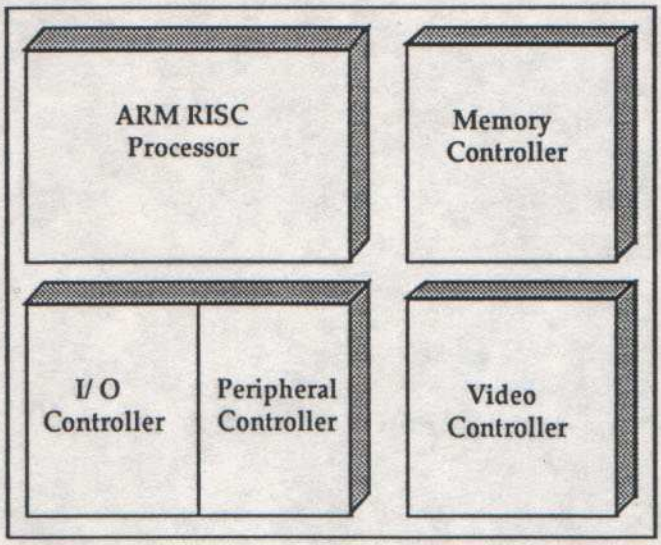
ARM250 - Integrated Controller

The ARM250 is a complete computer system on a chip - comprising a 32-bit RISC processor, a memory controller with DRAM interface, a bit-mapped video controller and an I/O controller. It is suitable for a wide range of cost-sensitive embedded control, portable and consumer games applications - particularly (but not only) those which require a video display.

The device is designed to drive up to 4 Mbytes of DRAM directly at 12 MHz, and at these speeds can sustain about 10 MIPS (peak).

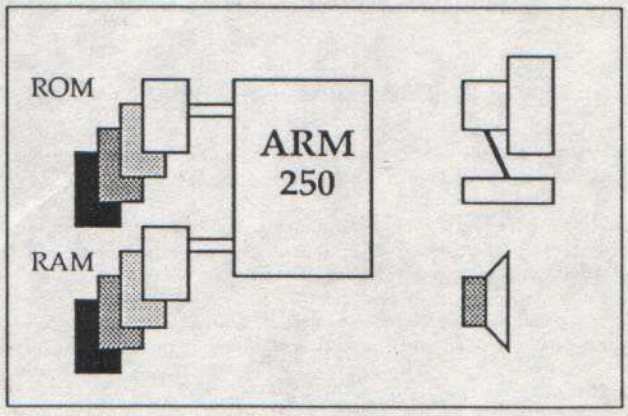


- ARM 32 bit RISC core
- Memory Controller
- Video Controller
- Flexible I/O Controller
- Reduced system cost
- 10 MIPS (peak) @ 12 MHz
- Small footprint 160 PQFP package
- Low power consumption

Applications

- Portable Computers
- Desktop Computers
- Games Consoles
- Palmtop Computers
- Low cost X terminals

Example - Video games console



Change Log	Issue	Date	By	Change
	A	11 Aug 92	ST/PM	Transferred to Frame DTP

Doc No : ARM DDI 0030A
Issued : Aug 1992

1. Overview

The ARM250 is a highly integrated microcontroller based on a *32 bit RISC processor.* In addition to the central processor the ARM250 includes :

a *Memory Controller* with direct interface to DRAM and ROM and support for memory mapping and protection.

a *Video controller* for bit-mapped display with integral palette. On-chip video and sound DACs allow interfacing to video monitors with minimal external circuitry..

an *I/O controller* interfaces to a wide range of standard peripheral chips. Provides timers, serial port, parallel port and a range of interrupt and configuration inputs.

The I/O subsystem contains some on-chip functions such as timers, and in addition can directly drive industry standard peripheral chips. The video subsystem is capable of a wide variety of display modes up to Super VGA resolutions (600 X 800 pixels), as well as adding impressive sound capability

The device is based on 4 standard chips from the ARM RISC chip set - ARM2aS, MEMC1a, VIDC and IOC. In order to reduce the size of this datasheet, the datasheets for these parts are not duplicated here as they retain identical functionality (but with improved performance) within ARM250. The relevant datasheets, available on request from Advanced RISC Machines Ltd, are as follows :

- ARM2aS Datasheet
- MEMC1a Datasheet
- VIDC Datasheet
- IOC Datasheet

The main blocks in ARM250 are the processor (ARM2aS), memory controller (MEMC), video controller (VIDC), input/output controller (IOC) and I/O extension block (IOEB). The first four of these blocks are based on standard chips from the ARM RISC chip set. Some extra logic, known as IOEB, is used in the I/O system and is documented in this datasheet.

A simplified block diagram appears in Figure 1. A list of relevant documentation is given later in this datasheet and a detailed block diagram of ARM250 appears at the end of this chapter.

1.1 Processor

The processor (ARM2aS) is an implementation of the Advanced RISC Machines' ARM processor which is a general purpose 32-bit single-chip microprocessor. The architecture is based on Reduced Instruction Set Computer (RISC) principles and provides a high instruction throughput and an excellent real-time interrupt response from a small silicon area.

Pipelining is employed so that all parts of the processing and memory systems can operate continuously. Typically, while one instruction is being executed, its successor is being decoded, and a third instruction is being fetched from memory.

The ARM instruction set has proved to be a good target for compilers of high-level languages. Where required for critical code segments, assembly code programming is also straightforward, unlike some RISC processors which depend on sophisticated compiler technology to manage complicated instruction inter-

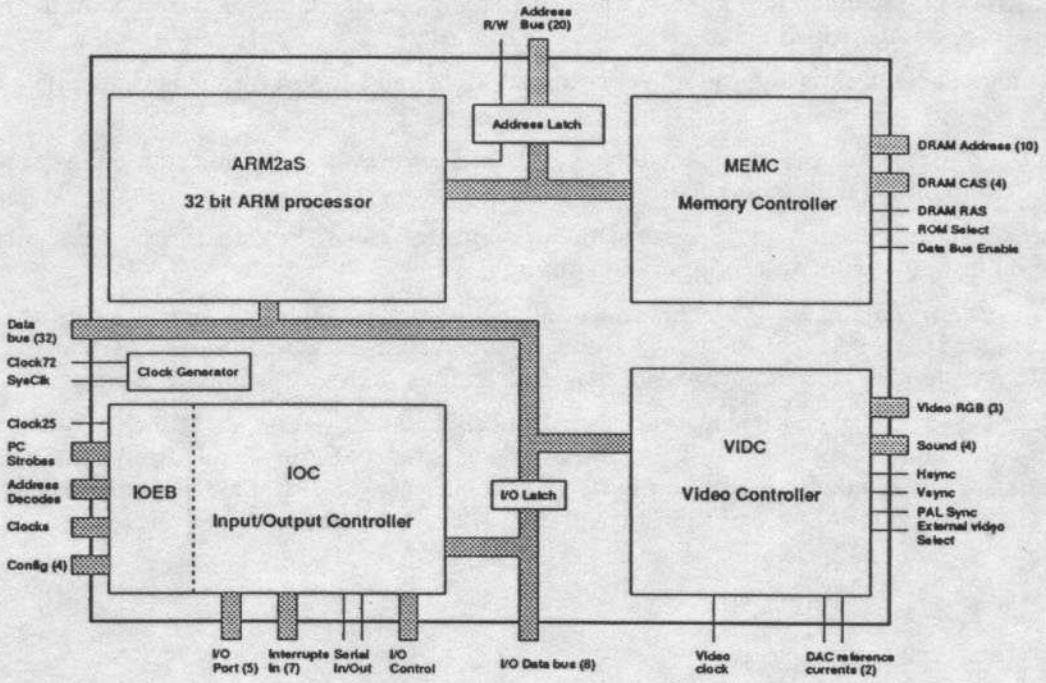


Fig 1 : Simplified Block Diagram

dependencies.

ARM Ltd provides a complete cross development toolkit for the ARM, consisting of the tools and documentation needed to develop software for ARM-based systems, including simulation and debug tools which allow ARM targeted software to be debugged and evaluated in more familar, fully hosted, software development environments.

1.2 Memory Controller

The memory controller (MEMC) acts as the interface between the ARM processor, the video controller, the I/O controller, read-only memories and dynamic memory devices, providing all the critical system timing signals and partitioning the memory map.

ARM250 is designed principally for use with two types of memory - read-only memory (ROM) and dynamic RAM (DRAM). In both cases, a 32 bit data bus is used to transfer memory data. A 20 bit address bus is used to supply addresses for ROMs and for I/O port memory mapping. Internally, a 26 bit address bus is used but only 20 bits are brought out to pins. To interface to up to 4 megabytes of DRAM, ARM250 provides multiplexed addresses and row and column strobes which drive the DRAM chips directly.

The memory controller also contains a Logical to Physical Address Translator which maps the physical memory into a 32MByte logical address space (with three levels of protection) allowing virtual memory and multi-tasking operations to be implemented. Fast "page-mode" DRAM accesses are used to maximise memory bandwidth.

The memory controller provides a Direct Memory Access (DMA) system for the video controller. It contains pointer registers to 3 separate RAM buffers which it automatically transfers to the video controller when required.

1.3 Video Controller

The video controller (VIDC) uses buffers in RAM to generate a video display with a hardware cursor and stereo sound. The memory controller is responsible for coordinating transfers from these buffers to the video controller by DMA. The video controller requests data from the memory when required, and holds it in one of three internal buffers before using it. Data from the internal buffers is serialised and then presented to a digital to analogue converter (DAC). There are DACs for both sound and video, allowing driving of video monitors and audio systems with minimal external buffering. The video data also passes through a colour look-up palette before being output.

