inmos

User Manual

IMS B014

User Manual and Reference Guide

Contents

IMPORTANT— READ THIS BEFORE PROCEEDING

The IMS B014 can be damaged by improper handling and incorrect installation in certain computers, particularly SUN workstations. You must read section B before installing the board.

Look for an errata sheet in the package, if present it may contain important information and must therefore be read.

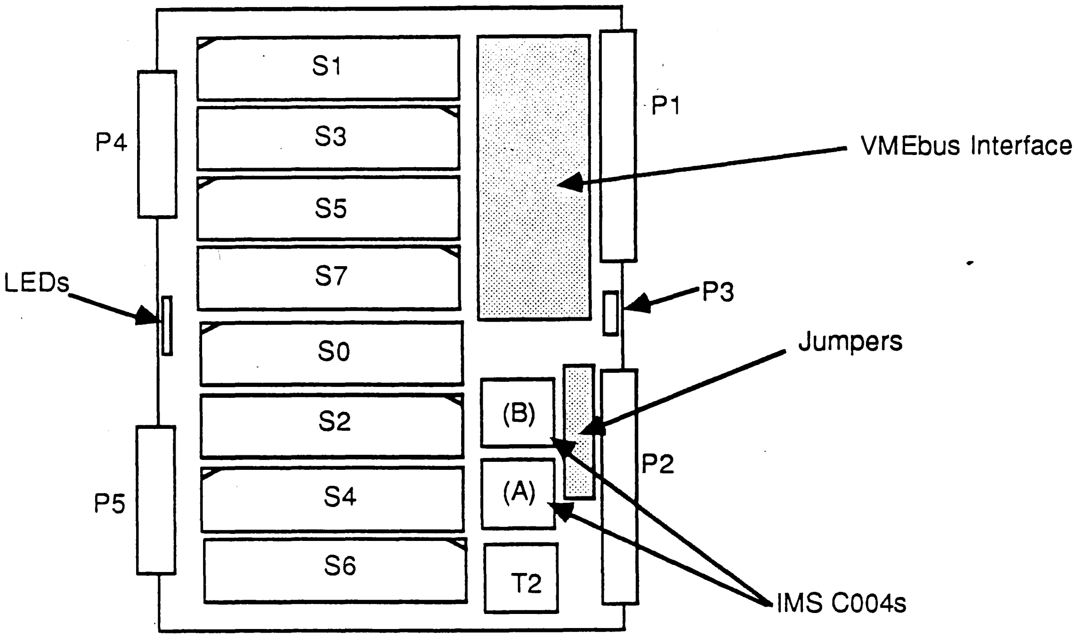


Figure 1: The IMS B014 board

2 The IMS B014

The VMEbus is a popular parallel bus originally developed for M68000 processors. TRAMs are INMOS-developed processor modules, designed to a published specification [3]. The IMS B014 brings these two standards together by allowing TRAMs to be integrated into VMEbus systems.

The IMS B014 is a 6U high standard-depth VMEbus board (see figure 1). It provides space for up to eight TRAMs (TRAnsputer Modules). The VMEbus interface is a slave interrupter which gives one INMOS serial link and a reset port. The TRAMs may be connected together in any network using the IMS B014's two IMS C004 link switches.

The IMS B014 can take a variety of TRAMs to give processing power of 7–80 MIPS and 32K–8Mbytes RAM per processor (using current TRAMs).

The VMEbus interface is designed for low cost and easy programming and allows VMEbus master devices to communicate with one of the TRAMs on the IMS B014 at a maximum data rate of about 300Kbytes/s.

3 Introduction to the VMEbus

The VMEbus, originally proposed by Motorola, Mostek and Philips, is now an IEC and IEEE standard. It provides a parallel 8, 16 or 32-bit bus with multi-master capabilities. Mechanical constraints are basically those of IEC 297 (eurocard). The VMEbus is now very popular and hundreds of boards are available from many vendors. The VMEbus industry is co-ordinated by VITA (VMEbus Industry and Trade Association) who publish useful information and organise exhibitions, conferences and committes on standards. [1]

[1] The current VMEbus standard recognised and the one to which the IMS B014 adheres is *The VMEbus specification REV C.1* [6]. IEC 821 is almost identical to REV C.1, however the new IEEE 1014 spec uses new text to describe the VMEbus and it is not yet clear whether any of the differences between the IEEE spec and REV C.1 are relavent to the IMS B014.

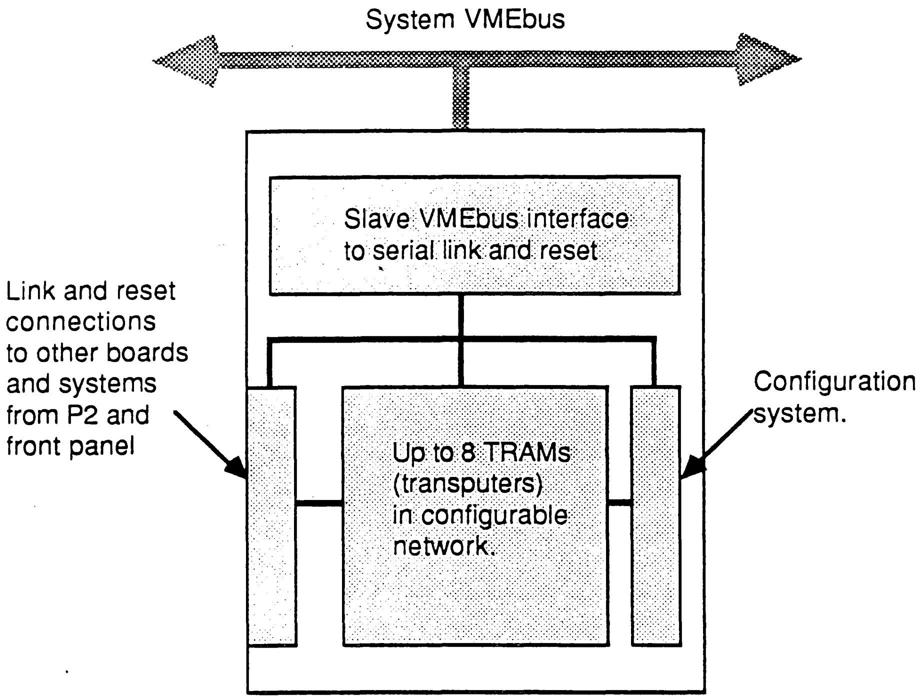

System VMEbus
Slave VMEbus interface to serial link and reset
Link and reset connections to other boards and systems from P2 and front panel
Up to 8 TRAMs (transputers) in configurable network.
Configuration system.

Figure 2: Block Diagram

3.1 VMEbus Terminology

The VMEbus Specification includes many special terms and abbreviations. These make descriptions of VMEbus boards and systems very precise. Unfortunately for the casual user, some of these terms must be defined in order that the description of the IMS B014's VMEbus interface can be precise—

1 There are different kinds of VMEbus data transfer cycle. The only cycles supported by the IMS B014 are called "A16:D08(O)". This means that a 16-bit address is decoded (sometimes called *short I/O addressing*), and that 8-bit data is transfered on the lower part of the basic 16-bit databus. The IMS B014 neither monitors nor drives the higher address lines and the higher databus bits.

2 The VMEbus supports "indivisable" read-modify-write cycles, these are called RMW cycles. The IMS B014 does not support RMW cycles.

3 Devices which can interrupt the VMEbus are called "interrupters". The IMS B014 is such a device. There are seven VMEbus interrupt lines. Since the IMS B014 can use any of the seven lines, it is called an "INT(1–7)" interrupter.

Exhaustive definitions of these and the other VMEbus terms are to be found in [6].

3.2 VMEbus Certification

INMOS has comitted to having the IMS B014 and all future VMEbus boards validated by VMElaboratories Inc. Denton, TX. The validation process ensures that a board has no design errors in the VMEbus interface section and will work in any VMEbus system.

At the time of writing, the IMS B014 has not been submitted for validation, contact INMOS for up-to-date information.

4 Transputer Background

The family of *transputers* available from INMOS includes 16 and 32-bit processors. These are all capable of communicating with each other and can be programmed when connected together to execute tasks in parallel. Full information on transputers can be found in *The Transputer Reference Manual* and in other INMOS publications. The IMS B014 uses one transputer on-board for network configuration and can, of course, support transputer arrays in the form of TRAMs.

4.1 INMOS Links

Transputers use special serial communication connections to talk to each other. These are called *INMOS links*, or *links*. Links use two wires to send data in two directions between two transputers (or other chips) at up to 20 Mbits/s. All communication between TRAMs on the IMS B014, and between the TRAMs and the VMEbus interface, is via links. For full information on links see the [2] and [4].

4.2 TRAMs

TRAMs are small circuit modules which communicate via INMOS serial links. A full TRAM background is to be found in [3]. The smallest TRAM (called "size 1") is about $3\frac{1}{2}$" by 1" and has eight IC-type pins along each of the short ends. These sixteen pins carry four INMOS links, reset signals, clock and power supplies. TRAMs are plugged into a motherboard (or baseboard) just like socketed ICs.

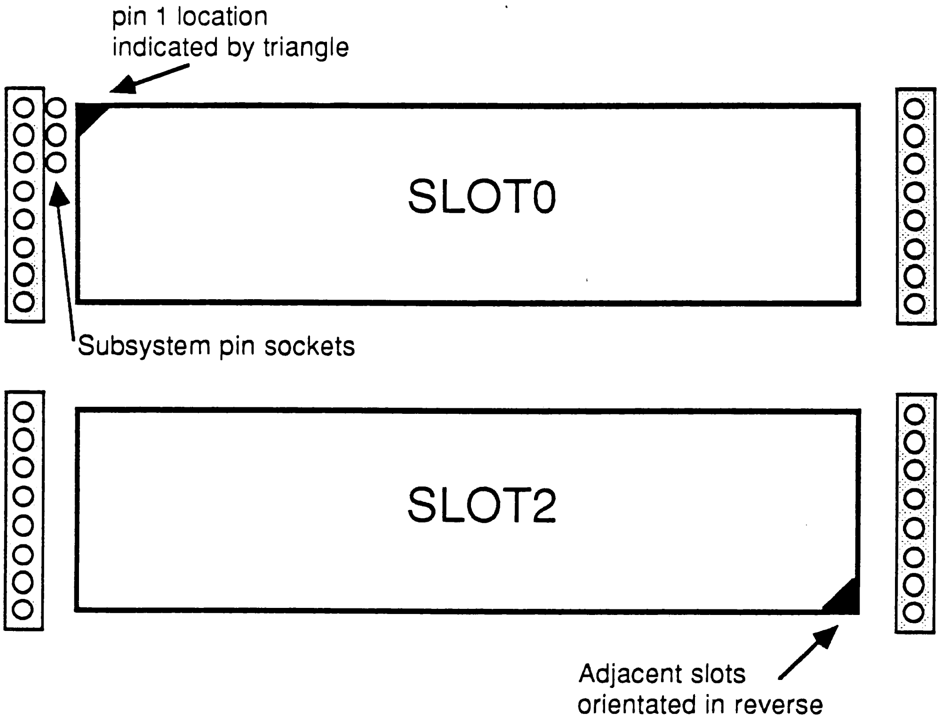


Figure 3: TRAM Slots

Larger TRAMs are simply a multiple of the size 1 footprint. Thus, a "size 2" TRAM occupies two of the sockets into which a size 1 TRAM will plug. In order to avoid confusion, discussions about motherboards (such as the IMS B014) always refer to "slots" (see figure 3). A slot is one position into which a size 1 TRAM may be plugged. So, for instance, the IMS B014 which has eight slots, may have eight size 1 TRAMS or four size 2 or two size 4 or one size 8 or even four size 1 and one size 4. There is no restriction on mixing the different sizes, as long as the total complement will fit.

In addition, some TRAMs have pins which are sockets at the top. This means that some TRAM combinations can be "double-stacked". For instance, if you had two size 4 (single transputer) TRAMs fitted to an IMS B014, only two slots would actually be used for the link signals. The remaining six slots would be covered by the TRAMs but are not used *electrically*. In some cases you can plug further TRAMS into the *already fitted* size 4 TRAMS and so make use of the unused slots. This double stacking is not always possible and usually makes the board too high to fit into a single VMEbus card slot.

5 VMEbus Interface

The VMEbus is a parallel bus which supports multiple masters and 8, 16 and 32-bit transfers. Full information on the VMEbus is contained in [6]. The IMS B014 implements a VMEbus SLAVE interface which allows one INMOS link and a subsystem port (for reset) to be driven from the VMEbus. The interface is constructed and driven ıg similar lines to other simple VMEbus slave interfaces, for instance an RS-232 interface board.

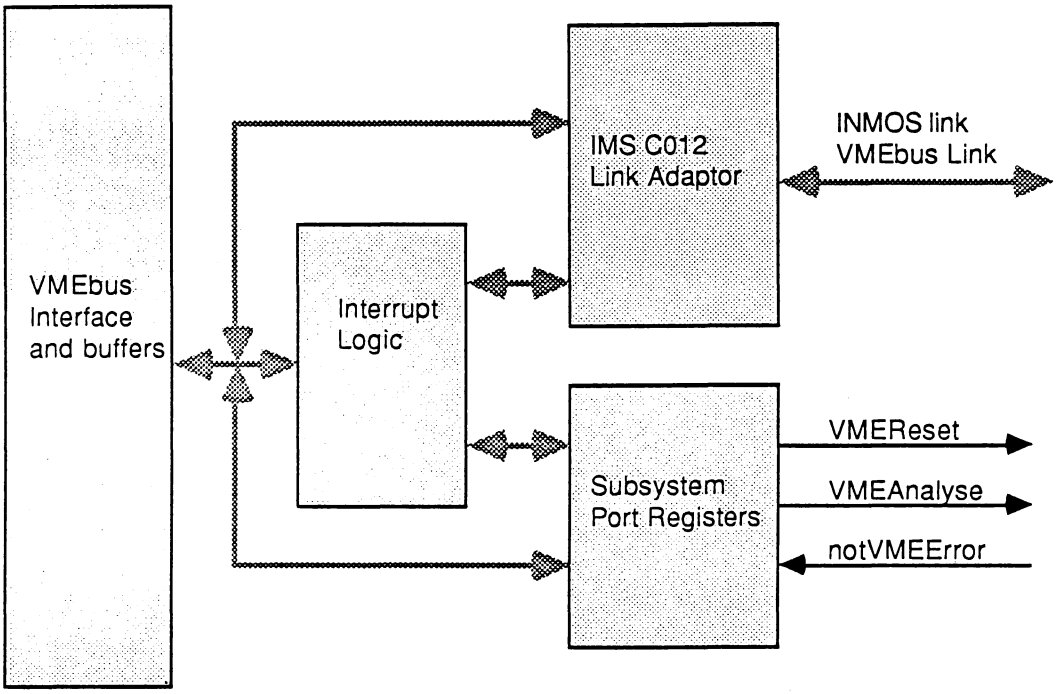


Figure 4: VMEbus Interface

The IMS B014's VMEbus interface is an 8-bit slave, driven by some master VMEbus device. The master can be any VMEbus board capable of generating A16:D08(0) VMEbus cycles. The IMS B014 can generate interrupts on any of the VMEbus interrupt lines or levels (the level is software-selectable). Interrupts can be programmed to occur on link transmiter empty, link receiver full and subsystem error signal active.

The only parameters for the VMEbus interface which must be selected by switches or jumpers on the board, rather than by software-programmed registers, are the VMEbus address and whether the board responds to VMEbus cycles.

5.1 Accessing the Board

The IMS B014's interface contains a number of 8-bit registers. The interface is programmed by accessing these registers from a master processor on the VMEbus. The master processor will see the IMS B014's registers as a part of its memory address space. The particular location of the registers in the processor's memory will depend upon the design and configuration of the master.

Note that the documentation for many 68020 and 68030-based processor cards will only mention "D32" and "D16" cycles. This is because they are confusing the processor bus size with the VMEbus cycle type. These boards *will* generate D08(O) cycles, suitable for addressing the IMS B014, when the processor performs a byte access at an odd address in the sixteen bit bus size address space.

The IMS B014 will respond with a BERR* (bus error) acknowlege if an attempt is made to perform an even byte cycle (D08(E)) or a D16 cycle. This is as recommended in the VMEbus specification. One result of this feature is that incorrect programs and device drivers which try to read 16-bit data from the board will cause a bus error. The action taken on bus error will depend upon the host system. For instance, a SUN-3 will suffer a kernel crash. Programmers, especially those using high-level languages, should bear this in mind. Programs running on a master board can be made to perform byte accesses either by writing sections in assember or by specifying the IMS B014's registers as type "char" in C.

Figure 6 shows the registers and their active bits.

5.2 Link Interface

The IMS B014's VMEbus interface provides one INMOS link, this is called the *VMEbusLink*. Unlike links on transputers, which are directly connected to the processor, this link is connected to a special INMOS interface chip called a *link adaptor*. This device (an IMS C012) functions very much like a UART does for an RS-232 interface.

The IMS C012 link adaptor has four registers (see figure 5). These are all available in the IMS B014's address space and can be read/written by a master device on the VMEbus.

The link adaptor registers function as follows—

Input Data Register holds the last data byte received from the VMEbus link. It only contains valid data whilst the *data present flag* is set in the input status register. It can not be assumed to contain valid data after it has been read; a double read may or may not return valid data on the second read. If the *data present* flag in the input status register is valid on a subsequent read it indicates new data is in the buffer. Writing to this register will have no effect.

Input Status Register This register contains the *data present* flag (bit 0, the least significant bit) and the *input interrupt enable* control bit (bit 1, the second least significant bit). The *data present* flag is set to indicate that a data byte in the input data register is valid. It is reset only when the input data register is read, or by a link reset condition (see section 5.6). When writing to this register, the *data present* flag and the unused bits 2–7 must be written as zero bits.

Output Data Register Data written to this register is transmitted out of the VMEbus link. Data must only be written to this register when the *output ready* bit in the output status register is set, otherwise data already being transmitted may be corrupted. Reading this register results in undefined data.

Output Status Register This register contains the *output ready* flag (bit 0, the least significant bit) and the *output interrupt enable* control bit (bit 1, the second least significant bit). The *output ready* flag is set to indicate that the data output buffer is empty. It is reset only when data is written to the output data register; it is set by by a link reset condition (see5.6)

When writing to this register, the *output ready* flag and the unused bits 2–7 must be written as zero bits. The *output interrupt enable* bit can be set and reset by writing to the input status register (note that the *output ready* flag must be written as a zero). Interrupts are explained in section 5.5.

Address	Register
#xx01	Link adaptor input data register (read only)
#xx03	Link adaptor ouptut data register (write only)
#xx05	Link adaptor input status register
#xx07	Link adaptor output status register

Figure 5: IMS C012 Registers

A link byte input proceeds as follows—
A data byte received on the VMEbus link sets the *data present* flag in the input status register. If interrupts are enabled, a link data input interrupt is generated. A VMEbus master device (processor) will, either in response to the interrupt or in a polling loop, examine the input status register. The *data present* flag will be set, signifying valid data in the inputs data register. The master then reads the data byte. A new data byte can now be received and the process repeats.

A link byte output proceeds as follows—
The master device (processor), either after receiving a data output interrupt or in a polling loop, reads the output status register. It will determine from the *output ready* flag that a byte may be written to the output

#xx01	DI7	DI6	DI5	DI4	DI3	DI2	DI1	DI0	Link Adaptor Input Data Register
#xx03	DO7	DO6	DO5	DO4	DO3	DO2	DO1	DO0	Link Adaptor Output Data Register
#xx05	☒	☒	☒	☒	☒	☒	In Int Enab	Data Pres	Link Adaptor Input Status Register
#xx07	☒	☒	☒	☒	☒	☒	Out Int Enab	Out Ready	Link Adaptor Output Status Register
#xx09	☒	☒	☒	☒	☒	☒	☒	Reset /Error	Sybsystem Error/Reset Register
#xx0B	☒	☒	☒	☒	☒	☒	☒	Analyse	Subsystem Analyse Register
#xx0D	☒	☒	☒	☒	In D Enab	Out D Enab	Errorint Enab	☒	Interrupt Enable Register
#xx0F	☒	☒	☒	☒	☒	Lev2	Lev1	Lev0	Interrupt Level Register
#xx11	SI7	SI6	SI5	SI4	SI3	SI2	SI1	SI0	Interrupt Status/ID Register
#xx13	Err7	Err6	Err5	Err4	Err3	Err2	Err1	Err0	TRAM Error Register

Figure 6: VMEbus Interface Registers

data register. It then writes the byte to the output data register. The byte is transmitted on the VMEbus link. When the link adaptor is next able to transmit a byte, the data output interrupt will be generated (if enabled).

5.3 Subsystem Interface

The VMEbus can provide "system services" to TRAMs (see section 7). This involves generating a reset signal (VMEreset) and analyse signal (VMEanalyse) as well as monitoring an error signal (notVMEError). This capability is provided via registers mapped into the IMS B014's address space (see figure 7).

