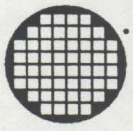


IMS B408 and IMS B409
User Manual

Contents

1 Introduction

The IMS B408 and IMS B409 are two transputer modules (TRAMs[1]). They are designed to be used together to build distributed computer graphics systems: that is, systems where the drawing task is shared by several processors. A system consists of one IMS B409 and one or more IMS B408s. They are connected together by means of a *pixel bus*. The IMS B409 generates the system timing and converts digital pixel data into analogue RGB signals suitable for driving a video monitor. Each IMS B408 has an IMS T800 floating point transputer and a block of dual ported memory. The IMS T800 draws into the memory via one port: the other port is connected to the pixel bus. The pixel bus distributes timing signals from the IMS B409 to the IMS B408s and combines the pixel data output from the IMS B408s.

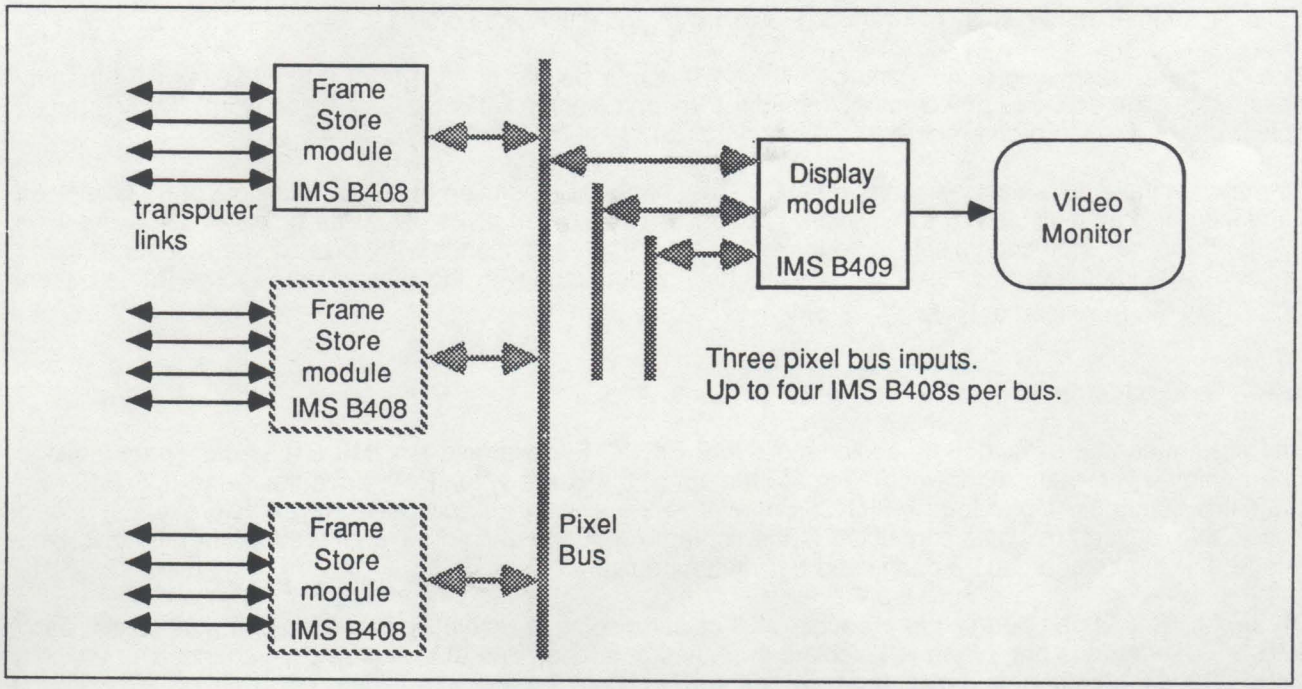


Figure 1.1 Distributed graphics system architecture

1.1 The Distributed graphics system

The graphics system is designed around the *pixel bus*. This is a 32-bit wide bus used to move pixel data at up to 100 MBytes/sec: the minimum bus cycle time is 40ns. It also distributes system timing signals.

A graphics system contains one display module (IMS B409). The IMS B409 has three pixel channels each of which can consume a stream of pixel data from a pixel bus and generate RGB output suitable for colour video monitors. It generates the synchronisation signals required by the monitor and also those which are distributed to the rest of the system via the pixel bus. The IMS B409 achieves pixel rates up to 64 MHz.

A system contains one or more frame store modules (IMS B408). A frame store module contains an IMS T800 32-bit transputer with on-chip FPU and 1.25 Mbytes of RAM which is dual ported to the pixel bus. This allows the transputer to draw into the RAM while a stream of pixel data flows through the dual port to the pixel bus. This pixel stream is synchronised to the timing signals distributed on the pixel bus and operates autonomously of the transputer.

[1] Further information on the INMOS TRAM standard can be found in appendix A and in [2]

1.2 Capabilities

A single IMS B408 with an IMS B409 supports a variety of display resolutions, numbers of frame buffers, number of displayable colours. A selection of possibilities is

- a single frame buffer of 1024 × 768 pixels with 256 displayed colours.
- a single frame buffer of 640 × 480 pixels with 262144 displayed colours.
- up to 4 frame buffers of 640 × 480 pixels with 256 displayed colours.
- an interlaced display of 512 × 575 pixels with 256 displayed colours.

The number of displayed colours can be either 256 from a palette of 262144 or the full 262144. In the first case, each pixel occupies one byte of store, in the second, each pixel occupies a 32 bit word. Thus, reduced colour displays take up one quarter the memory space of full colour displays.

Display resolution is arbitrary with the restriction that horiontal resolution must be a multiple of 64 pixels. An interlaced display is limited to a maximum horizontal resolution of 1024 byte-wide pixels or 256 word-wide pixels. For a non-interlaced display, the maximum horizontal resolution is 8192 pixels. The number of frame buffers is dependent on the resolution and the pixel width: the lower the resolution the more frame buffers are available.

1.3 Expansion

The system can be expanded by adding more IMS B408s. For example two IMS B408s can share the load in an animation system by drawing alternate frames at the same time. Both drive the same IMS B409 via the same pixel bus. Up to four IMS B408s may drive the same pixel bus, as shown in figure 1.2. It is also possible for several modules connected to the same pixel bus to each draw a different part of the displayed image: i.e. the screen can be partitioned between modules.

Three IMS B408s can be used to produce a full colour display at resolutions up to 1024 × 768 pixels: each IMS B408 generates one of the RGB colour primaries and feeds one of the three pixel channels of a single IMS B409 via three separate pixel busses, as shown in figure 1.2.

These expansion methods can be combined. Drawing bandwidth and processing power in an expanded system is proportional to the number of frame store modules (IMS B408s) in the system.

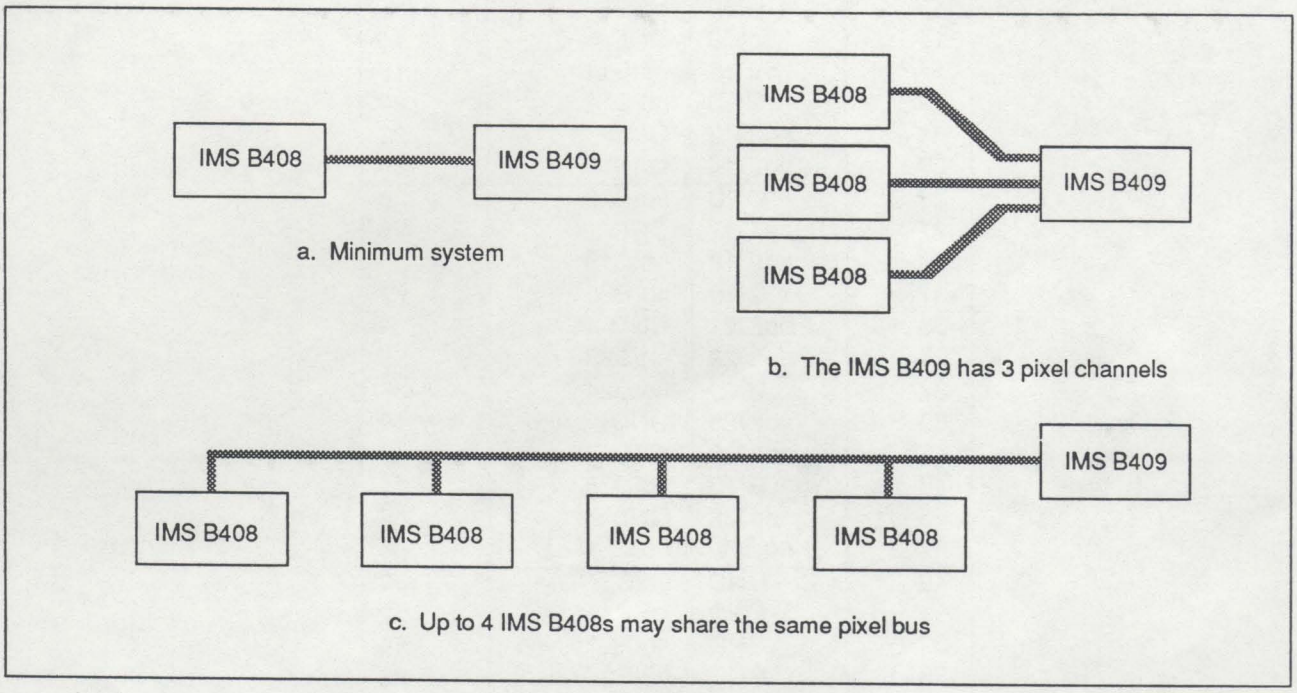


Figure 1.2 Basic and expanded systems

2 The Pixel Bus

The pixel bus is a synchronous bus which distributes system timing and carries 32-bit data at up to 25MHz. One end of the pixel bus should always end at a single pixel bus connector on an IMS B409, the other should always be terminated with the termination module supplied with the pixel bus cable.

2.1 Signals

Pin No.	Pin Name	Pin Name	Pin No.
1	GND	notD0	2
3	notD1	GND	4
5	notD2	notD3	6
7	GND	notD4	8
9	notD5	GND	10
11	notD6	notD7	12
13	GND	notD8	14
15	notD9	GND	16
17	notD10	notD11	18
19	GND	notD12	20
21	notD13	GND	22
23	notD14	notD15	24
25	GND	notD16	26
27	notD17	GND	28
29	notD18	notD19	30
31	GND	notD20	32
33	notD21	GND	34
35	notD22	notD23	36
37	GND	notD24	38
39	notD25	GND	40
41	notD26	notD27	42
43	GND	notD28	44
45	notD29	GND	46
47	notD30	notD31	48
49	GND	notSEQclk	50
51	GND	notRAMclk	52
53	GND	notFieldSync	54
55	GND	notEarlyBlank	56
57	GND	notEvenField	58
59	GND	SysReady	60

Table 2.1 Pixel Bus pin designations

notD0-notD31 Inverted pixel data. These wires are driven by the IMS B408 and monitored by the IMS B409. They are driven by 64mA *open collector* buffers and should be terminated by 200Ω to 3V at both ends of the bus: a suitable termination network is 470Ω to **GND** and 330Ω to **Vcc**. The IMS B409 has this termination built-in on these lines and, for the other end of the pixel bus cable, it is provided by the termination module supplied with the cable. This termination module should always be fitted.

notSEQclk This is the system timing reference and is driven continuously by the IMS B409. It has a maximum frequency of 25MHz and is normally a sub-multiple of the pixel clock used internally on the IMS B409. Data transfers and control strobe transitions are referenced to the falling edges of **notSEQclk**.

notRAMclk An output clock from the IMS B409 of the same frequency and phase as **notSEQclk** but gated so that it does not run during display blanking. It is used to clock pixel data from the IMS B408. In the off state it is high.

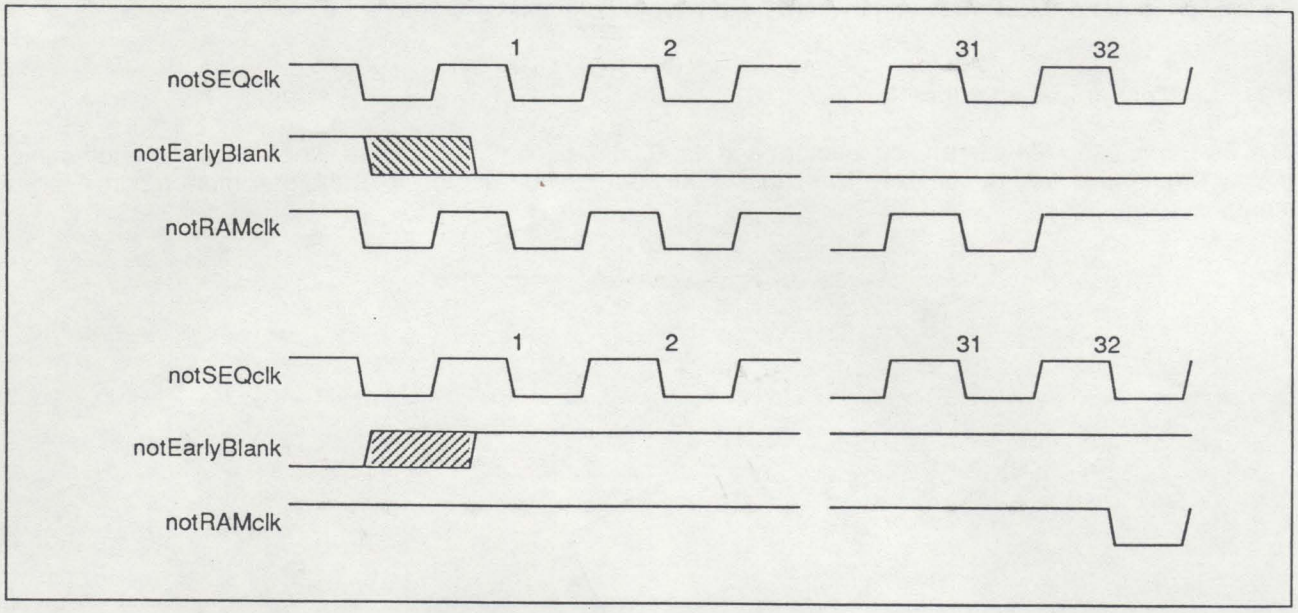


Figure 2.1 Relationship between notEarlyBlank and notRAMclk

notEarlyBlank Driven by the IMS B409. A time-advanced version of the display blanking signal which provides an early warning of when **notRAMclk** will be turned on and off. When **notEarlyBlank** goes low, there will be 31 falling edges of **notRAMclk** before **notRAMclk** is turned off. When **notEarlyBlank** goes high, there will be 31 falling edges of **notSEQclk** before **notRAMclk** is turned on. The timing relationship is shown in figure 2.1.

notFieldSync Driven by the IMS B409. Low for a minimum of 1 line-time during field flyback (vertical blanking) to indicate the start of a new display field. The timing of **notFieldSync** relative to **notSEQclk** is not specified.

notEvenField Driven by the IMS B409. Low during even fields when an interlaced display is being generated: high during odd fields, low during even fields. This signal may only change when **notFieldSync** is low.

