\overline{\text{H}(0)}$

NOTE. See note 1 to table 2.

Table 4. Address modifier codes to be recognized by slave devices of different widths sharing the same bus

Bus data width (bits)	Device data width (bits)	Address modifier code for recognition of address block							
		First block		Second block		Third block		Fourth block	
		AdM(1)	AdM(0)	AdM(1)	AdM(0)	AdM(1)	AdM(0)	AdM(1)	AdM(0)
16	8	Q	Q	Acv	Q	Acv	Acv	–	–
	16	Q	Acv	Acv	Q	Acv	Acv	Q	Q
24	8	Q	Q	Q	Acv	Acv	Acv	–	–
	16	Q	Acv	Acv	Acv	Q	Q	–	–
	24	Acv	Q	Acv	Acv	Q	Q	Q	Acv
32	8	Q	Q	Q	Acv	Acv	Q	–	–
	16	Q	Acv	Acv	Acv	Q	Q	–	–
	24	Acv	Q	Q	Q	Q	Acv	–	–
	32	Acv	Acv	Q	Q	Q	Acv	Acv	Q

NOTE. See note 1 to table 2.

Rule A3. The arbiter shall hold $\overline{\text{BusGr}}$ active if and only if:

(ai) the arbiter has made an allocation under rule A2; and

(aii) $\overline{\text{Rs}}$ is quiescent;

NOTE. Bus allocation is complete.

(bi) or, the arbiter is not holding a $\overline{\text{Rq}}$ line active; and

(bii) $\overline{\text{CcBn}}$ is active; and

(biii) $\overline{\text{CcRes}}$ is quiescent; and

(biv) $\overline{\text{Bus Deallocate}}$ is active; and

(bv) $\overline{\text{CcRes}}$ was quiescent when $\overline{\text{Bus Deallocate}}$ became active; and

(bvi) $\overline{\text{Rs}}$ is quiescent.

NOTE. The arbiter deallocates the cycle that the slave has refused.

Rule A4. A bidding device shall release its $\overline{\text{Rq}}$ line if and only if:

(a) $\overline{\text{BusGr}}$ is active;

(b) or, $\overline{\text{Rs}}$ is active.

Rule A5. A device shall hold $\overline{\text{BusAcq}}$ active, if and only if:

(ai) $\overline{\text{BusGr}}$ is active; and

(aii) its $\overline{\text{Rq}}$ line is active (and it is not itself holding this line active); and

(aiii) $\overline{\text{CcBn}}$ is quiescent; and

(aiv) $\overline{\text{CcRes}}$ is quiescent; and

(av) $\overline{\text{CcFin}}$ is quiescent; and

(avi) the device is not holding the $\overline{\text{It}}$ line active; and

(avii) $\overline{\text{CcAbort}}$ is quiescent; and

(aviii) $\overline{\text{Rs}}$ is quiescent;

NOTE. The master acquires the bus.

(bi) or, $\overline{\text{CcRes}}$ is quiescent; and

(bii) $\overline{\text{CcBn}}$ is active; and

(biii) the signals on the highway and byte mode lines correspond to an address recognized by the device; and

(biv) the device is not holding $\overline{\text{CcFin}}$ active; and

(bv) the device is about to hold $\overline{\text{Bus Deallocate}}$ active according to rule D1a or (after application of rule S1) rule D1b; and

(bvi) $\overline{\text{CcAbort}}$ is quiescent; and

(bvii) $\overline{\text{Rs}}$ is quiescent.

NOTE. The slave device holds $\overline{\text{BusAcq}}$ active before $\overline{\text{CcRes}}$, if it intends to ask the arbiter to remove allocation from the current master, or before $\overline{\text{Bus Deallocate}}$, if it intends to refuse the cycle.

Rule A6. Alternatively to rule A5, a device shall hold $\overline{\text{It}}$ active, and thus initiate an Interrupt cycle, if and only if:

(i) $\overline{\text{BusGr}}$ is active; and

(ii) its $\overline{\text{Rq}}$ line is active (and it is not itself holding this line active); and

(iii) it is not holding $\overline{\text{BusAcq}}$ active; and

(iv) $\overline{\text{CcAbort}}$ is quiescent; and

(v) $\overline{\text{Rs}}$ is quiescent.

NOTE. The master generates an interrupt to the arbiter.

Rule A7. A device shall become locked if and only if:

(a) it holds $\overline{\text{BusAcq}}$ active according to rule A5a;

(b) or, it holds $\overline{\text{It}}$ active according to rule A6.