Three forms of video synchronisation pulses are generated with programmable timing and polarity. The video clock may be selected from a variety of sources and appears on pins to allow the video to be "gen-locked" to an external signal.

DMA transfers move video, cursor and sound data in blocks of four 32-bit words, allowing efficient use of page-mode DRAM without locking up the system data bus for long periods. The video controller is programmable, offering a very wide choice of display formats. The pixel rate can be selected from a range of sources and programmable prescaling applied. Video data can be serialised to either 8, 4, 2, or 1 bit per pixel. The horizontal timing parameters can be controlled to units of 2 pixels, and the vertical timing parameters can be controlled to units of a raster. The colour look-up palette which drives the three on-chip video DACs is 13 bits wide, offering a choice of 4096 colours or an external video source.

The hardware cursor is 32 pixels wide and may be any number of rasters high. It can be positioned anywhere on the screen. Three simultaneous colours (again from a choice of 4096) are supported, and any pixel can be defined as transparent, making it possible to define cursors of many shapes.

The sound system incorporates an exponential DAC and stereo image table for the generation of high quality sound. Up to 8 channels are supported, each with a separate stereo position.

1.4 I/O Controller

The I/O controller is based on a chip known as IOC together with some additional logic, known as IOEB, which is used to provide additional functionality. The I/O system is based around an 8 bit I/O bus which connected to the main data bus by latches which allow the memory and I/O systems to operate asynchronously. This bus can be upgraded to 16 bits with the addition of two external latches.

The I/O controller contains three 16 bit timers which are clocked at 2 MHz. Two of the timers are general purpose and may be used to provide timed interrupts. The third provides the clock for the serial interface which provides a UART-like interface for asynchronous serial data at baud rates up to 31250.

The memory controller defines part of the memory map as containing memory mapped I/O devices and the I/O controller controls accesses made in this area of memory. It provides four different cycle times for use with peripheral devices of varying speeds. The I/O part of the memory map is further divided by address decoding to provide a number of pins which are active in various parts of the I/O space. One of these areas is designed for interfacing directly to one of the highly integrated PC combination chips which provides interfaces to hard and floppy discs, parallel printer port, etc. It is also possible to bypass the I/O controller for certain I/O accesses so that the cycle timing is determined by external logic.

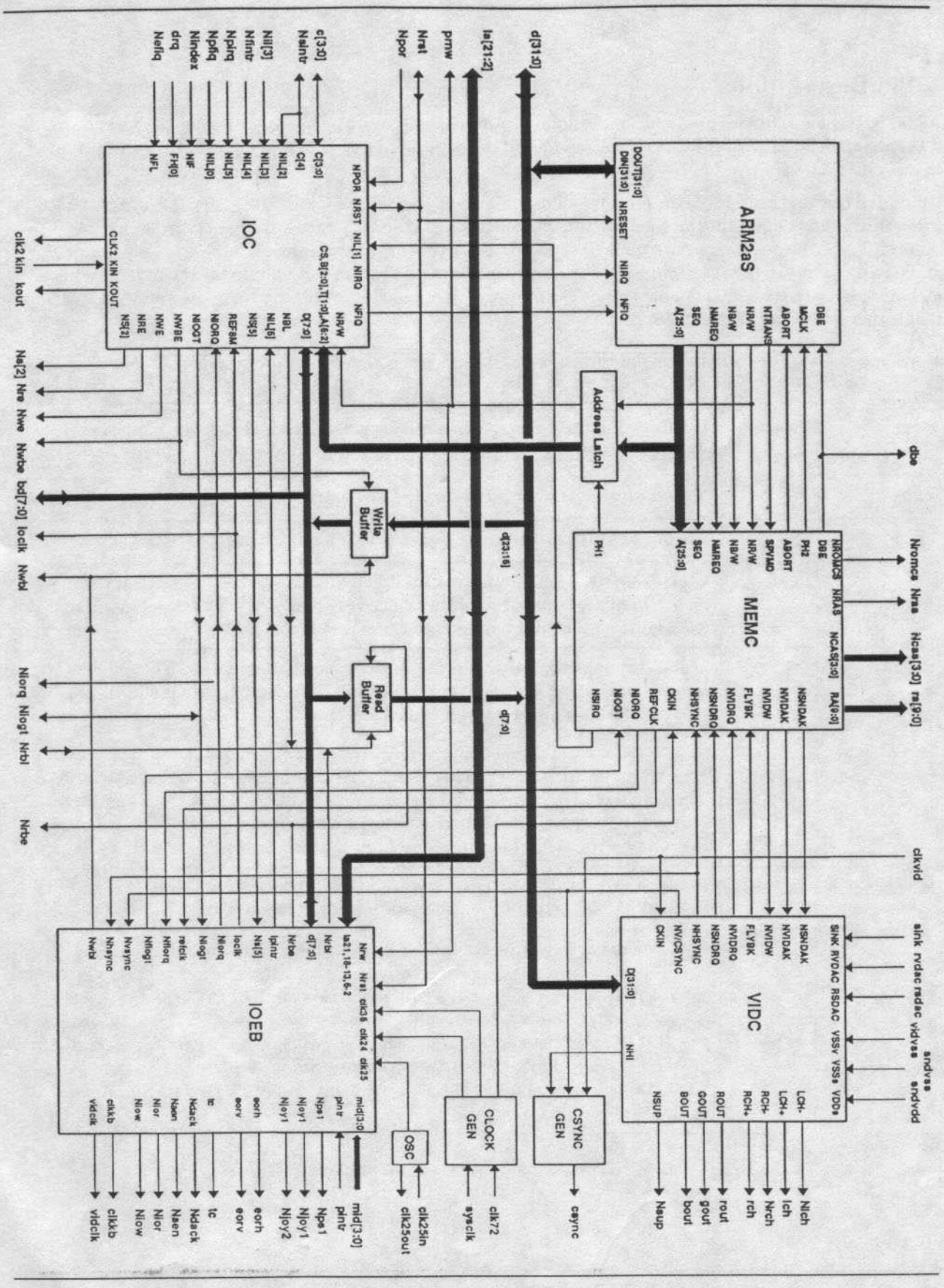
The I/O controller also coordinates interrupts for the whole system. There are a number of external interrupt pins, both edge and level triggered, and internal interrupts from the memory and video controllers, timers and serial interface. These interrupts can be selectively masked in the I/O controller and are routed to one of two interrupt inputs on the processor. These two interrupts differ only in priority; one is meant for use in low latency situations, the other where interrupt latency is less critical. There are also a number of pins which form I/O ports, readable and/or writeable by the processor. These may be used to read external status and control external logic.

1.5 System Clocks

There are a number of clocking options for ARM250. The basic clock is a 72 MHz input which is internally divided down to 24 and 36 MHz. These form two of the three possible video clock frequencies. The chip also has a pair of pins which form an oscillator with an external crystal in the range 1 to 26 MHz (typically 25.175 MHz) which is used to form the third video clock source. The 24 MHz signal is further divided to generate 12 and 8 MHz signals. These are both available on output pins and the 8MHz signal also forms the clock for the I/O system. The 36 MHz signal is divided by three in the memory controller to make a 12 MHz clock for the processor and memory system.

It is possible to bypass the 36 MHz signal to the memory controller and drive the processor and memory system with an external clock at a higher speed. For example, a 48 MHz clock would result in the processor being clocked at 16 MHz. This would result in a system capable of over 7 MIPS using standard DRAMs.

The video clocks of 24, 25 and 36 MHz provide for industry standard video displays up to SVGA resolution.


ARM2aS
DBE
MCLK
ABORT
NTRANS
NRW
NBW
NMREQ
SEQ
A[25:0]
DOUT[31:0]
DIN[31:0]
NRESET
NIRQ
NFIQ
MEMC
NROMCS
NRAS
NCAS[3:0]
RA[9:0]
PH1
PH2
SPVMD
NSNDAK
NVIDAK
NVIDW
FLYBK
NVIDRQ
NSNDRQ
NHSYNC
CKIN
REFCLK
NIORQ
NIOGT
NSRQ
IOC
CLK2 KIN KOUT
NPOR NRST NIL[1] NIRQ NFIQ
NRW
NBL
NIL[6]
NS[3]
REFSM
NIORQ
NIOGT
NWBE
NWE
NRE
NS[2]
IOEB
Nrw Nrst clk36 clk24 clk25
ipintr
retclk
Nfiorq
Nfiogt
Nvsync
Nhsync
Nvbl
Nrbl
eorh
eorv
tc
Ndack
Naen
Nior
Niow
clkkb
vidclk
mid[3:0]
pintr
Npst
Njoy1
Njoy2
VIDC
SINK RVDAC RSDAC VSSv VSSs VDDs
LCH-
LCH+
RCH-
RCH+
ROUT
GOUT
BOUT
NSUP
Nich
Ich
Nrch
rch
rout
gout
bout
Nsup
NVI/CSYNC
CKIN
NHI
CLOCK GEN
CSYNC GEN
OSC
Address Latch
Write Buffer
Read Buffer
dbe
clkvid
sink rvdac rsdac vidvss sndvss sndvdd
Nromcs Nras Ncas[3:0] ra[9:0]
clk2 kin kout
Ns[2] Nre Nwe Nvrbe bd[7:0] lockin Nvbl
Niorq Niogt Nrbl
Nrbe
c[3:0]
Nseintr
Nii[3]
Nfiintr
Npirq
Npfiq
Nindex
Nefiq
drq
Nfiq
NIF
FH[0]
NFL
prnw
Nrst
Npor
la[21:2]
d[31:0]
d[23:16]
d[7:0]
clk72
syncclk
clk25in
clk25out
csync

2. Pin Descriptions

ARM250 has a mixture of analogue and digital inputs and outputs. The analogue inputs (IA) and outputs (OA) are associated with DACs in VIDC and a full description of their use and characteristics is provided in the VIDC data sheet.