SysReady This signal is used as a syncronisation mechanism between multiple IMS B408s. It is neither driven nor monitored by the IMS B409. Each IMS B408 drives this signal from a writeable latch with an open-collector pull-down and a 4k7Ω pull-up resistor. Each IMS B408 can also read the state of this signal. Only when all IMS B408s have written 1 to their ready latch, is this signal pulled high. This condition can be used in combination with **notFieldSync** to interrupt the IMS T800 on the IMS B408. Thus it provides a means of synchronising frame flipping between multiple IMS B408s in a system.

This signal must not be terminated at either end of the pixel bus.

2.2 Electrical specification

The pixel data signals **notD0–notD31** should be terminated with 470Ω to **GND** and 330Ω to **Vcc** at both ends of the cable. This is equivalent to 200Ω to 3V at both ends of the cable. This is required to establish a TTL high logic level on the bus, and to prevent signal reflections from un-terminated cable ends. This termination is provided at one end of the pixel bus by the IMS B409 pixel data inputs, and at the other end of the bus by the termination module supplied with the IMS B409.

The clock and control signals: **notSEQclk**, **notRAMclk**, **notEarlyBlank**, **notFieldSync**, and **notEvenField**; should be terminated at the end of the pixel bus most distant from the driving module (IMS B409) with 100Ω to **GND**. This termination is provided by the termination module supplied with the IMS B409.

All IMS B408s and IMS B409s in a system must have a common **GND** reference.

2.3 Mechanical specification

The pixel bus uses 60-way IDC connectors and flat 0.05in. pitch ribbon cable. Other kinds of ribbon cable, e.g. twist-and-flat, are not suitable. The cable should be as short as possible: the maximum recommended length is 30cm.

3 The IMS B408

The IMS B408 implements the drawing and image storage parts of a medium to high performance graphics system. It incorporates a powerful 32-bit microprocessor with on-chip FPU, 1 Mbyte of workspace RAM and 1.25 Mbyte of display RAM accessable to the processor and dual ported to the Pixel Port. The pixel port is capable of sustaining continuous data transmission at up to 100 Mbytes/sec, independently of the processor, and under control of an autonomous address generator. The IMS B408 supports both interlaced and non-interlaced displays of arbitrary resolution up to 1024 × 768 pixels. At lower resolutions multiple frame buffers are supported; e.g. 4 frame buffers of 640 × 480 pixels.

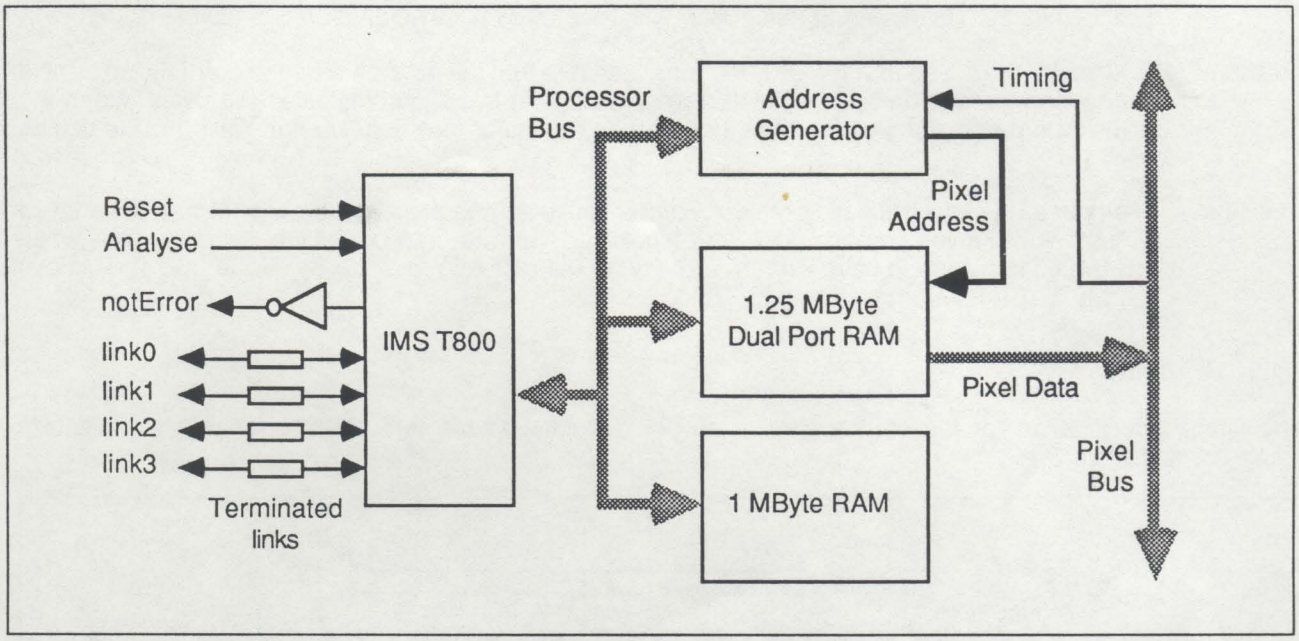


Figure 3.1 IMS B408 block diagram

The IMS B408 is designed to be used with the IMS B409 display driver TRAM. When connected to an IMS B409 via the INMOS Pixel Bus (and a suitable video monitor) a complete drawing and display system is formed. System performance is increased simply by adding more IMS B408s.

The IMS B408 performs the drawing function in such a system. The graphics processor is an IMS T800; a fast 32 bit processor with on-chip FPU. Image data is drawn into 1.25 Mbyte of dual port RAM; a further 1 Mbyte of RAM is provided for program/data storage. Image data is output through the pixel bus pixel port under the control of the dual port address generator. The address generator is programmable and responds to system timing signals from the pixel bus.

3.1 Pixel Port signals

The IMS B408 has a pixel data port in addition to the usual TRAM signals. This enables the TRAM to connect via the INMOS pixel bus to an IMS B409 display TRAM; and possibly several other IMS B408s. The pinout is defined in Table 2.1.

notD0–31 Pixel data is output on a 32 bit bus. New data is output after the falling edge of **notSEQclk**. The data bus is open collector and carries inverted data. This allows data from different serial port modules to be ORed on the Pixel Bus.

notSEQclk A continuous input clock used as the timing reference by the dual port address generator. It has a maximum frequency of 25MHz. Data and control strobe transitions are referenced to the falling edge of **notSEQclk**.

notRAMclk Used to clock pixel data from the dual port RAM onto the pixel bus. It is the same frequency and phase as **notSEQclk** but is gated so that it does not run during blanking. Thus, no data is lost during blanking. In the off state it is high.

notEarlyBlank A time-advanced version of the display blanking signal. Used by the dual port address generator as an early warning of when pixel data will be required and of when it should be turned off. After this signal goes low, there shall be 31 falling edges of **notRAMclk**. **notRAMclk** must then remain high until the 32nd falling edge of **notSEQclk** after **notEarlyBlank** goes high. The required relationship between **notEarlyBlank** and **notRamClock** is shown in figure 2.1.

notFieldSync Resets the dual port address generator at the start of each field. Low during field flyback (vertical blanking). The timing of this strobe with respect to **notSEQclk** is not critical.

notEvenField Used by the dual port address generator to ensure that the pixel data for the correct display field is output when generating an interlaced display. A low on this signal indicates the even field of the odd/even field pair making up an interlaced frame. This input may only change while **notFieldSync** is low.

SysReady This acts as a synchronisation mechanism for multiple modules. The IMS B408 has a writeable READY bit which drives an open collector output onto this wire, it also monitors its state. Only when all IMS B408s in a system have written 1 (ready) to their READY bits will **SysReady** be 1. This can be used to EVENT (interrupt) the IMS T800.

3.1.1 Timing

The timing specification for the control signals and output data on the IMS B408 pixel port are given in figure 3.2 and table 3.1.

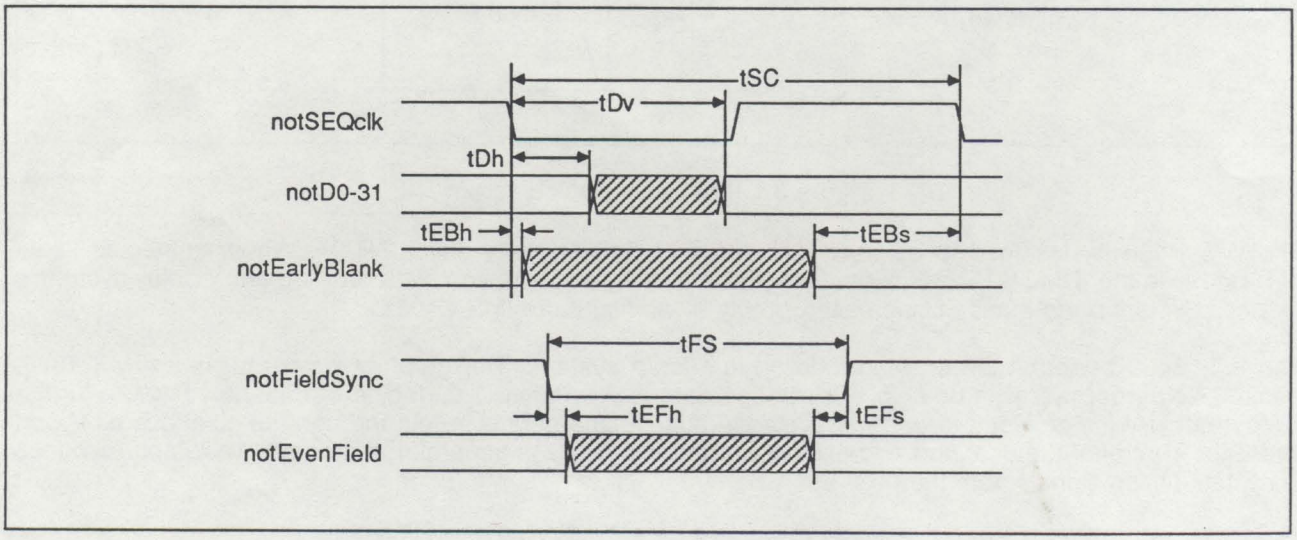


Figure 3.2 IMS B408 pixel port timing

3.1.2 Electrical Specification

The open collector drivers used for the data bus are capable of sinking 64mA and must be pulled up by an external resistor network. This network is part of the pixel data input structure on the IMS B409. The clock and control inputs have 4k7Ω pull up resistors to establish an idle condition on each input when the bus is disconnected.

Parameter	Description	time
tSC	min serial clock period	40ns
tDh	min data-hold time after clock	10ns
tDv	max time to data valid after clock	28ns
tEBh	min hold time after clock	0ns
tEBs	min set-up time prior to clock	20ns
tFS	min low time	100ns
tEFh	min hold time	0ns
tEFs	min set-up time	20ns

Table 3.1 IMS B408 pixel port timing

3.2 Pixel Port control registers

There are a small number of control registers associated with the pixel port and its address generator. These registers are located as shown in Table 3.2. All writeable bits in these registers are 0 after reset (note that there are many non-writeable bits which have no function and may be read as 1 or 0).

Register	read/write	Byte address
Display Start	write only	#00000000
Ready	read/write	#00040000
SysReady	read only	#00080000
Interlace Enable	read/write	#000C0000
Event Mode	read/write	#00100000
Output Enable	read/write	#00140000

Table 3.2 Control register locations

Display Start This register holds the address of the pixel at the top left hand corner of the displayed image. It can be used to implement flipping between multiple drawing buffers. The dual-ported memory can be thought of as being divided into twenty 64 kbyte blocks. Drawing buffers may only start on the boundaries of these blocks: i.e. their start addresses must be a multiple of 65536. The value written to the **Display Start** register is a word offset into the dual-ported RAM: ie it is

(drawing buffer byte address - #80100000) / 4

Interlace Enable Selects an interlaced or non-interlaced display. Writing 1 causes the address generator to produce interlace addressing; writing 0 causes it to produce non-interlaced addressing. When read, the upper 31 bits of this register are undefined.

Event Mode Selects the EVENT (interrupt) source to be either *FieldSync* (**Event Mode = 1**) or *FieldSync AND SysReady* (**Event Mode = 0**). When read, the upper 31 bits of this register are undefined.

Output Enable Enables and disables the pixel port output buffers. Writing 1 enables the data output buffers; writing 0 disables them. When read, the upper 31 bits of this register are undefined.

Ready Writing 0 drives SysReady low; writing 1 allows it to be pulled high. When read, the upper 31 bits of this register are undefined.

SysReady is a read only location which reflects the condition of the SysReady wire. Bit 0 is read as 0 if SysReady is low; 1 if SysReady is high. When read, the upper 31 bits of this register are undefined.

3.3 Memory Map

There are 2304 kbytes of memory. This consists of 4 kbytes of internal transputer memory and 2300 kbytes of external DRAM. The upper 1280 kbytes is dual ported to the pixel port. The lower 1024 kbytes would normally be used for program storage and the dual ported area as a drawing area (frame buffer). Table 3.3 shows

how the memory is mapped into the address space of the IMS T800 (the "#" sign indicates a hexadecimal number).

	Byte address	Cycle Time
IMS T800 on chip RAM	#80000000 - #80000FFF	50ns
External Workspace RAM	#80001000 - #800FFFFF	200ns
Dual port RAM	#80100000 - #8023FFFF	200ns

Table 3.3 Memory map of the IMS B408

3.3.1 Arrangement of pixel data in 8-bit Pixel mode

Pixel data is mapped in memory such that the pixel immediately to the right of the pixel at address m has address $m+1$: and so that the pixel below the pixel at address m has address $m+w$ where w is the number of pixels across the screen. Thus, placing a one dimensional array in memory, increasing the array index steps directly along each display line from left to right and then downwards from the top to the bottom of the display. Placing a two-dimensional array, in occam, results in the first array index counting from top to bottom and the second counting from left to right: i.e. the array should be dimensioned as `screen[height][width]`.