Rule A8. The arbiter shall release $\overline{\text{BusGr}}$ if and only if:

(ai) $\overline{\text{BusAcq}}$ is active; and

(aii) $\overline{\text{Bus Deallocate}}$ is quiescent:

NOTE. The arbiter acknowledges bus acquisition.

(bi) or, $\overline{\text{It}}$ is active; and

(bii) $\overline{\text{Bus Deallocate}}$ is quiescent;

NOTE. The arbiter acknowledges an interrupt.

(ci) or, it is not holding a $\overline{\text{Rq}}$ line active; and

(cii) $\overline{\text{BusAq}}$ is quiescent;

NOTE. The arbiter completes the bus deallocation handshake.

(di) or, it is holding $\overline{\text{CcAbort}}$ active; and

(dii) $\overline{\text{Bus Deallocate}}$ is quiescent;

(e) or, $\overline{\text{Rs}}$ is active.

Rule A9. The arbiter shall release any $\overline{\text{Rq}}$ line it is currently holding active if and only if:

(ai) $\overline{\text{BusGr}}$ is quiescent; and

(aii) $\overline{\text{BusAcq}}$ is quiescent; and

(aiii) $\overline{\text{It}}$ is quiescent; and

(aiv) the arbiter is not about to hold $\overline{\text{BusGr}}$ under rule A3;

NOTE. The last allocated master has finished using the bus.

(bi) or, $\overline{\text{BusGr}}$ is quiescent; and

(bii) a device holds its $\overline{\text{Rq}}$ line active; and

(biii) the arbiter is not about to hold $\overline{\text{BusGr}}$ under rule A3;

NOTE. A device requests allocation while the allocated master is still using the bus.

(c) or, $\overline{\text{Bus Deallocate}}$ is active;

NOTE. The arbiter asks the allocated master to release the bus for reallocation.

(d) or, it is holding $\overline{\text{CcAbort}}$ active;

(e) or, Reset is active.

Rule A10. A device that is holding $\overline{\text{BusAcq}}$ active shall release it if and only if:

(ai) the current bus cycle is established (rule M5); and

(aii) the device's $\overline{\text{Rq}}$ line is active; and

(aiii) it does not wish to hold the bus for further cycles;

NOTE. The master relinquishes the bus unilaterally.

(bi) or, the current bus cycle is established (rule M5); and

(bii) the device's $\overline{\text{Rq}}$ line is quiescent; and

(biii) it does not require to retain the bus for an indivisible operation;

NOTE. The arbiter asks the master to release the bus.

(ci) or, the current cycle is refused (rule M6); and

(cii) it is not holding $\overline{\text{CcBn}}$ active; and

(ciii) it is not holding $\overline{\text{CcFin}}$ active;

NOTE. The master releases the bus in a Deallocate cycle.

(di) or, it is holding $\overline{\text{Bus Deallocate}}$ active; and

(dii) it is slave for the current cycle;

NOTE. The onus is on the slave to guarantee that $\overline{\text{BusAcq}}$ is active when $\overline{\text{Bus Deallocate}}$ becomes active (see rules A10a, A10b, S1 and D1b).

(ei) or, it is holding $\overline{\text{Bus Deallocate}}$ active; and

(eii) it has refused to become slave for the current cycle; and

(eiii) $\overline{\text{BusGr}}$ is active;

(f) or, $\overline{\text{CcAbort}}$ is active;

(g) or, $\overline{\text{Rs}}$ is active.

Rule A11. A device that is holding $\overline{\text{It}}$ active shall release it if and only if:

(a) $\overline{\text{BusGr}}$ is quiescent;

NOTE. The handshake completes the Interrupt cycle.

(b) or, $\overline{\text{CcAbort}}$ is active;

(c) or, $\overline{\text{Rs}}$ is active.

Rule A12. A device shall cease to be locked if and only if:

(a) it completes a cycle as master (rules M9b, M9c and M11a) when $\overline{\text{BusAcq}}$ is quiescent;

NOTE. This is the end of a cycle in which the master released the bus for reallocation.

(b) or, it releases $\overline{\text{BusAcq}}$ (rules A10a and A10b) after completion of one cycle as master (rules M9b, M9c and M11a) and before commencement of another (rule M1);

NOTE. The master releases the bus for reallocation following a Hold cycle or a Retain cycle.

(c) or, it releases $\overline{\text{It}}$ under rule A11a;

NOTE. This is the end of Interrupt cycle.

(di) or, it completes a cycle as slave in a Write cycle or a Vector cycle (rules S4a and S4c); and

(dii) it is not holding $\overline{\text{It}}$ active;

NOTE. This condition releases the lock following the abortion of a master cycle.

(e) or, it releases $\overline{\text{BusAcq}}$ under rule A10c;

NOTE. This release is after deallocation.