The subsystem interface register work as follows—

Subsystem Reset/Error Register Writing Bit 0 of this register controls the state of VMEReset. Asserting the bit (writing a one) asserts VMEReset (a reset condition). This bit is cleared on VMEbus reset.

Reading Bit 0 of this register returns the state of notVMEError. If notVMEError is asserted (error condition) then the bit reads cleared (a zero).

The other unused bits must be written cleared (zero) and read as unpredictable values.

Subsystem Analyse Register Writing Bit 0 of this register controls the state of VMEAnalyse. Asserting the bit (writing a one) asserts VMEAnalyse (an analyse condition). This bit is cleared on VMEbus reset.

The other unused bits bust be written cleared (zero) and read as unpredictable values.

An active level on notVMEError can generate a subsystem error interrupt if enabled (see section 5.5).

Address	Register
#xx09	Subsystem Reset/Error register
#xx0B	Subsystem Analyse register

Figure 7: Subsystem Registers

5.4 Error Monitoring

The eight "error" signals from the eight TRAM slots on the IMS B014 can be read in a register, available in the IMS B014's address space at location **#xx13**. Bit 0 (least significant) contains the error signal from TRAM slot 0 and so on. If an error signal is active (signifying an error condition) then its corresponding bit in the TRAM error register is cleared. These error signals are visually monitored by LED indicators (see section 7.1).

Note that the contents of the TRAM error register need not bear any relation to the state of the Subsystem Error Register since they are derived from independent sources.

Note that transputers do not typically clear their error signals on reset, a program which explicitly clears the error signal must be run. Transputer programs usually have this code included automatically.

5.5 Interrupts

The IMS B014 can generate a VMEbus interrupt on any of the seven levels, the level is programmed into a register. An interrupt can be generated by three conditions—

1 A data output interrupt from the link adaptor.

2 A data input interrupt from the link adaptor.

3 An active level on the subsystem error signal (notVMEError).

All three interrupt sources are independently maskable via the interrupt enable register. Note that for data interrupts this is in addition to the interrupt enable bits in the link adaptor input/output status registers.

When a VMEbus interrupt is recognised and serviced by a VMEbus "interrupt handler" (usually a processor), the IMS B014 responds with a status/ID byte. This is VMEbus terminology for "interrupt vector". The status/ID byte will be used by the interrupt handler to determine what action to take in servicing the interrupt. The status/ID byte returned by the IMS B014 is stored in the status/ID register.

Address	Register
#xx0D	Interrupt enable Register
#xx0F	Interrupt Level register
#xx11	Interrrupt status/ID register

Figure 8: Interrupt Registers

The interrupt registers function as follows—

Interrupt enable register This contains three enable bits, one for each of the input data, output data and error active interrupt sources. After an interrupt condition has occured, an actual interrupt will only be generated if the appropriate enable bit is set.

If the interrupt enable register, or the interrupt enable bits in the link adaptor input and output status registers are used to disable an interrupt after that interrupt has been generated, and before the interrupt acknowlege cycle, then the IMS B014 will not respond to the interrupt acknowlege cycle. This will result in a bus timeout and probably some kind of spurious interrupt trap on the interrupt handler.

The enable bits may be read. The unused bits in this register read unpredictable values. On VMEbus reset all three bits are cleared, meaning that interrupts are disabled after a system reset.

Interrupt Level register This contains the VMEbus interrupt level to be used for interrupts. Seven levels (1–7) are possible. The level used is dependent upon the system into which the IMS B014 is installed. Bits 0–2 hold the level, as a binary number. If a level corresponding to zero is selected then interrupts will not be generated. This register must not be changed when interrupts are active.

The level can be read. The upper five bits return unpredictable data. The level is set to zero on VMEbus reset.

Interrupt status/ID register This register holds the status/ID byte returned on interrupt acknowlege cycles. This register can be read. On VMEbus reset this register contains unpredictable data. This register should not be changed when interrupts are active.

The particular number stored in this register depends upon the driving software and the host system. The IMS B014 will operate correctly with any status/ID byte.

It should be noted that there may be interrupt conditions valid when interrupts are disabled. For instance, the sybsystem error line could be permanently active or a byte may have been received by the link adaptor some time in the past (possible due to electrical noise during power-up). Before enabling interrupts the programmer should always check for pending interrupt conditions and if necessary, remove the condition. In the case of a data input interrupt reading a byte from the input data register will remove the condition.

5.6 VMEbus Reset and link reset

The VMEbus defines a signal (SYSRESET*) which is used to reset the entire system, usually after power-on or a system crash. The IMS B014 uses this signal to do the following—

- The VMEbus interface is reset to a known state, certain registers are initialized to known values. The IMS C012 link adaptor is reset, overcomming any locked up link condition.
- The IMS C004 link switch devices are reset, this disconnects all their link connections.
- None of the TRAMs are reset, the network configuration processor (T2) is not reset.

Throughout this document, this condition will be called "VMEbus reset".

When the VMEbus subsystem reset signal is asserted (by writing the appropriate values into the subsystem reset/error register), apart from reseting any TRAMs connected to the VMEbus subsytem port, the VMEbus interface link adaptor is reset. This allows link lock-ups to be overcome without totally resetting the entire system. Users must bear this in mind as it means that while VMEreset is held active, no VMEbuslink communication can occur. Also, after performing a VMEbus subsystem reset, the contents of the link adaptor registers will need to be re-initialized.

5.7 VMEbus Interface Configuration

The IMS B014 can respond to addresses in the A16 (short I/O) VMEbus address space. This means that the board will respond to addresses in the range **#xx00** to **#xxFF** where **xx** is determined by the address selection switches. In addition, the VMEbus interface can be completely disabled by removing jumper JP2. This is useful if more than one IMS B014 is to be fitted into a system and the VMEbus interface is not required on all the boards. Full information on how to configure the board is to be found in section 9.

5.8 Address Modifier Codes

The VMEbus uses extra "address lines" called *address modifiers* to specify certain things about data transfer cycles. The IMS B014 as shipped responds to codes **#29** and **#2D**. These correspond to the standard non-privileged and privileged short address codes. If other codes are required for special applications, consult INMOS for the appropriate information.

5.9 Programming the VMEbus Interface

This section gives example code fragments in C to give a flavour of how the IMS B014's interface may be programmed. An undrestanding of the C language is assumed.

Firstly, we declare pointers to the registers—

```
char *board_base;
char *data_in;
char *data_out;
char *status_in;
char *status_out;
char *reset;
char *error;
char *analyse;
char *int_enable;
char *int_level;
char *vector;
char *tram_error;

void setup_addresses()
{
    board_base = 0x00ff0001;
    data_in = board_base;
    data_out = board_base + 2;
    status_in = board_base + 4;
    status_out = board_base + 6;
    reset = board_base + 8;
    error = reset;
    analyse = board_base + 10;
    int_enable = board_base + 12;
    int_level = board_base + 14;
    vector = board_base + 16;
    tram_error = board_base + 18;
}
```

Here are some example setup routines—

```
void enable_out_ints()
{
    *status_out = 0x02;
    *int_enable |= 0x04;
}

void enable_in_ints()
{
    *status_in = 0x02;
    *int_enable |= 0x08;
}

void enable_error_ints()
{
    *int_enable |= 0x02;
}

void disable_out_ints()
{
    *status_out = 0x00;
    *int_enable &= 0xfb;
}

void disable_in_ints()
{
    *status_in = 0x00;
    *int_enable &= 0xf7;
}

void disable_error_ints()
{
    *int_enable &= 0xfd;
}

void set_int_level(level)
int level;
{
    int_level = (char) level;
}

void set_vector(vec)
int vec;
{
    *vector = (char) vec;
}
```

Polled link I/O procedures—

```
char read_link()
{
    while ( (*status_in & 1) != 1) ;
    return( *data_in );
}

void write_link(byte)
char byte;
{
    while ( (*status_out & 1) != 1 ) ;
    *data_out = byte;
}
```

5.10 Operating without the VMEbus interface

The IMS B014 has been designed so that correct operation of the TRAM slots is possible even if the board is not connected to a VMEbus backplane. What this means is that the VMEbus interface is completely independent of the TRAM slots, except for the VMEbus SYSRESET* signal. Since SYSRESET* is used for power-on initialization of the IMS C004s, non-VMEbus users of the IMS B014 should drive SYSRESET* with their power-on reset signal. Except for this consideration, all the other VMEbus signals can be un-driven and un-monitored and the non-VMEbus parts of the IMS B014 will function.

In this non-VMEbus role, the IMS B014 can be used as a double-eurocard TRAM motherbaord suitable for mounting in standard VMEbus card-cages (which are readily available fitted with power supplies).

6 TRAM slots

The IMS B014 has eight TRAM "slots". A slot is a place on the motherboard where a TRAM may be plugged. A slot has four links associated with it. Since TRAMs come in various sizes, the number of TRAMS on a motherboard may be less than the number of slots.

6.1 TRAM Network

The eight TRAM slots provided on the IMS B014 each have four INMOS links. These links are designated by their slot number and link number for the slot. Thus, slot 0, link 1 is link 1 of slot 0. Slots are numbered 0–7, links are numbered 0–3. This numbering scheme is used consistently throughout this manual. The slot posititions are shown in figure 1 and on the board silk screened legend.

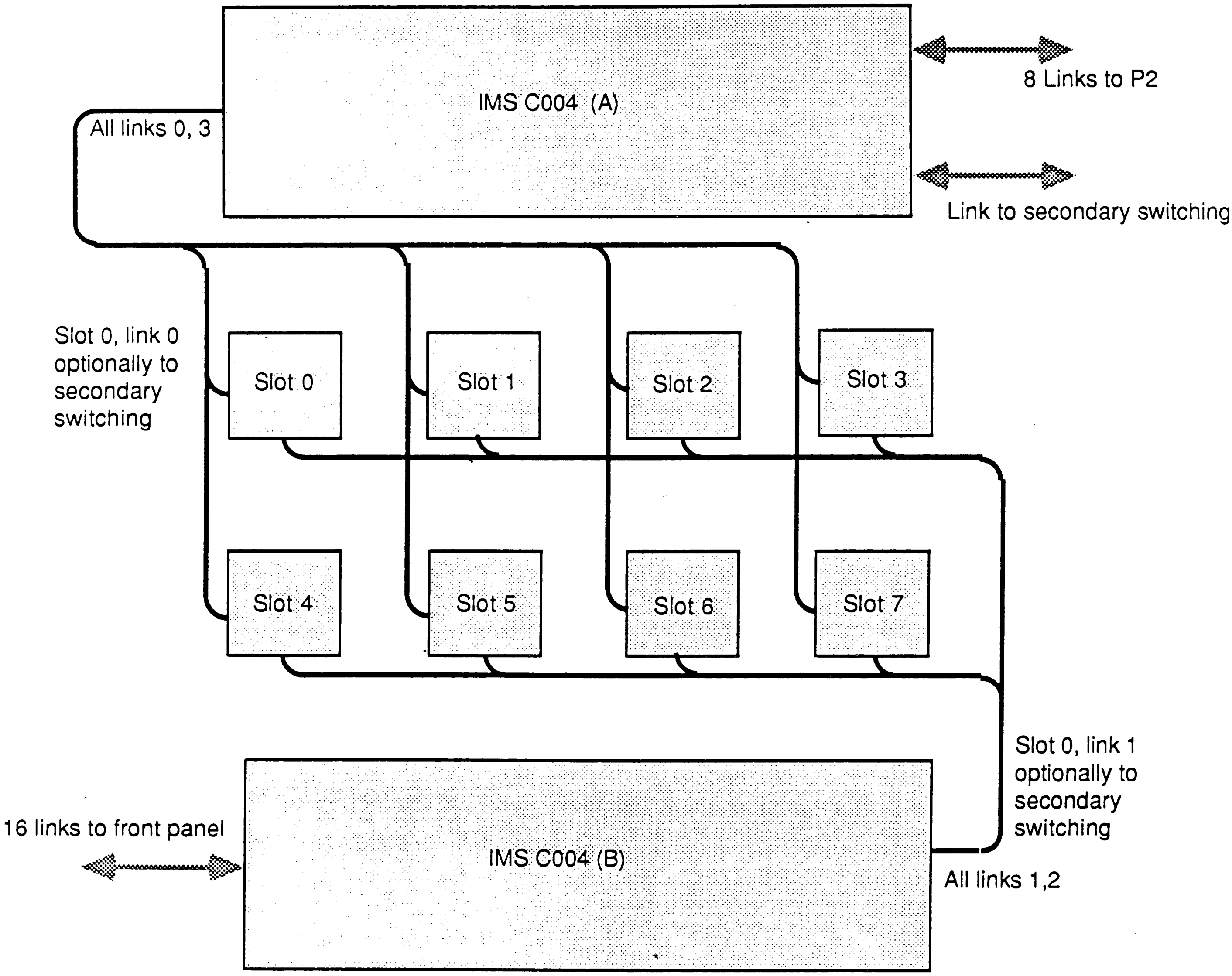


Figure 9: Basic Link Network

The eight slots, each with four links, give 32 links in total. The IMS B014 uses IMS C004 link switch chips to allow these links, along with INMOS links which come from the IMS B014's connectors, to be connected to each other. The links are used to send data between the transputer on the TRAMs and so the actual connection network used will be different for different applications. The basic network organisation is shown in figure 9.

The IMS B014 is capable of implementing any network between its eight slots. For instance, one application may need a mesh array, shown in figure 10 wheras another may need a tree network (see figure 11).

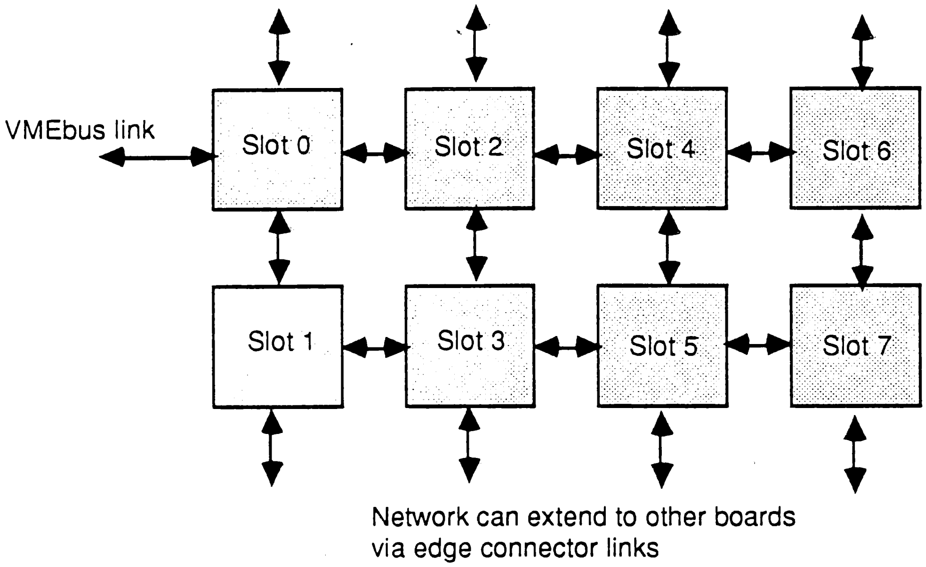


Figure 10: Example Matrix Network

All the connections made using IMS C004s are programmed by software, without the need to plug jumpers on the card. The connection arrangement can be changed without the card being removed from the card-cage and, with the right software it is even possible to change the connections dynamically (see section 6.7).

Some connections must be programmed by changing jumpers. These are connections which will carry data involved in actually configuring the IMS C004s. Obviously, if a TRAM in slot 0 is to configure the IMS C004s, it must communicate with them. The link carrying this communication can not itself go through either of the IMS C004s. The link configuring done by jumpers is called "secondary link switching" and is described in section 6.2.

The link connections and IMS C004 switches provided on the IMS B014 allow any network to be constructed between the TRAMs on the board. This is provided that the actual link used for any path between two TRAMs is not fixed. For instance, to connect between slot 0 and slot 1 there is a path for each link on slot 0 to get to a link on slot 1. However, if one considers slot 0, link 3 it may only be connected to links 0 and 3 on slot 1, and not to links 1 and 2. Usually this restriction is not a problem as the links are physically identical and the program running on the TRAMs may be written to use any link for a particular path.

There are a total of 24 links provided on the connectors so any multi-board network which needs 24 or fewer links off each board can be constructed. The connector links may be connected to any TRAM on the board with a similar restriction on the actual destination link as for inter-board connections.

6.2 Secondary Link Switching

Some links which may be required to carry data for configuring the IMS C004s, but nevertheless themselves require configuration, are switched by jumpers. Figure 12 shows the links involved. The grey lines indicate a possible connection. The nine possible connections are designated A–I, by letters in the diagram. These letters correspond with the letters in figure 24 attached to the jumper locations. If, for a particular letter, the

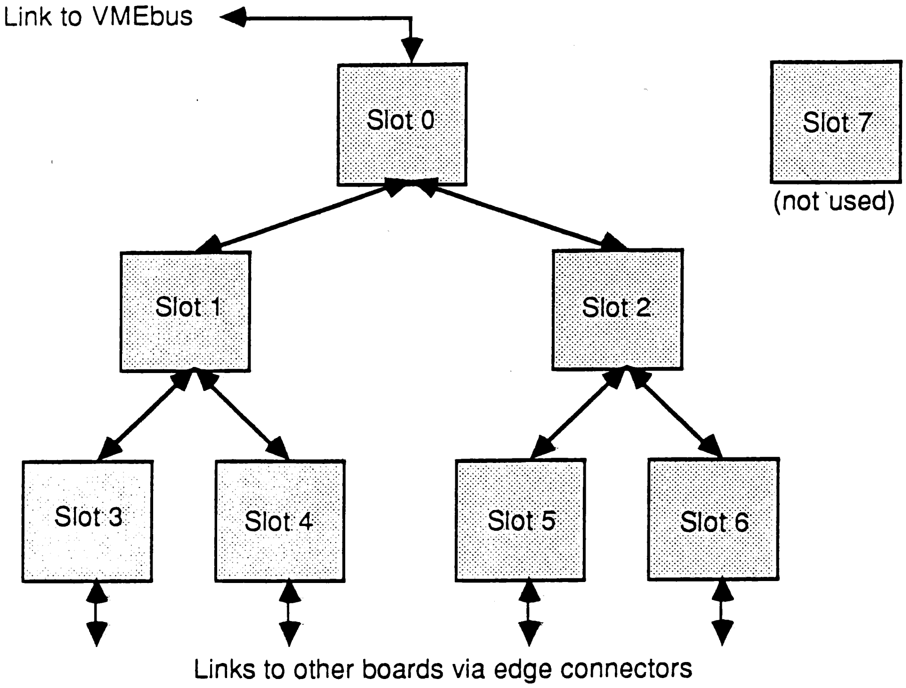


Figure 11: Example Tree Network

two jumpers are fitted then that connection is made. Care should be taken not to make mutually exclusive connections[2], for instance connecting A as well as I.

6.3 The IMS C004

The IMS C004 device used to switch links on the IMS B014 is described fully in *The Transputer Reference Manual* ([2]). The IMS C004 has 32 link ports. Any of these links may be "wired up" to any of the others on commands sent to the IMS C004 down a 33rd link, called *ConfigLink*. The commands sent down the ConfigLink are simple two or three byte messages, saying "connect link 1 to link 3" and so on. The ConfigLinks for both of the two IMS C004s on the IMS B014 are connected directly to two links of an IMS T212 (or IMS T222) transputer which is also on the IMS B014 (see figure 13). The IMS C004 control sequences are shown in appendix K.

This transputer (called the network configuration processor or T2), sends the configuration data to the IMS C004s based on commands which it receives on one of its other links.

6.4 Network Configuration Processor

The Network configuration processor is a 16-bit transputer (either an IMS T212 or IMS T222) whose purpose on the IMS B014 is to co-ordinate the flow of configuration data to the IMS C004 link switch chips. For easy reference, this processor is also called a "T2". It has no external memory and is connected via links as shown in figure 13.

In general the Network Configuration Processor will run a simple program (which is booted down its link 1)[3]. That program will then receive commands on link 1. These commands will come from some master computer which may be setting up the network for a number of motherboards. The T2 will send any appropriate configuration data to the IMS C004s on its links 0 and 3. Data which is not destined for its board will be

[2] Links are of course point-to-point connections so the effect of installing jumpers in exclusive positions will be to connect two outputs to each of the two link inputs involved. While this will result in a non-working link connection, it will not damage any circuitry.

[3] Transputers do not need ROMs since after reset they can receive a program directly down a link and run it.