All digital outputs are CMOS drivers and all digital inputs have either CMOS (IC), TTL (IT) or Schmitt (IS) input thresholds depending on their intended use. The output drivers have a range of current driving capabilities. The weakest drivers can source or sink 4mA (O4), the majority of outputs have 8mA capability (O8) and two outputs can provide 16mA (O16). Some of the 8mA drivers are slew-limited (OS8) to reduce instantaneous current requirements when many outputs are changing. Some outputs are open-drain (OD4,OD8), that is they only ever sink current .

Name	Type	Description
Nre	O8	**I/O Read Enable.** An active low output strobe which is used to control peripheral read accesses. It is driven by the IOC output NRE
kin	IC	**Keyboard In.** Serial input to the IOC keyboard input (KIN).
kout	O4	**Keyboard Out.** Serial output from the IOC keyboard output (KOUT).
clk72	IC	**Clock 72MHz.** This input provides the main clock for ARM250. It should be a 72 MHz square wave of 50% duty cycle (+/- 5%). It is divided down internally to 24 and 36 MHz and these frequencies are used in various parts of the device.
Ntest	IC	**Test.** A test input which should be high for normal operation. This pin has an internal pull-up resistor and so may be left unconnected. Driving this input low will put the chip into a test mode and this should not be done while the device is in use.
Nrst	IS/OD8	**Reset.** This bidirectional pin may be driven low as an input to reset ARM250. It is internally connected to the reset inputs of ARM2aS, MEMC, IOC and IOEB. The output is driven low whenever the power-on reset pin (**Npor**) is low and the pin will normally have an external pull-up resistor fitted.
Npor	IS	**Power-on Reset.** This active low input causes the **Nrst** pin to go low and so reset ARM250. It is also connected to the NPOR input of IOC.
sysclk	IC	**System Clock.** This input is optionally used to drive the MEMC clock input (CKIN). If the pin is held permanently high, the MEMC clock is derived from the internal 36 MHz source. If a square wave is applied to the pin with a frequency greater than 4 MHz (ie clk72 divided by 18) then this signal is used as the MEMC clock. It is anticipated that the frequency used will be greater than 36 MHz to make the processor and memory run faster than they would otherwise. This pin has an internal pull-up resistor and can be left unconnected if desired.

Name	Type	Description
clk25out	O4	**Clock25 Out.** This pin forms one of a pair (with **clk25in**) which form the basis of a crystal oscillator circuit. A resistor, two capacitors and a crystal are required to complete the circuit which will operate in the range 1 to 26 MHz.
clk25in	IC	**Clock25 In.** This pin forms an oscillator circuit with **clk25out**. If the oscillator function is not required, it may be driven by an external CMOS level signal. Internally, it is connected to the video clock selector circuit in IOEB.
mid[3:0]	IS	**Configuration Inputs.** These four pins provide four status inputs which may be read by the ARM processor. They are connected to a register in IOEB.
clkkb	O4	**Keyboard Clock.** This output is a 12 MHz square wave (ie **clk72** divided by 6).
eorh	O16	**Horizontal Sync.** This signal is the horizontal sync output of VIDC (NHSYNC) which has been passed through a programmable inversion in IOEB.
eorv	O16	**Vertical/Combined Sync.** This signal is the vertical/combined sync output of VIDC (NV/CSYNC) which has been passed through a programmable inversion in IOEB.
Njoy[1]	O4	**Joystick enable 1.** This active low signal is an address decode (in the I/O address space) produced by IOEB. It decodes an area which is effectively a single word. See the section on IOEB address decoding for more details.
Njoy[2]	O4	**Joystick enable 2.** Similar to **Njoy[1]** but decodes a different address.
Nras	OS8	**Row Address Strobe.** This is the NRAS signal from MEMC. It is used to clock a row address into DRAM devices.
Ncas[3:0]	OS8	**Column Address Strobes.** These four signals are the NCAS[3:0] outputs of MEMC. They are used to clock a column address into DRAM devices.
ra[9:0]	OS8	**RAM Addresses.** These ten signals are the RA[9:0] outputs of MEMC. They provide the row and column addresses for DRAM devices.
dbe	O4	**Data Bus Enable.** This output is active high while the ARM processor is driving the external data bus. It is the MEMC DBE signal and may be used with an external inverter to provide the (active low) write enable signal for DRAM devices.
d[31:0]	IT/OS8	**Data Bus.** These 32 bidirectional pins form the main data bus. They are active while the ARM is performing memory and I/O accesses and when DMA operations are in progress to move data from external RAM to VIDC.
Nromcs	O4	**ROM Select.** This active low output is the MEMC NROMCS signal. It is low while the processor is accessing that part of the memory map which is allocated to ROM.

Name	Type	Description
prnw	OS8	**Read/not Write.** This output is the ARM2aS NR/W signal which has been latched and inverted. It is intended as a read/write signal for peripheral devices and a high level indicates a read cycle.
la[21:2]	OS8	**Latched Addresses.** These outputs are a subset of the ARM2aS address lines which have been latched by the PH1 output of MEMC. They are valid throughout each memory access and are intended for use with ROM memories, I/O devices and external address decoding.
pintr	IT	**Printer Interrupt.** This level sensitive input is used to generate an interrupt input to IOC when a rising edge is applied. See the section on IOEB for more details.
Naen	O4	**PC Address Enable.** This active low output is generated by IOEB when a PC I/O access is performed in a certain address range. See the section on IOEB for more details.
Nior	O4	**PC I/O Read.** This active low output is generated by IOEB when a PC I/O read access is performed in a certain address range. See the section on IOEB for more details.
Niow	O4	**PC I/O Write.** This active low output is generated by IOEB when a PC I/O write access is performed in a certain address range. See the section on IOEB for more details.
Ndack	O4	**PC Data Acknowledge.** This active low output is generated by IOEB when a PC I/O access is performed in a certain address range. See the section on IOEB for more details.
tc	O4	**PC Terminal Count.** This active high output is generated by IOEB when a PC I/O access is performed in a certain address range. See the section on IOEB for more details.
rsdac	IA	**Sound DAC Reference Current.** This input is a reference current input for the VIDC sound DAC. It is connected to the VIDC RSDAC input.
sndvss	P	**Sound VSS.** This pin is the ground supply to the sound DAC in VIDC (VIDC pin VSSs). It must be externally connected to VSS.
sndvdd	P	**Sound VDD.** This pin is the power supply for the sound DAC in VIDC (VIDC pin VDDs). It must be at the same potential as VDD and decoupled to sndvss.
lch	OA	**Left Channel Sound.** This is one of a pair of outputs (with Nlch) forming the left channel sound output from VIDC (VIDC output LCH+).
Nlch	OA	**Inverse Left Channel Sound.** This is one of a pair of outputs (with lch) forming the left channel sound output from VIDC (VIDC output LCH-).

Name	Type	Description
rch	OA	**Right Channel Sound.** This is one of a pair of outputs (with Nrch) forming the right channel sound output from VIDC (VIDC output RCH+).
Nrch	OA	**Inverse Right Channel Sound.** This is one of a pair of outputs (with rch) forming the right channel sound output from VIDC (VIDC output RCH-).
rvdac	IA	**Video Reference Current.** This input is a reference current input for the VIDC video DAC. It is connected to the VIDC RVDAC input.
vidvss	P	**Video VSS.** This pin is the ground supply to the video DAC in VIDC (VIDC pin VSSv). It must be externally connected to VSS.
rout	OA	**Red Video.** This is the red output from the VIDC video DACs (VIDC output ROUT).
gout	OA	**Green Video.** This is the green output from the VIDC video DACs (VIDC output GOUT).
bout	OA	**Blue Video.** This is the blue output from the VIDC video DACs (VIDC output BOUT).
sink	IT	**Sink.** Active high input to the VIDC external synchronisation input (SINK).
Nsup	O4	**Supremacy Bit.** Active low output from the VIDC supremacy output (NSUP)
vidcclk	O4	**Video Clock Out.** This output comes from the video clock selector in IOEB. In most applications it will be connected directly to **clkvid** (pin 124).
clkvid	IC	**Video Clock In.** This input drives the VIDC clock input (CKIN). Note that this input has CMOS thresholds whereas the corresponding input on VIDC has TTL thresholds.
csync	O4	**Compensated Sync.** This output is a modified form of the vertical/combined sync output of VIDC. It is described in detail later in this datasheet.
Nindex	IT	**IF Interrupt.** This (falling) edge sensitive interrupt input is connected to an IOC interrupt input (NIF). It generates a slow interrupt (IRQ) when active.
drq	IT	**FH0 Interrupt.** This active high interrupt input is connected to an IOC interrupt input (FH[0]). It generates a fast interrupt (FIQ) when active.
Npirq	IT	**IL5 Interrupt.** This active low interrupt input is connected to an IOC interrupt input (NIL[5]). It generates a slow interrupt (IRQ) when active.
Npfiq	IT	**IL0 Interrupt.** This active low interrupt input is connected to an IOC interrupt input (NIL[0]). It can be programmed to generate a slow interrupt (IRQ) and/or a fast interrupt (FIQ) when active.
Nfintr	IT	**IL4 Interrupt.** This active low interrupt input is connected to an IOC interrupt input (NIL[4]). It generates a slow interrupt (IRQ) when active.