3.3.2 Arrangement of pixel data in Interlaced mode

In interlaced mode, the two fields of a frame are correctly interleaved in memory but are scanned out separately. The address offset between vertically adjacent pixels is always 1024. This means that the maximum screen width in interlaced mode is 1024 pixels in 8-bit pixel mode and 256 pixels in 24-bit pixel mode. The screen width in non-interlaced display mode is a maximum of 8192 pixels.

3.3.3 Arrangement of pixel data in 24-bit Pixel mode

When a full colour display is being produced, each pixel occupies a 32-bit word. The least-significant byte of a word is the red component of the pixel, the next least-significant byte of a word is the green component of the pixel, and the next least-significant byte of a word is the blue component of the pixel. The most significant byte is not used and can be written with anything. Words are stored in memory with the least-significant byte occupying the lowest byte address. The address of a word is the address of its least-significant byte and is always a multiple of four.

3.4 Multiple drawing buffers

Multiple drawing buffers for animation are supported by the **Display Start** register. The dual-ported memory is divided into twenty 64kbyte blocks. A drawing buffer may start at any block boundary: i.e. the start address of a drawing buffer must be a multiple of 64kbytes. The size and number of frame buffers is limited only by the amount of dual-ported memory on the IMS B408. Flipping between drawing buffers is done simply by writing a new value to the **Display Start** register. The value written to the **Display Start** register is

(drawing buffer byte address – #80100000) / 4

3.5 Syncronisation using Ready and SysReady

Multiple IMS B408s in a system can be syncronised by means of the **Ready** and **SysReady** control bits. **SysReady** is a readable bit which reflects the state of the **SysReady** wire on the pixel bus. **Ready** is a read/writeable bit which when written with 0 pulls the **SysReady** wire low: when written with 1 it causes this IMS B408 not to drive the **SysReady** wire. Thus, any IMS B408 in a system can pull the **SysReady** wire low and cause the **SysReady** bits of all IMS B408s to be read as 0, simply by writing 0 to its own **Ready** bit. Only when all IMS B408s in a system have written 1 to their **Ready** bits do **SysReady** and the **SysReady** bits on all the IMS B408s become 1.

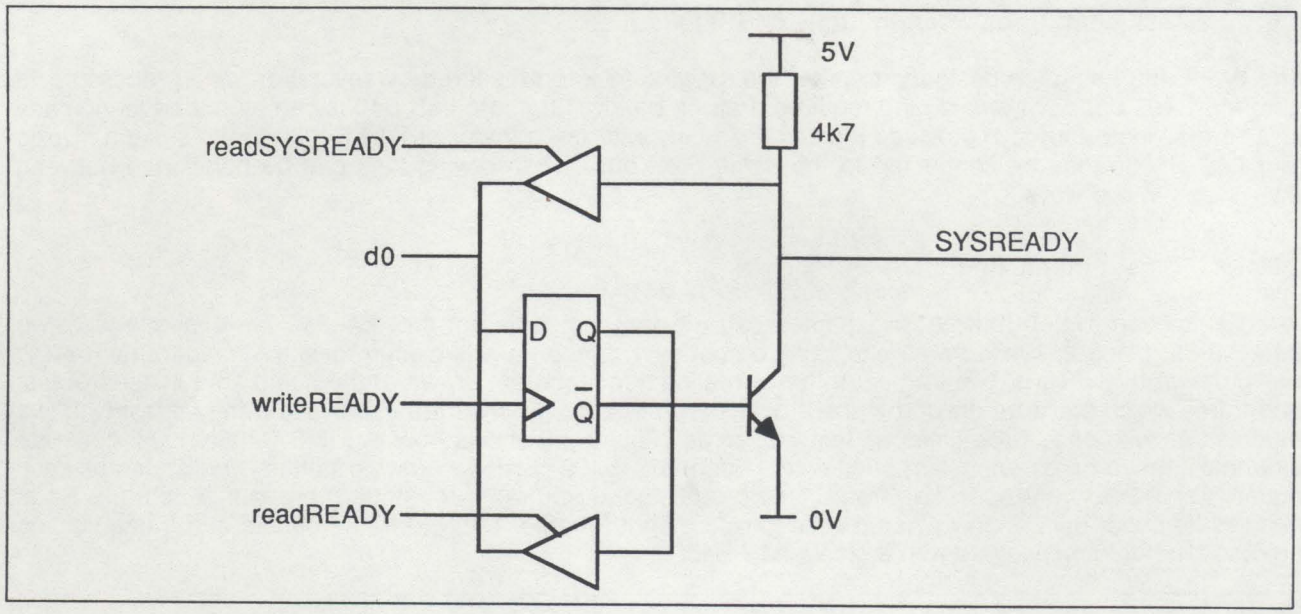


Figure 3.3 SysReady Logic

Writing 0 to the **event mode** register causes the IMS T800 to be interrupted (evented) only on **notFieldSync** pulses when **SysReady** is high: i.e. during field flyback when all IMS B408s in a system have written 1 to their **Ready** bits. This is intended as a means of syncronising frame-flipping between multiple IMS B408s.

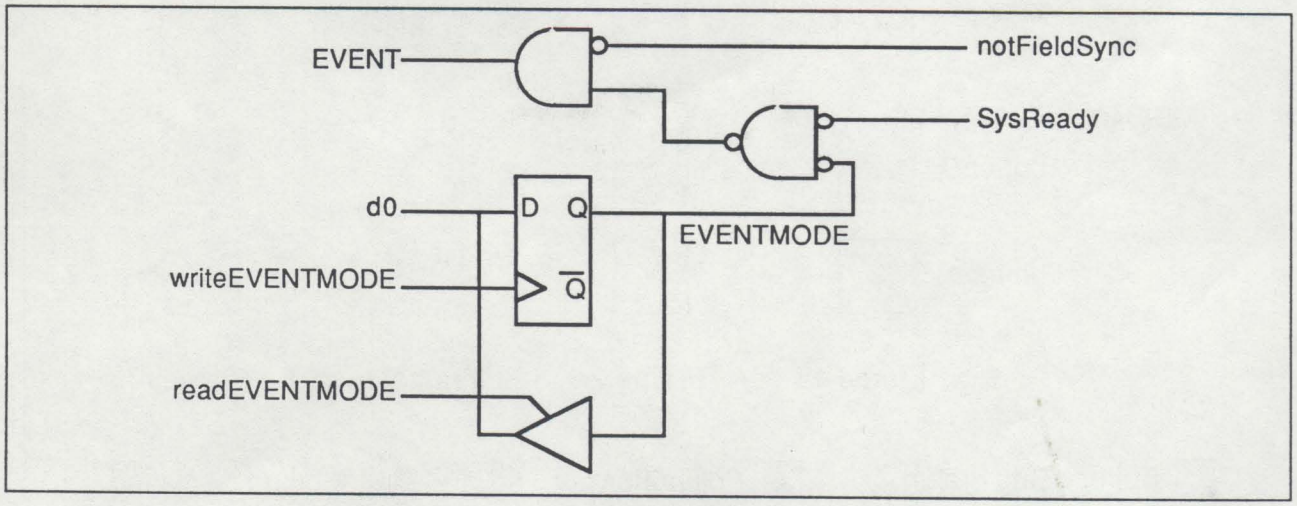


Figure 3.4 Event Logic

Before starting to draw a new image frame on an invisible screen, each IMS B408 should write 0 to its **Ready** bit. It should also write 0 to its **Event Mode** register. When it has finished drawing its part of the new image, it writes 1 to its **Ready** bit. When all of the IMS B408s have completed their drawing and written 1 to their **Ready** bits, the **SysReady** wire will go high and the next **notFieldSync** pulse will event the IMS T800s on all of the IMS B408s. An event handler on each IMS B408 can then flip to the new image simultaneously, simply by loading the **Display Start** register with the start address of the new image. The frame-flipping occurs during vertical blanking because it is driven by **notFieldSync**.

3.6 Task-sharing between multiple IMS B408s

The IMS B408 has been designed to allow the drawing task to be shared by several drawing processors. If one IMS B408 does not provide the required drawing bandwidth, more IMS B408s can be added to increase it. The drawing bandwidth provided increases linearly with the number of IMS B408s in the system. Up to four IMS B408s may be connected to the same pixel bus. The drawing task can be partitioned between IMS B408s in two ways.

3.6.1 Time multiplexing

In an animation system, successive frames can be drawn by different IMS B408s. For example with two IMS B408s, 1 and 2. While frame n is being output by 1, frame $n+1$ is being drawn by 2 and frame $n+2$ is being drawn by 1. Thus, two frames of the animation sequence are drawn at the same time instead of just one frame which could be drawn by one processor on its own. Frames are generated twice as fast with two modules as with one, three times as fast with three. Figure 3.5 shows how two IMS B408s could generate alternate frames of an animation sequence. Each IMS B408 has two drawing buffers, the status of each buffer is shown for four succesive frame times: after the fourth frame the pattern repeats. Each IMS B408 enables its ouput buffers only when it is its turn to output a frame. Multiple frame buffers and frame flipping are used in the same way as with a single IMS B408.

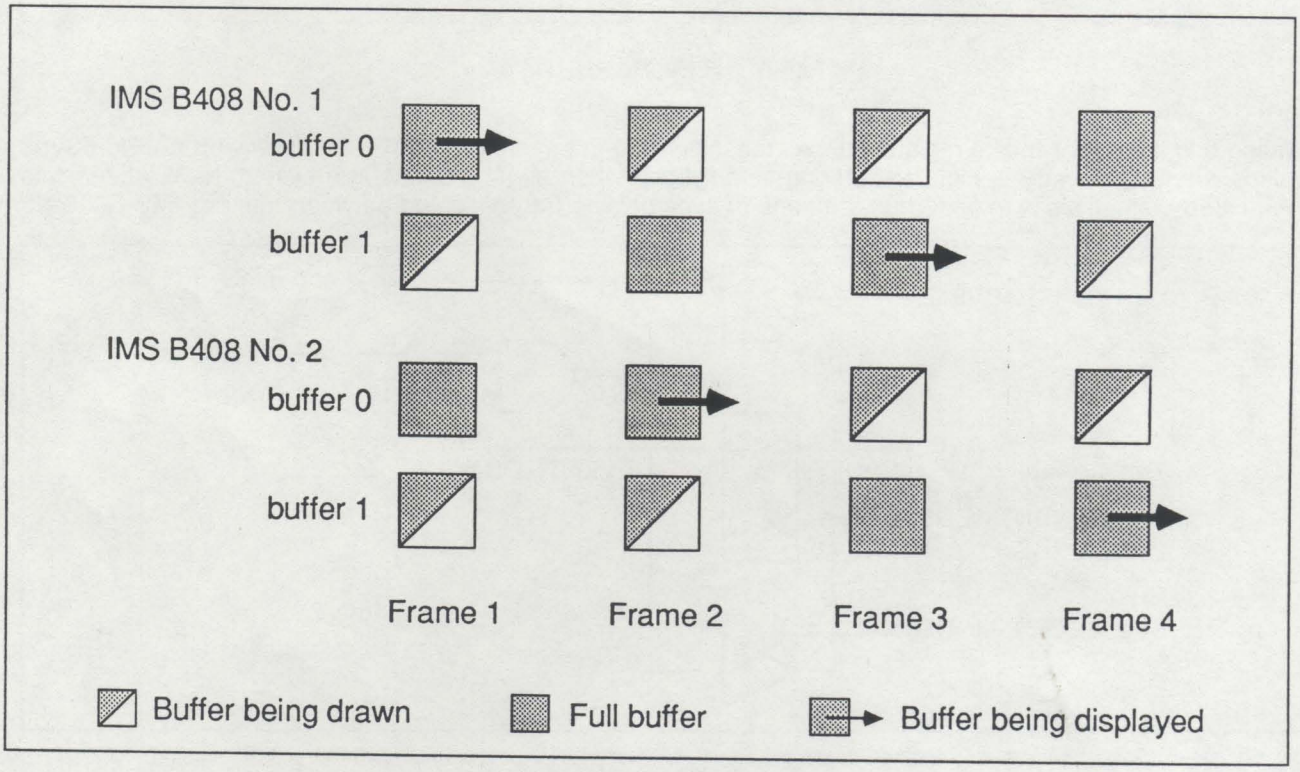


Figure 3.5 Time multiplexing between two IMS B408s

3.6.2 Screen partitioning

The screen can be partitioned between IMS B408s so that each draws a different part of the image. This is done by creating a frame-buffer on each IMS B408 which is the size of the whole screen. Each IMS B408 then writes 0 to all of the parts of the screen which it is not going to draw to. Thus, the way in which the screen is partitioned can be completely arbitrary. Figure 3.6 shows a simple partition between two IMS B408s. Each IMS B408 draws only to its own part of the drawing buffer, the pixel data from the IMS B408s is combined on the pixel bus to produce the whole image. Multiple frame buffers and frame flipping can be used in the same way as with a single IMS B408.

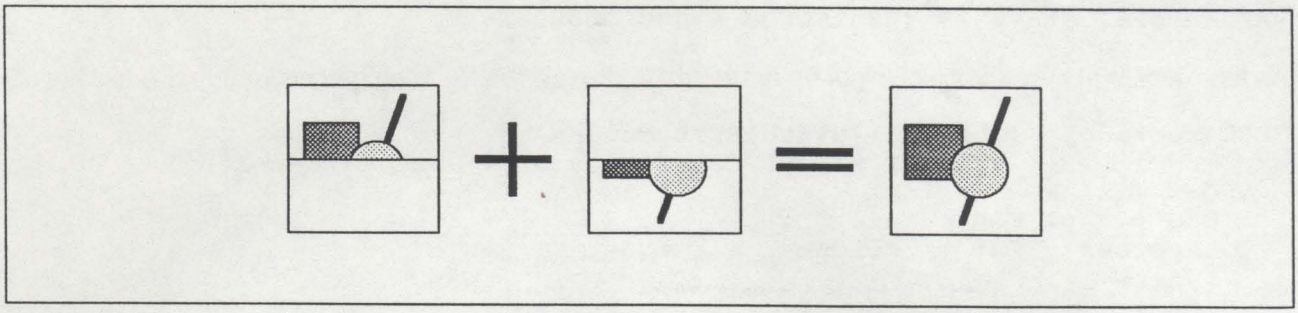


Figure 3.6 Screen partitioned between two IMS B408s

So that the drawing work-load is shared most equally between the IMS B408s, it is usually best that where there are n boards, each board draws every nth scan-line. This means that areas of the image with many objects and areas with few objects are distributed evenly between boards.

