Hardware Interface 18

18.1 Detailed Pin Descriptions

This chapter describes the embedded Pentium® processor pins that interface to the system. Both the embedded Pentium processor and the embedded Pentium processor with MMX technology have the same logical hardware interface. The embedded Pentium processor with MMX technology has one extra signal, VCC2DET#.

The processor, when operating in dual processing mode, modifies the functionality of the following signals:

- A20M#, ADS#, BE4#–BE0#, CACHE#, D/C#, FERR#, FLUSH#, HIT#, HITM#, HLDA, IGNNE#, LOCK#, M/IO#, PCHK#, RESET, SCYC, SMIACT#, W/R#

18.1.1 A20M#

A20M#	Address 20 Mask
	Used to emulate the 1 Mbyte address wraparound on the 8086
	Asynchronous Input

Signal Description

When the address 20 mask (A20M#) input is asserted, the processor masks physical address bit 20 (A20) before performing a lookup to the internal caches or driving a memory cycle on the bus. A20M# is provided to emulate the address wraparound at 1 Mbyte which occurs on the 8086.

Note: A20M# must be asserted only when the processor is in real mode. *The effect of asserting A20M# in protected mode is undefined and may be implemented differently in future processors.*

Inquire cycles and writebacks caused by inquire cycles are not affected by this input. Address bit A20 is not masked when an external address is driven into the processor for an inquire cycle. Note that if an OUT instruction is used to modify A20M#, this does not affect previously prefetched instructions. A serializing instruction must be executed to guarantee recognition of A20M# before a specific instruction.

The processor, when configured as a Dual processor, ignores the A20M# input.

When Sampled/Driven

A20M# is sampled on every rising clock edge. A20M# is level sensitive and active low. This pin is asynchronous, but must meet setup and hold times for recognition in any specific clock. To guarantee that A20M# will be recognized before the first ADS# after RESET, A20M# must be asserted within two clocks after the falling edge of RESET

Note: As the performance of embedded Pentium processors continues to improve, code sequences are executed faster. As a result, some code sequences that rely upon hardware timing may fail. Specifically when a keyboard controller is used to toggle the A20M# pin and the keyboard

controller is slow in response, data or code may be read from a wrong address at some point in a code sequence. Therefore, you should ensure that the keyboard controller switches the A20M# signal fast enough to match the execution speed of the processor. Software should be written to synchronize code execution with the toggling of the A20M# signal.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
A20	When asserted, A20M# masks the value of address pin A20.
CPUTYP	When strapped to V_{CC}, the processor ignores the A20M# input.

18.1.2 A31–A3

A31–A3	Address Lines
	Defines the physical area of memory or I/O accessed.
	Input/Output

Signal Description

As outputs, the Address Lines (A31–A3) along with the byte enable signals (BE7#–BE0#) form the address bus and define the physical area of memory or I/O accessed.

The embedded Pentium processor is capable of addressing 4 gigabytes of physical memory space and 64 Kbytes of I/O address space.

As inputs, the address bus lines A31–A5 are used to drive addresses back into the processor to perform inquire cycles. Since inquire cycles affect an entire 32-byte line, the logic values of A4 and A3 are not used for the hit/miss decision, however A4 and A3 must be at valid logic level and meet setup and hold times during inquire cycles.

When Sampled/Driven

When an output, the address is driven in the same clock as ADS#. The address remains valid from the clock in which ADS# is asserted until the earlier of the last BRDY# or the clock after NA#, or until AHOLD is asserted.

When an input, the address must be returned to the processor to meet setup and hold times in the clock in which EADS# is sampled asserted.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
A20M#	When asserted, A20M# causes address pin A20 to be masked.
ADS#	A31–A3 are driven with ADS# (except when a external inquire cycle causes a writeback before AHOLD is deasserted, see Chapter 19, “Bus Functional Description”).
AHOLD	A31–A3 are floated one clock after AHOLD is asserted.
AP	Even address parity is driven/sampled with the address bus on AP.
APCHK#	The status of the address parity check is driven on the APCHK# pin.
BE7#–BE0#	Completes the definition of the physical area of memory or I/O accessed.
BOFF#	A31–A3 are floated one clock after BOFF# is asserted.
EADS#	A31–A5 are sampled with EADS# during inquire cycles.
HIT#	HIT# is driven to indicate whether the inquire address driven on A31–A5 is valid in an internal cache.
HITM#	HITM# is driven to indicate whether the inquire address driven on A31–A5 is in the modified state in the data cache.
HLDA	A31–A3 are floated when HLDA is asserted.
INV	INV determines whether the inquire address driven to the processor on A31–A5 should be invalidated or marked as shared if it is valid in an internal cache.