Pin Descriptions

Name	Type	Description
Nsintr	IT/OD4	**IL2/C4 Interrupt.** The input side of this pin is connected to both the C[4] control register input and the NIL[2] interrupt input of IOC. It may be used to generate slow and/or fast interrupts. The output side is controlled by the C[4] bit in the IOC control register.
Nefiq	IT	**FL interrupt.** This active low interrupt input is connected to an IOC interrupt input (NFL). It generates a fast interrupt (FIQ) when active.
clk2	O8	**Peripheral Clock.** This output is a 2MHz square wave generated by IOC (CLK2 output) which divides the I/O clock by 4.
c[3:0]	IT/OD4	**I/O Port.** These bidirectional pins behave like the IOC pins C[3:0].
Niorq	O8	**I/O Request.** This active low output indicates that an I/O access is requested. It is generated by IOEB and derived from the NIORQ output of MEMC.
Niogt	IT/ OD8	**I/O Grant.** As an input, this bidirectional pin is driven low to indicate that an I/O device has finished an access cycle which was started by the Niorq signal. It drives an input to IOEB which passes a modified form back to the MEMC NIOGT input. As an output, it is driven low when the IOC NIOGT output is low.
Nps1	O8	**Peripheral Select 1.** This active low output is an address decode generated by IOEB. It is decribed in detail later.
Nil[3]	IT	**IL3 Interrupt.** This active low interrupt input is connected to an IOC interrupt input (NIL[3]). It generates a slow interrupt (IRQ) when active.
Ns[2]	O8	**Peripheral Select 2.** This active low output is an address decode generated by IOC (NS[2] output).
ioclk	O8	**I/O Clock.** This output is an 8MHz square wave generated by IOEB. It is the reference clock for the I/O system and is derived from the clk72 input by dividing by 9.
bd[7:0]	IT/O8	**I/O Data Bus.** These bidirectional pins form the I/O data bus. Internally, they are connected to IOEB and IOC and also to the main ARM data bus via latches. The I/O system is described in more detail later.
Nwbe	O4	**Write Buffer Enable.** This output comes from the IOC NWBE output. It is used to enable an optional external latch which expands the I/O bus from 8 to 16 bits.
Nrbe	O4	**Read Buffer Enable.** This output comes from IOEB. It is used to enable an optional external latch which expands the I/O bus from 8 to 16 bits.
Nwbl	O4	**Write Buffer Latch.** This output comes from IOEB. It is used to control an optional external latch which expands the I/O bus from 8 to 16 bits.

Name	Type	Description
Nrbl	IT/OD8	**Read Buffer Latch.** The active low output of this bidirectional pin is controlled by IOEB and IOC. The input controls the internal read latch on the I/O data bus. Its function is described in more detail later.
Nwe	O8	**I/O Write Enable.** An active low output strobe which is used to time peripheral write accesses. It is driven by the IOC output NWE.

The following pins provide power to the device. Note that there are also specialised power pins for the analogue circuitry which are individually listed above.

VSS - 9, 18, 28, 49, 67, 89, 107, 126, 146

VDD - 10, 29, 50, 68, 90, 108, 127, 147

3. Block Level Wiring

This section describes the way in which the main blocks of ARM250 are wired. It should be read with reference to the block level diagram shown earlier in this document and the appropriate data sheets.

3.1 IOC wiring

Signal	Connects	Comment
REF8M	IOEB/ioclk, ioclk pin	Main I/O clock input
CLK8	<none>	not used
CLK2	clk2 pin	Peripheral timing clock (2 MHz)
NBL	Nrbl pin	Read buffer latch output
D[7:0]	I/O data bus	I/O data bus
NIORQ	IOEB/Niorq	I/O request input
NIOGT	Niogt pin	I/O grant output
T[1:0]	Addr latch (la[20:19])	Cycle type select inputs
NR/W	Addr latch	Read/write input (latched ARM/NR/W)
NSEXT	<none>	not used
NS[7:6]	<none>	not used
NS[5]	IOEB/Ns[5]	Peripheral select output (bank 5)
NS[4:3]	<none>	not used
NS[2]	Ns[2] pin	Peripheral select output (bank 2)
NS[1]	<none>	not used
B[2:0]	Addr latch (la[18:16])	Bank select inputs
CS	Addr latch (la[21])	Chip select input
A[6:2]	Addr latch (la[6:2])	Register select inputs
NRBE	<none>	not used
NWBE	Nwbe pin	Write buffer enable output
NRE	Nre pin	Read enable output
NWE	Nwe pin	Write enable output

Signal	Connects	Comment
NRST	Nrst pin	Reset input
NPOR	Npor pin	Power-on reset input
NIL[7]	tied HIGH	
NIL[6]	IOEB/Ipintr	Interrupt input
NIL[5]	Npirq pin	Interrupt input
NIL[4]	Nfintr pin	Interrupt input
NIL[3]	Nil3 pin	Interrupt input
NIL[2]	Nsintr pin	Interrupt input (see also C4)
NIL[1]	MEMC/NSIRQ	Interrupt input
NIL[0]	Npfiq pin	Interrupt input
NIF	Nindex pin	Interrupt input
IR	MEMC/FLYBK, VIDC/FLYBK	Interrupt input
FH[1]	tied LOW	
FH[0]	drq pin	Interrupt input
C[5]	input tied LOW	not used
C[4]	Nsintr pin	I/O port (see also NIL2)
C[3:0]	c[3:0] pins	I/O port
NIRQ	ARM/NIRQ	Slow interrupt output
NFIQ	ARM/NFIQ	Fast interrupt output
BAUD	<none>	not used
KIN	kin pin	Keyboard (serial) input
KOUT	kout pin	Keyboard (serial) output

3.2 MEMC wiring

Signal	Connects	Comment
A[25:0]	ARM/A[25:0], Addr latch	Address bus inputs
NR/W	ARM/NR/W, Addr latch	Read/write input
NB/W	ARM/NB/W	Byte/word input
NMREQ	ARM/NMREQ	Memory request input
SEQ	ARM/SEQ	Sequential access input
SPVMD	ARM/NTRANS	Supervisor mode input
PH1	Address latch	Phase 1 clock
PH2	ARM/MCLK	Phase 2 clock
DBE	ARM/DBE, dbe pin	Data bus enable output
ABORT	ARM/ABORT	Abort input
NIORQ	IOEB/Nfiorq	I/O request output
NIOGT	IOEB/Nfiogt	I/O grant input
CKIN	Clock generator	Main memory/processor clock input
REFCK	IOEB	Reference clock output
RESET	Nrst pin	Reset input (inverted)
RA[9:0]	ra[9:0] pins	RAM address outputs
NRAS	Nras pin	RAM row address strobe output
NCAS[3:0]	Ncas[3:0] pins	RAM column address strobe outputs
NROMCS	Nromcs pin	ROM select output
NVIDW	VIDC/NVIDW	Video controller write output
FLYBK	VIDC/FLYBK, IOC/IR	Flyback input
NHSYNC	VIDC/NHSYNC,IOEB/Nhsync	Horizontal sync input
NVIDRQ	VIDC/NVIDRQ	Video data request input
NVIDAK	VIDC/NVIDAK	Video data acknowledge output
NSNDRQ	VIDC/NSNDAK	Sound data request input
NSNDAK	VIDC/NSNDRQ	Sound data acknowledge output
NSIRQ	IOC/NIL[1]	Sound interrupt output

3.3 VIDC wiring

Signal	Connects	Comment
CKIN	clkvid pin	Video clock input
NVIDW	MEMC/NVIDW	Register write input
D[31:0]	d[31:0] pins	Data bus inputs
NVIDRQ	MEMC/NVIDRQ	Video data request output
NVIDAK	MEMC/NVIDAK	Video data acknowledge input
NSNDRQ	MEMC/NSNDRQ	Sound data request output
NSNDAK	MEMC/NSNDAK	Sound data acknowledge input
FLYBK	MEMC/FLYBK, IOC/IR	Flyback output
SINK	sink pin	External synchronisation input
NHI	CSYNC generator	Interlace marker output
NSD[3:0]	<none>	not used
NIBSEL	tied LOW	
NL/R	<none>	not used
RVDAC	rvdac pin	Video DAC reference current
ROUT,GOUT ,BOUT	rout,gout,bout pins	RGB Video outputs
NSUP	Nsup pin	Supremacy output
NHSYNC	MEMC/NHSYNC,IOEB/ Nhsync	Horizontal sync output
NV/CSYNC	IOEB/Nvsync, CSYNC gen.	Vertical/Combined sync output
NVED[3:0]	<none>	not used
RSDAC	rsdac pin	Sound DAC reference current
LCH+, LCH-	lch, Nlch pins	Left channel sound output
RCH+, RCH-	rch, Nrch pins	Right channel sound output
VSSs	sndvss pin	Sound DAC ground
VDDs	sndvdd pin	Sound DAC power
VSSv	vidvss pin	Video DAC ground

3.4 ARM2aS wiring

Signal	Connects	Comment
MCLK	MEMC/PH2	Processor clock input
NWAIT	tied HIGH	
NIRQ	IOC/NIRQ	Slow interrupt input
NFIQ	IOC/NFIQ	Fast interrupt input
NRESET	Rst pin	Reset input
ABORT	MEMC/ABORT	Abort input
DIN/ DOUT[31:0]	d[31:0] pins	Data bus (input/output)
NENOUT	d[31:0] pins	Data bus output enable
DBE	MEMC/DBE, dbe pin	Data bus enable input
NB/W	MEMC/NB/W	Byte/word output
NM[1:0]	<none>	Not used
A[25:0]	MEMC/A[25:0],Addr latch	Address bus outputs
ALE	tied HIGH	
NR/W	MEMC/NR/W, Addr latch	Read/write output
NMREQ	MEMC/NMREQ	Memory request output
NTRANS	MEMC/SPVMD	Translate address output
NOPC	<none>	Not used
SEQ	MEMC/SEQ	Sequential access output
NCPI	<none>	Not used
LOCK	<none>	Not used
CPA, CPB	tied HIGH	No coprocessors possible

4. The Input/Output System and IOEB

The majority of the I/O system functionality in ARM250 is provided by IOC. A block of extra logic known as IOEB (Input/Output Extension Block) provides a number of extra I/O facilities such as control registers and extra address decoding. A major function of IOEB is the decoupling of the I/O system from the memory system. This allows the memory and processor to run with a fast clock speed while the I/O system runs with a slower clock.