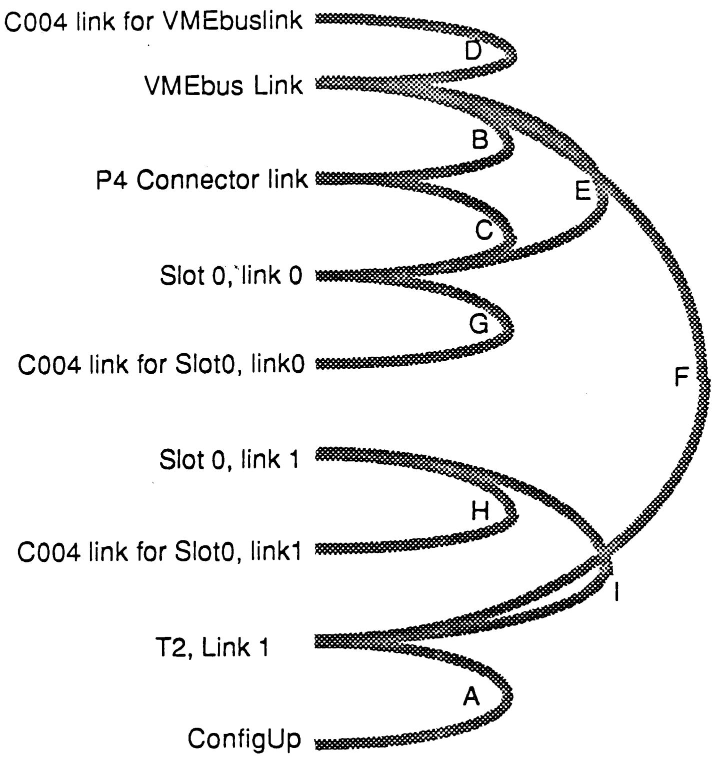


Figure 12: Secondary link switching jumpers

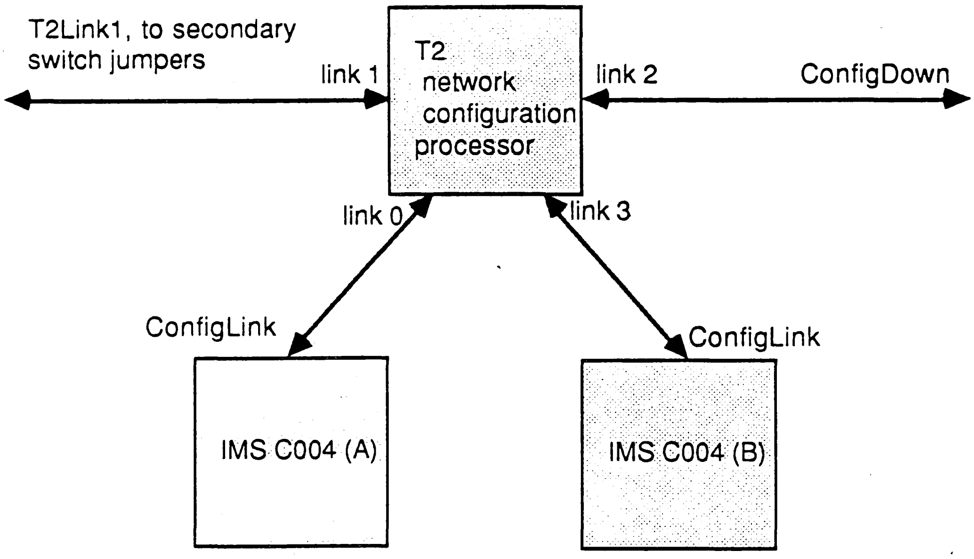


Figure 13: The Network Configuration Processor and IMS C004s

re-transmitted on link 2. This data will be received on link 1 of the network configuration processor on the next board (if there is one). The program which runs on the network configuration processor can be user-defined, but INMOS supplies suitable software.

The two links 1 and 3 of the network configuration processor are designed to be connected as a daisy-chain in multiple board systems. When links 1 and 2 of the T2 reach the edge connectors, they are called "ConfigUp" and "ConfigDown" respectively. In general, ConfigUp of each board is connected to ConfigDown of the adjacent board. ConfigUp of the first board is connected to the processor which is configuring the network (see figure 14).

The reason for using a processor to communicate with the link switches is that infinitely large multi-board systems may be constructed. Such systems may be configured from one link (the head of a large ConfigUp–ConfigDown daisy-chain) and may contain different board types since all INMOS TRAM motherboards use the same scheme.

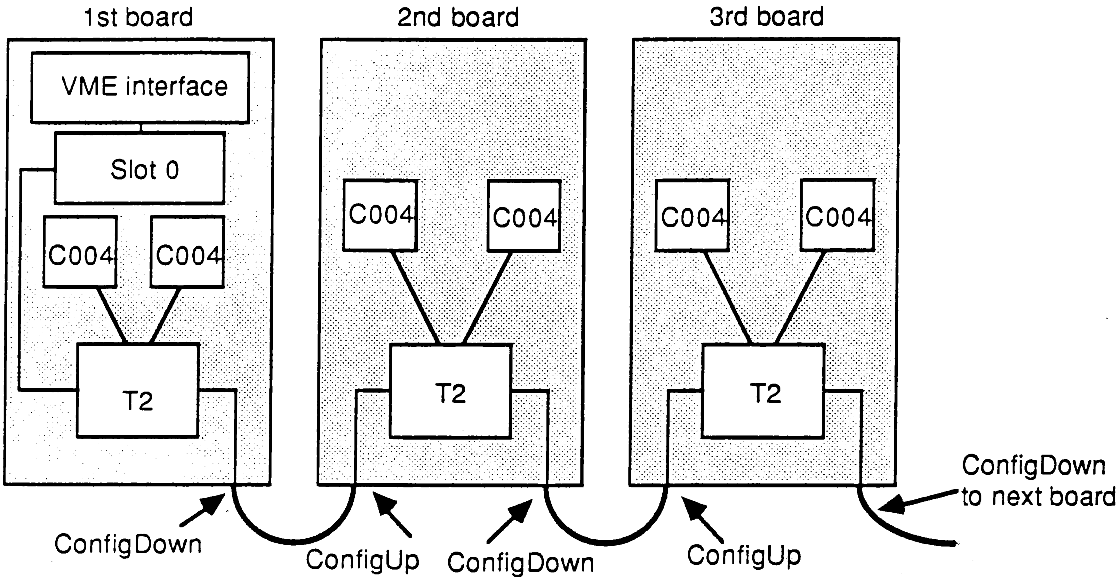


Figure 14: ConfigUp/ConfigDown Daisy-Chain

The IMS B014 provides flexibility in the connections to the T2. This is achieved via the secondary link switching jumpers (see figure 12) and jumpers which allow either the front or the back edge connectors to be used for ConfigUp and ConfigDown (see section 9 and figure 17).

With the secondary link switching arrangement on the IMS B014, it is possible to send configuration data to the T2 from—

1 Slot 0, Link 1—This means that a "master" TRAM on the IMS B014 can configure the TRAM array.

2 The VMEbusLink—This means that a VMEbus master can configure the TRAM array.

3 The "ConfigUp" connections on the edge connectors (either the front or back of the board, selected by jumpers)—This means that some other board can configure the TRAM array.

6.5 C004 Reset

The two IMS C004 link switches have a reset pin that is driven by a power-on-reset circuit. The IMS C004s can be also soft-reset by a command from the T2.

The IMS T212 has 2 Kbytes of on-chip RAM (4 Kbytes in the case of the IMS T222). It also has an external memory interface. Circuitry on the IMS B014 is connected to the T2's external memory interface which allows the reset signal to the IMS C004s to be controlled from the T2. By writing a *one* into bit position *zero* in any external memory word, the reset signal to the IMS C004s is *asserted*. Similarly, by writing a *zero* into bit position *zero* in any external memory word, the reset signal to the IMS C004s is *de-asserted*.

The porpose of this feature is to allow the C004s to be hard-reset without resetting the whole VMEbus system. This may be useful in systems requiring some kinds of fault tolerance. Note that if the IMS T212 reads from any location in its external memory space then the IMS C004 reset signal will be set to an unpredictable level.

6.6 Affect of the IMS C004 on link bandwidth

Depending upon the type of transputers on the TRAMs at both ends of a link which passes through an IMS C004, the link bandwith may be lower than for a simple transputer-to-transputer connection.

For instance, two IMS T800s connected directly will give a unidirectional link bandwith of 1.7 Mbytes/s. However, with one IMS C004 switching that link, the link bandwith is 1.3 Mbytes/s. With two IMS C004s switching the link, as is the case with a board-to-board link using IMS B014s, the link bandwith will be 800 Kbytes/s.

If an application calls for full link bandwith and this is not achievable using the IMS C004s then, providing the desired network does not need to change, header-circuits can be plugged in place of the IMS C004s. These will connect the links across directly using pieces of wire and buffer chips where necessary. Contact INMOS for information on these.

6.7 Dynamic Connection Changing

In theory it is possible to change the configuration of the IMS C004s while a program is executing on the TRAM array. This may be useful, for example, in a system which needs a particular network during a data gathering phase but a completely different network during a data processing phase. Although this is possible, it is not easy to organise and should only be attempted by experienced users who have a complete understanding of what needs to be done. For those who still want to proceed, here are some guidelines—
The basic idea is that providing there is no traffic on a link, you can switch the path it takes through an IMS C004. After switching, processing can proceed using the new network.

Obviously this requires careful synchronisation between all the programs in all the TRAMs—something which is usually achieved via the links which are being switched.

7 System Services

Transputers and TRAMs need to be fed reset and analyse signals (these allow the processor to be reset and placed in a debugging mode) and they generate an error signal which needs to be monitored. These three signals (reset, analyse and error) are together called *system services*. The system services for any TRAM are treated as one conceptual signal, even although there are actually three. System services are generated by a "subsytem port". A subsystem port will generate a reset and analyse sugnal and monitor an error signal. The VMEbus interface of the IMS B014 provides a subsystem port. Some TRAMs also have subsystem ports (in this case, the TRAM is said to have *subsystem capability*) on three extra pins (see section B.3). Slot 0 of the IMS B014 can take a TRAM with subsystem capability. System services are used by TRAMs. On the IMS B014, slot 0 is separated from the other seven TRAMs for system services—this is to allow the TRAM in slot 0 to provide the system services for the other seven. System services can also come from off-board via the ServicesUp connection on the connectors. The system services going to slots 1–7 are propagated to the ServicesDown connection on the connectors—it can provide the ServicesUp for the next board in the card-cage (see figure 15).

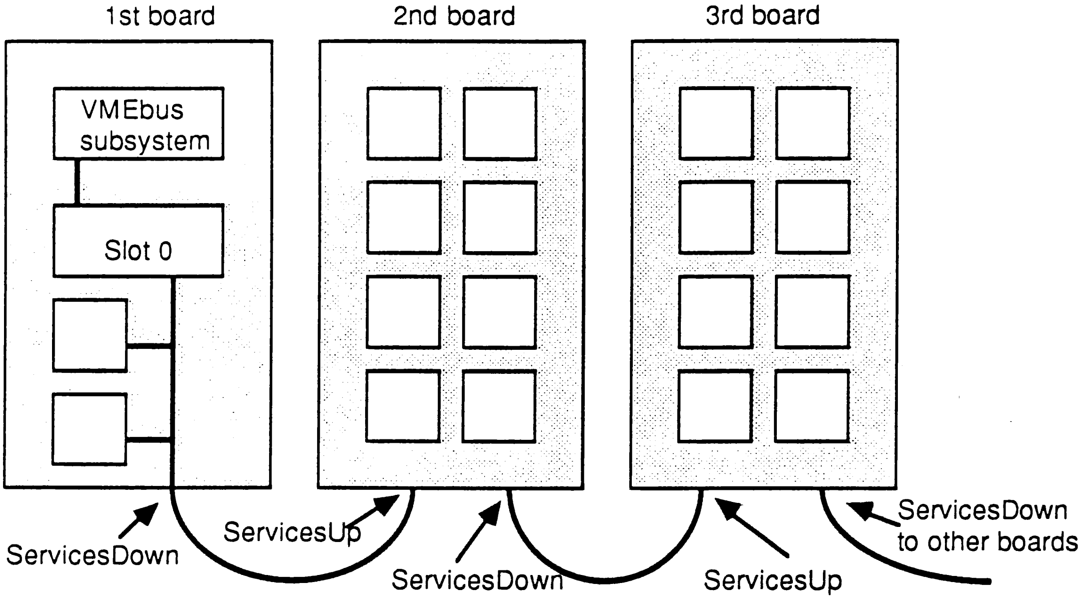


Figure 15: Services Daisy-Chain

These ServicesUp and ServicesDown connections on the edge connectors are designed to allow multiple board systems to be connected together in a daisy-chain. That is, the ServicesUp for a board comes from the ServicesDown of the previous one. These connections are compatible with other INMOS boards.

The arrangement of system services connections is shown in figure 16. The "switches" shown in the figure allow the user to select where the system services for parts of the board are driven from. In reality, these switches are electronic and are controlled by jumpers—one jumper per switch (note that in figure 16 the switch positions correspond to the respective jumpers being "installed").

7.1 Error LEDs

There are eight LED indicator lamps situated along the middle of the IMS B014's front panel. These monitor the error signals from all eight slots on the board. This monitoring is completely independent from the error signals which go to the connectors and to either of the "subsystem" ports—those error signals are combined from a number of slots. When the board is fitted in a card-cage the top LED monitors slot 0, the next lower slot 1 and so on down to slot 7 at the bottom. These LEDs are useful for error diagnosis because they allow

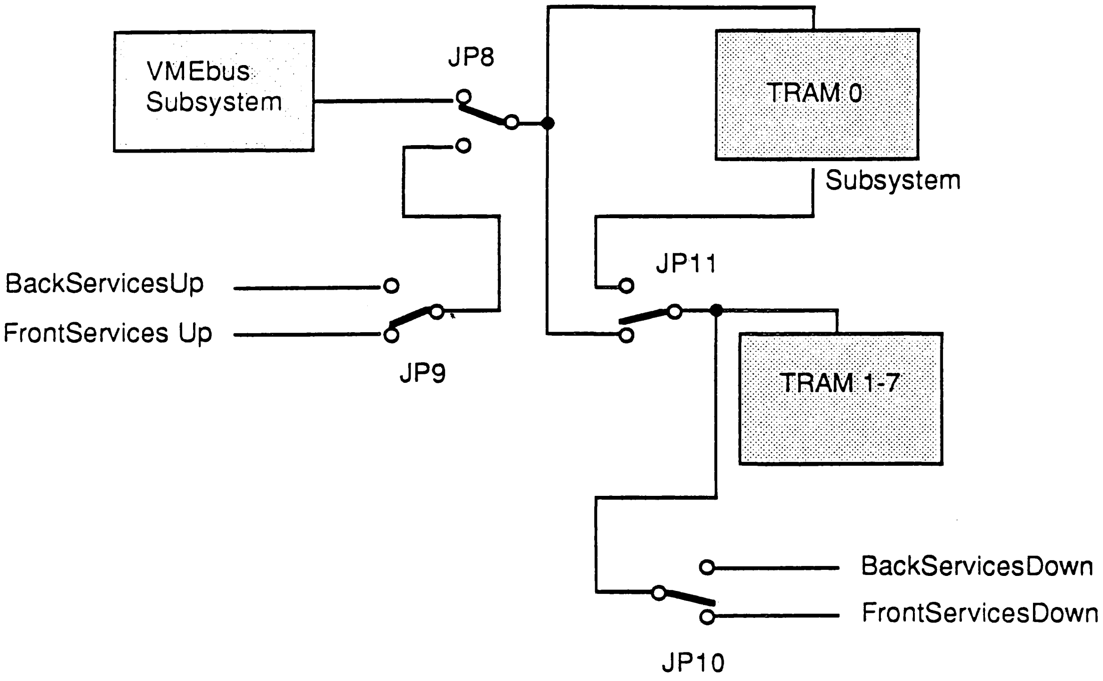


Figure 16: System Services Configuration

errors to be tracked down to an individual TRAM after an error is detected by one of the subsystem ports. Full information on the transputer error signal can be found in [2]. Note that error signals are not usually de-asserted on reset so these LEDs may be lit when the board is switched on.

7.2 Driving Services Signals

The three services signals—Reset, Analyse and Error— are detailed in [2]. However, because the signals are passed through circuitry on the IMS B014, it is necessary to allow time for the signals to be propagated through the system. Always allow 100ms between the assertion of a services signal and the time when it actually reaches its destination.

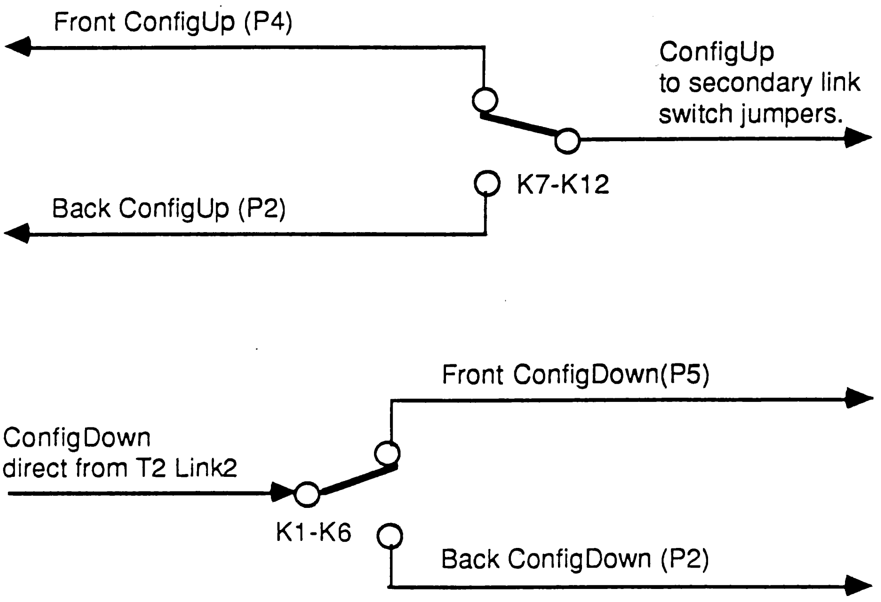


Figure 17: ConfigUp link switching

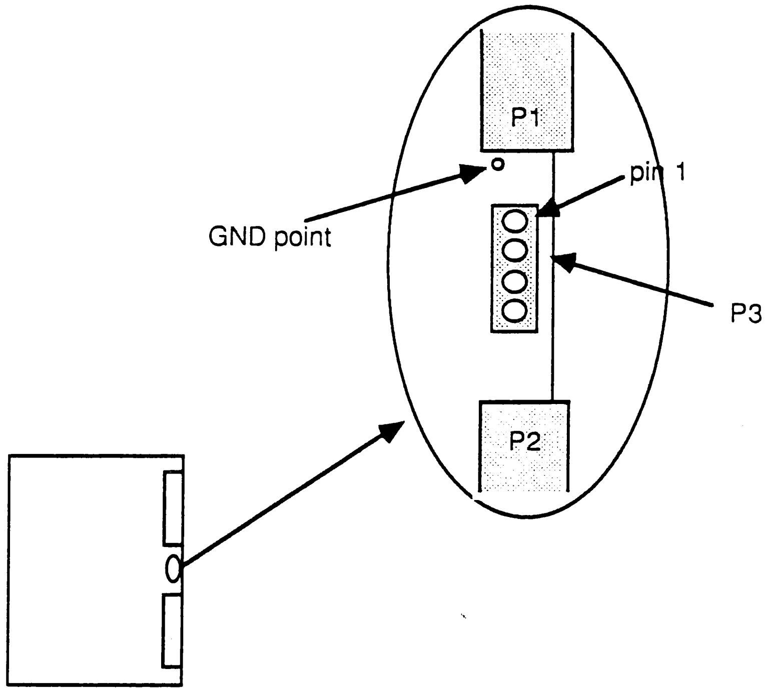


Figure 18: Position of P3 and GND point

8 Connectors

The IMS B014 has five connectors. Two of these are the VMEbus P1 and P2 connectors (these designations match those in the VMEbus specification). Another two connectors are on the front panel and are called P4 and P5. The remaining connector (P3) is a four-pin power connector. P3 allows the ±12v power rails from the VMEbus to be used by TRAMs, for example the IMS B407 Ethernet TRAM which needs +12v. The position of P3 is shown in figure 18. Mechanical details for the connectors are given in section C.