18.1.3 ADS#

ADS#	Address Strobe
	Indication that a new valid bus cycle is currently being driven by the processor.
	Synchronous Input/Output

Signal Description

The Address Strobe output indicates that a new valid bus cycle is currently being driven by the processor. The following pins are driven to their valid level in the clock ADS# is asserted: A31–A3, AP, BE7#–BE0#, CACHE#, LOCK#, M/IO#, W/R#, D/C#, SCYC, PWT, PCD.

ADS# is used by external bus circuitry as the indication that the processor has started a bus cycle. The external system may sample the bus cycle definition pins on the next rising edge of the clock after ADS# is driven active.

ADS# floats during bus HOLD and BOFF#. ADS# is not driven low to begin a bus cycle while AHOLD is asserted unless the cycle is a writeback due to an external invalidation. An active (floating low) ADS# in the clock after BOFF# is asserted should be ignored by the system.

This signal is normally identical to the ADSC# output. When operating in dual processing mode, the processor uses this signal for private snooping.

When Sampled/Driven

ADS# is driven active in the first clock of a bus cycle and is driven inactive in the second and subsequent clocks of the cycle. ADS# is driven inactive when the bus is idle.

This signal becomes an Input/Output when two embedded Pentium processors are operating together in Dual Processing Mode.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
ADSC#	ADS# is identical to the ADSC# output.
APCHK#	When operating in dual processing mode, APCHK# is driven in response to ADS# for a private snoop.
D/P#	When operating in dual processing mode, D/P# should be sampled with an active ADS#.
SMIACT#	When operating in dual processing mode, SMIACT# should be sampled with an active ADS# and qualified by D/P#.

18.1.4 ADSC#

ADSC#	Additional Address Strobe
	Indicates that a new valid bus cycle is currently being driven by the processor.
	Synchronous Output

Signal Description

This signal is identical to the ADS# output. This signal can be used to relieve tight board timings by easing the load on the Address Strobe signal.

When Sampled/Driven

Refer to the ADS# signal description.

Note: ADSC# is not tested and timings are not specified. It is recommended that ADSC# not be used.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
ADS#	ADSC# is identical to the ADS# output.

18.1.5 AHOLD

AHOLD	Address Hold
	Floats the address bus so an inquire cycle can be driven to the processor.
	Synchronous Input

Signal Description

In response to the Address Hold request input, the processor stops driving A31–A3 and AP in the next clock. This pin is intended to be used for running inquire cycles to the processor. AHOLD allows another bus master to drive the processor address bus with the address for an inquire cycle. Since inquire cycles affect the entire cache line, although A31–A3 are floated during AHOLD, only A31–A5 are used by the processor for inquire cycles (and parity checking). Address pins 3 and 4 are logically ignored during inquire cycles but must be at a valid logic level when sampled.

While AHOLD is active, the address bus is floated, but the remainder of the bus can remain active. For example, data can be returned for a previously driven bus cycle when AHOLD is active. In general, the processor does not issue a bus cycle (ADS#) while AHOLD is active; the only exception to this is that writeback cycles due to an external snoop are driven while AHOLD is asserted.

Since the processor floats its bus immediately (in the next clock) in response to AHOLD, an address hold acknowledge is not required.

When AHOLD is deasserted, the processor drives the address bus in the next clock. It is the responsibility of the system designer to prevent address bus contention. This can be accomplished by ensuring that other bus masters have stopped driving the address bus before AHOLD is deasserted. Note the restrictions to the deassertion of AHOLD discussed in the inquire cycle section of the Chapter 19, “Bus Functional Description.”

AHOLD is recognized during RESET and INIT. Note that the internal caches are flushed as a result of RESET, so invalidation cycles run during RESET are unnecessary.