4.1 IOC Addressing

The address line wiring used in ARM250 ensures that IOC is selected when **Ia[21]** is high. The address lines **Ia[20:19]** determine the cycle type and **Ia[18:16]** determine which bank is selected. When bank 0 is selected to access the IOC internal registers, **Ia[6:2]** are used to select which register is accessed. Note that only banks 2 and 5 are useful in ARM250 as the other bank select signals are not used. The decode signal for bank 2 is available on a pin (Ns[2]). Bank 5 is further decoded in IOEB.

Address (hex)	Accessed device
3200000	IOC internal reg 0 (control reg)
3200004	IOC internal reg 1 (serial I/O)
3200008	IOC internal reg 2 (unused)
...	
3220000	Bank 2 slow
32A0000	Bank 2 medium
3320000	Bank 2 fast
33A0000	Bank 2 synchronous
...	
3250000	Bank 5 slow
32D0000	Bank 5 medium
3350000	Bank 5 fast
33D0000	Bank 5 synchronous

4.2 I/O Decoupling

All I/O accesses in ARM250 are memory mapped. MEMC predefines a part of the address space for I/O and when an access is made in this area, an I/O cycle is started. The I/O region of memory extends from 3000000 to 33FFFFF (hex) and an access in this range causes MEMC to assert its NIORQ signal. The I/O controller uses this to start the I/O cycle and then signals back to MEMC when the I/O cycle is complete by asserting the NIOGT signal on MEMC. MEMC deasserts its NIORQ signal and the I/O cycle is then com-

plete. IOC has a pair of signals which interface directly to the MEMC signals but this will only work if MEMC and IOC share the same clock. Some extra logic is required to run MEMC with a different (faster) clock than IOC and this logic is implemented in IOEB.

The NIORQ signal from MEMC is connected to IOEB (the Nfiorq input) and when it is asserted IOEB commences an I/O cycle. Two types of cycle are possible, PC I/O cycles and normal I/O cycles. The former are synchronised to the MEMC clock (REFCLK) and provide a restricted interface to an external peripheral chip. The latter are synchronised to the I/O clock and provide a more general interface. The two cycles are differentiated by address decoding. An I/O access in the range 3010000 to 302FFFF (hex) is a PC I/O access. This area actually appears four times in the memory map, the other regions beginning at 3090000, 3110000 and 3190000 (hex). All other accesses in the I/O space are normal I/O cycles.

The PC I/O accesses affect a set of 5 pins on ARM250 (**Ndack, Naen, tc, Nior** and **Niow**). The first three of these are effectively address decodes within the PC I/O region while the latter two are strobes for read and write accesses. These 5 pins remain in the idle state unless a PC I/O access is being made.

Range	Start	End	Ndack	Naen	tc	Nior	Niow
(Idle)		1	1	1	0	1	1
AEN	3010000	3011FFF	1	0	0	R	W
DACK	3012000	302BFFF	0	1	0	R	W
TC	302A000	302BFFF	0	1	1	R	W
LCD	302C000	302FFFF	1	1	0	1	1

The **Nior** and **Niow** pins indicate a read or write cycle respectively and are active low (ie R = 0 in read cycles, W = 0 in write cycles). Some timing diagrams for PC I/O cycles are shown below. The first two are read cycles the first of which shows a read cycle with the **Naen** signal active while the second one has the **Ndack** signal active. Note the **Nrbl** signal which falls near the end of the **Nior** strobe to hold the data in the read data latch. The second diagram shows two write cycles, the first with the **Ndack** signal active. The second write has the tc signal active as well as Ndack as the decoded areas for these two signals overlap. Note that in the write cycles, the **Nwbl** signal falls early in the cycle to hold the write data in the write data latch.

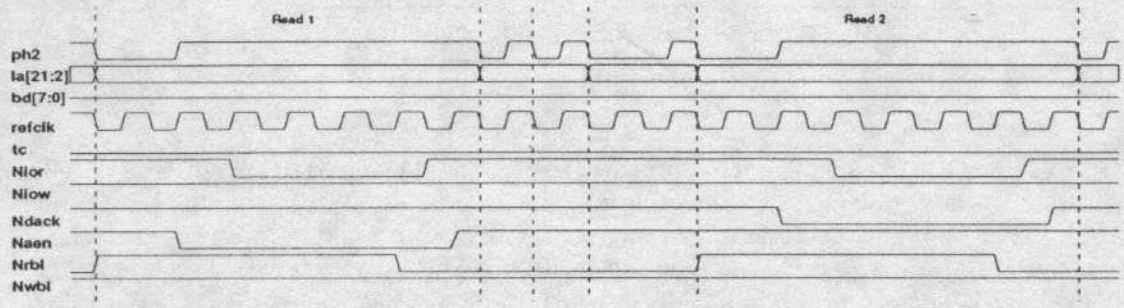


Fig 3 : PC I/O read cycles

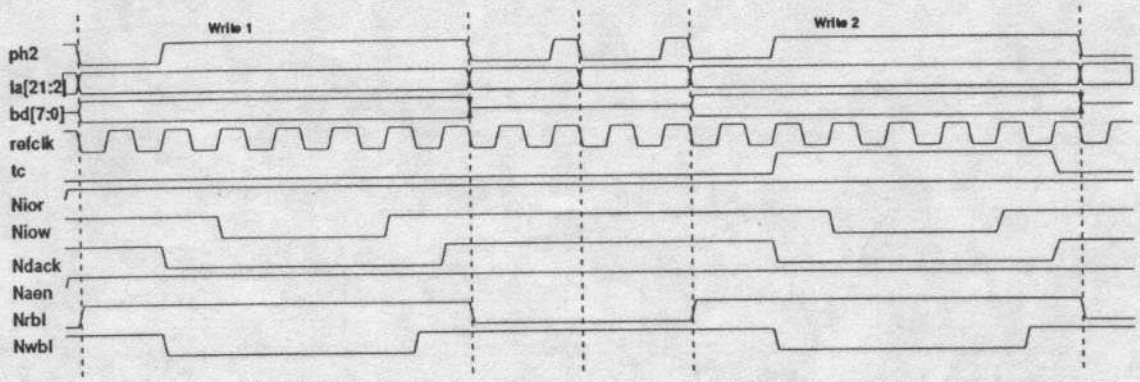


Fig 4: PC I/O write cycles

Normal I/O accesses are more complex than PC I/O accesses. The NIORQ signal from MEMC is synchronised to the I/O clock in IOEB and the sychronised signal drives the Niorq pin and the IOC NIORQ input. If the I/O address specifies that IOC should control the cycle (**la[21]** is high), then IOC will do that and generate NIOGT when the cycle is complete. This signal is then resynchronised in IOEB to the MEMC clock and fed back to MEMC's NIOGT input to terminate the cycle.

The two diagrams below show first a read and then a write cycle using the **Ns[2]** output from IOC. In this case two clocks are shown, **refclk** is the MEMC clock running at 12 MHz and **ioclk** is the I/O clock running at 8 MHz. These I/O cycles are longer than the PC I/O cycles because of the need to synchronise signals between these clocks at the beginning and end of the cycle. All signals connected to the peripheral device are synchronised to the 8 MHz clock. The first diagram shows a read cycle with data being gated onto the I/O bus by the **Nre** strobe. This data is latched by the **Nrbl** signal to hold it till the end of the cycle where it will be read by the processor.

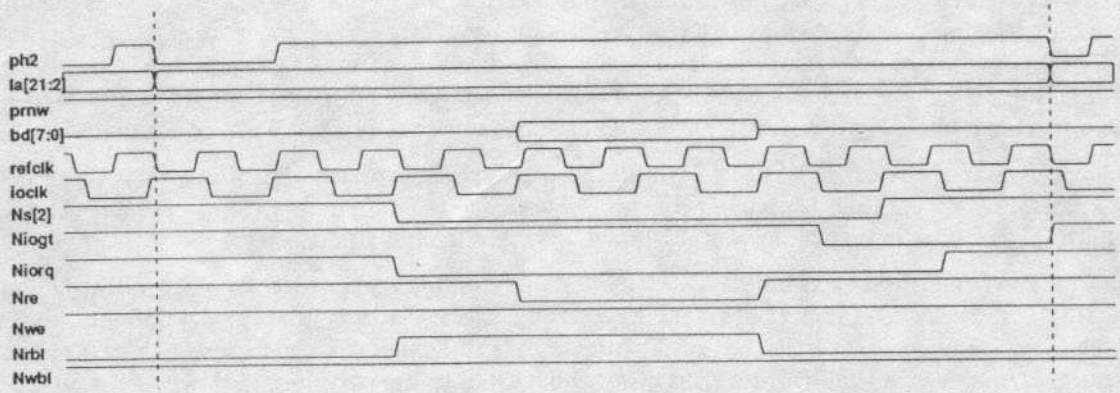


Fig 5: Normal I/O read cycle

The write cycle diagram shows **Nwbl** going low early in the cycle to hold the data in the write latch. The signal **Nwe** may be used as a strobe to clock the data into a peripheral device. The diagram also shows the **Niorq** and **Niogt** pins. In this case, the signal **Niogt** is being provided by IOC.