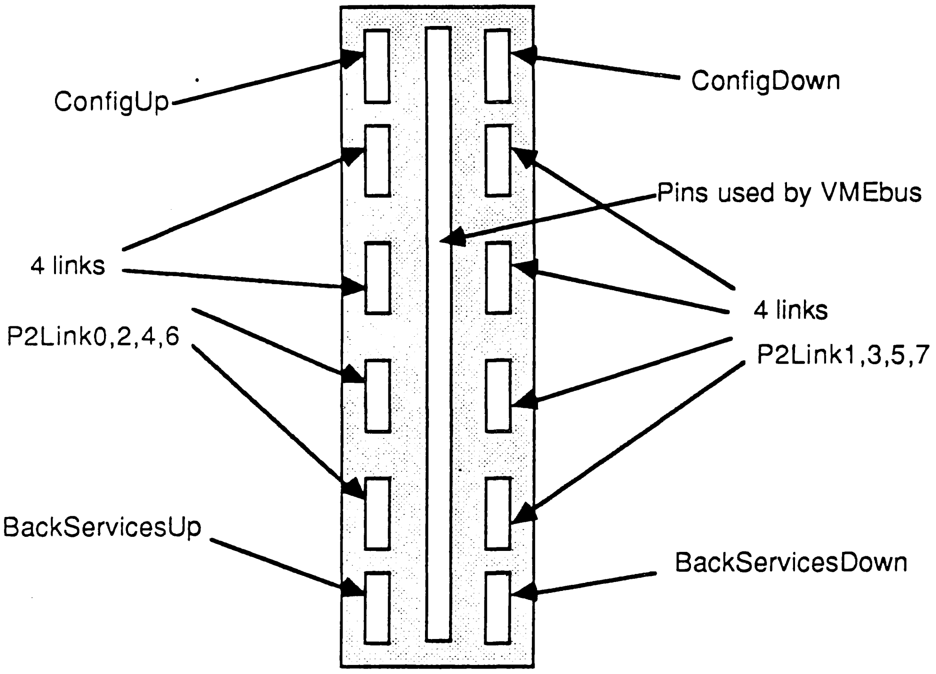


Figure 19: Back Connector (P2)

When signals other than the VMEbus are to be connected to a VMEbus board there are two possibilities—either use the uncommited P2 pins (see below) or connectors on the front panel. P2 connections are mechanically neat but are difficult to access in a card-cage and give only 64 pins. Front panel connectors are easy to access but are too untidy in some applications. The IMS B014 allows either approach to be used. Either the front connectors (P4 and P5) or the rear connector (P2) may be used. Provision is made for configuring some signals to connect via either the front or the rear connectors.

The VMEbus defines all of the pins on P1 and 32 of the 96 pins on P2. Since the IMS B014 does not support any of the 32-bit VMEbus capabilities, the VMEbus-defined pins on P2 are un-connected, except for power pins. In order to function on a VMEbus, only P1 need be connected—to a J1 VMEbus backplane. If the full power dissipation of the IMS B014 is to be realised then the power connections on P2 will need to be conected to a J2 backplane.

The remaining 64 pins on P2 which the VMEbus does not use (these are user-defined in the VMEbus specification) are, on the IMS B014, used to carry INMOS links and reset signals to other boards in a card-cage. Note that if conections to P2 are inconvenient then connectors P4 and P5 can be used for any signal which is available on P2 (indirectly in the case of links).

The user-defined pins on P2 are used by many VMEbus boards to take I/O signals out of the card-cage and also to carry "secondary" busses such as VMX and VSB. Because of this, some card-cages may have these pins pre-wired. It is important to ensure that the IMS B014 is not plugged into a card-cage which has these signals wired for a different board (see section B).

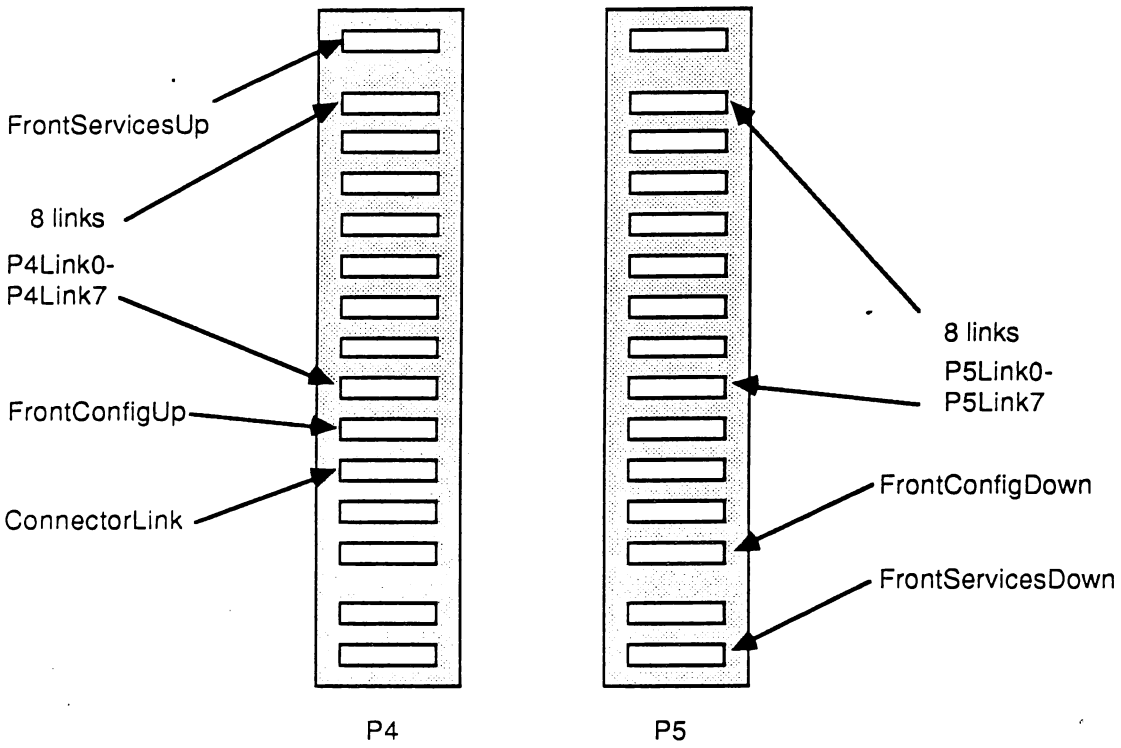


Figure 20: Front Connectors

8.1 Cables

INMOS has developed a standard cable set for evaluation boards. Link cables have 5-pin single-in-line connectors at each end with a key pin. Services cables have 5-pin single-in-line connectors at each end with two key pins. The keyed pins prevent the connectors being inserted incorrectly. The cables are available in a range of lengths (see fig 39). Using this system it is quick and easy to connect links and services signals from board to board and system to system. These cables must not be used between equipment on different mains circuits and ideally, to ensure noise-free operation, should only be used within one system operating from the same 5V power supply.

The connectors on all INMOS boards and modules are designed to be compatible with these cables. The IMS B014 is provided with a cable set which includes many short link cables (which can be used to link edge links to each other), some standard and long link cables (which can be used to connect multiple boards together), some services cables (reset, analyse and error) and a DIN 41612 connector which when plugged into P2 allow cables to be connected to the board.

This DIN 41612 connector is a back-to-back plug with some pins cut short and some pins sleeved (shown in figure 21). These are to mate with the keying on INMOS cables. This assembly fits onto P2 or onto the back of a J2 VMEbus backplane.

A "breakout" plug is provided for P4 and P5 (see figure 22). This allows standard INMOS cables to be used with these front panel connectors.

If extra INMOS cables are required, contact either INMOS or your local Sales Office for ordering details.

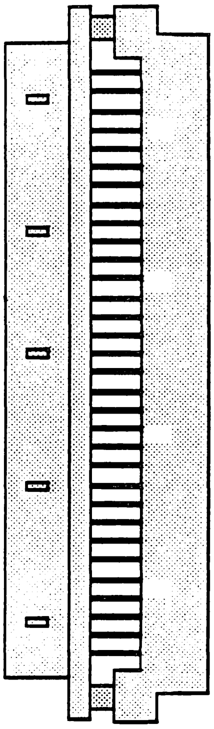


Figure 21: "Back-to-back" DIN Connector

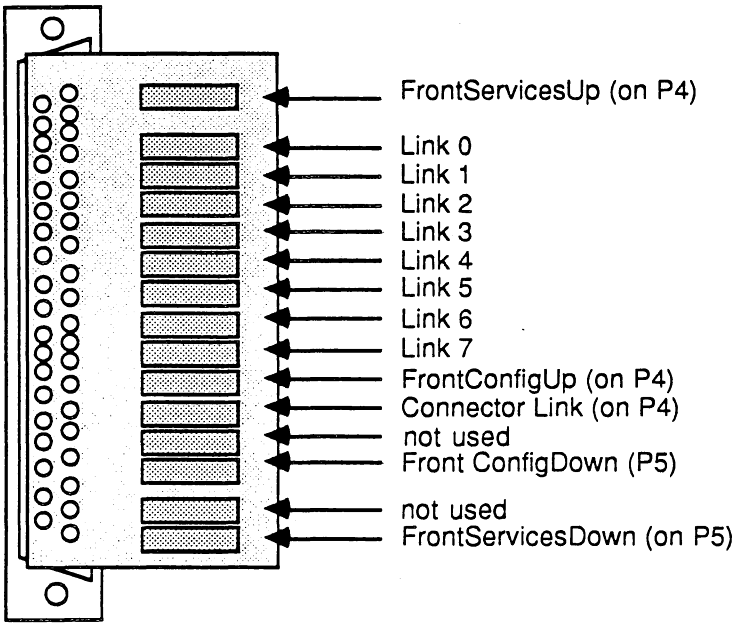


Figure 22: Front Connector "breakout board"

9 Configuration

This section covers all user-configurable aspects of the IMS B014—switches and jumpers.

Jumpers are labeled either JPx, where a jumper is either installed or absent between two pin posts; or Kx–Ky where jumpers are installed over a centre pin and either of two other pins (making a kind of change-over switch). The locations of all the jumpers are shown in figure 24. It is not necessary to wire-wrap any connections between jumper pins.

The IMS B014 is shipped with no jumpers installed—therefore it is necessary to perform some configuration before using the board.

9.1 VMEbus address selection

The VMEbus interface can be completely disabled by removing JP2. This means, of course that JP2 must be fitted for the VMEbus interface to work.

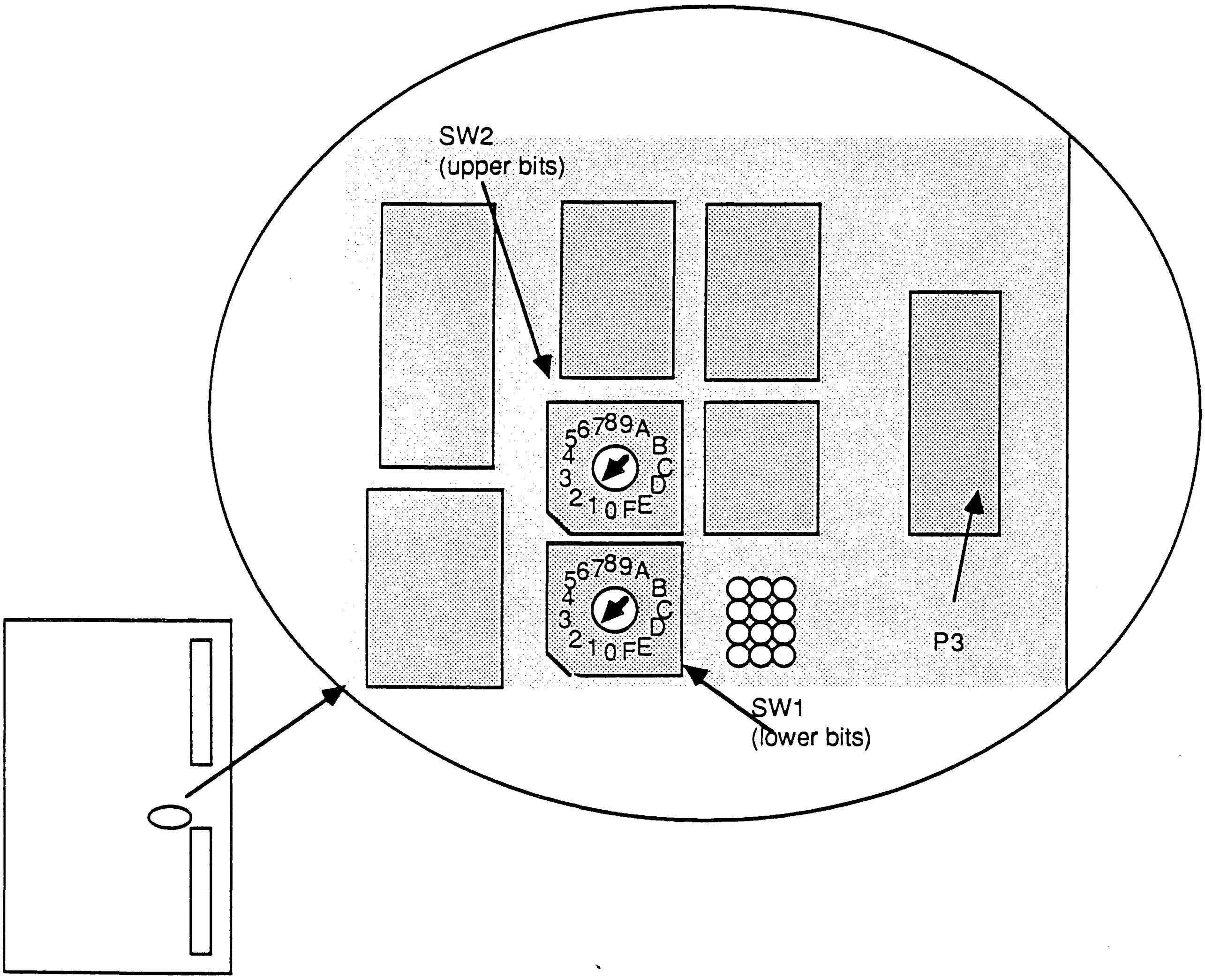


Figure 23: VMEbus address switches

The VMEbus address is selected by two hex switches (see figure 23). In setting the VMEbus address, you are selecting a unique 8-bit binary number which will be compared with the VMEbus addresses. The upper four bits of this number are set by SW2 while the lower four bits are set by SW1. A small screwdriver or trim-tool can be used to rotate the pointers on SW1,2. The pointer must be rotated to the desired hex character. Here are some example settings—

We want to use addresses **#DB00–#DBFF**. This means that the 8-bit number to be compared is **#DB**. Its upper four bits are **#D**, its lower four bits are **#B**. Accordingly we set SW2 to "D" and SW1 to "B".

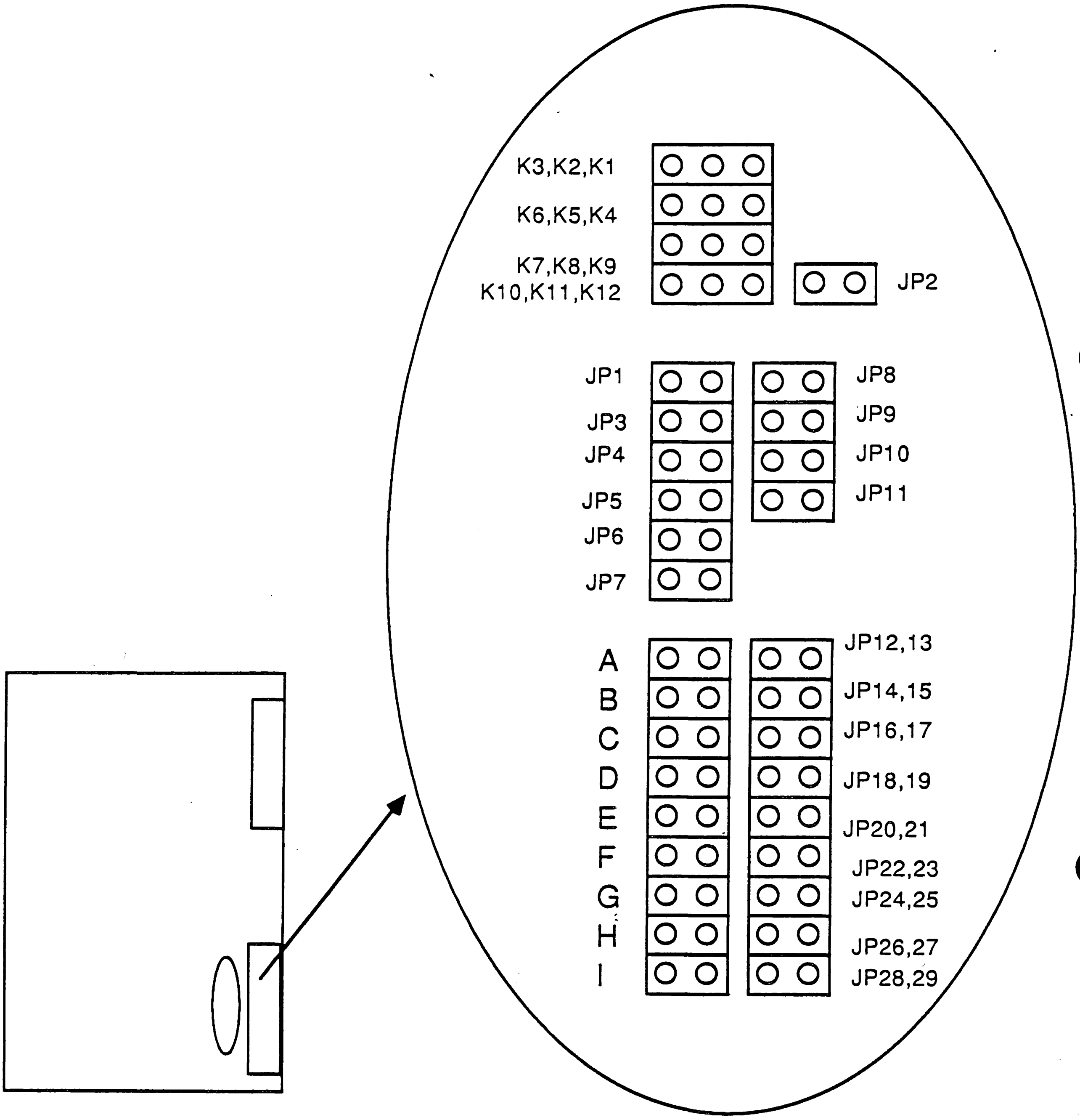


Figure 24: Configuration Jumpers

9.2 Link Speeds

The links available on the IMS B014 (from C004s, link adaptor, TRAM slots and the T2) can be set to work at different speeds. For a link port on one device to talk succesfully to another device, they must be set to the same speed. Current technology allows link speeds of 10 or 20Mbits/s. Note that if a link is being switched by one of the IMS C004 link switch devices then the C004 link speed must be the same as the link's speed.

The IMS B014 allows the user to set the link speeds independently for different parts of the board. This is to allow for situations where the board is communicating with other systems at different speeds. For most situations, the whole board would be configured to operate at 20Mbits/s.

The jumpers which control link speed are JP1, JP3, JP4, JP5, JP6 and JP7. For complete 20Mbits/s operation all these jumpers must be removed. For complete 10Mbits/s operation all these jumpers must be fitted. Figure 25 shows the full details. Note that TRAM link speeds are not defined by the IMS B014, but by the TRAM's interpretation of its "LinkSpeedA" and "LinkSpeedB" pins. Current TRAMs implement these speed selections as shown in figure 26.

Jumper	Use
JP1	Link speed for VMEbus link. Installed for 10Mbits/s, empty for 20Mbits/s.
JP3	Link speed for C004s and T2. Installed for 10Mbits/s, empty for 20Mbits/s.
JP4	LinkSpeedA for TRAMslot 0. Installed for "low", empty for "high".
JP5	LinkSpeedB for TRAMslot 0. Installed for "low", empty for "high".
JP6	LinkSpeedA for TRAMslots 1–7. Installed for "low", empty for "high".
JP7	LinkSpeedB for TRAMslots 1–7. Installed for "low", empty for "high".

Figure 25: Jumpers for link speed

LinkSpeedA	LinkSpeedB	Link 0 speed	Links 1-3 speed
low	low	10Mbits/s	10Mbits/s
low	high	10Mbits/s	20Mbits/s
high	low	20Mbits/s	10Mbits/s
high	high	20Mbits/s	20Mbits/s

Figure 26: Current TRAM link speeds

9.3 ConfigUp and ConfigDown

The IMS B014 allows the user to select whether the network configuration pipeline links (ConfigUp and ConfigDown) connect to the front connectors (P4 and P5) or to the rear connector (P2). Jumpers K1–K12 perform this selection, as detailed in figure 27.

So, for example if ConfigUp is to be connected via P4, install K7–K8 and K9–K10. If ConfigDown is to be connected via P2, install K1–K2 and K3–K4.

Jumper	Function
K1–K2,K4–K5	Installed—ConfigDown Goes to P2 (rear)
K2–K3,K5–K6	Installed—ConfigDown Goes to P5 (front)
K7–K8,K10–K11	Installed—ConfigUp Comes From P4 (front)
K8–K9,K11–K12	Installed—ConfigUp Comes From P2 (rear)

Figure 27: Jumpers for ConfigUp/ConfigDown

9.4 Services

The services signals (reset, analyse and error) can be switched via jumpers JP8–11 as detailed in figure 28. Figure 16 shows the services organisation.