(f) or, $\overline{\text{Rs}}$ is active.

5.4.3 *Rules for devices acting as master*

Rule M1. The master shall commence a cycle if and only if:

(i) it is holding $\overline{\text{BusAcq}}$ active; and

(ii) $\overline{\text{CcBn}}$ is quiescent; and

(iii) $\overline{\text{CcRes}}$ is quiescent; and

(iv) $\overline{\text{CcFin}}$ is quiescent; and

(v) $\overline{\text{CcAbort}}$ is quiescent; and

(vi) $\overline{\text{Rs}}$ is quiescent.

NOTE. The cycle starts after bus free.

Rule M2. The master shall hold the highway and byte mode lines to form an address if and only if:

(i) the cycle is commenced (rule M1); and

(ii) $\overline{\text{CcAbort}}$ is quiescent; and

(iii) $\overline{\text{Rs}}$ is quiescent.

Rule M3. The master shall hold $\overline{\text{CcFin}}$ active if and only if:

(ai) the cycle is commenced (rule M1); and

(aii) $\overline{\text{CcAbort}}$ is quiescent; and

(aiii) it wishes to perform a Write cycle; and

(aiv) $\overline{\text{Rs}}$ is quiescent;

NOTE. This is the beginning of a Write cycle.

(bi) or, the cycle is established (rule M5); and

(bii) it is not holding an address; and

(biii) it wishes to perform a Read cycle; and

(biv) $\overline{\text{CcAbort}}$ is quiescent; and

(bv) $\overline{\text{Rs}}$ is quiescent.

NOTE. The address has been removed from the bus in a Read cycle.

Rule M4. The master shall hold $\overline{\text{CcBn}}$ active if and only if:

(ai) it is holding $\overline{\text{BusAcq}}$ active; and

(aii) it is holding an address; and

(aiii) it is holding $\overline{\text{CcFin}}$ active; and

(aiv) it wishes to perform a Write cycle; and

(av) $\overline{\text{CcAbort}}$ is quiescent; and

(avi) $\overline{\text{Rs}}$ is quiescent;

NOTE. The Write address is on the bus.

(bi) or, it is holding $\overline{\text{BusAcq}}$ active; and

(bii) it is holding an address; and

(biii) it wishes to perform a Read cycle; and

(biv) $\overline{\text{CcAbort}}$ is quiescent; and

(bv) $\overline{\text{Rs}}$ is quiescent;

NOTE. The Read address is on the bus.

(ci) or, it is holding $\overline{\text{BusAcq}}$ active; and
(cii) it is holding an address; and
(ciii) it wishes to perform a Vector cycle; and
(civ) $\overline{\text{CcAbort}}$ is quiescent; and
(cv) $\overline{\text{Rs}}$ is quiescent.

NOTE. The Vector address is on the bus.

Rule M5. The master shall regard the cycle commenced under rule M2 as established if and only if $\overline{\text{CcRes}}$ is active.

NOTE. The slave has accepted the address.

Rule M6. The master shall regard the cycle commenced under rule M1 as refused if and only if:

(i) the master's $\overline{\text{Rq}}$ line is quiescent; and
(ii) $\overline{\text{BusGr}}$ is active.

NOTE. This is a bus deallocation.

Rule M7. The master shall release an address if and only if:

(a) the cycle is established (rule M5);
(b) or, the cycle is refused (rule M6);
(c) or, $\overline{\text{CcAbort}}$ is active;
(d) or, $\overline{\text{Rs}}$ is active.

Rule M8. The master shall hold the highway lines to form Write data if and only if:

(i) the cycle is established (rule M5); and
(ii) it wishes to perform a Write cycle; and
(iii) $\overline{\text{CcAbort}}$ is quiescent; and
(iv) $\overline{\text{Rs}}$ is quiescent.

Rule M9. The master shall release $\overline{\text{CcBn}}$ if and only if:

(ai) it is not holding an address; and
(aii) it is holding Write data on the highway;

NOTE. This event occurs in a Write cycle.

(bi) or, it is not holding $\overline{\text{CcFin}}$ active; and
(bii) $\overline{\text{CcFin}}$ is active; and
(biii) $\overline{\text{CcRes}}$ is quiescent; and
(biv) it has accepted the signals on the highway as Read data; and
(bv) $\overline{\text{CcAbort}}$ is quiescent;

NOTE. This event occurs in a Read cycle.

(ci) or, it is not holding an address; and
(cii) it wishes to perform a Vector cycle; and
(ciii) $\overline{\text{CcRes}}$ is active; and
(civ) $\overline{\text{CcAbort}}$ is quiescent;

NOTE. This event occurs in a Vector cycle.