3.7 Programming examples

This section contains some simple sections of program, in occam, which illustrate how the IMS B408 can be programmed. The control registers can be placed in the following manner.

```
INT display.start:
PLACE display.start AT (#00000000/4) + #20000000:
INT ready:
PLACE ready AT (#00040000/4) + #20000000:
INT sys.ready:
PLACE sys.ready AT (#00080000/4) + #20000000:
INT enable.interlace:
PLACE enable.interlace AT (#000C0000/4) + #20000000:
INT event.mode:
PLACE event.mode AT (#00100000/4) + #20000000:
INT output.enable:
PLACE output.enable AT (#00140000/4) + #20000000:
```

A program will usually begin by initialising these registers to some useful values.

```
PROC initialise()
  SEQ
    display.start := 0
    ready := 0
    enable.interlace := 0
    event.mode := 0
    output.enable := 1
:
```

The next example shows how two drawing buffers, **screen0** and **screen1**, can be placed in specific areas of the dual-ported memory. Note that **screen1** does not immediately follow **screen0** but is placed at the next 64kbyte address boundary. **screen0.start** and **screen1.start** are the values which will be written to the **display.start** register to select which screen is output to the pixel bus.

```
VAL width IS 640:
VAL height IS 480:
[height][width]BYTE screen0:
[height][width]BYTE screen1:
PLACE screen0 AT (#80100000/4) + #20000000:
PLACE screen1 AT (#80150000/4) + #20000000:
VAL screen0.start IS (#80100000 - #80100000)/4:
```

```
VAL screen1.start IS (#80150000 - #80100000)/4:
```

As an example of how an object might be drawn to the display, the following procedure draws a rectangle.

```
PROC rectangle ([height][width]BYTE screen,
                VAL INT x1, y1, x.size, y.size, VAL BYTE colour)
  SEQ x = x1 FOR x.size
    SEQ y = y1 FOR y.size
      screen[y][x] := colour
:
```

4 The IMS B409

The IMS B409 implements the timing generation and display driver parts of a medium to high performance graphics system. It consists of three pixel channels and a programmable video timing generator (VTG), controlled by an IMS T222. Each pixel channel consists of a 4-1 byte multiplexer and an IMS G176 colour look-up table (CLUT). Input to each pixel channel is by a separate pixel bus input and each channel generates a set of RGB outputs. The IMS B409 supports both interlaced and non-interlaced displays of arbitrary resolution up to a maximum dot rate of 64MHz.

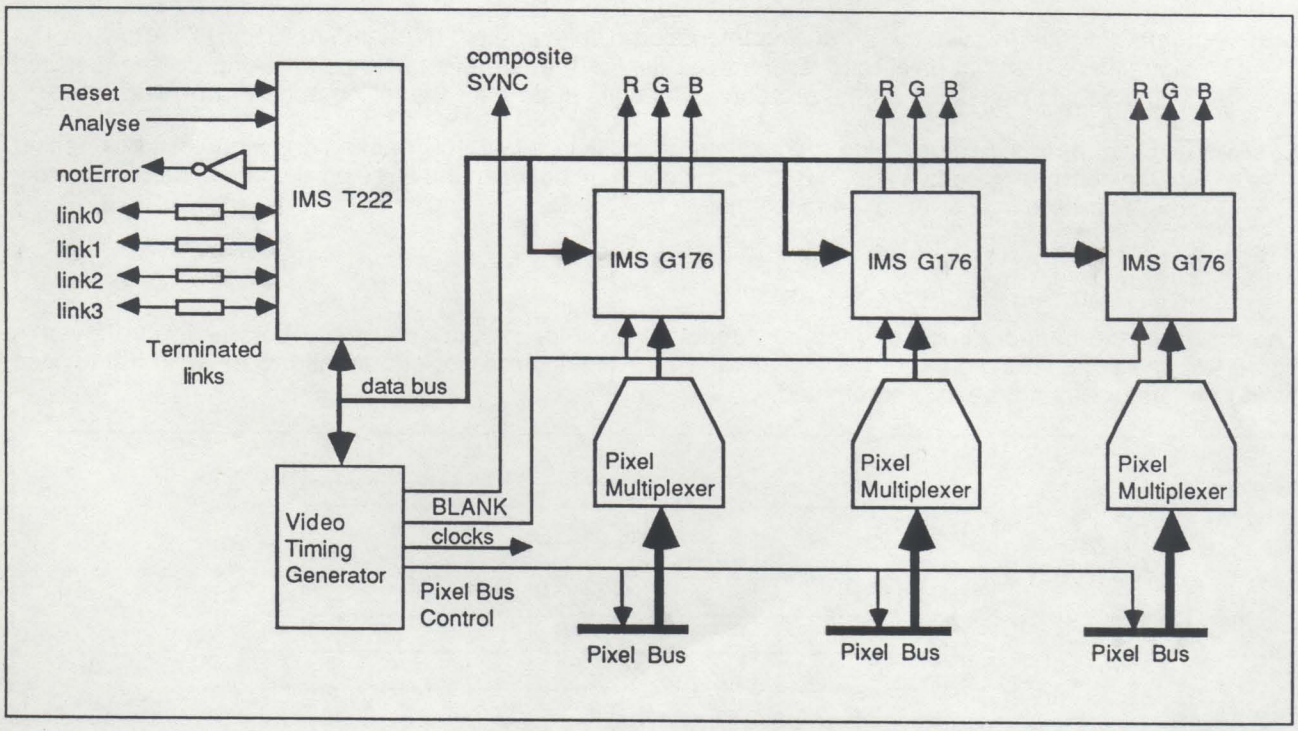


Figure 4.1 IMS B409 block diagram

The IMS B409 is designed to be used in conjunction with one or more IMS B408 frame store TRAMs. When connected to an IMS B408 via the INMOS Pixel Bus (and a suitable video monitor) a complete drawing and display system is formed. System performance is increased simply by adding more IMS B408s.

The IMS B409 has three pixel channels. Each channel inputs a 32 bit wide pixel stream from an INMOS Pixel Bus and processes it into a form suitable for display by a colour monitor. The IMS B409 also generates system timing and control signals and outputs them on each pixel bus.

4.1 Pixel Port Connectors

The IMS B409 has three pixel data ports in addition to the usual TRAM signals (see section A). This enables the IMS B409 to connect via the INMOS pixel bus to one or more IMS B408s. The Pixel Bus uses 60-way IDC connectors and flat ribbon cable; the pinout of each port is defined in Table 2.1. The clock and control outputs are driven by high current buffers capable of driving into 100Ω loads.

notD0–31 Pixel input data is latched on the falling edge of **notSEQclk**. The data bus is open collector and carries inverted data. This allows data from different serial port modules to be ORed on the Pixel Bus. Each data input is terminated with 330Ω to **Vcc** and 470Ω to **GND**.

notSEQclk A continuous output clock for use as the system timing reference. It has a maximum frequency of 25MHz. Data and control strobes transitions are referenced to the falling edge of **notSEQclk**.

notRAMclk An output clock of the same frequency and phase as **notSEQclk** but gated so that it does not run during display blanking. It is used to clock pixel data from the IMS B408s onto the pixel bus so that no data is lost during blanking. In the off state it is high.

notEarlyBlank A time-advanced version of the display blanking signal. It provides early warning of when pixel data will be required and of when it should be turned off. The IMS B409 outputs **notEarlyBlank** with the relationship to **notRamClock** defined in figure 2.1.

notFieldSync Output low during field flyback (vertical blanking) to indicate the start of a new field.

notEvenField For use in systems producing interlaced displays; e.g. TV standard displays. A low on this signal indicates that pixel data for the even field of the odd/even field pair making up an interlaced frame should be placed on the pixel bus. Changes state after the falling edge of **notFieldSync**.

SysReady Used as a synchronisation mechanism by multiple IMS B408 frame store modules. It is neither driven nor monitored by the IMS B409 but is common between the three pixel channel bus connectors to support the synchronisation mechanism.

4.1.1 Timing

The timing of the control strobe outputs, and required pixel input data set-up and hold times are given in figure 4.2 and table 4.1. Note that the signals and their timings are not guaranteed to be valid if the video timing generator has not been programmed.

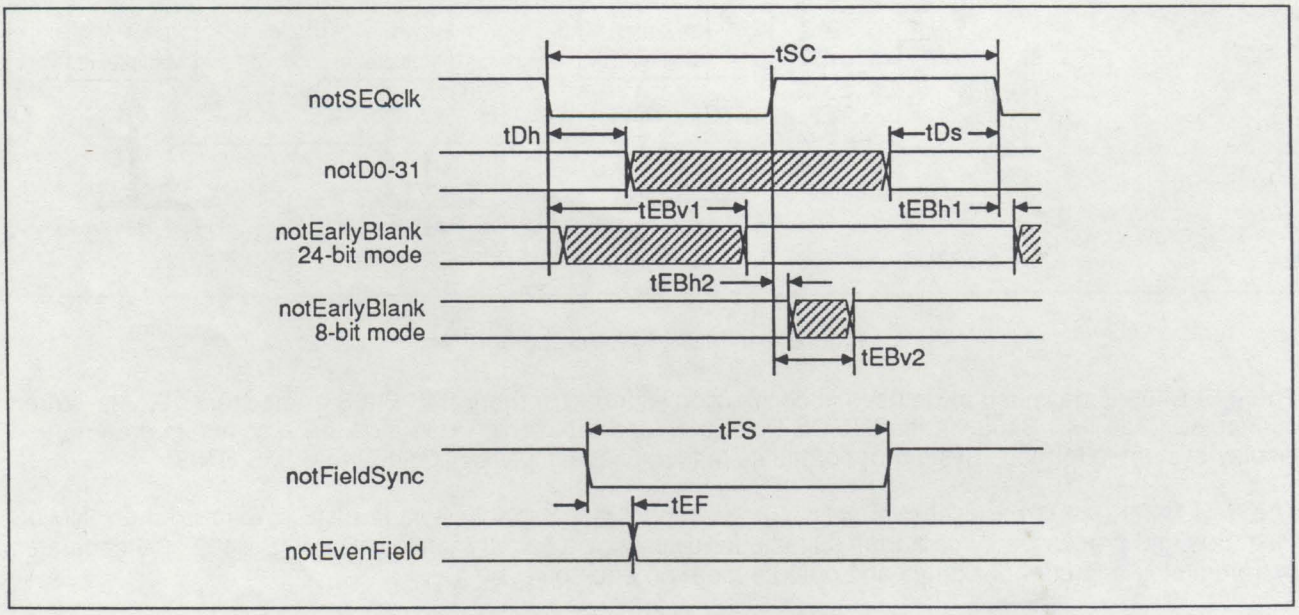


Figure 4.2 IMS B409 pixel port timing

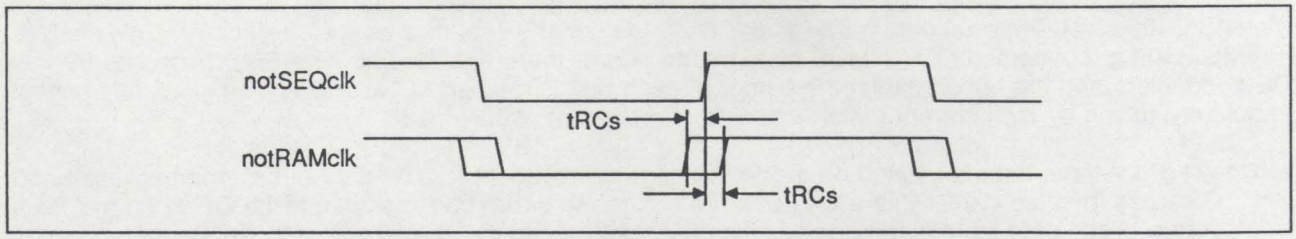


Figure 4.3 Skew between pixel bus clocks

Parameter	Description	time (8-bit mode)	(18-bit mode)
fPclk	max pixel clock frequency	64MHz	25MHz
tSC	min serial clock period	62ns	40ns
tDs	min pixel input data set-up	20ns	10ns
tDh	min pixel input data hold	10ns	10ns
tEBv1	max time to notEarlyBlank valid	–	20ns
tEBh1	min notEarlyBlank hold time	–	0ns
tEBv2	max time to notEarlyBlank valid	10ns	–
tEBh2	min notEarlyBlank hold time	0ns	–
tFS	min notFieldSync low period	2500ns	2500ns
tEF	max time to notEvenField valid	10ns	10ns
tRCs	max skew between clocks	5ns	5ns

Table 4.1 IMS B409 pixel port timing

4.2 The Pixel channels

The IMS B409 has three pixel channels: A, B and C. Each channel consists of a 4-1 byte multiplexer and an IMS G176 colour look-up table (CLUT). Input to a channel is through a pixel bus connector, output is from a set of RGB video outputs. There are two operating modes.

4.2.1 8-bit pixel mode

Each channel accepts 32 bit pixel data from a pixel bus connector at 1/4 the pixel rate. This is multiplexed down to an 8 bit wide stream at pixel rate which is fed to the CLUT pixel data inputs. Thus, the pixel bus runs at 1/4 the pixel rate. The pixel rate may be up to 64MHz. Each channel can provide a separate display with up to 256 colours on each screen. It is not necessary to use all three channels. Since the three channels are synchronised it is possible to use each channel to generate one of the RGB primaries. Skew between any two pixel channels on the same IMS B409 is less than 5ns.

4.2.2 18-bit pixel mode

In this mode the IMS B409 provides a single display of up to 262144 colours. It allows a full colour display to be produced by an IMS B409 with a single IMS B408. This mode requires a 32 bit word to be supplied to the channel A pixel bus input for every pixel displayed. The channel B and channel C pixel bus inputs cannot be used.

The least significant byte from the channel A input is routed direct to the channel A CLUT, the second least significant byte is routed direct to the channel B CLUT, and the third least significant byte is routed direct to the channel C CLUT. Each CLUT is loaded with a ramp so that the DAC output level is proportional to the pixel address (since the CLUTs use 6-bit DACs, only the lower 6 bits of each byte are significant). Thus, the video output from channel A is controlled by byte 0 of the pixel word, the output from channel B is controlled by byte 1, and the output from channel C is controlled by byte 3. Thus, a single input word can specify directly the red, green and blue components of a pixel. Output skew between any two pixel channels on the same IMS B409 is less than 5ns.