If, for example, services for slot 0 are to come from P4 and services for slots 1–7 are to come from slot 0's subsystem pins and ServicesDown is to be propagated out P2 then install jumpers as follows—
Install JP9 to set the ServicesUp source, remove JP8 to set slot 0's services, remove JP11 to set the services for slots 1–7 and remove JP10 to select P2 for ServicesDown.

Jumper	Function
JP8	Installed—Services for Slot 0 come from VMEbus otherwise from ServicesUp.
JP9	Installed—ServicesUp come from P4 otherwise from P2.
JP10	Installed—ServicesDown go to P5 otherwise to P2.
JP11	Installed—Services for Slots 1–7 come from the same source as for Slot 0, otherwise from Slot 0's subsystem port.

Figure 28: Jumpers for Services Configuration

9.5 Secondary Link Switching

The idea of the secondary link switching jumpers is that they allow the user to configure the links which are potentially involved in the programming of the IMS C004 link switch devices.

Various links are switchable via the secondary link switching jumpers, as shown in figure 12. Each connection arc in figure 12 is lettered A–I. Each lettered connection has associated with it two jumpers in the secondary link switching jumper block. These are numbered JP12–JP29 and are labeled with the connection letters A–I on the board (see figure 24). When the two jumpers associated with a connection are installed, the connection is made.

Jumper	Function
JP12,JP13	Secondary link switching, link connection "A" made when installed.
JP14,JP15	Secondary link switching, link connection "B" made when installed.
JP16,JP17	Secondary link switching, link connection "C" made when installed.
JP18,JP19	Secondary link switching, link connection "D" made when installed.
JP20,JP21	Secondary link switching, link connection "E" made when installed.
JP22,JP23	Secondary link switching, link connection "F" made when installed.
JP24,JP25	Secondary link switching, link connection "G" made when installed.
JP26,JP27	Secondary link switching, link connection "H" made when installed.
JP28,JP29	Secondary link switching, link connection "I" made when installed.

Figure 29: Jumpers for Secondary Link Switching

10 Example Setup

To help first-time users here is a complete example setup for the IMS B014.

This configuration is typical for a system where the IMS B014 is used to host a transputer development system. A TRAM with subsystem capability is installed in slot 0.

Starting with no jumpers installed:

1 The VMEbus interface is enabled by installing JP2.

2 All links are set to 20Mbits/s speed by leaving JP1–JP7 empty.

3 Install JP8 to generate system services for slot 0 from the VMEbus subsystem.

4 JP9 is left empty since its state is not important.

5 JP10 is installed so ServicesDown go to the front panel connector (P5).

6 JP11 is left empty so services for slots 1–7 come from slot 0's subsystem.

7 Install K2–K3 and K5–K6 so that ConfigDown goes to the front panel connector (P5).

8 Install JP20 and JP21 so that slot 0, link 0 is connected to the VMEbus link.

9 Install JP28 and JP29 so that slot 0, link 1 is connected to link 1 of the nextwork configuration processor.

10 Finally, configure the VMEbus address switches to an address appropriate for your system.

A Glossary

Jumper These are the little plastic blocks, about an $\frac{1}{8}$" by $\frac{1}{8}$" by $\frac{1}{10}$" which are to be found in a small bag in the packaging and also plugged into the card. They have a metal insert which shorts two pin-posts together when the jumper is fitted to them. Fitting is accomplished by aligning the two holes in the jumper with the two posts which it is to be plugged into; next push the jumper fully home. Removal is usually best done with the aid of a small pair of pilers—it is possible to break your nails when removing a jumper. Note that jumpers sometimes fail "open" so if you are suspicious, it may be worth checking or replacing the jumpers on a non-functional board.

Card-cage The metal frame which holds a set of circuit cards (boards or PCBs) together. Usually has an associated backplane and power supply.

VITA The "VMEbus Industry Trade Association" is an organisation of VMEbus vendors and users which promotes the VMEbus and produces standards and literature.

IEC297 The mechanical standard which specifies the "eurocard" board sizes and corresponding card-cages.

RS-232 A "standard" asynchronous serial interface used for communication between computers and video terminals and peripherals.

UART Stands for Universal Asynchronous Receiver/Transmitter. This is a standard device, dating back to discrete-transistor circuits in DEC computers, which provides an interface between an asynchronous serial communcation line (such as RS-232) and a TTL parallel port.

Link Lockup INMOS serial links use an interlocked acknowledge protocol. This means that if for some reason an acknowledge packet is lost and does not arrive at the sending transputer or link adaptor then no more communication will occur. This is because the sending device is perpetually waiting for the acknowledge. The acknowlege packet can be lost due to noise (unlikely in a correctly designed system) or cables becoming disconnected.

The only way to resolve a link lockup is to reset the link ciruitry in the devices at both ends of the link.

B Handling

The unpacking note in the shipping carton will give details on how to unpack the IMS B014. Standard anti-static precautions should be observed since the IMS B014 contains MOS devices which are susceptable to static-discharge damage.

Some VMEbus compatible card-cages, notably SUN workstations, make use of the user-defined pins on connector P2. It is extremely important that the IMS B014 is not plugged into such a card-cage because permanant destruction of the IMS B014 and/or SUN can result. This restriction only applies to some slots in some kinds of SUN (and probably other card-cages) but users should always be aware of the danger. The solution required for the SUN is to use a special holding frame which isolates the P2 user-defined pins from the backplane. INMOS believes that the frame required has SUN part number 501-1220. Contact INMOS if you are unsure of the situation.

B.1 Installing the IMS B014 in a VMEbus card-cage

Before installing any board, first make sure that the power is turned off.

Inspect the VMEbus connectors for bent pins. Next align the corners of the board with the ends of the card-guides. The TRAM slot side of the board should be to the right. Slide the card home—if resistance is encountered the board is probably not aligned properly with the card guides. Now push firmly on the handles until the board is fully home. The front panel will fit against the card-cage. Lastly screw in the retaining bolts at the top and bottom of the front panel.

Remember that if you are using interrupts, the interrupt daisy-chain jumpers on the VMEbus should be configured correctly. Please consult the documentation for your VMEbus system to find out how to do this.

B.2 TRAM Installation

Although TRAMs can be plugged and un-plugged from the IMS B014 many times without contact wear, great care should be taken when fitting and removing TRAMS.

When fitting a TRAM to the IMS B014, inspect the TRAM for bent pins, then make sure that it is the correct way round (match up the yellow triangle on the TRAM with the yellow triangle on the IMS B014), then line up the pins with their sockets. When you are sure that the pins are aligned, gently push the TRAM home. Excessive force is not necessary and probably means that the pins are misaligned.

If vibration resistance is needed then you will need to bolt the TRAM in place using the M2.5 nylon bolts provided with the IMS B014.

TRAMs must be kept as flat as possible when they are removed from the IMS B014. Gently ease up one end of the TRAM a little, then ease up the other end by the same amount. Continue this until the TRAM comes free from its socket.

If you are fitting a TRAM in slot 0 to run a transputer development system it is likely that you need to fit the subsystem pins. Fitting a TRAM with subsystem pins is covered in section B.3.

B.3 Installation of a TRAM with subsystem pins

Some TRAMs have a "subsystem" port (see section 7). The signals for the subsystem port come through three extra socket pins (other than the standard TRAM pins) on the underside of the TRAM. When fitting a TRAM which uses its subsystem port to a motherboard such as the IMS B014, you must install the special pin-strip in the TRAM's subsystem pin sockets. This procedure is shown in figure 30. Some of these pin-strips are provided with the IMS B014. The TRAM is then plugged into the IMS B014 in the same way as described above in section B.2, taking care to align the subsystem pins with their respective holes in the IMS B014. These subsystem socket pins on the IMS B014 are shown in figure 31.

Note that subsystem pins can only be fitted to a TRAM which has subsystem capability and which is plugged into slot 0.

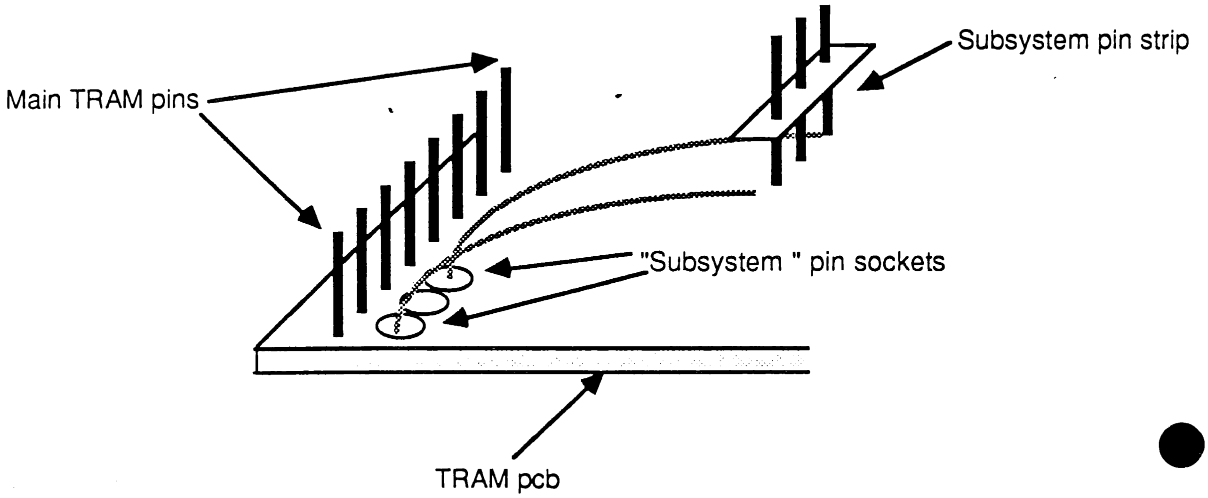


Figure 30: Subsystem pins installation

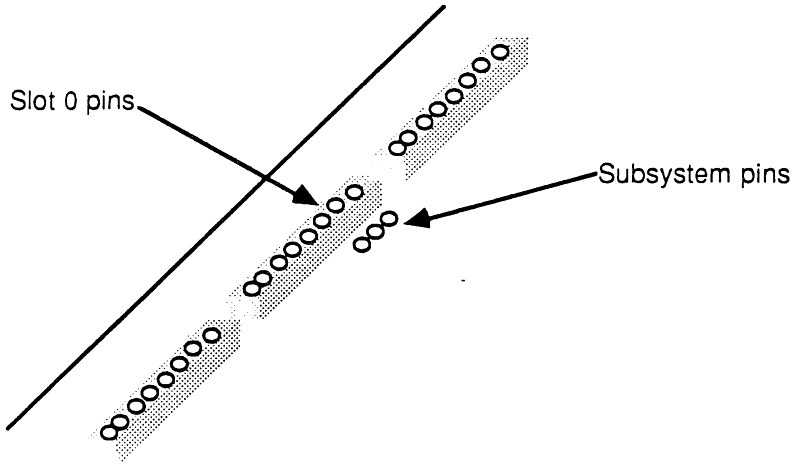


Figure 31: Subsystem pin sockets

C Mechanical and Thermal Details

The IMS B014 is designed to accord with DIN 41494 and IEC 297 standards. The board is nominally 160mm by 233.35mm. Figure 41 shows all board dimensions. Nominal board thickness is 1.6mm. The supplied front panel width is 4HP (approx 20mm). This is compatible with a board-to-board pitch in a card cage of 0.8". M2.5 fastening bolts are provided on the front panel, these mate with tapped holes in the card cage and fix the board securely. Front panel handles allow the board to be removed from the card cage (by un-screwing the retaining bolts and pulling hard on the handles). Note that the front panel is *required* when operating the IMS B014 in a card cage, both for mechanical rigidity and to give correct cooling air flow.

No components protrude more than 2.47mm below the surface of the board. To fit in a 0.8" pitch card-cage, no component should protrude more than 13.7mm above the surface of the board. This is a function of the TRAMs installed on the board.

Adequate cooling air flow must be provided to maintain the components on the board within their operating temperature. Air flow should run parallel to the board surface and parallel to the front panel. The amount of heat dissipated by the board depends upon the TRAMs fitted. With no TRAMs the IMS B014 dissipates no more than 5W. With TRAMs fitted, the maximum dissipation allowed (from 5v supply) is 18.75W when only using a J1 backplane and 37.5W when using a J1/J2 backplane[4]. It is essential that the user ensures that the maximum power dissipation is not exceeded. The cooling air flow required for a particular application will probably need to be determined empirically.

A single board operating in static air at room temperature (and not in a card-cage) will usually not need forced air cooling. This kind of set-up should only be used for lab and development work. High reliability is not to be expected from boards which are not provided with adequate cooling. Some TRAM types may require forced air cooling if they have high power dissipation components such as very fast PALs and fast A-D converters.

The two DIN 41612 (603-2-IEC-C096Mx-xxx) connectors (P1 and P2) have class 2 contact finish (1 micron gold) and are specified to give 400 mating cycles minimum.

If a TRAM with special I/O capabilities (such as the IMS B407 Ethernet TRAM or the IMS B409 Graphics TRAM) is fitted to the IMS B014, consideration will have to be given to the TRAM's cabling requirements. One way to handle cabling from TRAMs is to make a special wide front panel for the IMS B014. Connectors can be mounted through the panel and cabled via flying leads to the TRAMs. This gives a neat removable unit but takes up card-cage space. A drawing of the hole positions in the IMS B014's front panel is shown in figure 40.

C.1 Mating Connectors

Connectors to mate with the connectors on the IMS B014 are as follows—

P1/P2 — 96-way DIN41612 female connectors which are available from most connector manufacturers.

P3 — Molex 8981/70156 series (Disk Drive Power Connection System), for instance shell 891-4P (order number 15-24-4048) with crimps order number 02-08-1202.

P4/P5 — D-subminiature 37-way female Connectors, available in solder tail, crimp and solder bucket form from most connector manufacturers.

Jumpers — 0.1" pitch jumpers or programming shunts, again available from many manufacturers. Some types are very stiff-fitting and suit fixed configurations, others are slacker and are more useful when configurations are changed often.

Note that INMOS does not guarantee that these connector descriptions and part numbers are correct.

[4] J1 is the minimum VMEbus backplane and mates with P1 connectors on VMEbus boards. J2 mates with P2 connectors and is sometimes called a 32-bit backplane because it is needed for 32-bit VMEbus operations. Combined J1/J2 backplanes mate with both P1 and P2 and are needed for reliable operation of fast 32-bit VMEbus transfers.

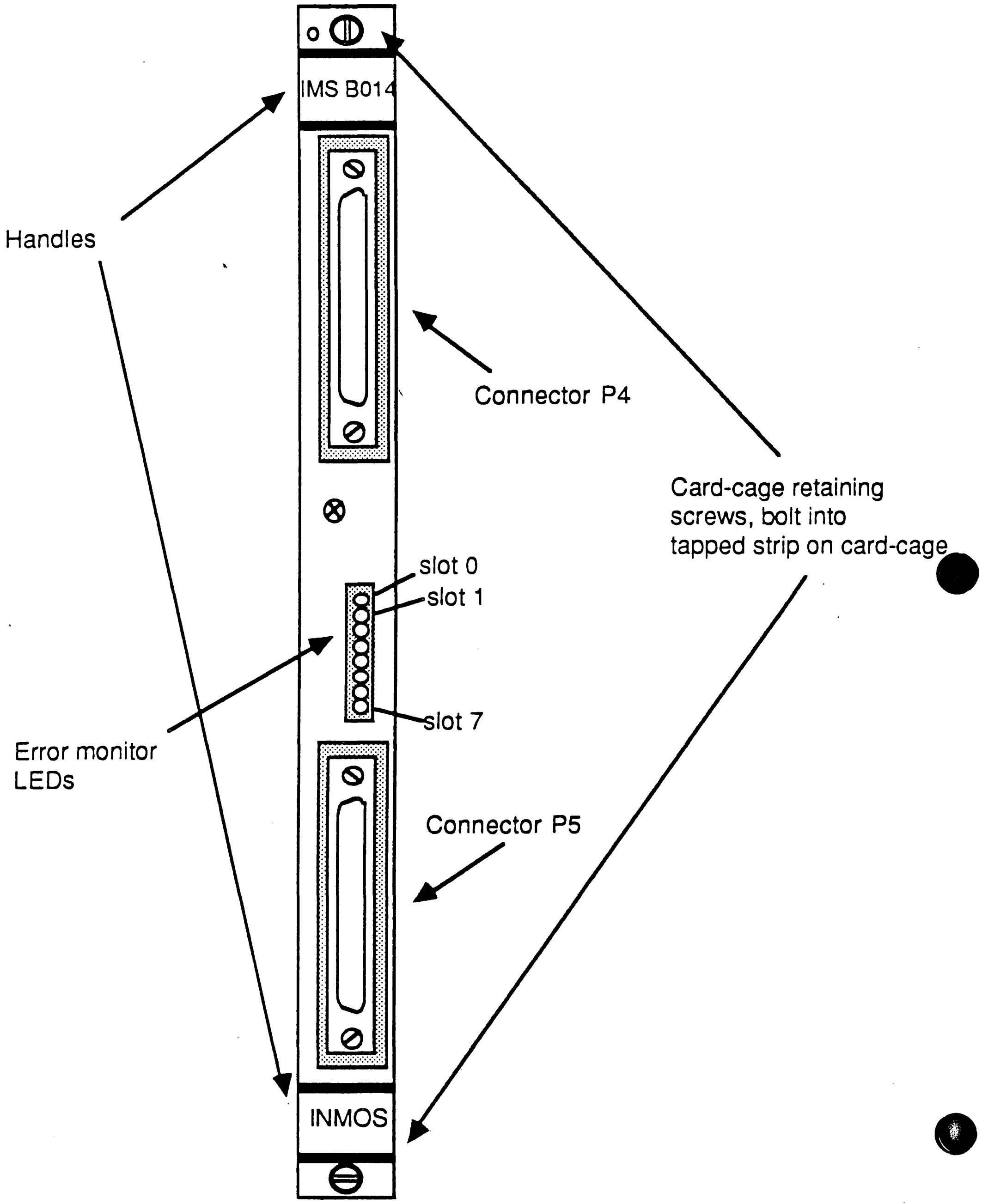


Figure 32: Front Panel Identification

	Operating	Storage
Temperature	0 to +40°C ambient air	−55 to +85°C
Relative Humidity	95% non condensing	95% non condensing
Thermal Shock	< 0.08°C/s	< 0.15°C/s
Altitude	−300 to +3000m	−300 to +16000m

Figure 33: Environmental Details

D Electrical Details

D.1 Power Supply

The IMS B014 requires power supply voltages in accordance with the VMEbus specification. That is, the +5V dc supply must be between 4.875V and 5.25V and have less than 50mV pk-pk noise and ripple between dc and 10MHz. The IMS B014 does not incorporate protection against incorrect power supplies. Major damage can result from operating the board outside its power supply range.

D.2 Board-to-Board Link Connections

The INMOS serial links provided on the IMS B014's connectors may be used to communicate directly with other boards in the same electrical environment (same DC supply, ground reference and low electromagnetic noise). However, these link connections *must* be made with transmission lines of characteristic impedence 100Ω.

The following are examples of valid connection methods—

1 A printed circuit trace of about 0.008" width over a ground plane on a multi-layer board.

2 Twisted pair ribbon cable.

3 INMOS link cables.

4 Twisted pair wire-wrap wire.

Single wires, PCB traces without ground planes and coaxial cable should *not* be used. It is recommended that link connections are not longer than 1m. Although long connections will function, enhanced reliability will be gained by making longer connections using buffered connections.

Services connections are not subject to the same conditions and can be carried on simple wires.

D.3 Non-Local Link Connections

Links may be used to communicate between TRAMs and transputers (and other boards containing transputers, even boards not manufactured by INMOS) wherever their location. However, when using links to communicate between TRAMs or transputers which are not powered from the same DC supply some special considerations need to be addressed—

1 Electrical noise may corrupt link data when ordinary INMOS link cables are used over long distances and between equipment which is powered from different supplies.

2 Since the INMOS cables and boards use single-ended, common ground signaling, earth loops between equipment can lead to signal corruption (not necessarily on the link signals but perhaps on the services signals causing spurious system resets).

3 The common ground in the cables can contravene electrical safety regulations which prevent earth signals being connected between equipment more than a certain distance appart. These regulations prevent, among other things, fire damage during electrical storms.

INMOS suggests that when such inter-box connections are required that diferential, double-ended balanced connections are used. These may be constructed from ordinary 26LS31 and 26LS32 driver/receiver devices for 10Mbits/s link operation and from ECL drivers/receivers for 20Mbits/s links. New driver/receiver devices designed for high-speed disk drives allow non-ECL 20Mbits/s operation. Signals should be receiver terminated to a floating ground and not to the ground at the receiving end. The services signals must be treated in a similar way although the timing constraints are not critical. Background information on this subject is to be found in [1], [4] and [5].

INMOS are currently composing, in conjunction with a number of other companies, a standard method for implementing differential link and services connections.