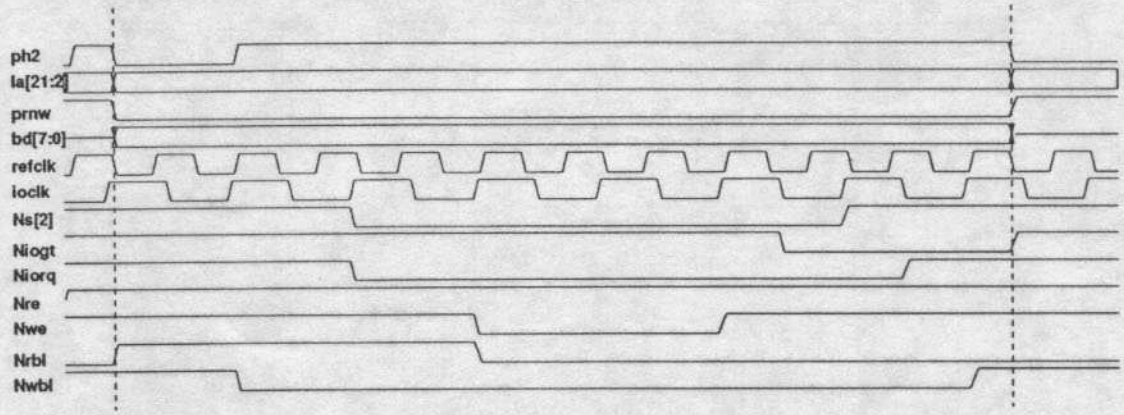


Fig 6 : Normal I/O write cycle

If the I/O address does not specify a PC I/O cycle or a normal I/O cycle controlled by IOC then it is assumed that external logic will control the cycle and provide the NIOGT signal to terminate the cycle. For this reason, the **Niogt** pin on ARM250 is bidirectional, allowing it to be driven externally. An external I/O controller can drive this pin low to terminate an I/O cycle. The range of addresses available for such controllers is any address in which **la[21]** is low and the address is not in one of the PC I/O ranges given above. Note that accessing an address for which no controller exists can cause the system to hang as no NIOGT will be provided to MEMC.

4.3 The I/O Data Bus

The I/O data bus in ARM250 is 8 bits wide and is internally connected to the main data bus by two 8 bit latches. The I/O bus can be expanded to 16 bits by the addition of two external 8 bit latches. In the unexpanded (8 bit) case, the latches are internal. One latch is for read data and connects the I/O bus to bits 7 through 0 of the main data bus. The other latch is for write data and connects to main data bus bits 23 through 16. If the expansion latches are fitted, the read latch connects externally to data bus bits 15 through 8 and the write latch to bits 31 through 24. The latches should be of type 74HCT573 or equivalent.

Reads from I/O space provide data in bits 0 to 7 of the word which is read and optionally in bits 8 through 15 if an expansion latch is fitted. Writes to I/O space must have the data in bits 16 through 23 of the word written with further data in bits 24 through 31 if an expansion latch is fitted.

The latch function is required because MEMC can interrupt an I/O cycle (which may be arbitrarily long) to perform a DMA cycle for the video controller. For a write cycle, this means that the data on the processor bus may change during the I/O cycle and so the output data is latched early in the cycle to ensure that the data on the I/O bus remains valid. For a read cycle, the data from the peripheral is latched late in the I/O cycle so that it remains valid while IOEB synchronises the NIOGT signal back to MEMC.

The figure below shows the configuration necessary to expand the I/O data bus to 16 bits. The Nrbl pin has an open drain driver and so requires an external pull-up resistor. External I/O controllers must drive this pin low when they have put valid read data on the I/O bus.

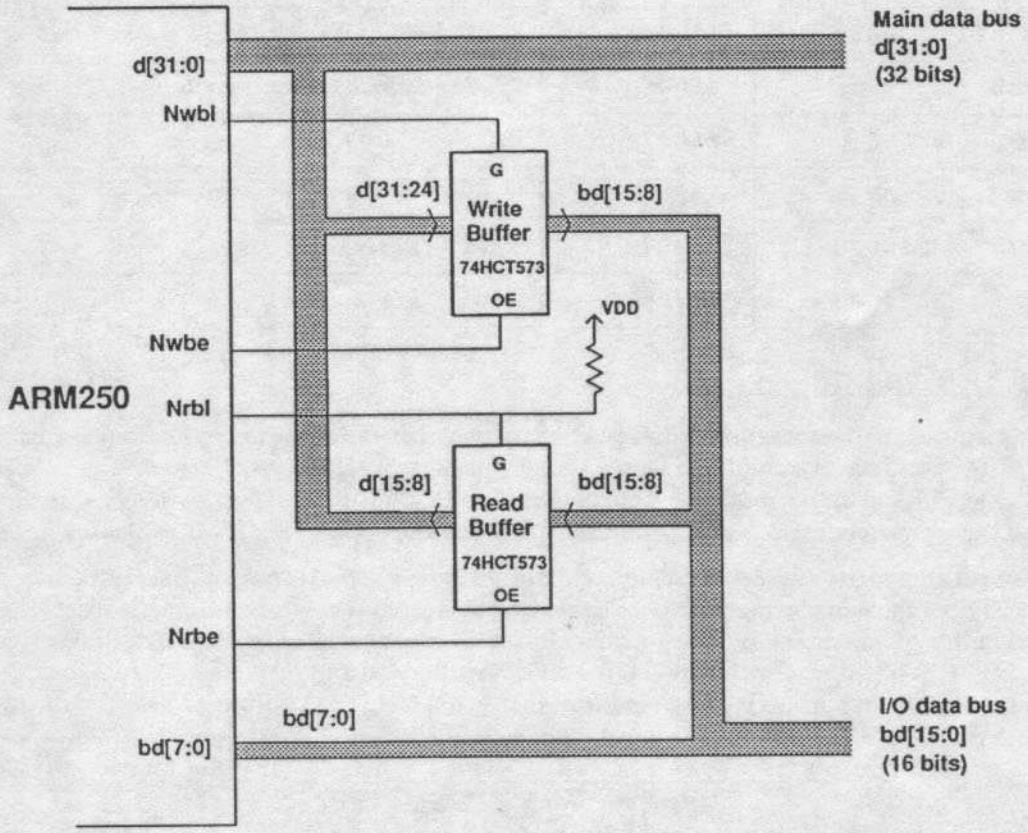


Fig 7: Expanding the I/O bus to 16 bits

4.4 IOEB Address Decoding

All decoding in IOEB is done using the latched processor address lines (from the address latch). The lines involved are **la[21]**, **la[18:13]** and **la[6:2]**. Some of the decoding also involves the NS[5] (bank 5) output of IOC and so the length of these cycles may be controlled by high order address lines **la[20:19]**.

The following decoding is done in IOC bank 5 using **la[6:2]** and the latched read/write signal from ARM. Note that complete decoding of all address lines is not used so the strobes may repeat at intervals through the bank 5 I/O space. It is recommended that they only be accessed at the addresses shown. All of the strobes below may have their duration controlled by using **la[21:19]**. The internal strobes (Ns5xxx), which control access to registers in IOEB, may all be accessed with fast IOC cycles.

Strobe	Base Address	End Address	Type	Used
Ns5w48	3250048	325004B	Write	IOEB control register
Ns5r50	3250050	3250053	Read	IOEB ID register
Ns5x58	3250058	325005B	R/W	IOEB interrupt latch
Ns5r70	3250070	3250073	Read	IOEB MID register
Njoy1	3250078	325007B	Read	Pin Njoy[1]
Njoy2	325007C	325007F	Read	Pin Njoy[2]
Nps1	3344000	3347FFF	R/W	Pin Nps1

4.5 IOEB Internal Registers

IOEB implements a small number of registers which may be read and written by the processor. There are two read registers, one read by the **Ns5r50** strobe and the other by the **Ns5r70** strobe. The former simply places the value F5 (hex) in the low order byte of word which is read. The latter places Fx (hex) in the low order byte of the word, where "x" is the current state of the **mid[3:0]** pins (**mid[0]** is the least significant bit.

There is one writeable register which is written with the **Ns5w48** strobe. It contains four bits which are taken from bits 16 to 19 of the word being written. The register is cleared whenever the chip is reset. The register controls part of the video system. Bit 3 (from data bus bit 19) controls the **eorv** output. If the bit is one, **eorv** outputs the NV/CSYNC signal from VIDC. If the bit is zero, **eorv** outputs inverted NV/CSYNC. Bit 2 of the register controls the **eorh** output. If the bit is zero, **eorh** is the NHSYNC output of VIDC. If the bit is one, **eorh** is inverted NHSYNC. Bits 1 and 0 control what is fed to the **vidclk** output as follows :

Bit 1	Bit 0	vidclk
0	0	24 MHz clock
0	1	25 MHz clock
1	0	36 MHz clock
1	1	24 MHz clock

The 24 and 36 MHz clocks are derived from the **clk72** input by dividing by 3 and 2. The 25 MHz clock comes from the **clk25in** input.

4.6 IOEB Interrupt Latch

The **pintr** pin is connected to an interrupt latch in IOEB. A rising edge on **pintr** causes an interrupt to be latched in IOEB. The latch output is connected to the NIL[6] interrupt input on IOC and goes low when the rising edge is detected. The interrupt is cleared (NIL[6] is set high) by resetting the chip or by the **Ns5x58** strobe.