(di) or, the current cycle is refused (rule M6); and
(dii) it is not holding $\overline{\text{CcFin}}$ active; and
(diii) it is not holding an address;

NOTE. This event occurs in bus deallocation.

(ei) or, $\overline{\text{CcAbort}}$ is active; and
(eii) it is not holding $\overline{\text{CcFin}}$ active; and
(eiii) it is not holding an address;
(fi) or, $\overline{\text{CcAbort}}$ is active; and
(fii) it is holding Write data;
(gi) or, $\overline{\text{Rs}}$ is active; and
(gii) it is not holding $\overline{\text{CcFin}}$; and
(giii) it is not holding an address;
(hi) or, $\overline{\text{Rs}}$ is active; and
(hii) it is holding Write data.

Rule M10. The master shall release Write data from the highway if and only if:

(ai) $\overline{\text{CcRes}}$ is quiescent; and
(aii) $\overline{\text{CcBn}}$ is quiescent;

NOTE. The slave accepts Write data.

(b) or, $\overline{\text{CcAbort}}$ is active;
(c) or, $\overline{\text{Rs}}$ is active.

Rule M11. The master shall release $\overline{\text{CcFin}}$ if and only if;

(ai) it is not holding $\overline{\text{CcBn}}$ active; and
(aii) it is not holding Write data; and
(aiii) the cycle is established (rule M5); and
(aiv) $\overline{\text{CcAbort}}$ is quiescent;

NOTE. This is the final handshake in a Write cycle.

(bi) or, it is holding $\overline{\text{CcBn}}$ active; and
(bii) $\overline{\text{CcRes}}$ is quiescent; and
(biii) the cycle is established (rule M5);

NOTE. This event occurs in the middle of a Read cycle.

(ci) or, the current cycle is refused (rule M6); and
(cii) it is not holding an address;

NOTE. This event occurs while a Write cycle is being deallocated.

(di) or, $\overline{\text{CcAbort}}$ is active; and
(dii) it is not holding an address;
(ei) or, $\overline{\text{CcAbort}}$ is active; and
(eii) it is not holding Write data;
(fi) or, $\overline{\text{Rs}}$ is active; and
(fii) it is not holding an address;
(gi) or, $\overline{\text{Rs}}$ is active; and
(gii) it is not holding Write data.

5.4.4 *Rules for devices acting as slave*

Rule S1. A device shall hold $\overline{\text{CcRes}}$ active if and only if:

(i) $\overline{\text{CcBn}}$ is active; and
(ii) $\overline{\text{BusGr}}$ is quiescent; and
(iii) the signals on the highway and byte mode lines correspond to an address recognized by the device; and
(iv) the device is not refusing to become slave for the current cycle (rule D1a); and
(v) the device is not holding $\overline{\text{CcFin}}$ active; and
(vi) $\overline{\text{CcAbort}}$ is quiescent; and
(vii) $\overline{\text{Rs}}$ is quiescent.

Rule S2. The slave shall hold Read data on the highway if and only if:

(i) $\overline{\text{CcFin}}$ is active; and
(ii) $\overline{\text{CcFin}}$ was not active when it held $\overline{\text{CcRes}}$ active; and
(iii) $\overline{\text{CcAbort}}$ is quiescent; and
(iv) $\overline{\text{Rs}}$ is quiescent.

NOTE. This event occurs in a Read cycle.

Rule S3. The slave shall hold $\overline{\text{CcFin}}$ active if and only if:

(i) $\overline{\text{CcFin}}$ is active; and
(ii) $\overline{\text{CcFin}}$ was not active when it held $\overline{\text{CcRes}}$ active; and
(iii) $\overline{\text{CcAbort}}$ is quiescent; and
(iv) $\overline{\text{Rs}}$ is quiescent.

NOTE. This event occurs in a Read cycle.

Rule S4. The slave shall release $\overline{\text{CcRes}}$ if and only if:

(ai) it is not holding $\overline{\text{CcFin}}$ active; and
(aii) $\overline{\text{CcFin}}$ is active; and
(aiii) $\overline{\text{CcFin}}$ was active when it held $\overline{\text{CcRes}}$ active; and

(aiv) CcBn is quiescent; and
(av) it is not holding BusAcq active; and
(avi) it has accepted the signals on the highway as Write data; and
(avii) CcAbort is quiescent;

NOTE. The slave has accepted the data in a Write cycle.

(bi) or, it is holding Read data on the highway; and
(bii) it is holding CcFin active; and
(biii) it is not holding BusAcq active;

NOTE. The slave signals the presence of Read data on the bus.