When Sampled

AHOLD is sampled on every rising clock edge, including during RESET and INIT.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
A31–A3	A31–A3 are floated as a result of the assertion of AHOLD.
ADS#	ADS# is not driven if AHOLD is asserted (except when a external inquire cycle causes a writeback before AHOLD is deasserted). See Chapter 19, “Bus Functional Description.”
AP	AP is floated as a result of the assertion of AHOLD.
EADS#	EADS# is recognized while AHOLD is asserted.

18.1.6 AP

AP	Address Parity
	Bidirectional address parity pin for the address lines of processor.
	Input/Output

Signal Description

This is the bidirectional Address Parity pin for the address lines of processor. There is one address parity pin for the address lines A31–A5. Note that A4 and A3 are not included in the parity determination.

When an output, AP is driven by the processor with even parity information on all processor generated cycles in the same clock as the address driven. (Even address parity means that there are an even number of HIGH outputs on A31–A5 and the AP pins.)

When an input, even parity information must be returned to the processor on this pin during inquire cycles in the same clock in which EADS# is sampled asserted to ensure that the correct parity check status is driven on the APCHK# output.

The value read on the AP pin does not affect program execution. The value returned on the AP pin is used only to determine even parity and drive the APCHK# output with the proper value. It is the responsibility of the system to take appropriate actions if a parity error occurs. If parity checks are not implemented in the system, AP may be connected to V_{CC} through a pull-up resistor and the APCHK# pin may be ignored.

When Sampled/Driven

When an output, AP is driven by the processor with even parity information on all processor generated cycles in the same clock as the address driven. The AP output remains valid from the clock in which ADS# is asserted until the earlier of the last BRDY# or the clock after NA#, or until AHOLD is asserted.

When an input, even parity information must be returned to the processor on this pin during inquire cycles in the same clock that EADS# is sampled asserted to guarantee that the proper value is driven on APCHK#. The AP input must be at a valid level and meet setup and hold times when sampled.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
A31–A5	The AP pin is used to create even parity with the A31–A5 pins.
ADS#	AP is driven with ADS# (except when a external inquire cycle causes a write-back before AHOLD is deasserted, see Chapter 19, “Bus Functional Description”).
AHOLD	AP is floated one clock after AHOLD is asserted.
APCHK#	The status of the address parity check is driven on the APCHK# output.

BOFF#	AP is floated one clock after BOFF# is asserted.
EADS#	AP is sampled with EADS# during inquire cycles.
HLDA	AP is floated when HLDA is asserted.

18.1.7 APCHK#

APCHK#	Address Parity Check
	The status of the address parity check is driven on this output.
	Asynchronous Output

Signal Description

APCHK# is asserted two clocks after EADS# is sampled active if the processor detects a parity error on the A31–A5 during inquire cycles.

Driving APCHK# is the only effect that bad address parity has on the processor. It is the responsibility of the system to take appropriate action if a parity error occurs. If parity checks are not implemented in the system, the APCHK# pin may be ignored.

Address parity is checked during every private snoop between the Primary and Dual processors. Therefore, APCHK# may be asserted due to an address parity error during this private snoop. If an error is detected, APCHK# will be asserted two clocks after ADS# for one processor clock period. The system can choose to acknowledge this parity error indication at this time or do nothing.

When Sampled/Driven

APCHK# is valid for one clock and should be sampled two clocks following ADS# and EADS# assertion. At all other times it is inactive (high). APCHK# is not floated with AHOLD, HOLD, or BOFF#. The APCHK# signal is glitch-free.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
ADS#	When operating in dual processing mode, APCHK# is driven in response to a private snoop.
AP	Even address parity with the A31–A5 should be returned to the processor on the AP pin. If even parity is not driven, the APCHK# pin is asserted.
A31–A5	The AP pin is used to create even parity with A31–A5. If even parity is not driven to the processor, the APCHK# pin is asserted.
EADS#	APCHK# is driven in response to an external snoop.

18.1.8 APICEN

APICEN	APIC Enable
	This pin enables the APIC on the processor.
	Synchronous Configuration Input
	Needs external pull-up resistors.