In this mode, the pixel bus runs at the pixel rate. The pixel rate is therefore limited to 25MHz.

4.2.3 Pixel Channel Mode select

The pixel channel mode is set by writing to the Pixel Channel Mode Select register. Writing 1 selects multiplexed (8 bits/pixel) mode: writing 0 selects non-multiplexed (18 bits/pixel) mode. The register location is given in table 4.2. This register is write only.

Register	Byte Address
Channel Mode Select	#B000

Table 4.2 Channel Mode Select register location

4.3 The Colour look-up tables

There are three IMS G176 colour look-up tables: one for each of the three pixel channels. Each of these devices combines a 256 word, 18 bit wide RAM and three 6 bit DACs. 8 bit data applied to the device's pixel inputs addresses a location in the RAM. 6 bits of the addressed data are applied to each of the DACs which generate the red, green, and blue (RGB) outputs. Thus, the device can display up to 256 colours, selectable from a palette of 262144. The RAM contents are writeable and readable by the IMS T222.

Ordinary accesses to the CLUT registers should be made at the addresses shown in table 4.3. These registers are mapped as the lower 8 bits of a 16 bit word addressed at that location. They can be written and read either as 16 bit words or as bytes addressed at the given locations. If written as 16 bit words, the upper 8 bits are ignored; if read as 16 bit words, the upper 8 bits are read undefined.

Register	Byte Address
Channel A Pixel Address (write mode)	#0000
Channel A Colour Value	#0400
Channel A Pixel Mask	#0800
Channel A Pixel Address (read mode)	#0C00
Channel B Pixel Address (write mode)	#1000
Channel B Colour Value	#1400
Channel B Pixel Mask	#1800
Channel B Pixel Address (read mode)	#1C00
Channel C Pixel Address (write mode)	#2000
Channel C Colour Value	#2400
Channel C Pixel Mask	#2800
Channel C Pixel Address (read mode)	#2C00

Table 4.3 IMS G176 registers

Block moves to and from the colour value registers should be made to the regions defined in table 4.4. In this region, the colour value register appears as an 8 bit wide register repeated at each byte address. Thus, byte arrays of colour values can be block copied to and from these areas. Correct results for block writes are not guaranteed for pixel clock speeds of less than 16MHz. Correct results for block reads are not guaranteed for pixel clock speeds of less than 28MHz. With pixel clock frequencies lower than these, reads and writes should be made one at a time to the colour value registers at the locations in table 4.3.

Register	Byte Address
Channel A Colour Value	#4400-#47FF
Channel B Colour Value	#5400-#57FF
Channel C Colour Value	#6400-#67FF

Table 4.4 IMS G176 block move areas

4.3.1 The Pixel Mask Register

The Pixel Mask Register is ANDed with the incoming pixel data and is normally be set to #FF. Writing different values splits the palette of 256 colours into a number of sub-palettes. This can be used to create flashing objects and simple animation. The Pixel Mask Register can be read.

4.3.2 The Pixel Address Register

Each CLUT has a single Pixel Address Register but it is accessed at two locations. Writing to the Pixel Address Register at the *write mode* location indicates that a colour is to be assigned to a location in the table. Writing to the Pixel Address Register at the *read mode* location indicates that the colour at the addressed location is to be read. A read from either the read mode or write mode location returns the same result: the contents of the Pixel Address Register.

4.3.3 Assigning Colours to Pixel Values

The pixel value for which a colour is to be assigned is written to the write mode Pixel Address register. The 18 bit colour value to be assigned is then written to the Colour Value register as three bytes. The 6 least-significant bits of the first byte written are the RED component of the assigned colour, those of the second byte written are the GREEN component, and those of the third are the BLUE component.

After the third write to the Colour Value register, the contents of the Pixel Address register are automatically incremented. A further three writes to the Colour Value register assign a colour to the new location pointed to by the Pixel Address register and the Pixel Address register is incremented again. Thus, it is possible to assign the entire contents of the CLUT simply by writing 0 to the Pixel Address register and performing 768 byte writes to the Colour Value Register. This allows transputer block moves to be used to assign colours to pixel values.

The IMS T222 is evented on field flyback so that updates of the colour look-up tables can be made to occur only during vertical blanking (field flyback).

4.3.4 Reading the CLUT Contents

The address of the location to be read is written to the Pixel Address register at the read mode location. The colour assigned to this address can then be read from the Colour Value register as three bytes. The first read returns the RED component, the second the GREEN, the third the BLUE. After the third read, the Pixel Address register is incremented and the contents of the newly addressed location copied to the Colour Value register. Thus, blocks of assigned colours can be read in the same way as they can be written.

4.3.5 Pixel Address inversion

The pixel data on the pixel bus is inverted. This means that pixels written with value n in the memory of the IMS B408 have value $255 - n$ at the pixel inputs of the colour look-up tables. This means that when a colour table is loaded into the look-up tables on the IMS B409, the colour for pixel value n should be written to *pixel address* $255 - n$. Otherwise, the expected colours will not be reproduced.

4.4 Video Outputs

Each pixel channel has a set of RGB outputs brought out on three SMB connectors. The outputs are current sources. Each output will drive 0.7V pk-pk into a 75Ω load. The outputs are d.c. coupled: 0.3V is blanking level and 1.0V (on load) is peak white. The connector positions are shown in figure 5.2.

4.5 Syncronisation Signals

Sync is not composited with the video signals but is available from a separate sync output (also an SMB connector). The sync output will also drive into 75Ω and is d.c. coupled: 5V is the idle level, sync pulses are 0V. The same sync output is shared by the three pixel channels.

4.6 Memory Map

The IMS T222 on the IMS B409 is able to access 4 kbytes of internal transputer memory. This is sufficient memory to contain the small amount of code and data required to set up the colour look-up tables and the

VTG. The internal memory on the IMS T222 has a 50ns access cycle time; i.e. a single processor cycle. The IMS T222 has a 64 kbyte address space with addresses ranging from #8000 to #7FFF where # indicates a hexadecimal number.

	Byte Address
IMS T222 internal RAM	#8000-#8FFF

Table 4.5 IMS B409 memory location

4.7 The video timing generator

The video timing generator is an NEC D7220. It is mapped into the IMS T222's address space as shown in Table 4.6. It is used only as a programmable timing generator and performs no drawing functions. Line frequency, field frequency and resolution can be programmed (horizontal resolution must be a multiple of 64 pixels) and displays may be either interlaced or non-interlaced.

Register	read/write	Byte address
Parameter FIFO	write only	#A000
Status Register	read only	#A000
Command FIFO	write only	#A002
FIFO read	read only	#A002

Table 4.6 NEC D7220 register locations

On this board, the NEC 7220 has no function other than as a programmable source of syncronisation and blanking pulses. The NEC 7220 has two writeable locations: the *Command Register* and the *Parameter Register.* Values written to both locations enter the same FIFO, commands and parameters being distinguished according to which location they were written. These registers are mapped as the lower 8 bits of a 16 bit wide word at the addresses given. They can be read and written either as 16-bit words or as bytes at the addresses given. If written as words, the upper 8 bits are ignored: if read as 16-bit words, the upper 8 bits are read undefined.

The only commands to the NEC 7220 which are of interest in this application are the RESET command and the BCTRL command. The RESET command should be followed by eight parameter writes to the Parameter FIFO. These specify the timing of the sync and blanking pulses. The BCTRL command is used to unblank the display following a RESET command.

After any read or write to the 7220, software should wait for a minimum of 2μs before reading or writing the 7220 again. Otherwise correct operation of the 7220 is not guaranteed.

4.7.1 The Status Register

The 7220 status register contains several status flags, summarised in table 4.7. Bit 0 is the least-significant bit of the register. The following are the status flags which are useful in programming the 7220 on the IMS B409.

FIFO full 1 when the command/parameter FIFO is full: 0 otherwise. Software must not attempt to write to the command FIFO or to the parameter FIFO when this flag is 1.

FIFO empty 1 when the all of the commands and parameters written to the command/parameter FIFO have been processed: 0 when the 7220 is still busy with the last command.

Vertical Sync Active 1 during the vertical sync pulse. Can be polled to syncronise colour look-up table modifications to vertical blanking. The IMS T222 is evented by vertical sync which is a better way of ensuring syncronisation.

Horizontal Blank Active 1 during horizontal blanking. Can be polled to syncronise colour look-up table modifications to horizontal blanking.

Status Flag	Bit number
Data Ready	0
FIFO full	1
FIFO empty	2
Drawing in Progress	3
DMA Execute	4
Vertical Sync Active	5
Horizontal Blank Active	6
Light Pen Detect	7

Table 4.7 NEC D7220 status register

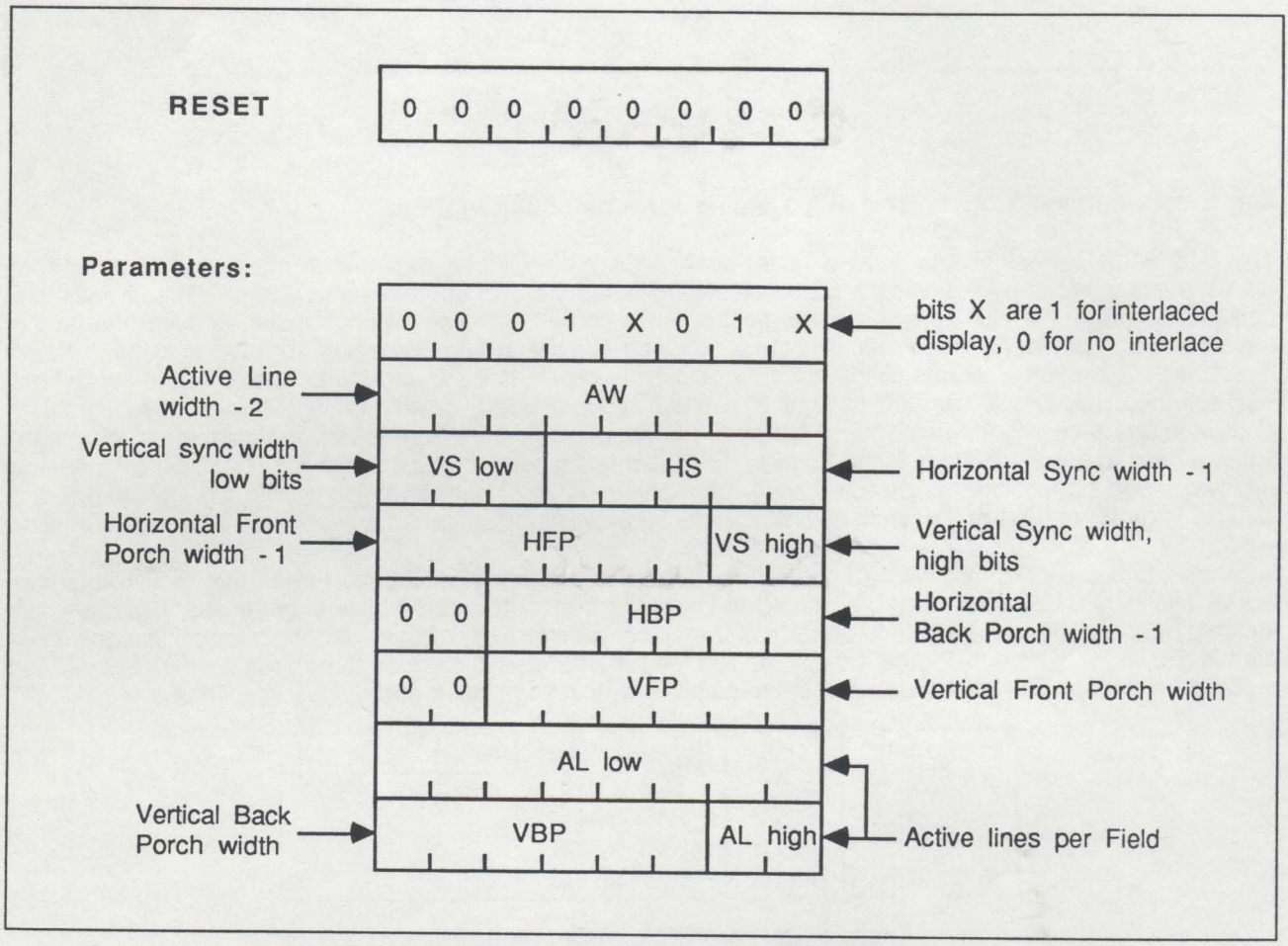


Figure 4.4 Parameters for the RESET command

4.7.2 RESET

The RESET command is used to set the sync and blanking periods. It is followed by 8 parameter bytes. The format of the command is shown in figure 4.4. The horizontal timing parameters are specified in periods of PixelClock/32. The vertical timing parameters are specified in lines. For vertical parameters, a zero value selects an interval of 2^n lines where n is the number of bits in the parameter field. The number of active words per line must be an even number. Each parameter defines the length of part of the video waveforms. Their meanings are shown in figure 4.6.

4.7.3 BCTRL

The BCTRL command is used to unblank the display following a reset command; it can also be used to blank the display if required. It has no parameters, but a 1 in the least significant bit will unblank the display, a 0 will blank it. The command format is shown in figure 4.5.

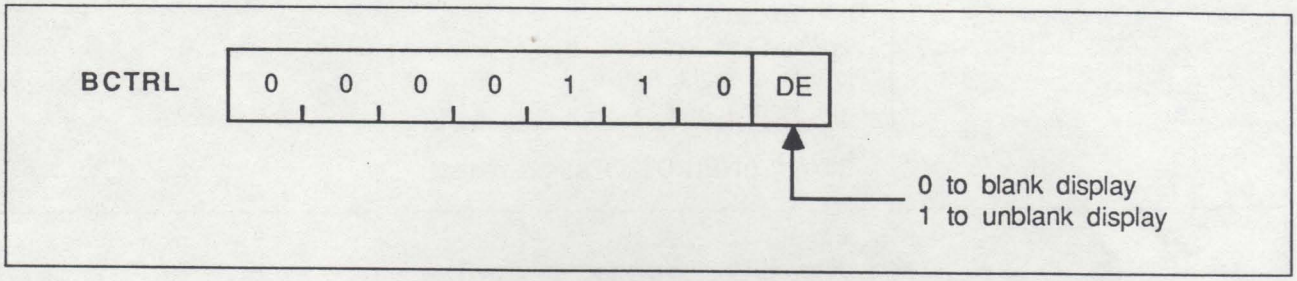


Figure 4.5 The BCTRL command

4.8 Determining the correct video timing for a particular monitor

The IMS B409 is designed to drive a raster-scan display device: i.e most video monitors. A raster-scan display device works by scanning a beam of electrons across a phosphorescent screen. When the beam strikes the screen it emits light: the more electrons, the more light. The beam is scanned down the face of the screen at a constant speed, usually taking between 15ms and 20ms to travel from top to bottom. When it reaches the bottom it returns to the top very quickly, in only a few μs, and starts again. At the same time, it is scanned repeatedly from left to right at a much higher speed, usually taking 15μs to 30μs to do so. Thus, the beam covers the screen in a series of horizontal scan lines from left to right, top to bottom, many times a second. At the bottom of the screen, the beam is caused to return to the top (*fly-back*) by a *vertical syncronisation* signal from the device driving the monitor. At the right-hand side of the screen, the beam is caused to return to the left (*fly-back*) by a *horizontal syncronisation* signal from the device driving the monitor.