E Connector Pin Assignments

The connection assignments for P1, P2, P3, P4 and P5 are shown, as they appear when looking at the connectors. Note that this is not in strict alphanumeric order.

Pin	row c	row b	row a
1	D08	BBSY*(nc)	D00
2	D09	BCLR*(nc)	D01
3	D10	ACFAIL*(nc)	D02
4	D11	BG0IN*	D03
5	D12	BG0OUT*	D04
6	D13	BG1IN*	D05
7	D14	BG1OUT*	D06
8	D15	BG2IN*	D07
9	GND	BG2OUT*	GND
10	SYSFAIL*(nc)	BG3IN*	SYSCLK(nc)
11	BERR*	BG3OUT*	GND
12	SYSRESET*	BR0*(nc)	DS1*
13	LWORD*	BR1*(nc)	DS0*
14	AM5	BR2*(nc)	WRITE*
15	A23(nc)	BR3*(nc)	GND
16	A22(nc)	AM0	DTACK*
17	A21(nc)	AM1	GND
18	A20(nc)	AM2	AS*
19	A19(nc)	AM3	GND
20	A18(nc)	GND	IACK*
21	A17(nc)	SERCLK(nc)	IACKIN*
22	A16(nc)	SERDAT(nc)	IACKOUT*
23	A15	GND	AM4
24	A14	IRQ7*	A07
25	A13	IRQ6*	A06
26	A12	IRQ5*	A05
27	A11	IRQ4*	A04
28	A10	IRQ3*	A03
29	A09	IRQ2*	A02
30	A08	IRQ1*	A01
31	+12V	+5VSTDBY(nc)	−12V
32	+5V	+5V	+5V

Figure 34: Connector P1 pin assignments

Pin	row c	row b	row a
1	GND	VCC	GND
2	nc	GND	nc
3	BackConfigUpOut	RESERVED (nc)	BackConfigDownOut
4	BackConfigUpIn	A24 (nc)	BackConfigDownIn
5	GND	A25 (nc)	GND
6	GND	A26 (nc)	GND
7	nc	A27 (nc)	nc
8	P2Link0Out	A28 (nc)	P2Link1Out
9	P2Link0In	A29 (nc)	P2Link1In
10	GND	A30 (nc)	GND
11	GND	A31 (nc)	GND
12	nc	GND	nc
13	P2Link2Out	VCC	P2Link3Out
14	P2Link2In	D16 (nc)	P2Link3In
15	GND	D17 (nc)	GND
16	GND	D18 (nc)	GND
17	nc	D19 (nc)	nc
18	P2Link4Out	D20 (nc)	P2Link5Out
19	P2Link4In	D21 (nc)	P2Link5In
20	GND	D22 (nc)	GND
21	GND	D23 (nc)	GND
22	nc	GND	nc
23	P2Link6Out	D24 (nc)	P2Link7Out
24	P2Link6In	D25 (nc)	P2Link7In
25	GND	D26 (nc)	GND
26	GND	D27 (nc)	GND
27	nc	D28 (nc)	nc
28	notBackUpReset	D29 (nc)	notBackDownReset
29	notBackUpAnalyse	D30 (nc)	notBackDownAnalyse
30	notBackUpError	D31 (nc)	notBackDownError
31	GND	GND	GND
32	GND	VCC	GND

Figure 35: Connector P2 pin assignments

1	+12V
2	−12V
3	0V
4	+5V

Figure 36: Connector P3 pin assignments

Pin		Pin	
20	notUpReset	1	GND
21	notUpError	2	notUpAnalyse
22	P4Link0In	3	P4Link0Out
23	P4Link1In	4	P4Link1Out
24	P4Link2Out	5	GND
25	P4Link3Out	6	P4Link2In
26	P4Link4Out	7	P4Link3In
27	GND	8	P4Link4In
28	P4Link5In	9	P4Link5Out
29	P4Link6In	10	P4Link6Out
30	P4Link7In	11	P4Link7Out
31	FrontConfigUpOut	12	GND
32	ConnectorLinkOut	13	FrontConfigUpIn
33	nc	14	ConnectorLinkIn
34	nc	15	nc
35	nc	16	nc
36	nc	17	nc
37	nc	18	nc
		19	nc

Figure 37: Connector P4 pin assignments

Pin		Pin	
20	nc	1	GND
21	nc	2	nc
22	P5Link0In	3	P5Link0Out
23	P5Link1In	4	P5Link1Out
24	P4Link2Out	5	GND
25	P5Link3Out	6	P5Link2In
26	P5Link4Out	7	P5Link3In
27	GND	8	P5Link4In
28	P5Link5In	9	P5Link5Out
29	P5Link6In	10	P5Link6Out
30	P5Link7In	11	P5Link7Out
31	nc	12	GND
32	nc	13	nc
33	nc	14	nc
34	nc	15	nc
35	nc	16	nc
36	FrontConfigDownIn	17	FrontConfigDownOut
37	notDownAnalyse	18	notDownReset
		19	notDownError

Figure 38: Connector P5 pin assignments

F Cable types

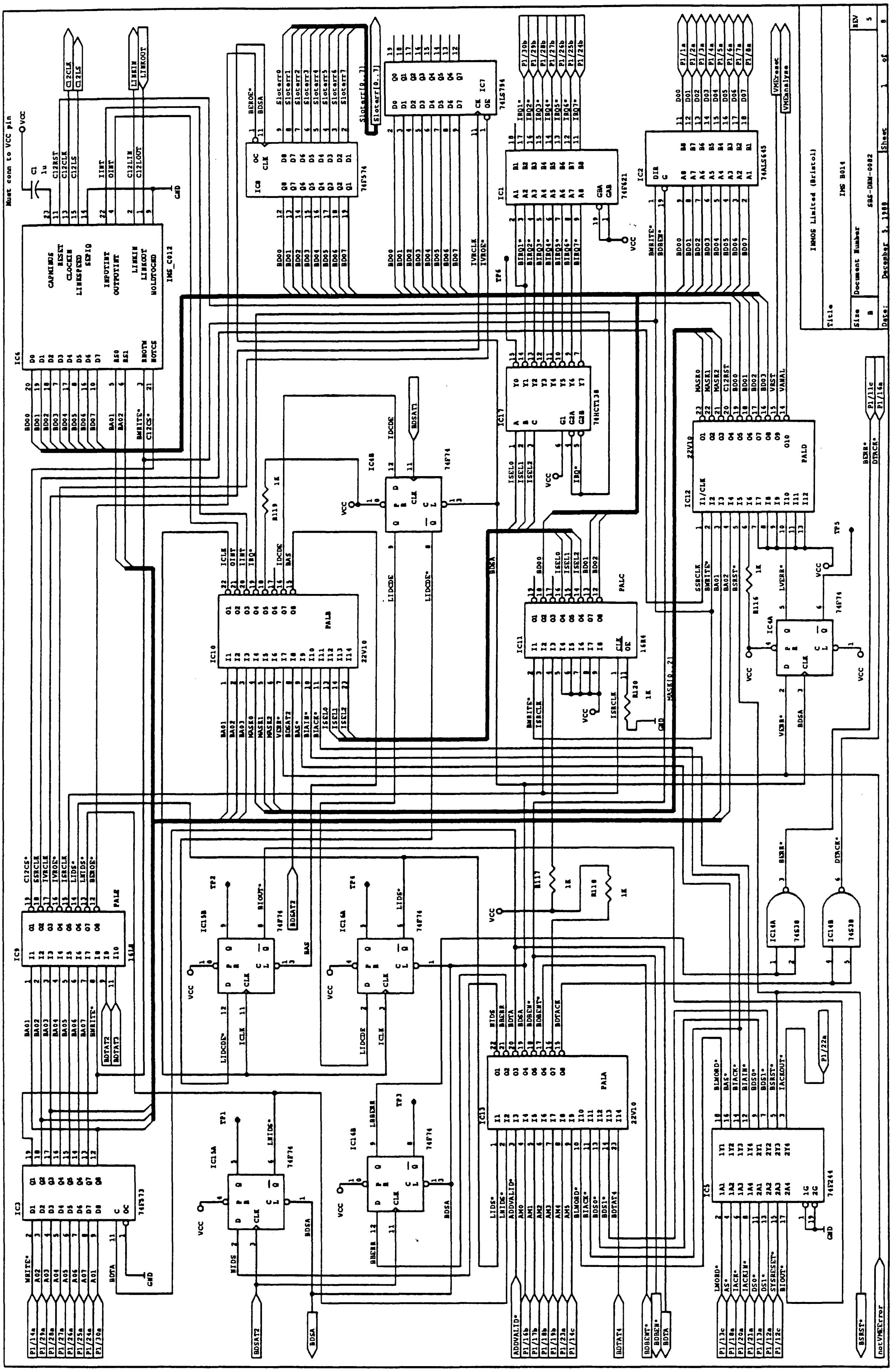
Figure 39 shows the reference numbers for INMOS cables relavent to the IMS B014. INMOS does not supply cables separately from boards but INMOS cables are available in most countries from a third-party vendor.

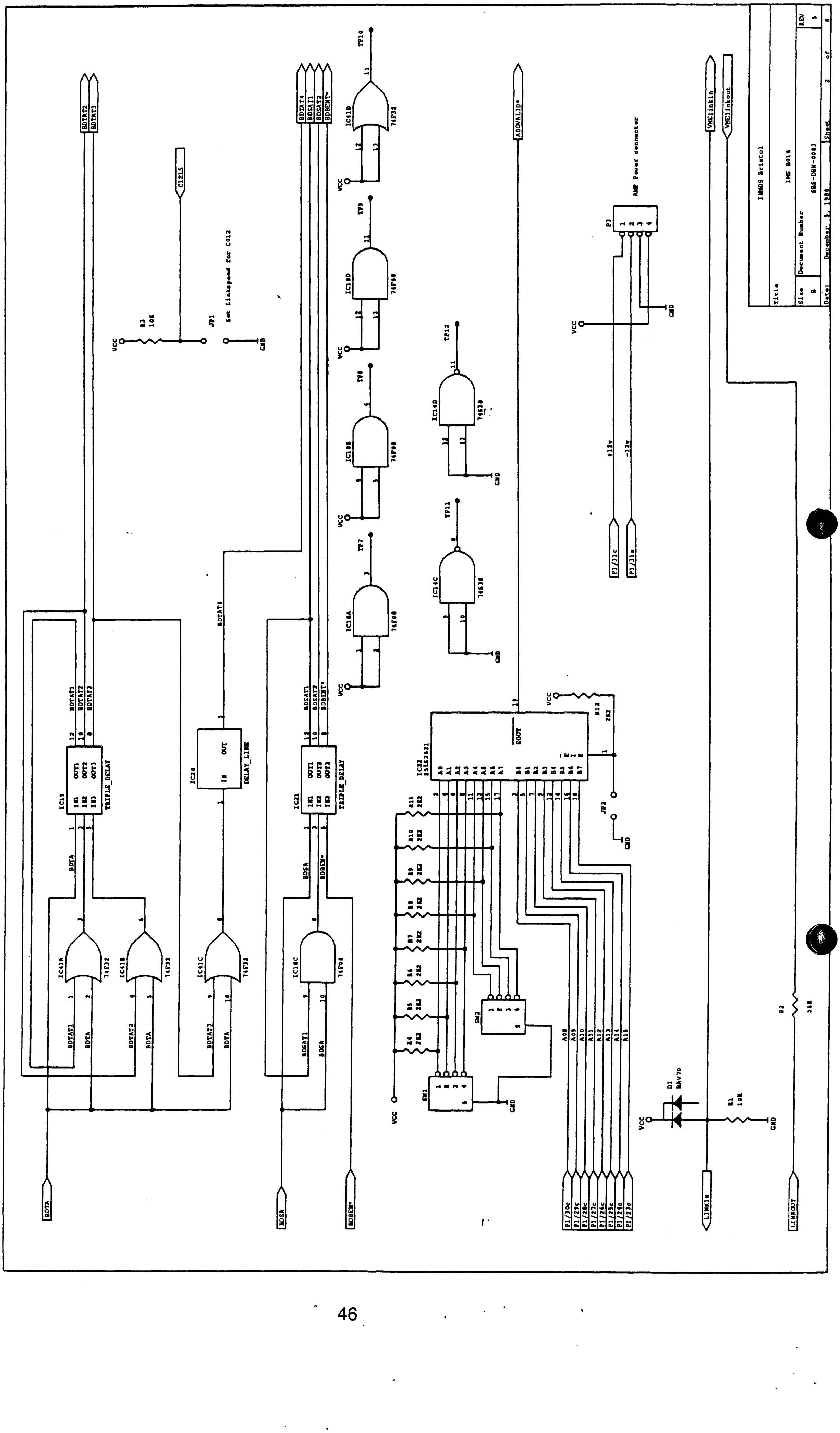
Reference Number	Description
221-CAB-0600	Short link cable (100mm)
221-CAB-0700	Standard link cable (500mm)
221-CAB-0800	Extra long link cable (2m)
221-CAB-0801	Long link cable (1m)
221-CAB-1000	Short services cable (100mm)
221-CAB-1100	Standard services cable (500mm)
221-CAB-1200	Extra long services cable (2m)
221-CAB-1201	Long services cable (1m)
221-SUB-0002	Breakout plug assembly
221-CAB-1601	Back-to-back DIN 41612 assembly
221-PIN-0008	Subsystem pin strip

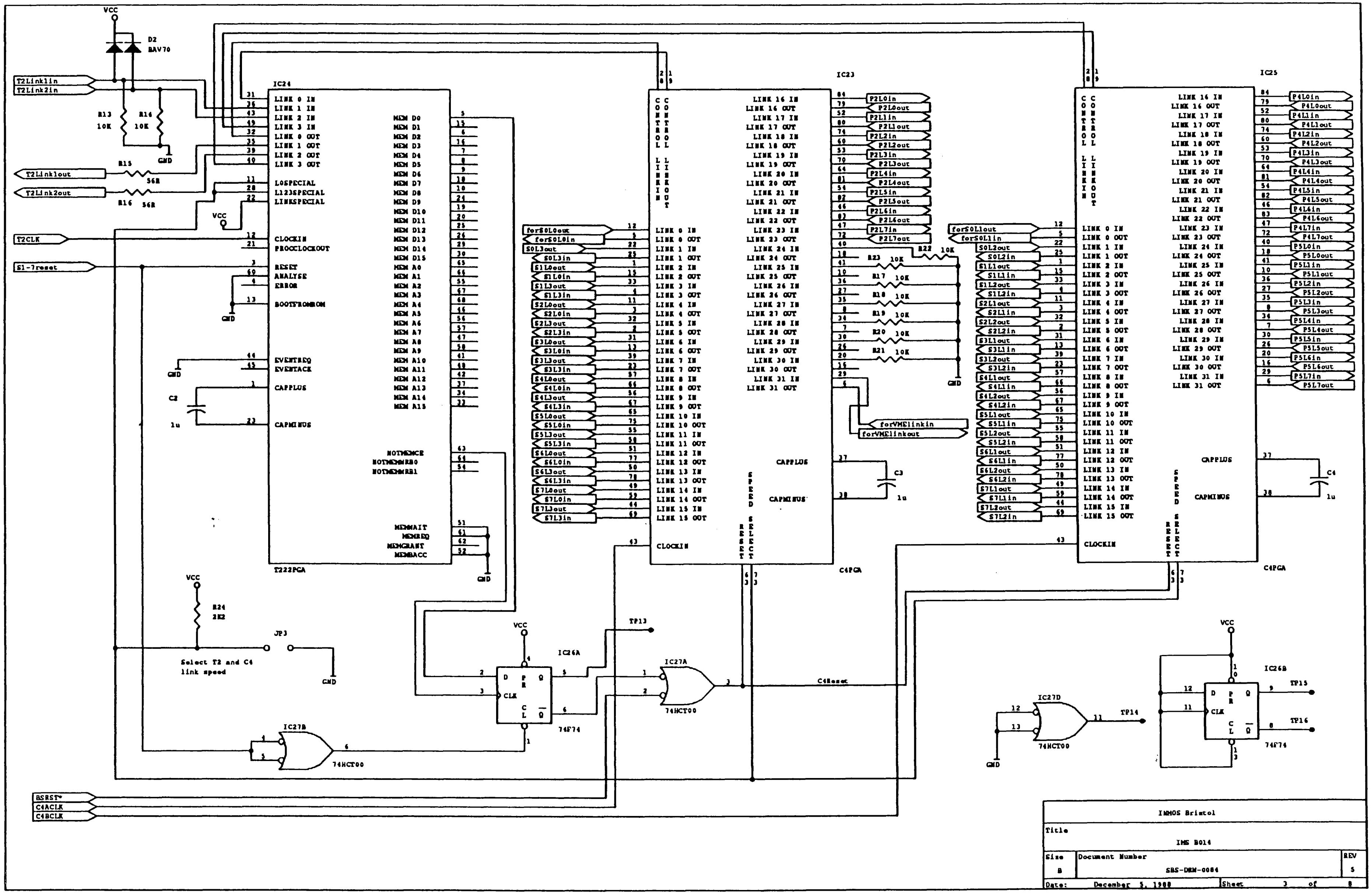
Figure 39: Cable Reference Numbers

G Schematic

The schematic data presented here is intended to supplement this manual in explaining the IMS B014's operation. It is not intended to be used by service personnel or as a basis for user modifications to the board (which will invalidate the warranty).


INMOS Limited (Bristol)
Title
IMS B014
Size
B
Document Number
SBS-DRW-0082
REV
5
Date: December 5, 1988
Sheet 1 of 8
Must conn to VCC pin
IC4 IMS_C012
IC8 74F574
IC7 74LS794
IC1 74F621
IC2 74ALS645
IC17 74HCT138
IC12 22V10 PALD
IC10 22V10 PALB
IC11 16R4 PALC
IC9 16L8 PAL2
IC3 74F573
IC13 22V10 PALA
IC5 74F244
IC14A 74S38
IC14B 74S38
IC15A 74F74
IC15B 74F74
IC16A 74F74
IC16B 74F74
IC4A 74F74
IC4B 74F74
VMEreset
VMEanalyse
notVMEError
BSRST*
Slotarr[0..7]

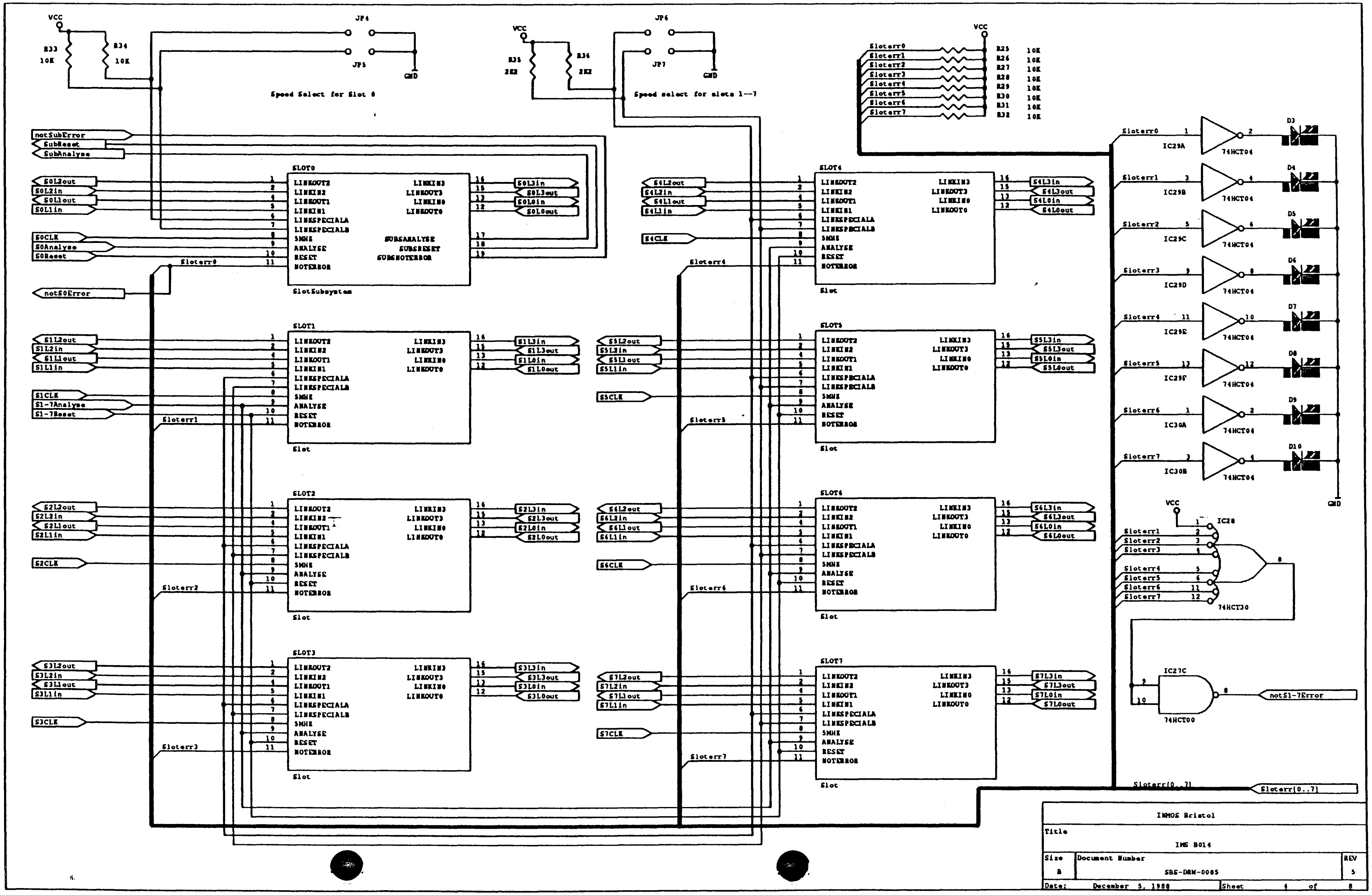

INMOS Bristol
Title
IMS B014
Size
B
Document Number
SRS-DRW-0083
REV
5
Date:
December 5, 1988
Sheet
2
of
8
BDTAT2
BDTAT3
BDTAT4
BDSAT1
BDSAT2
BDBEN*
C12LS
R3 10K
JP1
Set Linkspeed for C012
VCC
GND
IC41D
74F32
TP10
IC18D
74F08
TP9
IC18B
TP8
IC18A
TP7
IC14D
74F38
TP12
IC14C
TP11
ADDVALID*
VMElinkin
VMElinkout
P3
AMP Power connector
+12v
-12v
P1/31c
P1/31a
IC19
TRIPLE_DELAY
IN1
IN2
IN3
OUT1
OUT2
OUT3
IC20
DELAY_LINE
IN
OUT
IC21
IC22
25LS2521
A0
A1
A2
A3
A4
A5
A6
A7
B0
B1
B2
B3
B4
B5
B6
B7
EOUT
R12
2K2
JP2
IC41A
IC41B
IC41C
IC18C
BDTA
BDSA
BDBEN*
R4 2K2
R5 2K2
R6 2K2
R7 2K2
R8 2K2
R9 2K2
R10 2K2
R11 2K2
SW1
SW2
A08
A09
A10
A11
A12
A13
A14
A15
P1/30c
P1/29c
P1/28c
P1/27c
P1/26c
P1/25c
P1/24c
P1/23c
D1
BAV70
R1
10K
R2
56R
LINKIN
LINKOUT