5. Miscellaneous Features

5.1 The CSYNC output

The **csync** output is a modified form of the vertical/combined sync (NV/CSYNC) output from VIDC. Some extra logic is employed to allow this signal to be modified so that it contains a negative edge at the start of every raster thus making it suitable for input to a PAL encoder.

The logic allows a negative going pulse to be injected into the NV/CSYNC signal when it is high. The width of the pulse is fixed at 72 cycles of the video clock. It will usually be used with the 24 MHz clock option in which case the pulse is approximately 3µs wide. The pulse is triggered by the rising edge of the NHI output of VIDC and this occurs at the end of the horizontal display period.

The following figure shows the behaviour of the **csync** output. The upper two traces are internal signals showing the timing of vertical and horizontal syncs. The third shows the output from the NV/CSYNC output from VIDC when configured for combined sync. The fourth trace shows NHI and the bottom trace is the **csync** output which is the NV/CSYNC signal modified by the injection of negative going pulses and then inverted. The pulses are present in each raster during the vertical synchronisation pulse.

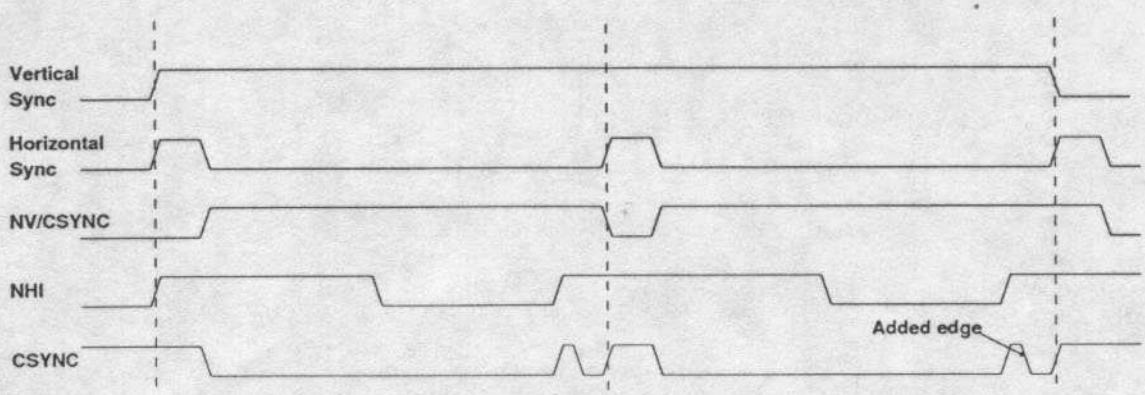


Fig 8: CSYNC timing

5.2 External Memory Interface

ARM250 can interface directly to both ROM and dynamic RAM memories. The ROM system will typically be made up of byte-wide ROMs in sets of four. The data pins are connected directly to the ARM250 data bus and the address lines to the latched address bus (**la[21:2]**). The 20 address lines allow a maximum ROM size of 4 Mbytes when using 1M by 8 devices. The **Nromcs** line is connected to the chip select pin of the ROMs. In a single bank system, the output enable inputs of the ROMs can be wired permanently on. If multiple banks of ROMs are required, the output enables may be generated by decoding high order address lines. The duration of ROM accesses can be controlled by a register in MEMC.

A dynamic RAM memory system will typically use 8 DRAM chips in the "by 4" configuration. The largest DRAM system of 4 megabytes will use eight 1M by 4 DRAM chips. The DRAM address lines are connected to the **ra[9:0]** pins. The four column address strobes (**Ncas[3:0]**) are each connected to the CAS- pins of two

DRAM chips and the row address strobe (Nras) is connected to the RAS- pins of all eight DRAM chips. The write enable (WE-) input of the DRAMs is driven by an inverter whose input is connected to the **dbe** output of ARM250. MEMC provides a number of transparent mechanisms to refresh the DRAM.

5.3 Using the Crystal Oscillator

The pins **clk25in** and **clk25out** are used with an external crystal to form an oscillator as shown in the diagram below. The bias resistor R1 is typically 1 MΩ and the capacitors C1 and C2 are typically 10 pF. The crystal (X1) should be a series resonant, fundamental mode type with a frequency in the range 1 to 26 MHz.

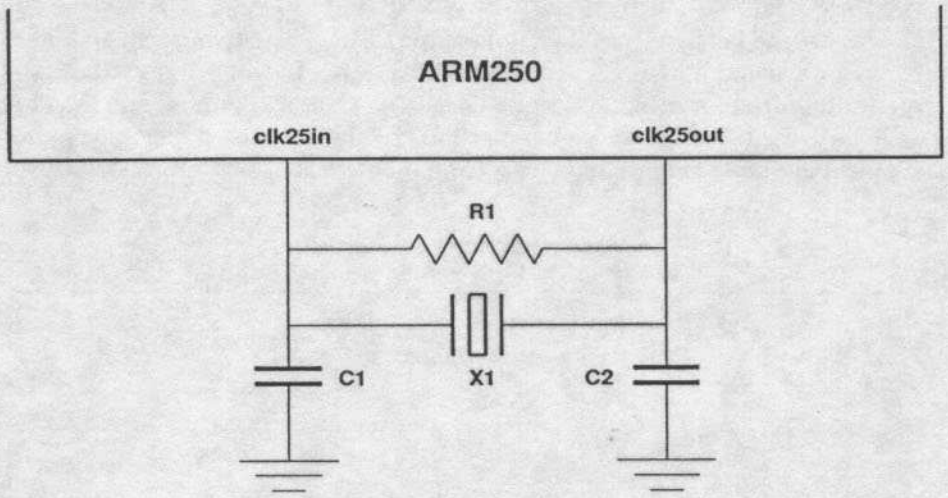


Fig 9 : Crystal Oscillator Circuit

5.4 The Ntest pin

The **Ntest** pin is driven low to place the chip into one of a number of test modes. This facility is used to test newly manufactured devices. One of the test modes may be useful for testing assembled boards containing ARM250. In this mode, all output drivers with one exception are placed in the high impedance state so that signals that they are connected to may be driven by an external tester. To place the chip in this mode the pins **mid[3]** and **mid[2]** must be low and the pins **mid[1]** and **mid [0]** must be high. The outputs may then be made high impedance by driving the **Ntest** pin low. The **clk25out** pin is not affected by test mode and its output is always enabled. It may be controlled, however, as it is always the inverse of the **clk25in** pin.

5.5 Compatibility of ARM2aS and ARM2

The ARM2aS processor used in ARM250 is largely code compatible with the earlier ARM2 processor although the architecture of ARM250 means that it is not possible to attach coprocessors. There are some minor software differences as follows.

- ARM2aS implements a swap instruction (SWP) which takes the place of one of the undefined instructions in ARM2.
- The internal timing associated with mode changes has been improved on ARM2aS and a banked register may now be accessed immediately after a mode change For compatibility with ARM2, it is recommended that this feature is not exploited in any new code.
- The implementation of the coprocessor data operation (CDP) instruction on ARM2 causes a Software Interrupt (SWI) to take the Undefined Instruction trap if the SWI was the next instruction after the CDP. This is no longer the case on ARM2aS, but the sequence CDP followed by SWI should be avoided for program compatibility.
- On ARM2, a coprocessor instruction for an absent coprocessor in the last word of a page which preceeds a page that causes an abort will enter the prefetch abort handler before taking the undefined instruction trap. ARM2aS will enter the undefined instruction handler first, and only take the prefetch abort trap when the undefined instruction handler returns control to the instruction in the aborting page. This is particularly relevant for ARM250 which can never have coprocessors.
- In the case of an address exception trap where the first word of a store multiple instruction writes to an illegal area of memory, the first write will be suppressed. Unlike on ARM2, however, subsequent writes will not be suppressed. It is therefore possible for a STM instruction which targets an illegal address (Ta) to corrupt memory at a lower address (Ta MOD 040000000 (hex)).

7. Package and Pinout

ARM250 is packaged in a 160 pin plastic Japanese quad flat package (PJQFP) which is designed to be surface mounted. The package outline and pin numbering is shown in the figure below.

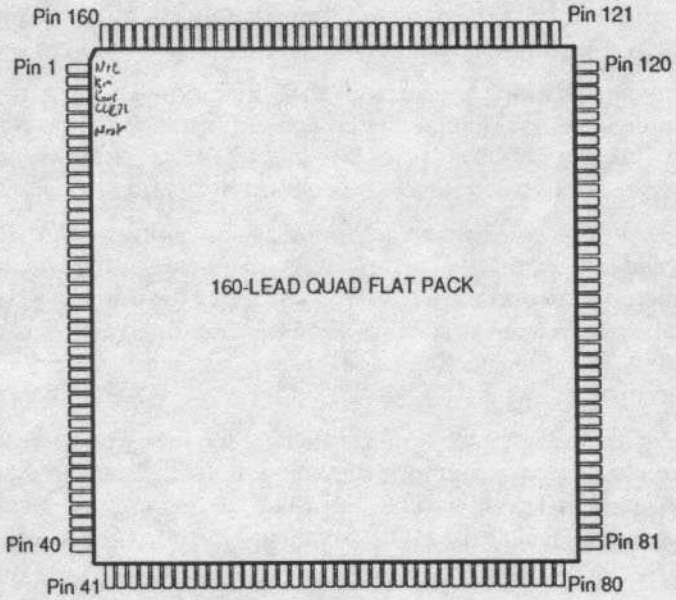


Fig 10 : ARM250 PJQFP Package

The pin functions are shown in the following table. The Type column shows the pin type as follows :

I - input
O - output
B - bidirectional
P - power

The Input column shows the input threshold type for input pins as follows :

C - CMOS threshold
CP - CMOS threshold with pull-up resistor
T - TTL threshold
S - Schmitt thresholds
A - Analogue input

The Output column shows the output type. All digital outputs have CMOS drivers. The number indicates the drive strength in mA and the SL suffix indicates a slew-limited output.