Most monitors are designed with a limited range of horizontal and vertical scan rates over which they can work. The IMS B409 must be programmed to generate horizontal and vertical syncronisation signals with suitable frequency for the monitor. The form of the video waveforms showing syncronisation pulses and video data is shown in figure 4.6. Figure 4.6 also shows the parameters which need to be programmed to establish a particular timing. Table 4.8 gives some examples of typical scan rates and display resolutions.

Horizontal	Vertical	Typ Resolution	Notes
15.625 kHz	50 Hz	512 × 575	interlaced display
35 kHz	50 Hz	640 × 480	
50 kHz	60 Hz	1024 × 768	
64 kHz	70 Hz	1024 × 768	

Table 4.8 Typical scan rates

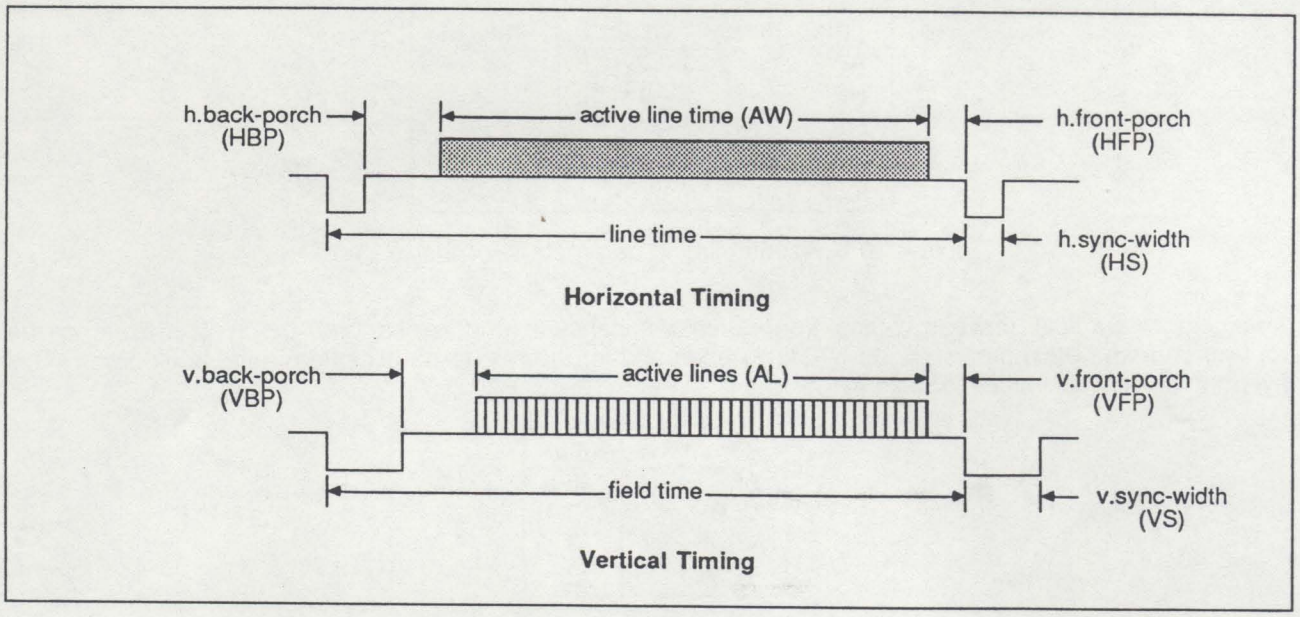


Figure 4.6 Video waveforms and timing parameters

4.8.1 Example

As an example of how to determine a suitable set of parameters for your particular monitor, suppose that it has the specification given in table 4.9 and you want a display of 640 by 480 pixels.

Parameter	Value
Horizontal scan rate	30-35 kHz
Vertical scan rate	48-62 Hz
Horizontal flyback time (min)	5.5μs

Table 4.9 Example monitor timing

The first step is to determine the pixel clock frequency required. Taking the horizontal scan rate to be 32.5kHz, the horizontal scan time is simply the inverse of this which is 30.8μs. Subtracting the horizontal flyback time gives 25.3μs[1]. This is the time in which one horizontal row of pixels is displayed (*horizontal display period*) so dividing by 640 gives the period of the pixel clock. Hence, the pixel clock frequency required is 25.3MHz. Now, oscillators are only readily available with certain frequencies. The nearest to this calculated value is 25MHz so we choose this. This gives the pixel clock period, Tp = 40ns.

The next step is to use this value to determine the actual horizontal display period and scan time. The horizontal display period is 640 $\times$ Tp = 25.6μs.

The horizontal front-porch, back-porch, and sync period are programmed in steps of 32 $\times$ Tp = 1.28μs. Each of these parameters should not be less than the minimum allowed by the specification of the monitor but if this is not available they can be estimated as follows. The total of their lengths is about 20% of the line scan time, and the length of the horizontal front-porch plus the length of the horizontal sync pulse should be about the same as the length of the horizontal back-porch. This suggests table 4.10 as a suitable set of parameters for the horizontal timing.

The total horizontal scan time is 33.28μs which means that the horizontal scan rate is Fh = 30.05kHz, which is within range. If it had not been in range, a different pixel clock frequency would need to be chosen.

Knowing the horizontal scan time allows the number of scan lines per field to be determined. This is simply $\frac{Fh}{Fv}$ which for a vertical scan rate of 60Hz is 30048/60 = 501. 480 of these are active display lines, the

[1] If the horizontal flyback time is not known it can be estimated as 20% of the horizontal scan time

Parameter	Value	Time (μs)
AW	20 – 2 = 18	25.6
HFP	1 – 1 = 0	1.28
HS	2 – 1 = 1	2.56
HBP	3 – 1 = 2	3.84
total		33.28

Table 4.10 Horizontal timing parameters for the example

remaining 21 are split between vertical front-porch, vertical sync, and vertical back-porch. It is usual for the front-porch and the sync pulse to be 3–5 lines each and for the rest to be allocated to the back-porch. This gives the parameter values in table 4.11.

Parameter	Value
AL	480
VFP	3
VS	3
VBP	15

Table 4.11 Vertical timing parameters for the example

4.9 Programming Examples

This section contains some example code, written in occam, which shows how the VTG and colour look-up tables can be programmed. The following two procedures can be used to write commands and parameters to the 7220. Status bits are polled until they indicate that the new command or parameter can be written. An important point is the use of delays before each access to the 7220 to ensure that the 7220 has enough recovery time between accesses. The delay time is one tick of the low priority process timer (64μs).

```
PROC write.command(VAL INT command)
  INT status.reg, command.reg:
  PLACE status.reg  AT (#A000 / 2) PLUS #4000:
  PLACE command.reg AT (#A002 / 2) PLUS #4000:
  VAL fifo.empty IS 4:
  TIMER time:
  INT now:
  SEQ
    time ? now
    time ? AFTER (now PLUS 1)
    WHILE (status.reg/\fifo.empty) <> fifo.empty  --wait for fifo to empty
      SEQ
        time ? now
        time ? AFTER (now PLUS 1)
    time ? now
    time ? AFTER (now PLUS 1)
    command.reg := command
:

PROC write.parameter(VAL INT data)
  INT status.reg, parameter.reg:
  PLACE status.reg AT (#A000 / 2) PLUS #4000:
  PLACE parameter.reg AT (#A000 / 2) PLUS #4000:
  VAL fifo.full IS 2:
  TIMER time:
  INT now:
  SEQ
    time ? now
```

```
    time ? AFTER (now PLUS 1)
    WHILE (status.reg /\ fifo.full) = fifo.full  --wait if fifo is full
      SEQ
        time ? now
        time ? AFTER (now PLUS 1)
    time ? now
    time ? AFTER (now PLUS 1)
    parameter.reg := data
:
```

The following is all that is required to load one of the colour look-up tables. The addresses used in the example are those of the channel A CLUT. This procedure waits until it is evented by a vertical sync pulse before writing to the CLUT, this ensures that the accesses happen during verical blanking and no interference appears on the display. The first input from the **event** channel clears a possible pending event due previous field sync pulses. The second input from the *event* channel causes the synchronisation.

```
PROC load.table([768]BYTE colours)
  INT pixel.mask, pixel.addr.write:
  [768]BYTE colour.val.block:
  PLACE pixel.mask AT (#0800 / 2) PLUS #4000:
  PLACE pixel.addr.write AT (#0000 / 2) PLUS #4000:
  PLACE colour.val.block AT (#4400 / 2) PLUS #4000:
  SEQ
    event ? dummy
    event ? dummy
    pixel.mask := #FF
    pixel.addr.write := 0
    colour.val.block := colours
:
```

If the pixel clock frequency is too low to allow block moving to the colour value register to be used, the array **colours** should be copied one byte at a time to the colour value register.

5 Installation

This chapter covers the installation of the IMS B408 and the IMS B409, covering the theory of installation and giving some examples.

Each installation requires

- one IMS B409
- one or more IMS B408s
- one pixel bus cable
- one pixel bus termination module
- one set of video leads

5.1 Pixel Bus Cabling

The IMS B408s, IMS B409 and pixel bus termination module must be cabled together so that: the pixel bus cable ends on one and only one of the IMS B409's pixel bus connectors, and at the other end on the pixel bus termination module. The IMS B408s are connected at any mechanically convenient points between them.

The pixel bus termination module and the IMS B408 have been designed so that the pixel bus termination module can be mounted on the IMS B408, as shown in figure 5.1. This would be the IMS B408 furthest along the pixel bus from the IMS B409.

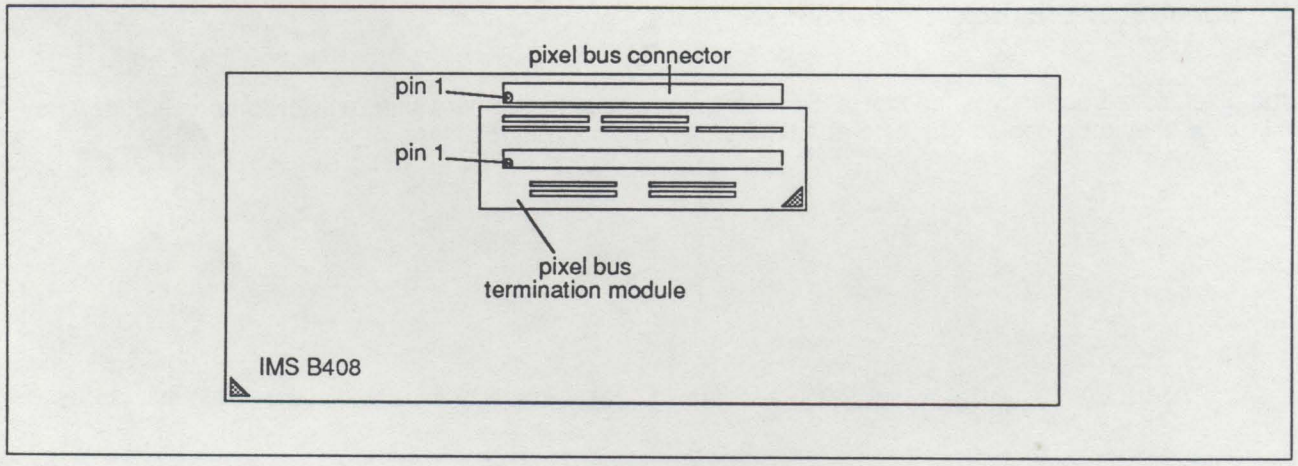


Figure 5.1 IMS B408 and termination module

5.2 Connecting to a video monitor

A cable is required with four SMB plugs at one end and a suitable connector or connectors for your monitor at the other. The monitor needs to have a separate sync signal input: this must be connected to the IMS B409's **sync** ouput. There may be a switch on the outside or inside of the monitor's case to select separate sync: this must be in the correct position, otherwise the monitor cannot syncronise to the IMS B409's ouput. The positions of the video connectors on the IMS B409 are shown in figure 5.2. The *sync* output is shared between the three video channels. When operating in 8-bit pixel mode, the monitor must be connected to the video outputs corresponding to the pixel bus input which is being used: e.g. the channel A video outputs if the channel A pixel bus input is being used. When operating in 18-bit pixel mode: the monitor's *red* video input should be connected to one of the channel A video outputs, the *green* video input should be connected to one of the channel B video outputs, and the *blue* video input should be connected to one of the channel C video outputs.