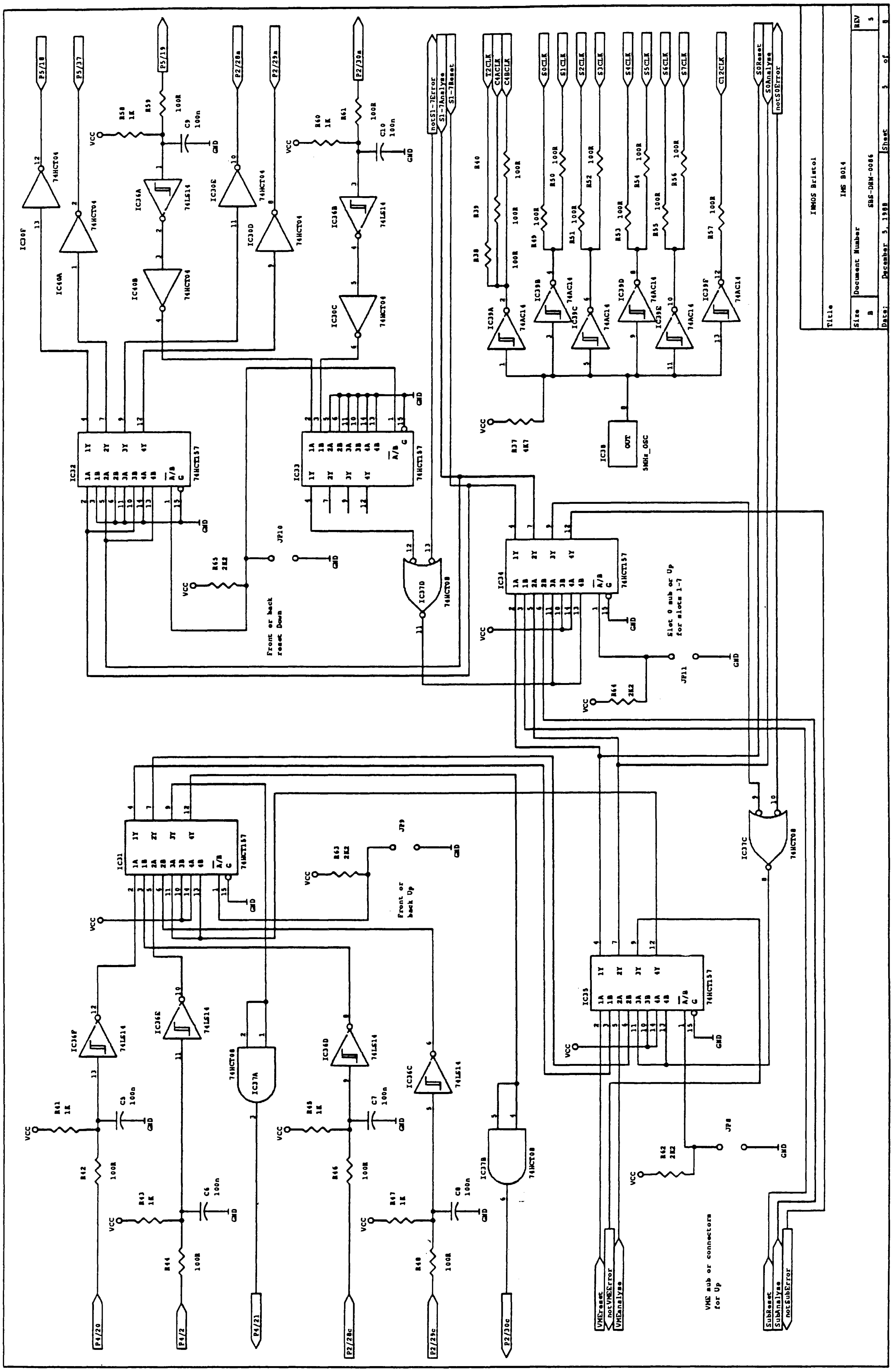


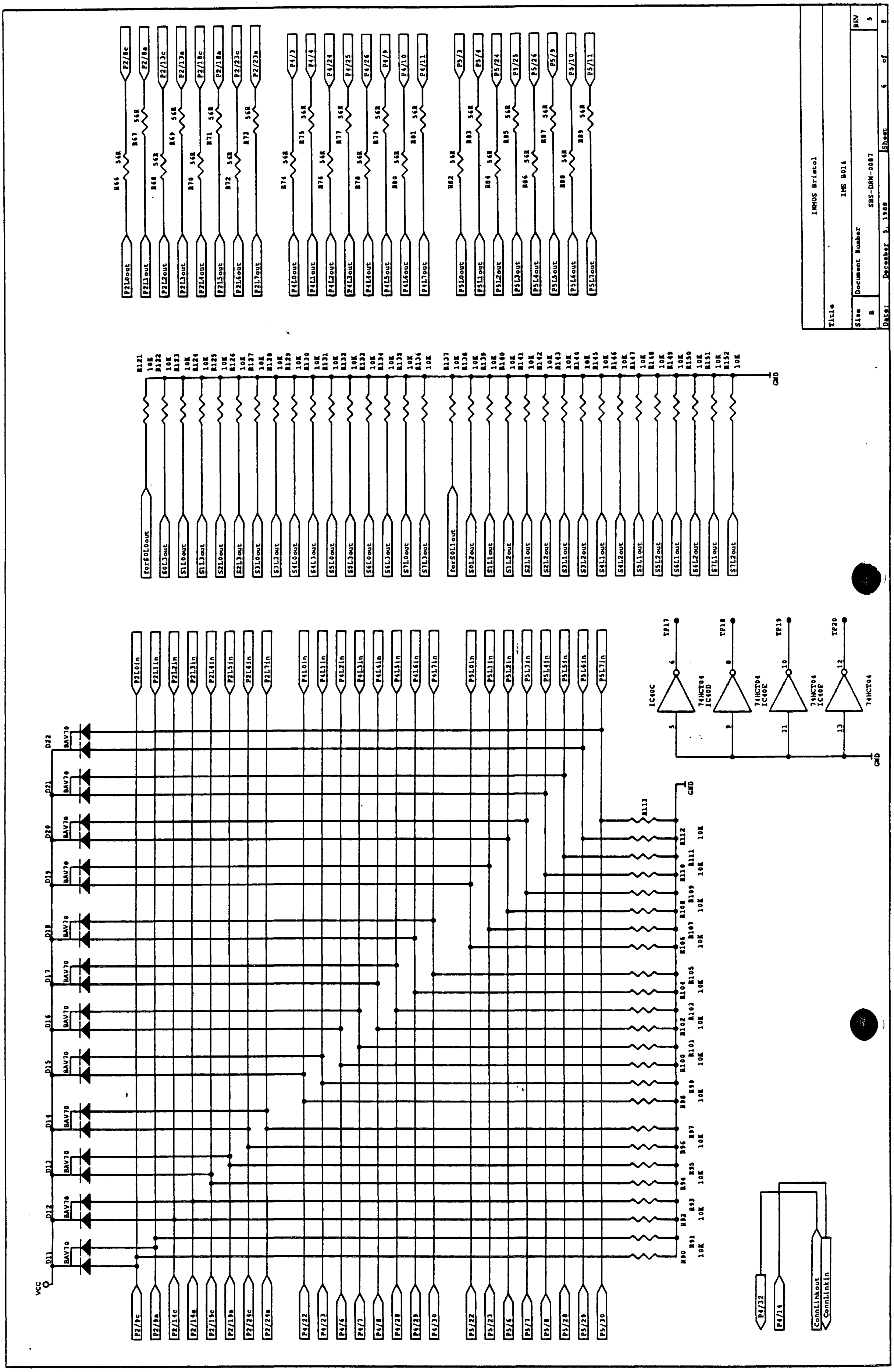
INMOS Bristol

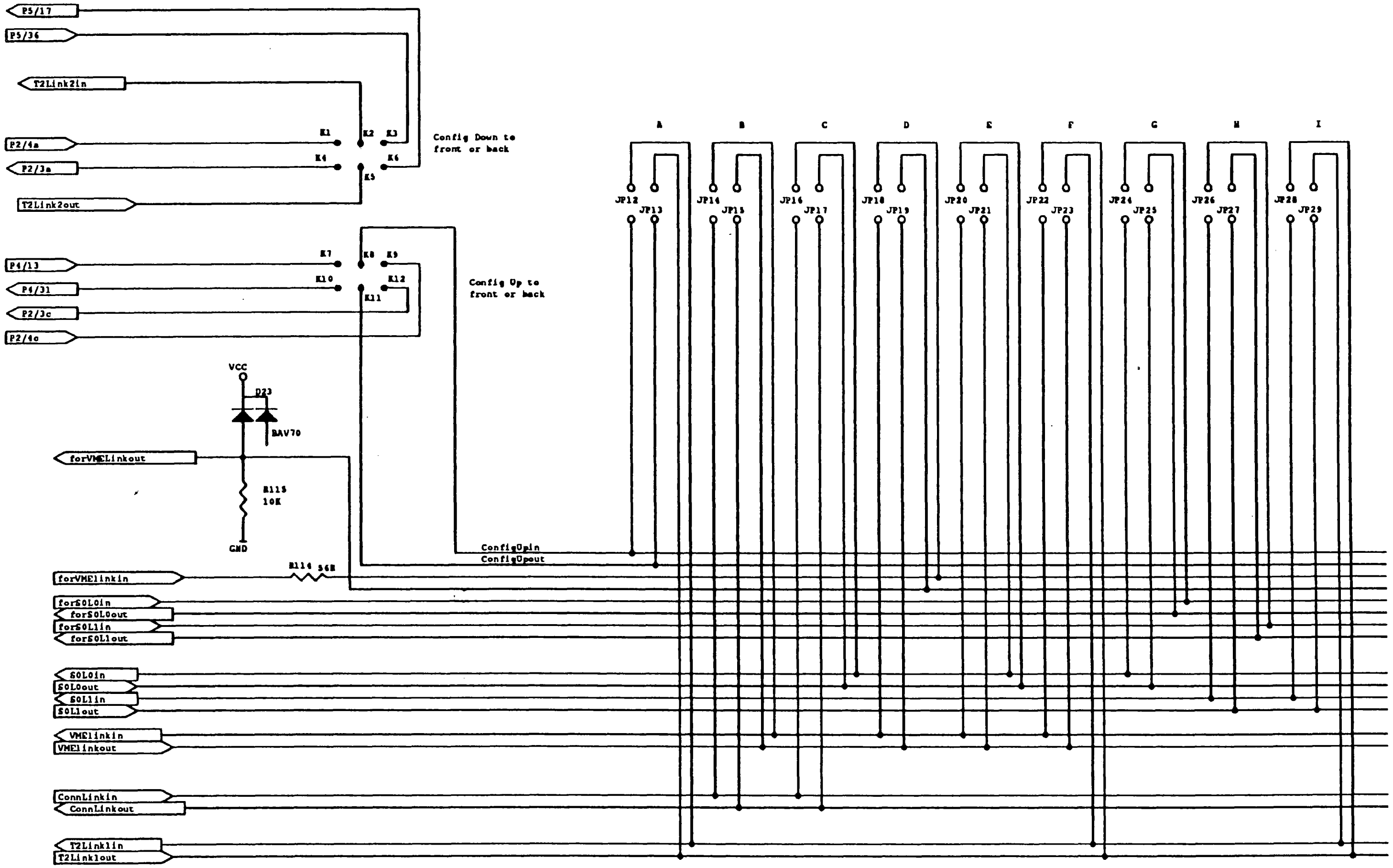
Title	IMS B014	
Size B	Document Number SRS-DRM-0084	REV 5
Date: December 5, 1988	Sheet 3 of 8	

DRAFT

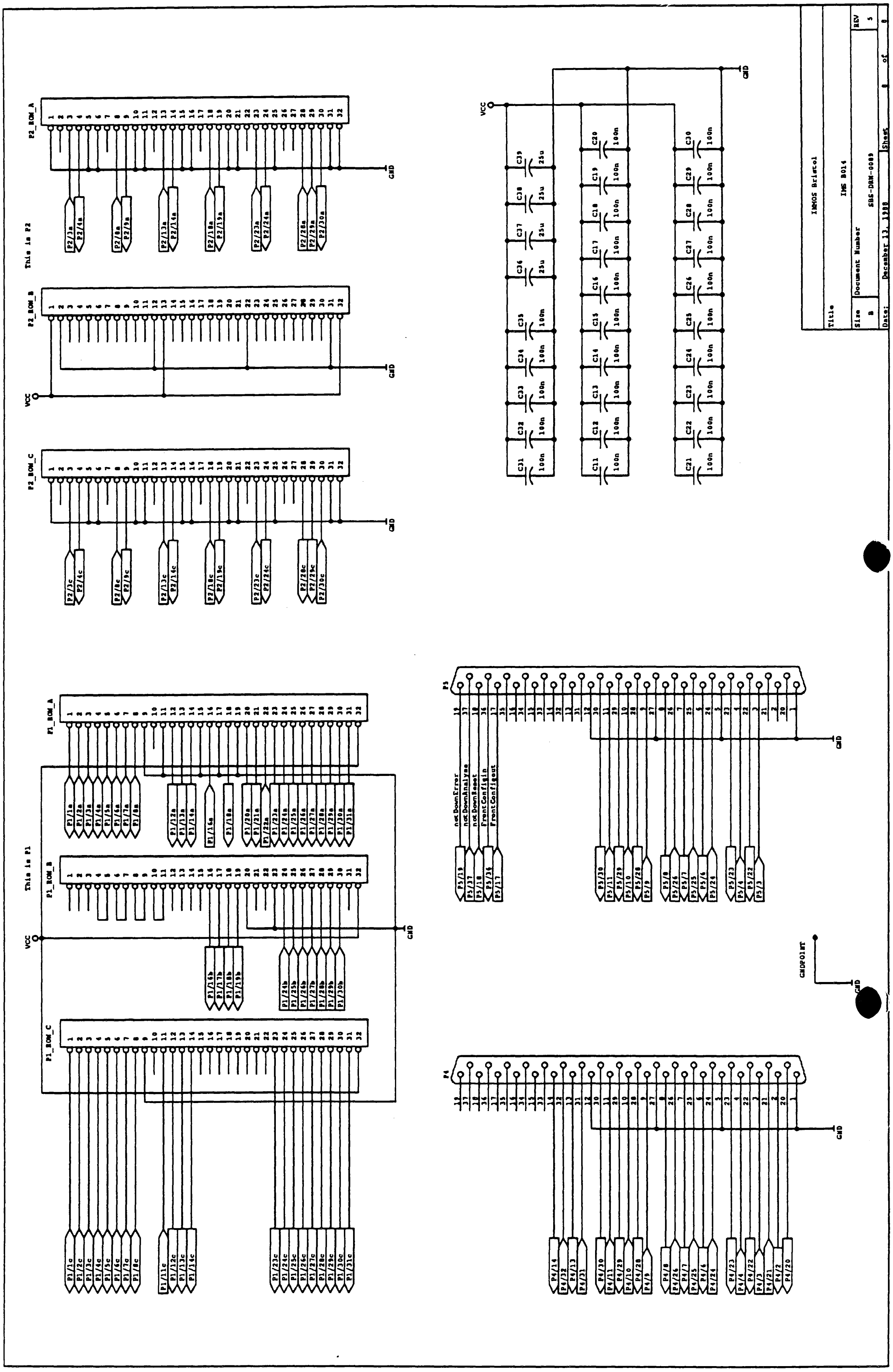

VCC
R33 10K
R34 10K
JP4
JP5
GND
Speed Select for Slot 0
VCC
R35 2K2
R36 2K2
JP6
JP7
GND
Speed select for slots 1--7
Sloterr0
Sloterr1
Sloterr2
Sloterr3
Sloterr4
Sloterr5
Sloterr6
Sloterr7
R25 10K
R26 10K
R27 10K
R28 10K
R29 10K
R30 10K
R31 10K
R32 10K
notSubError
SubReset
SubAnalyse
S0L2out
S0L2in
S0L1out
S0L1in
S0CLK
S0Analyse
S0Reset
notS0Error
SLOT0
LINKOUT2
LINKIN2
LINKOUT1
LINKIN1
LINKSPECIALA
LINKSPECIALB
SMHE
ANALYSE
RESET
NOTERROR
LINKIN3
LINKOUT3
LINKIN0
LINKOUT0
SUBSANALYSE
SUBSRESET
SUBSNOTERROR
SlotSubsystem
S0L3in
S0L3out
S0L0in
S0L0out
SLOT1
SLOT2
SLOT3
SLOT4
SLOT5
SLOT6
SLOT7
Slot
S1L2out
S1L2in
S1L1out
S1L1in
S1CLK
S1-7Analyse
S1-7Reset
S1L3in
S1L3out
S1L0in
S1L0out
S2L2out
S2L2in
S2L1out
S2L1in
S2CLK
S2L3in
S2L3out
S2L0in
S2L0out
S3L2out
S3L2in
S3L1out
S3L1in
S3CLK
S3L3in
S3L3out
S3L0in
S3L0out
S4L2out
S4L2in
S4L1out
S4L1in
S4CLK
S4L3in
S4L3out
S4L0in
S4L0out
S5L2out
S5L2in
S5L1out
S5L1in
S5CLK
S5L3in
S5L3out
S5L0in
S5L0out
S6L2out
S6L2in
S6L1out
S6L1in
S6CLK
S6L3in
S6L3out
S6L0in
S6L0out
S7L2out
S7L2in
S7L1out
S7L1in
S7CLK
S7L3in
S7L3out
S7L0in
S7L0out
IC29A
IC29B
IC29C
IC29D
IC29E
IC29F
IC30A
IC30B
74HCT04
D3
D4
D5
D6
D7
D8
D9
D10
GND
VCC
IC28
74HCT30
IC27C
74HCT00
notS1-7Error
Sloterr[0..7]
INMOS Bristol
Title
IMS B014
Size
B
Document Number
SBS-DMM-0085
REV
5
Date:
December 5, 1988
Sheet
4
of
8