(ci) or, CcFin is quiescent; and
(cii) CcBn is quiescent; and
(ciii) CcAbort is quiescent; and
(civ) it is not holding BusAcq active;

NOTE. This is the final handshake in a Vector cycle.

(d) or, CcAbort is active;
(e) or, Rs is active.

Rule S5. The slave shall release Read data from the highway if and only if:

(a) CcBn is quiescent;

NOTE. The master has accepted the Read data.

(bi) or, CcAbort is active; and
(bii) CcRes is quiescent;
(ci) or, Rs is active; and
(cii) CcRes is quiescent.

Rule S6. The slave shall release CcFin if and only if:

(ai) CcBn is quiescent; and
(aii) it is not holding Read data; and
(aiii) CcAbort is quiescent;

NOTE. This is the final handshake in a Read cycle.

(bi) or, it is not holding Read data; and
(bii) CcAbort is active;
(ci) or, it is not holding Read data; and
(cii) Rs is active.

5.4.5 *Rules for abortion of cycles*

Rule C1. The arbiter shall hold CcAbort active if and only if it detects that:

(ai) there has been an infringement of any of the rules A5, A6, A10 or A11 (see **5.4.2**) and
(aii) CcBn is quiescent; and
(aiii) CcRes is quiescent; and
(aiv) CcFin is quiescent; and
(av) Rs is quiescent;
(bi) or, there has been an infringement of any of the rules of **5.4.3** or **5.4.4**; and
(bii) It is quiescent; and
(biii) Rs is quiescent;
(ci) or, there has been an infringement of any of the rules A5, A6, A10 or A11; and
(cii) there has been an infringement of any of the rules of **5.4.3** or **5.4.4**; and
(ciii) Rs is quiescent.

Rule C2. The arbiter shall release CcAbort if and only if:

(ai) it is not holding any Rq line active; and
(aii) it is not holding BusGr active; and
(aiii) BusAcq is quiescent; and
(aiv) It is quiescent; and
(av) CcBn is quiescent; and
(avi) CcRes is quiescent; and
(avii) CcFin is quiescent; and
(aviii) Bus Deallocate is quiescent;
(b) or, Rs is active.

Rule C3. Failure to comply with a rule within the time specified by the system designer for a particular implementation of an arbiter shall be an infringement of that rule.

NOTE. The specified time may be included in the qualifying information in the bus designation (see clause 3).

Rule C4. A device shall regard a Read, Write or Vector cycle or an Interrupt as valid if and only if it has completed its action under rule A11a, M9b, M9c, M11a, S4a, S4c, or S6a, as appropriate.

NOTE. Rule C1 does not imply that any given Eurobus arbiter need be capable of detecting all possible infringements of the rules.

5.4.6 *Rules for bus deallocation*

Rule D1. A device shall hold Bus Deallocate active if and only if:

(ai) BusGr is quiescent; and
(aii) it is not holding CcRes active; and
(aiii) it is holding BusAcq active under rule A5b; and
(aiv) the device is refusing to become (and is not already) slave for the current cycle; and
(av) CcAbort is quiescent; and
(avi) Rs is quiescent;

NOTE. The device asks the arbiter to deallocate the bus.

(bi) or, it is holding BusAcq active under rule A5b; and
(bii) CcRes is active; and
(biii) CcAbort is quiescent; and
(biv) Rs is quiescent.

NOTE. The device asks the arbiter to remove allocation from the master.

Rule D2. A device shall release Bus Deallocate if and only if BusAcq is quiescent.

5.4.7 *Rules for data transfer.* Data shall be transferred over the least significant *w* highway lines where *w* is the notional width of the data in bits. The more significant highway lines shall be quiescent.

For example, on a 16-bit bus (*N* in table 1 is 15), 8-bit bytes shall be transferred over the least significant 8 bus lines, H(0) to H(7) inclusive, and the remaining lines, H(8) to H(15), shall be quiescent.

6. Electrical and timing requirements

6.1 Electrical requirements

6.1.1 *General.* Eurobus A shall operate over an electrically unbalanced voltage interface using a positive logic convention. All signals shall be referred to the bus 0 V connections which therefore need to meet stringent noise limits.

NOTE 1. The requirements for line and spur characteristics and connector pin allocation ensure good 0 V integrity.

NOTE 2. When devices are connected to the bus, the shunt impedance of each spur alters both line impedance and propagation velocity. Assuming that the shunt impedance is mainly capacitive, the line impedance and propagation velocity fall.

6.1.2 *Power supplies to the backplane*

6.1.2.1 For the commercial version, the power supply shall be 5 ± 0,15 V.