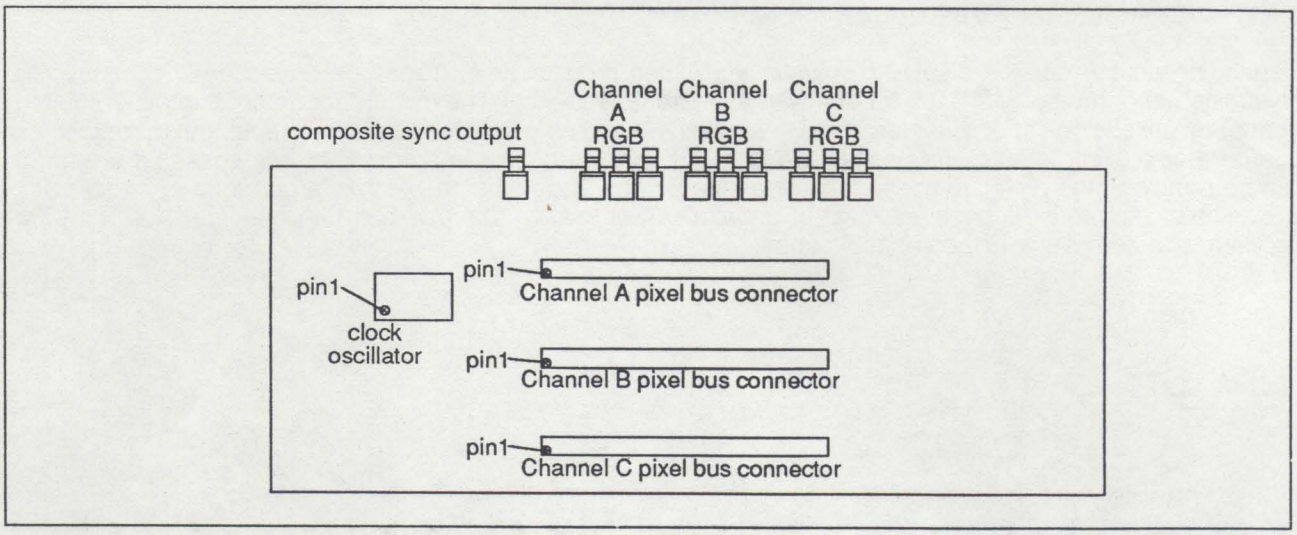


Figure 5.2 IMS B409 connector positions

5.3 Example

Figure 5.3 shows how a system containing two IMS B408s should be cabled. One end of the pixel bus cable is connected to the channel A pixel input on the IMS B409: this allows the system to operate in 8-bit pixel or 18-bit pixel mode. The other end is connected to the pixel bus termination module: this is mounted on one of the IMS B408s. In figure 5.3 the video cables are shown connected to the channel A video outputs for 8-bit pixel operation.

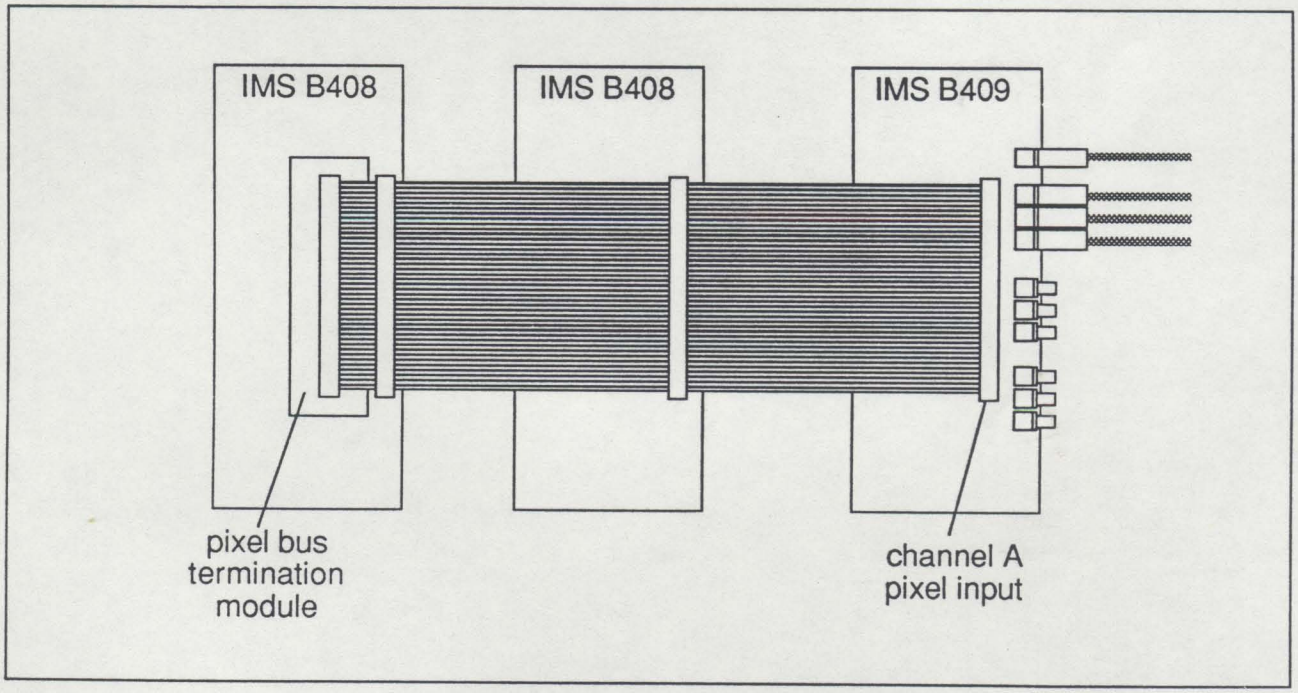


Figure 5.3 Example installation

5.4 Changing the Oscillator on the IMS B409

Depending on the desired display resolution and video monitor type, it may be necessary to change the crystal oscillator on the IMS B409 for one with a different frequency. The method for determining the required frequency (pixel clock speed) is described in section 4.8.1. The crystal oscillator is a rectangular, metal can, about 0.5 × 0.8 in. in size. Its position on the IMS B409 is shown in figure 5.2. It is socketed so that it can be removed and replaced easily. This is a standard package type, in which a large range of frequencies is available. Appendix D contains a list of addresses from which this part can be obtained. Note that the oscillator should have a TTL compatible output.

Appendices

A Links and TRAMS

A.1 Links

The transputer link is a point to point, 10 or 20 Mbit/s serial communications interface. A link connection between two devices is made by connecting a LinkOut/LinkIn pair on one device to a LinkOut/LinkIn pair on the other device: figure A.1.

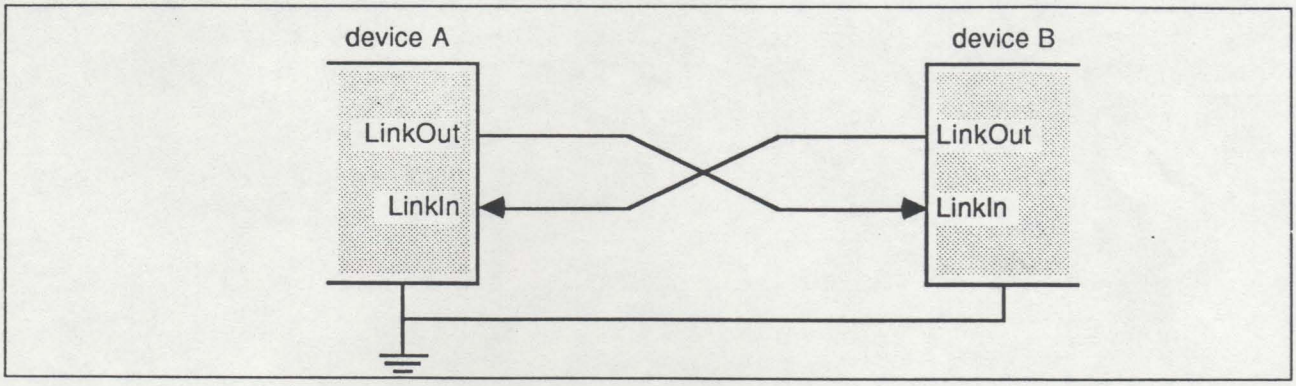


Figure A.1 A link

The two devices must have a common ground reference. Unbuffered links should only be used for connections between devices on the same circuit board, or between devices powered by the same supply within the same chassis. It is possible to connect unbuffered links between devices powered by separate supplies, or in different chassis, as long as the devices share a common ground reference and ground loops are eliminated. You must take care not to disconnect protective grounds.

A.1.1 Link protocol

Each byte sent is acknowledged by the receiving device. When device A sends a byte to device B, device B responds by sending an acknowledge packet to device A. The link is full duplex: both devices may transmit at the same time, data bytes and acknowledges being interspersed.

Pin	In/Out	Function	Pin No.
System Services			
Vcc, GND		Power supply and return	3,14
ClockIn	in	5MHz clock signal	8
Reset	in	Transputer reset	10
Analyse	in	Transputer error analysis	9
notError	out	Transputer error indicator	11
Links			
LinkIn0-3	in	INMOS serial link inputs to transputer	13,5,2,16
LinkOut0-3	out	INMOS serial link outputs from transputer	12,4,1,15
LinkspeedA,B	in	Transputer link speed selection	6,7

Table A.1 TRAM Pin designations

Notes:

1 Signal names are prefixed by **not** if they are active low; otherwise they are active high.

2 Details of the physical pin locations can be found in fig. B.2 and fig. C.2.

A.2 Transputer Modules (TRAMs)

TRAMS are small subassemblies of transputers and other components. They interface to each other via INMOS links, have a standard pinout, and come in a range of standard sizes. A full specification of the standard is available as INMOS Technical Note 29: *Dual InLine Transputer Modules (TRAMs)*. The standard TRAM interface signals are described in table A.1.

LinkOut0-3 Transputer link output signals. These outputs are intended to drive into transmission lines with a characteristic impedance of 100Ω. They can be connected directly to the **LinkIn** pins of other transputers or TRAMs.

LinkIn0-3 Transputer link input signals. These are the link inputs of the transputer on the IMS B407. Each input has a 10kΩ resistor to **GND** to establish the idle state, and a diode to **Vcc** as protection against ESD. They can be connected directly to the **LinkOut** pins of other transputers or TRAMs.

LinkSpeedA, LinkSpeedB These select the speeds of **Link0** and **Link1,2,3** respectively. Table A.2 shows the possible combinations for both the IMS B408 and the IMS B409.

LinkSpeedA	LinkSpeedB	Link0	Link1,2,3
0	0	10 Mbits/s	10 Mbits/s
0	1	10 Mbits/s	20 Mbits/s
1	0	20 Mbits/s	10 Mbits/s
1	1	20 Mbits/s	20 Mbits/s

Table A.2 Link speed selection

ClockIn A 5MHz input clock for the transputer. The transputer synthesises its own high frequency clocks. **ClockIn** should have a stability over time and temperature of 200ppm. **ClockIn** edges should be monotonic within the range 0.8V to 2.0V with a rise/fall time of less than 8ns.

Reset Resets the transputer, and other circuitry. **Reset** should be asserted for a minimum of 100ms. After **Reset** is deasserted a further 100ms should elapse before communication is attempted on any link. After this time, the transputer on the TRAM is ready to accept a boot packet on any of its links.

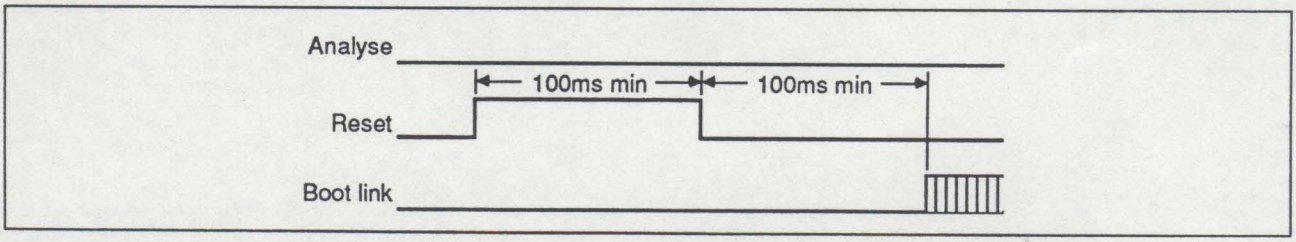


Figure A.2 Reset timing

Analyse is used, in conjunction with **Reset**, to stop the transputer. It allows internal state to be examined so that the cause of an error may be determined. **Reset** and **Analyse** are used as shown in figure A.3. A processor in analyse mode can be interrogated on any of its links.

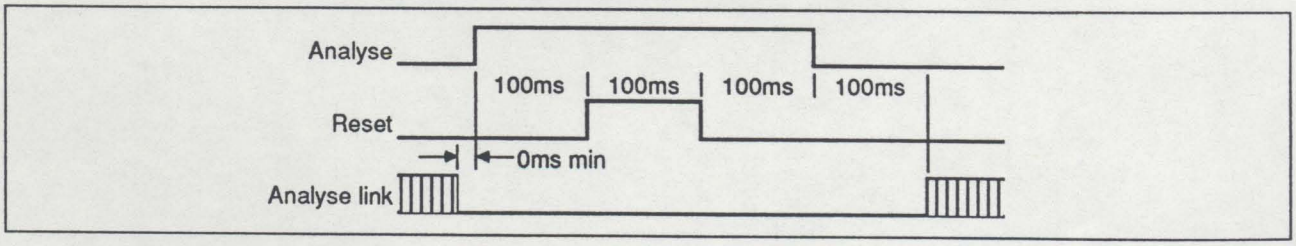


Figure A.3 Analyse timing

notError An open collector output which is pulled low when the transputer asserts its Error pin. **notError** should be pulled high by a 10kΩ resistor to **Vcc**. Up to 10 **notError** signals can be wired together. The combined error signal will be low when any of the contributing signals is low.

B IMS B408 Specification

B.1 Mechanical details

Figure B.1 indicates the vertical dimensions of a single IMS B408 and Figure B.2 is an outline drawing the of the IMS B408. Note that the component height includes the height taken up by a cable plugged into the pixel port connector. This means that the IMS B408 on a motherboard occupies more than one card slot in a 0.8in pitch card cage.