C - CMOS output
CD - CMOS open-drain output
CZ - CMOS tri-state output
A - Analogue output

Pin name	Number	Function	Type	Input	Output
Nre	1	I/O read enable	O		C 8
kin	2	Keyboard in	I	C	
kout	3	Keyboard out	O		C 4
clk72	4	Clock 72MHz	I	C	
Ntest	5	Test	I	CP	
Nrst	6	Reset	B	S	CD 8 SL
Npor	7	Power-on reset	I	S	
sysclk	8	System clock	I	CP	
VSS1	9	Power	P		
VDD1	10	Power	P		
clk25out	11	Clock25 out	O		C 4
clk25in	12	Clock25 in	I	C	
mid[0]	13	Config 0	I	S	
mid[1]	14	Config 1	I	S	
mid[2]	15	Config 2	I	S	
mid[3]	16	Config 3	I	S	
clkkb	17	Keyboard clock	O		C 4
VSS2	18	Power	P		
eorh	19	Horizontal sync	O		C 16
eorv	20	Vertical/combined sync	O		C 16
Njoy[1]	21	Joystick enable 1	O		C 4
Njoy[2]	22	Joystick enable 2	O		C 4
Nras	23	Row Address Strobe	O		C 8 SL
Ncas[0]	24	Column Address Strobe 0	O		C 8 SL
Ncas[1]	25	Column Address Strobe 1	O		C 8 SL
Ncas[2]	26	Column Address Strobe 2	O		C 8 SL

Pin name	Number	Function	Type	Input	Output
Ncas[3]	27	Column Address Strobe 3	O		C 8 SL
VSS3	28	Power	P		
VDD2	29	Power	P		
ra[0]	30	RAM address 0	O		C 8 SL
ra[1]	31	RAM address 1	O		C 8 SL
ra[2]	32	RAM address 2	O		C 8 SL
ra[3]	33	RAM address 3	O		C 8 SL
ra[4]	34	RAM address 4	O		C 8 SL
ra[5]	35	RAM address 5	O		C 8 SL
ra[6]	36	RAM address 6	O		C 8 SL
ra[7]	37	RAM address 7	O		C 8 SL
ra[8]	38	RAM address 8	O		C 8 SL
ra[9]	39	RAM address 9	O		C 8 SL
dbe	40	Data bus enable	O		C 8 SL
d[0]	41	Data bus 0	I/O	T	CZ 8 SL
d[1]	42	Data bus 1	I/O	T	CZ 8 SL
d[2]	43	Data bus 2	I/O	T	CZ 8 SL
d[3]	44	Data bus 3	I/O	T	CZ 8 SL
d[4]	45	Data bus 4	I/O	T	CZ 8 SL
d[5]	46	Data bus 5	I/O	T	CZ 8 SL
d[6]	47	Data bus 6	I/O	T	CZ 8 SL
d[7]	48	Data bus 7	I/O	T	CZ 8 SL
VSS4	49	Power	P		
VDD3	50	Power	P		
d[8]	51	Data bus 8	I/O	T	CZ 8 SL
d[9]	52	Data bus 9	I/O	T	CZ 8 SL
d[10]	53	Data bus 10	I/O	T	CZ 8 SL

Pin name	Number	Function	Type	Input	Output
d[11]	54	Data bus 11	I/O	T	CZ 8 SL
d[12]	55	Data bus 12	I/O	T	CZ 8 SL
d[13]	56	Data bus 13	I/O	T	CZ 8 SL
d[14]	57	Data bus 14	I/O	T	CZ 8 SL
d[15]	58	Data bus 15	I/O	T	CZ 8 SL
d[16]	59	Data bus 16	I/O	T	CZ 8 SL
d[17]	60	Data bus 17	I/O	T	CZ 8 SL
d[18]	61	Data bus 18	I/O	T	CZ 8 SL
d[19]	62	Data bus 19	I/O	T	CZ 8 SL
d[20]	63	Data bus 20	I/O	T	CZ 8 SL
d[21]	64	Data bus 21	I/O	T	CZ 8 SL
d[22]	65	Data bus 22	I/O	T	CZ 8 SL
d[23]	66	Data bus 23	I/O	T	CZ 8 SL
VSS5	67	Power	P		
VDD4	68	Power	P		
d[24]	69	Data bus 24	I/O	T	CZ 8 SL
d[25]	70	Data bus 25	I/O	T	CZ 8 SL
d[26]	71	Data bus 26	I/O	T	CZ 8 SL
d[27]	72	Data bus 27	I/O	T	CZ 8 SL
d[28]	73	Data bus 28	I/O	T	CZ 8 SL
d[29]	74	Data bus 29	I/O	T	CZ 8 SL
d[30]	75	Data bus 30	I/O	T	CZ 8 SL
d[31]	76	Data bus 31	I/O	T	CZ 8 SL
Nromcs	77	ROM select	O		C 4
prnw	78	Read/not Write	O		C 8 SL
la[2]	79	Latched address 2	O		C 8 SL
la[3]	80	Latched address 3	O		C 8 SL

Pin name	Number	Function	Type	Input	Output
la[4]	81	Latched address 4	O		C 8 SL
la[5]	82	Latched address 5	O		C 8 SL
la[6]	83	Latched address 6	O		C 8 SL
la[7]	84	Latched address 7	O		C 8 SL
la[8]	85	Latched address 8	O		C 8 SL
la[9]	86	Latched address 9	O		C 8 SL
la[10]	87	Latched address 10	O		C 8 SL
la[11]	88	Latched address 11	O		C 8 SL
VSS6	89	Power	P		
VDD5	90	Power	P		
la[12]	91	Latched address 12	O		C 8 SL
la[13]	92	Latched address 13	O		C 8 SL
la[14]	93	Latched address 14	O		C 8 SL
la[15]	94	Latched address 15	O		C 8 SL
la[16]	95	Latched address 16	O		C 8 SL
la[17]	96	Latched address 17	O		C 8 SL
la[18]	97	Latched address 18	O		C 8 SL
la[19]	98	Latched address 19	O		C 8 SL
la[20]	99	Latched address 20	O		C 8 SL
la[21]	100	Latched address 21	O		C 8 SL
pintr	101	Printer interrupt	I	T	
Naen	102	PC address enable	O		C 4
Nior	103	PC I/O read	O		C 4
Niow	104	PC I/O write	O		C 4
Ndack	105	PC data acknowledge	O		C 4
tc	106	PC terminal count	O		C 4
VSS7	107	Power	P		

Pin name	Number	Function	Type	Input	Output
VDD6	108	Power	P		
rsdac	109	Sound DAC ref current	I	A	
sndvss	110	Sound DAC ground	P		
sndvdd	111	Sound DAC power	P		
lch	112	Left channel sound	O		A
Nlch	113	Inverse left ch sound	O		A
rch	114	Right channel sound	O		A
Nrch	115	Inverse right ch sound	O		A
rvdac	116	Video DAC ref current	I	A	
vidvss	117	Video DAC ground	P		
rout	118	Red video	O		A
gout	119	Green video	O		A
bout	120	Blue video	O		A
sink	121	Video reset	I	T	
Nsup	122	Supremacy bit	O		C4
vidclk	123	Video clock out	O		C4
clkvid	124	Video clock in	C		
csync	125	Compensated sync	O		C 4
VSS8	126	Power	P		
VDD7	127	Power	P		
Nindex	128	IF Interrupt	I	T	
drq	129	FH0 Interrupt	I	T	
Npirq	130	IL5 Interrupt	I	T	
Npfiq	131	IL0 Interrupt	I	T	
Nfintr	132	IL4 Interrupt	I	T	
Nsintr	133	IL2/C4 Interrupt	B	T	CD 4
Nefiq	134	FL Interrupt	I	T	

Pin name	Number	Function	Type	Input	Output
clk2	135	Peripheral clock	O		C 8
c[0]	136	I/O port 0	B	T	CD 4
c[1]	137	I/O port 1	B	T	CD 4
c[2]	138	I/O port 2	B	T	CD 4
c[3]	139	I/O port 3	B	T	CD 4
Niorq	140	I/O request	O		C 8
Niogt	141	I/O grant	B	T	CD 8
Nps1	142	Peripheral select 1	O		C 8
Nil[3]	143	IL3 Interrupt	I	T	
Ns[2]	144	Peripheral select 2	O		C 8
ioclk	145	I/O clock	O		C 8
VSS9	146	Power	P		
VDD8	147	Power	P		
bd[0]	148	I/O data bus 0	I/O	T	CZ 8
bd[1]	149	I/O data bus1	I/O	T	CZ 8
bd[2]	150	I/O data bus 2	I/O	T	CZ 8
bd[3]	151	I/O data bus 3	I/O	T	CZ 8
bd[4]	152	I/O data bus 4	I/O	T	CZ 8
bd[5]	153	I/O data bus 5	I/O	T	CZ 8
bd[6]	154	I/O data bus 6	I/O	T	CZ 8
bd[7]	155	I/O data bus 7	I/O	T	CZ 8
Nwbe	156	Write buffer enable	O		C 4
Nrbe	157	Read buffer enable	O		C 4
Nwbl	158	Write buffer latch	O		C 4
Nrbl	159	Read buffer latch	B	T	CD 8
Nwe	160	I/O Write enable	O		C 8