INMOS Bristol
Title
IMS B014
Size
B
Document Number
SBS-DRN-0086
REV
5
Date:
December 5, 1988
Sheet
5
of
8
P5/18
P5/37
P5/19
P2/28a
P2/29a
P2/30a
notS1-7Error
S1-7Analyse
S1-7Reset
T2CLK
C4ACLK
C4BCLK
S0CLK
S1CLK
S2CLK
S3CLK
S4CLK
S5CLK
S6CLK
S7CLK
C12CLK
S0Reset
S0Analyse
notS0Error
IC30F
IC40A
IC36A
IC40B
IC30E
IC30D
IC36B
IC30C
74HCT04
74LS14
74HCT157
74HCT08
74AC14
IC32
IC33
IC34
IC31
IC35
IC37D
IC37C
IC37A
IC37B
IC36F
IC36E
IC36D
IC36C
IC39A
IC39B
IC39C
IC39D
IC39E
IC39F
IC38
OUT
5MHz_OSC
R37
4K7
R58
1K
R59
100R
C9
100n
R60
1K
R61
100R
C10
100n
R38
R39
R40
R49
R50
R51
R52
R53
R54
R55
R56
R57
100R
R65
2K2
JP10
Front or back
reset Down
JP11
R64
2K2
Slot 0 sub or Up
for slots 1-7
JP9
R63
2K2
Front or
back Up
R41
1K
C5
100n
R42
100R
R43
1K
C6
100n
R44
100R
R45
1K
C7
100n
R46
100R
R47
1K
C8
100n
R48
100R
JP8
R62
2K2
VME sub or connectors
for Up
VCC
GND
P4/20
P4/2
P4/21
P2/28c
P2/29c
P2/30c
VMEReset
notVMEError
VMEAnalyse
SubReset
SubAnalyse
notSubError


INMOS Bristol
Title
IMS B014
Size
B
Document Number
SBS-DRN-0087
REV
5
Date: December 5, 1988
Sheet 6 of 8
P2L0out
P2L1out
P2L2out
P2L3out
P2L4out
P2L5out
P2L6out
P2L7out
P4L0out
P4L1out
P4L2out
P4L3out
P4L4out
P4L5out
P4L6out
P4L7out
P5L0out
P5L1out
P5L2out
P5L3out
P5L4out
P5L5out
P5L6out
P5L7out
P2/8c
P2/8a
P2/13c
P2/13a
P2/18c
P2/18a
P2/23c
P2/23a
P4/3
P4/4
P4/24
P4/25
P4/26
P4/9
P4/10
P4/11
P5/3
P5/4
P5/24
P5/25
P5/26
P5/9
P5/10
P5/11
56R
forS0L0out
S0L3out
S1L0out
S1L3out
S2L0out
S2L3out
S3L0out
S3L3out
S4L0out
S4L3out
S5L0out
S5L3out
S6L0out
S6L3out
S7L0out
S7L3out
forS0L1out
S0L2out
S1L1out
S1L2out
S2L1out
S2L2out
S3L1out
S3L2out
S4L1out
S4L2out
S5L1out
S5L2out
S6L1out
S6L2out
S7L1out
S7L2out
10K
GND
P2L0in
P2L1in
P2L2in
P2L3in
P2L4in
P2L5in
P2L6in
P2L7in
P4L0in
P4L1in
P4L2in
P4L3in
P4L4in
P4L5in
P4L6in
P4L7in
P5L0in
P5L1in
P5L2in
P5L3in
P5L4in
P5L5in
P5L6in
P5L7in
TP17
TP18
TP19
TP20
IC40C
IC40D
IC40E
IC40F
74HCT04
VCC
D11
D12
D13
D14
D15
D16
D17
D18
D19
D20
D21
D22
BAV70
P2/9c
P2/9a
P2/14c
P2/14a
P2/19c
P2/19a
P2/24c
P2/24a
P4/22
P4/23
P4/6
P4/7
P4/8
P4/28
P4/29
P4/30
P5/22
P5/23
P5/6
P5/7
P5/8
P5/28
P5/29
P5/30
P4/32
P4/14
ConnLinkout
ConnLinkin



INMOS Bristol		
Title IMS B014		
Size B	Document Number SBS-DRW-0088	REV 5
Date: December 5, 1988	Sheet 7 of 8	


IMMOS Bristol
Title
IMS B014
Size
B
Document Number
SBS-DRN-0089
REV
5
Date:
December 13, 1988
Sheet
8
of
8
This is P2
P2_ROW_A
P2_ROW_B
P2_ROW_C
This is P1
P1_ROW_A
P1_ROW_B
P1_ROW_C
P3
P4
VCC
GND
GNDPOINT
notDownError
notDownAnalyse
notDownReset
FrontConfigIn
FrontConfigOut
100n
25u

DRAFT

H MMS2 Hardwire Fold

MMS2 is a network configuration program, available from INMOS, which can be used to configure the IMS B014's link switches. Here is the "hardwire fold" which MMS2 uses to gain knowlege about the IMS B014.

```
DEF B014
  {{{  definition
  {{{  sizes
  SIZES
    T2   1
    C4   2
    SLOT 8
    EDGE 27
  END
  }}}
  {{{  t2chain
  T2CHAIN
    T2 0, LINK 0  C4 0
    T2 0, LINK 3  C4 1
  END
  }}}
  {{{  hardwires
  HARDWIRE
  {{{  TRAMs to C004s
  {{{  TRAM 0
  C4 1, LINK  1 TO SLOT 0, LINK 2
  C4 0, LINK  1 TO SLOT 0, LINK 3
  }}}
  {{{  TRAM 1
  C4 0, LINK  2 TO SLOT 1, LINK 0
  C4 1, LINK  2 TO SLOT 1, LINK 1
  C4 1, LINK  3 TO SLOT 1, LINK 2
  C4 0, LINK  3 TO SLOT 1, LINK 3
  }}}
  {{{  TRAM 2
  C4 0, LINK  4 TO SLOT 2, LINK 0
  C4 1, LINK  4 TO SLOT 2, LINK 1
  C4 1, LINK  5 TO SLOT 2, LINK 2
  C4 0, LINK  5 TO SLOT 2, LINK 3
  }}}
  {{{  TRAM 3
  C4 0, LINK  6 TO SLOT 3, LINK 0
  C4 1, LINK  6 TO SLOT 3, LINK 1
  C4 1, LINK  7 TO SLOT 3, LINK 2
  C4 0, LINK  7 TO SLOT 3, LINK 3
  }}}
  {{{  TRAM 4
  C4 0, LINK  8 TO SLOT 4, LINK 0
  C4 1, LINK  8 TO SLOT 4, LINK 1
  C4 1, LINK  9 TO SLOT 4, LINK 2
  C4 0, LINK  9 TO SLOT 4, LINK 3
  }}}
  {{{  TRAM 5
  C4 0, LINK 10 TO SLOT 5, LINK 0
  C4 1, LINK 10 TO SLOT 5, LINK 1
  C4 1, LINK 11 TO SLOT 5, LINK 2
  C4 0, LINK 11 TO SLOT 5, LINK 3
  }}}
  {{{  TRAM 6
  C4 0, LINK 12 TO SLOT 6, LINK 0
  C4 1, LINK 12 TO SLOT 6, LINK 1
  C4 1, LINK 13 TO SLOT 6, LINK 2
  C4 0, LINK 13 TO SLOT 6, LINK 3
  }}}
  {{{  TRAM 7
  C4 0, LINK 14 TO SLOT 7, LINK 0
  C4 1, LINK 14 TO SLOT 7, LINK 1
  C4 1, LINK 15 TO SLOT 7, LINK 2
  C4 0, LINK 15 TO SLOT 7, LINK 3
  }}}
  }}}
  {{{  edges  0...7 (P2) to C004 A (0)
  C4 0, LINK 16 TO EDGE  0
  C4 0, LINK 17 TO EDGE  1
  C4 0, LINK 18 TO EDGE  2
  C4 0, LINK 19 TO EDGE  3
  C4 0, LINK 20 TO EDGE  4
  C4 0, LINK 21 TO EDGE  5
  C4 0, LINK 22 TO EDGE  6
  C4 0, LINK 23 TO EDGE  7
  }}}
  {{{  edges  8..15 (P4) to C004 B (1)
  C4 1, LINK 16 TO EDGE  8
  C4 1, LINK 17 TO EDGE  9
  C4 1, LINK 18 TO EDGE 10
  C4 1, LINK 19 TO EDGE 11
  C4 1, LINK 20 TO EDGE 12
  C4 1, LINK 21 TO EDGE 13
  C4 1, LINK 22 TO EDGE 14
  C4 1, LINK 23 TO EDGE 15
  }}}
  {{{  edges 16..23 (P5) to C004 B (1)
  C4 1, LINK 24 TO EDGE 16
  C4 1, LINK 25 TO EDGE 17
  C4 1, LINK 26 TO EDGE 18
  C4 1, LINK 27 TO EDGE 19
  C4 1, LINK 28 TO EDGE 20
  C4 1, LINK 29 TO EDGE 21
  C4 1, LINK 30 TO EDGE 22
  C4 1, LINK 31 TO EDGE 23
  }}}
  {{{  edges 24..26 (virtual edges) to C004s
  C4 0, LINK  0 TO EDGE 24
  C4 0, LINK 31 TO EDGE 25
  C4 1, LINK  0 TO EDGE 26
  }}}
  END
  }}}
  }}}
PIPE B014 END
```

I VMEbus capability

For easy description of VMEbus boards, the VMEbus specification defines a number of "capability" abreviations. The relavent capabilities for the IMS B014 are listed here—

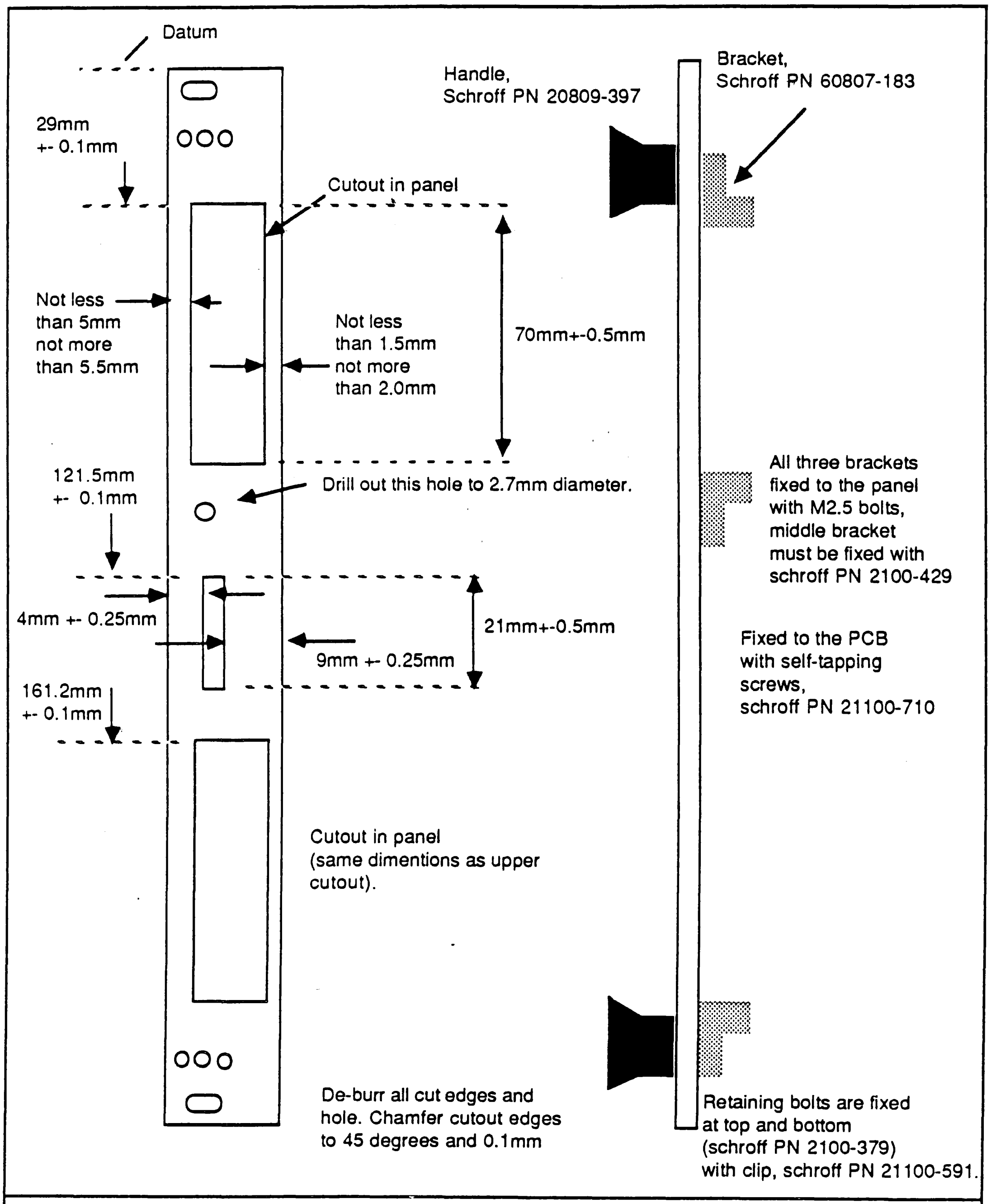
1 A16:D08(O) SLAVE

2 INT(1–7):D08(O) INTERRUPTER

3 6U high—double height board

VMEbus access time will be no longer than 170ns from DSA* to DTACK*.
The time to propagate a non-participating interrupt acknowlege cycle is no longer than 100ns from IACKIN* to IACKOUT*.
The time to respond to a participating interrupt acknowlege cycle will be no longer than 170ns from IACKIN* to DTACK*.

The IMS B014 propagates the BUSREQ* daisy-chain signals on the board, thus there is no need for jumpers on the backplane at the slot which the IMS B014 is plugged into.



Drawing number SBS-DRW-0103 Issue 2 INMOS Bristol
Title Front Panel assembly for IMS B014 Date 3 January 1989

Figure 40: Front Panel Mechanical Drawing

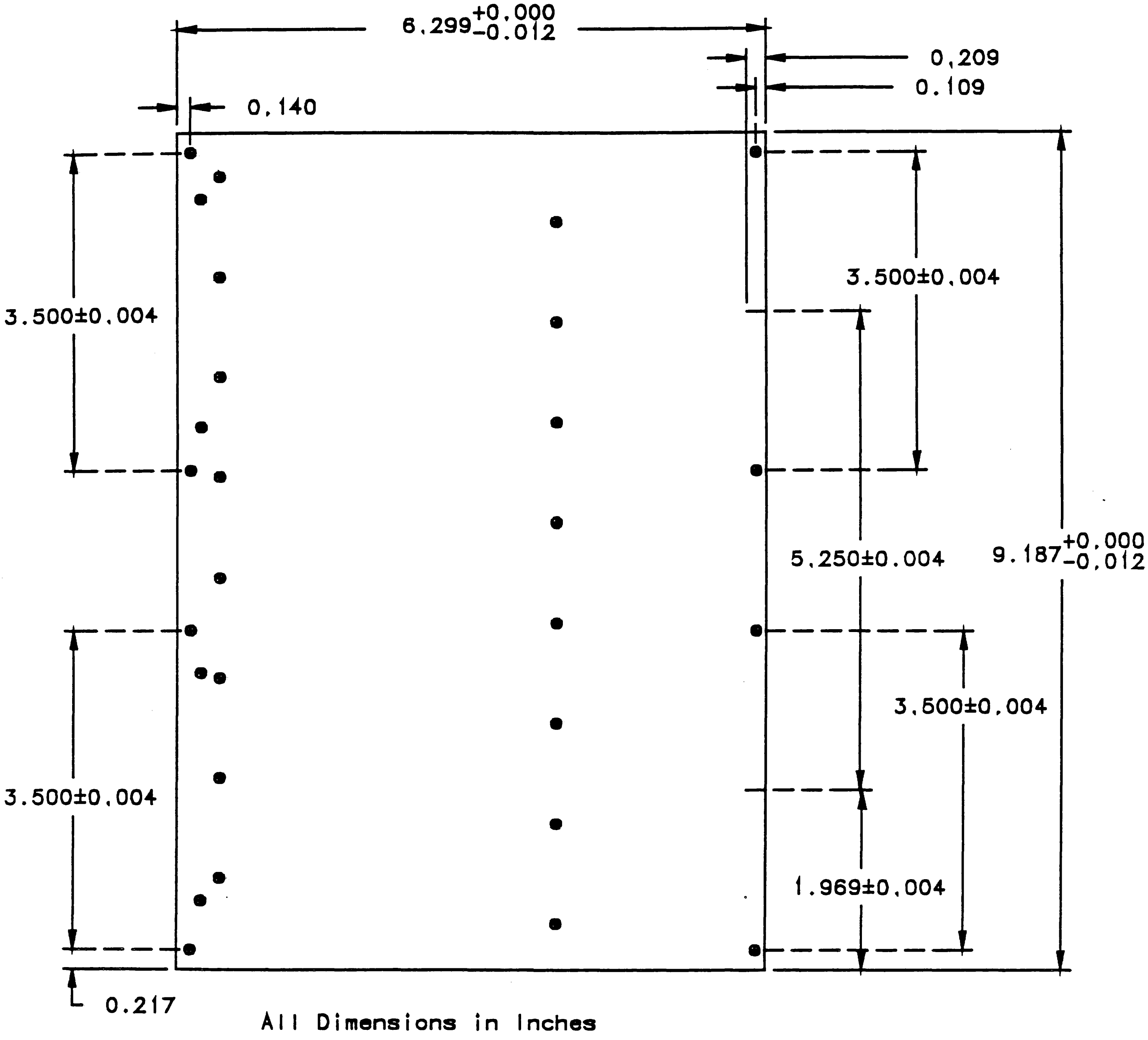


Figure 41: Board Outline Drawing

K IMS C004 Programming Sequences

Configuration Message	Function
[0] [input] [output]	Connects **input** to **output**
[1] [link1] [link2]	Connects **link1** to **link2**
[2] [output]	Enquires which input the **output** is connected to.
[3]	Must be sent at the end of every configuration sequence.
[4]	Resets the device.
[5] [output]	Output **output** is disconnected.
[6] [link1] [link2]	Disconnects the outputs **link1** and **link2.**

Figure 42: IMS C004 Programming Sequences

L Jumpers

Jumper	Function
JP1	Link speed for VMEbus link. Installed for 10Mbits/s, empty for 20Mbits/s.
JP2	Enable VMEbus interface when installed.
JP3	Link speed for C004s and T2. Installed for 10Mbits/s, empty for 20Mbits/s.
JP4	LinkSpeedA for TRAMslot 0. Installed for "low", empty for "high".
JP5	LinkSpeedB for TRAMslot 0. Installed for "low", empty for "high".
JP6	LinkSpeedA for TRAMslots 1–7. Installed for "low", empty for "high".
JP7	LinkSpeedB for TRAMslots 1–7. Installed for "low", empty for "high".
JP8	Installed—Services for Slot 0 come from VMEbus otherwise from ServicesUp.
JP9	Installed—ServicesUp come from P4 otherwise from P2.
JP10	Installed—ServicesDown go to P5 otherwise to P2.
JP11	Installed—Services for Slots 1–7 come from the same source as for Slot 0, otherwise from Slot 0's subsystem port.
K1–K2,K4–K5	Installed—ConfigDown Goes to P2 (rear)
K2–K3,K5–K6	Installed—ConfigDown Goes to P5 (front)
K7–K8,K10–K11	Installed—ConfigUp Comes From P4 (front)
K8–K9,K11–K12	Installed—ConfigUp Comes From P2 (rear)
JP12,JP13	Secondary link switching, link connection "A" made when installed.
JP14,JP15	Secondary link switching, link connection "B" made when installed.
JP16,JP17	Secondary link switching, link connection "C" made when installed.
JP18,JP19	Secondary link switching, link connection "D" made when installed.
JP20,JP21	Secondary link switching, link connection "E" made when installed.
JP22,JP23	Secondary link switching, link connection "F" made when installed.
JP24,JP25	Secondary link switching, link connection "G" made when installed.
JP26,JP27	Secondary link switching, link connection "H" made when installed.
JP28,JP29	Secondary link switching, link connection "I" made when installed.

Figure 43: List of all Jumpers

M Memory Map

Address	Register
#xx01	Link adaptor input data register (read only)
#xx03	Link adaptor ouptut data register (write only)
#xx05	Link adaptor input status register
#xx07	Link adaptor output status register
#xx09	Subsystem Reset/Error register
#xx0B	Subsystem Analyse register
#xx0D	Interrupt Enable Register
#xx0F	Interrupt Level register
#xx11	Interrrupt status/ID register
#xx13	TRAM error register

Figure 44: Board Memory Map

References

[1] Harold S. Stone, *Microcomputer Interfacing*
Addison-Wesley, 1982.

[2] INMOS, *Transputer Reference Manual*
Prentice Hall, 1988.

[3] Paul Walker, *INMOS Technical Note 29, Dual Inline Transputer Modules (TRAMs)*
INMOS, 1987.

[4] Trevor Watson and Michel Rygol, *INMOS Technical Note 18— Connecting INMOS Links*
INMOS, 1987.

[5] William R. Blood, Jr., *MECL System Design Handbook*
Motorola Inc., 1983.

[6] VITA, *VMEbus Specification Manual Revision C.1*
Printex Publishing, Inc, 1985.