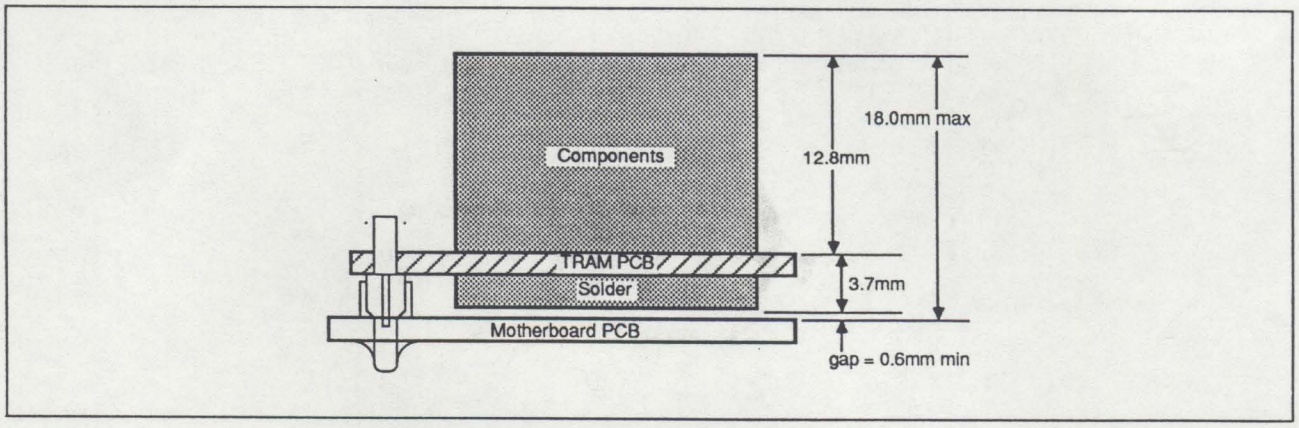


Figure B.1 IMS B408 height specification

B.2 Specification

TRAM feature		Unit	Notes
Transputer type	IMS T800-20		
Number of INMOS serial links	4		
Amount of DRAM	2.25	Mbyte	
DRAM "wait states"	1		
Memory cycle time	200	ns	
Subsystem controller	No		
Peripheral circuitry	Pixel Port		
Parity	No		
Size (TRAM size)	8		
Length	3.66	inch	
Pitch between pins	3.30	inch	
Width	8.75	inch	
Component height above PCB	12.8	mm	1
Component height below PCB	3.0	mm	2
Weight	215	g	
Storage temperature	0–70	deg C	
Operating temperature	10–40	deg C	3
Power supply voltage (Vcc)	4.75–5.25	Volt	
Power consumption	18	W	4

Table B.1 IMS B408 specification

Notes:

1 This dimension is larger than is normally stated for TRAMs because of the requirement to connect to the pixel bus.

2 This dimension includes the thickness of the PCB.

3 The figure quoted refers to the ambient air temperature.

4 The power consumption is the worst case value obtained when a sample of IMS B408 TRAMs were tested (running a program that utilised all four links and accessed memory simultaneously) at a supply voltage (Vcc) of 5.25 V.

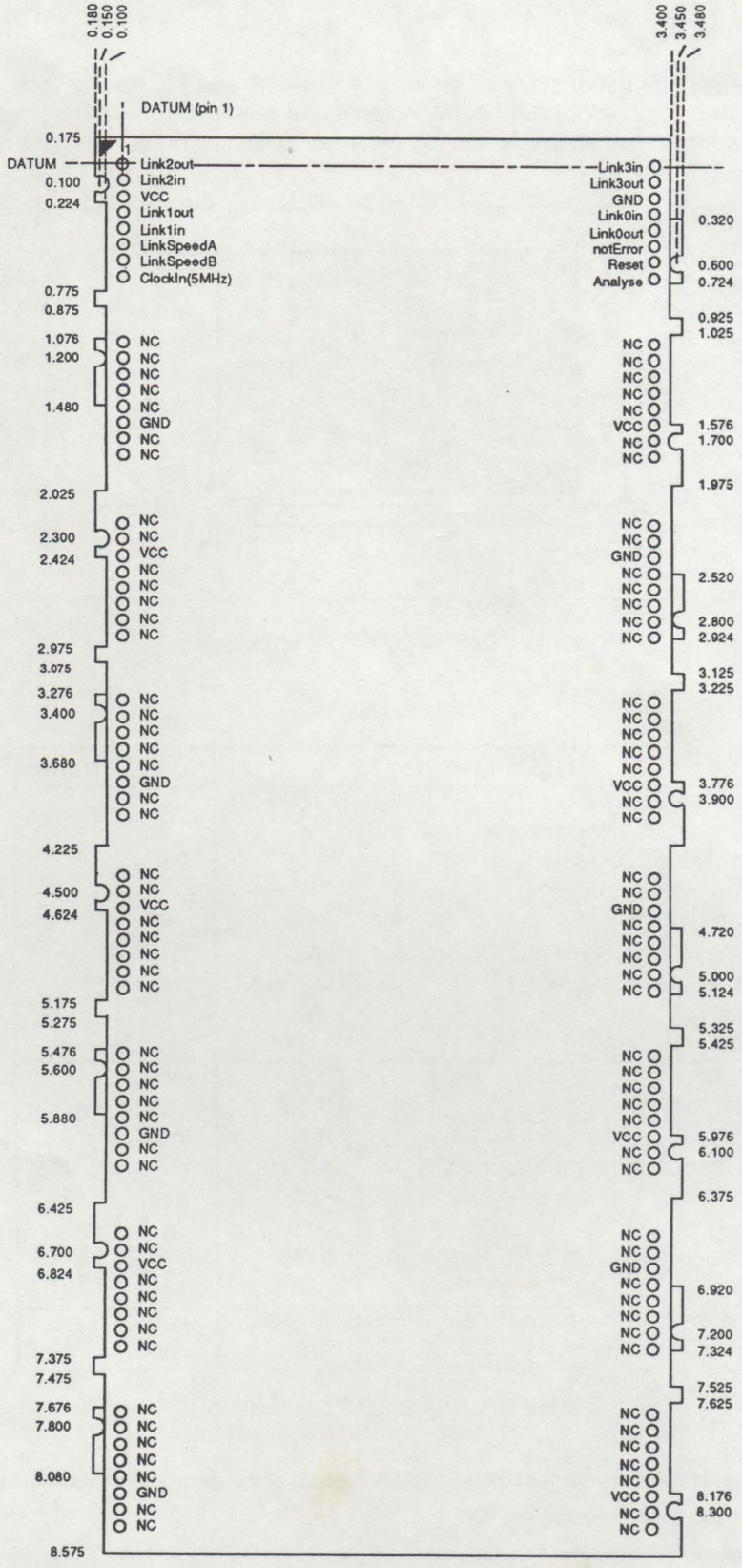


Figure B.2 IMS B408 outline drawing (All dimensions in inches)

C IMS B409 Specification

C.1 Mechanical details

Figure C.1 indicates the vertical dimensions of a single IMS B409 and Figure C.2 is an outline drawing the of the IMS B409. Note that the component height includes the height taken up by a cable plugged into a pixel bus input. This means that the IMS B409 on a motherboard occupies more than one card slot in a 0.8in. pitch card cage.

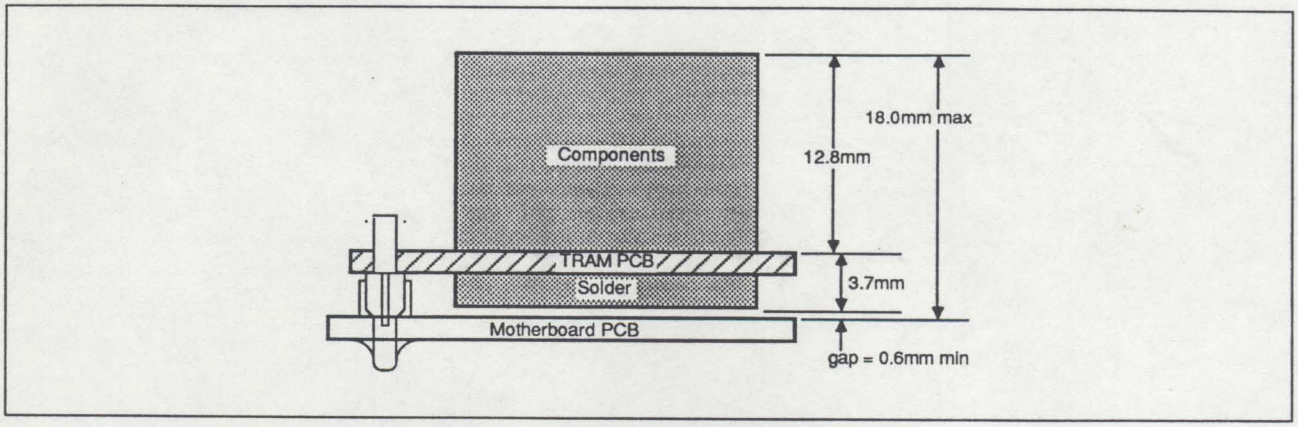


Figure C.1 IMS B409 height specification

C.2 Specification

TRAM feature		Unit	Notes
Transputer type	IMS T222-20		
Number of INMOS serial links	4		
RAM size	4	kbyte	
Memory cycle time	50	ns	
Subsystem controller	No		
Peripheral circuitry	VTG 3 Display channels		
Parity	No		
Size (TRAM size)	8		
Length	3.66	inch	
Pitch between pins	3.30	inch	
Width	8.75	inch	
Component height above PCB	12.8	mm	1
Component height below PCB	3.0	mm	2
Weight	185	g	
Storage temperature	0–70	deg C	
Operating temperature	10–40	deg C	3
Power supply voltage (Vcc)	4.75–5.25	Volt	
Power consumption	18	W	4

Table C.1 IMS B409 specification

Notes:

1 This dimension is larger than is normally stated for TRAMs because of the requirement to connect to the pixel bus.

2 This dimension includes the thickness of the PCB.

3 The figure quoted refers to the ambient air temperature.

4 The power consumption is the worst case value obtained when a sample of IMS B409 TRAMs were tested (running a program that utilised all four links and accessed memory simultaneously) at a supply voltage (Vcc) of 5.25 V.

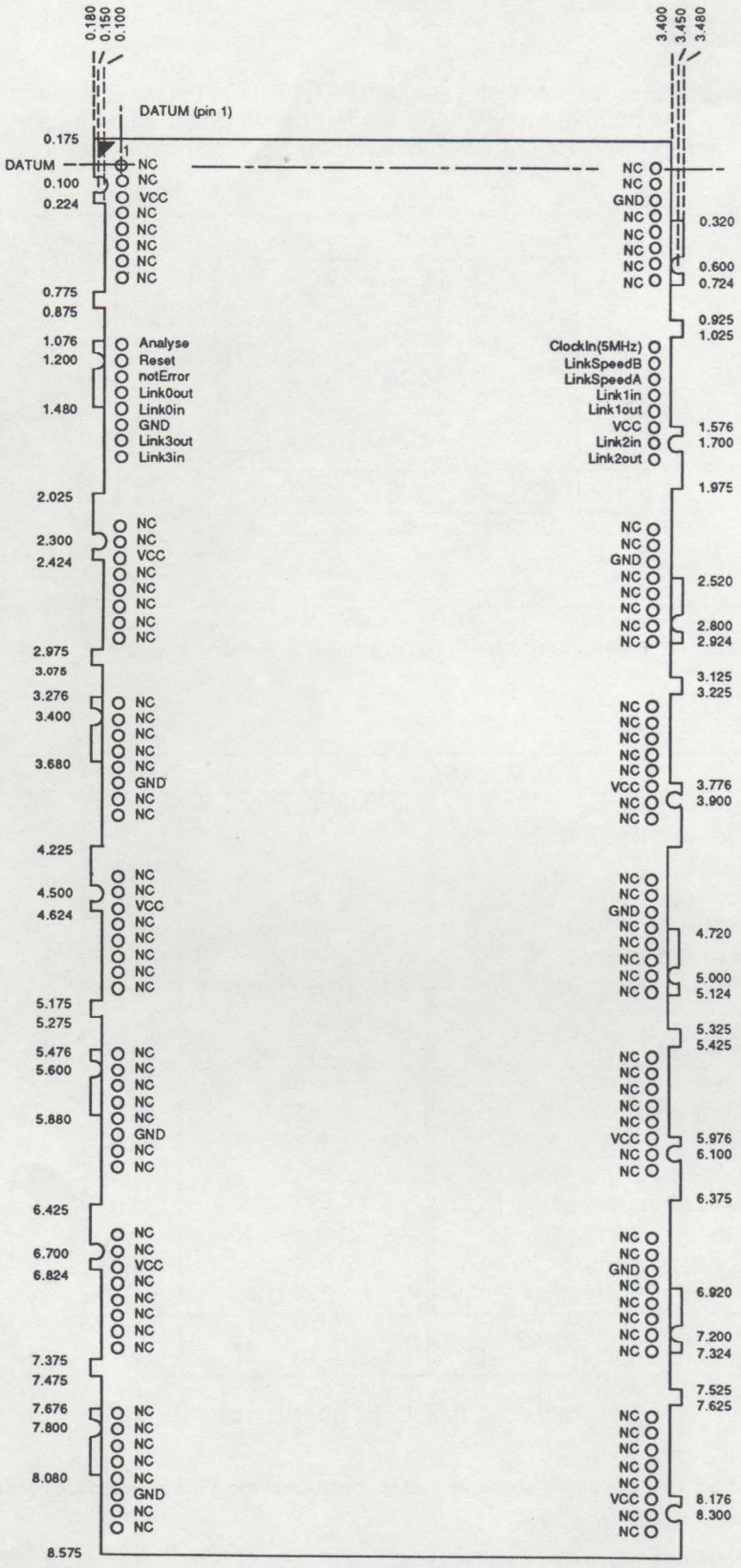


Figure C.2 IMS B409 outline drawing (All dimensions in inches)

D Availability of mating parts

This is not an exhaustive list: note that INMOS does not guarantee that these descriptions and part numbers are correct.

Crystal Oscillators In the UK, crystal oscillators are available from (among others) *IQD Ltd., North Street, Crewkerne, Somerset, TA18 7AR, England.* It should be possible to find a local supplier in most contries as this type of device is widely used.

SMB connectors Suitable parts are: *Greenpar 65201C22*, and *Radiall 114-005*. Similar parts which may also be suitable can be obtained from other connector manufacturers.

IDC connectors 60-way, 0.1in. pitch IDC connectors (and cable) can be obtained from most connector manufacturers. There is a variety of polarisation styles: most are suitable for use with the IMS B408 and IMS B409.

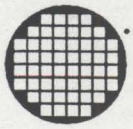
E References

1 *Transputer Reference Manual*
INMOS Limited
Prentice Hall 1988

2 *Technical Note 29: Dual Inline Transputer Modules (TRAMs)*
Paul Walker
INMOS Limited 1987

3 *Technical Note 18: Connecting INMOS Links*
Michael Rygol and Trevor Watson
INMOS Limited 1987

inmos



INMOS Limited
1000 Aztec West
Almondsbury
Bristol BS12 4SQ
UK
Telephone (0454) 616616
Telex 444723

INMOS Corporation
PO Box 16000
Colorado Springs
CO 80935
USA
Telephone (303) 630 4000
TWX 910 920 4904

INMOS GmbH
Danziger Strasse 2
8057 Eching
Munich
West Germany
Telephone (089) 319 10 28
Telex 522645

INMOS Japan K.K.
4th Floor No 1 Kowa Bldg
11-14 Akasaka 1-chome
Minato-ku
Tokyo 107
Japan
Telephone 03-505-2840
Telex J29507 TEI JN
Fax 03-505 2844

INMOS SARL
Immeuble Monaco
7 rue Le Corbusier
SILIC 219
94518 Rungis Cedex
France
Telephone (1) 46.87.22.01
Telex 201222