Signal Description

APICEN, if sampled high at the falling edge of RESET, enables the on-chip APIC. If it is sampled low, then the on-chip APIC is not enabled and the processor uses the interrupts as if the APIC was not present (Bypass mode).

APICEN must be driven by the system. This pin has an internal pulldown resistor and is sampled at the falling edge of RESET. When using an active circuit to override the internal pulldown resistor, the driver should have an internal effective pullup resistance of 1 KOhms or less.

When Sampled/Driven

APICEN should be valid and stable two clocks before and after the falling edge of RESET.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
BE3#–BE0#	When APICEN is sampled active, BE3#–BE0# are used to sample the APIC ID.
INTR/LINT0	When APICEN is sampled active, this input becomes the APIC local interrupt 0.
NMI/LINT1	When APICEN is sampled active, this input becomes the APIC local interrupt 1.
PICCLK	PICCLK must be tied or driven high when APICEN is sampled low at the falling edge of RESET.
PICD1	APICEN shares a pin with PICD1.
RESET	APICEN is sampled at the falling edge of RESET.

18.1.9 BE7#–BE0#

BE7#–BE0#	Byte Enable Outputs / APIC ID Inputs
	When operating in dual processing mode, BE4# is used to transfer information between the Dual and Primary processors during the atomic Flush operation.
	At RESET, the BE3#–BE0# pins read the APIC ID bits for the processor.
	After RESET, these pins are byte enables and help define the physical area of memory to I/O accessed.
	BE4#: Synchronous Input/Output, Dual Processing Mode. BE3#–BE0#: Synchronous Configuration Inputs, during RESET. BE3#–BE0#: Synchronous Outputs, following RESET.

Signal Description

As outputs, the byte enable signals are used in conjunction with the address lines to provide physical memory and I/O port addresses. The byte enables are used to determine which bytes of data must be written to external memory, or which bytes were requested by the processor for the current cycle.

- BE7# applies to D63–D56
- BE6# applies to D55–D48
- BE5# applies to D47–D40
- BE4# applies to D39–D32
- BE3# applies to D31–D24
- BE2# applies to D23–D16
- BE1# applies to D15–D8
- BE0# applies to D7–D0

In the case of cacheable reads (line fill cycles), all 8 bytes of data must be driven to the processor regardless of the state of the byte enables. If the requested read cycle is a single transfer cycle, valid data must be returned on the data lines corresponding to the active byte enables. Data lines corresponding to inactive byte enables need not be driven with valid logic levels. Even data parity is checked and driven only on the data bytes that are enabled by the byte enables.

The local APIC module on the embedded Pentium processor loads its 4-bit APIC ID value from the four least significant byte-enable pins at the falling edge of RESET. The following table shows the four pins that comprise the APIC ID.

APIC ID Register Bit	Pin Latched at RESET
bit 24	BE0#
bit 25	BE1#
bit 26	BE2#
bit 27	BE3#

Loading the APIC ID should be done with external logic that drives the proper address at reset. If the BE3#–BE0# signals are not driven, the APIC ID value defaults to 0000 for the embedded Pentium processor and 0001 for the Dual processor.

BE3#–BE0# pins establish the APIC ID for the processor and are input/output pins. These pins have strong internal pull down resistors and typically high external capacitive loading. A strong pullup on BE3#–BE0# is needed to make sure that the pins reach the correct value. In addition, since these pins are also outputs, a large resistive load would degrade the signal output during normal operation. A 50-Ohm three-state driver is recommended to drive these pins during RESET only.

Warning: An APIC ID of all 1s is an APIC special case (i.e., a broadcast) and must not be used. Because the Dual processor inverts the lowest order bit of the APIC ID placed on the lowest four BE pins, the value “1110” must not be used when operating in Dual Processing mode.

In a dual-processor configuration, the OEM socket and Socket 5/Socket 7 should have the four byte enable pairs tied together. The Primary processor loads the value seen on these four pins at RESET. The Dual processor loads the value seen on these pins and automatically inverts bit 24 of the APIC ID register. Thus, the two processors will have unique APIC ID values.

The Primary and Dual processors incorporate a mechanism to present an atomic view of the cache flush operation to the system when in dual processing mode. The Dual processor performs the cache flush operation and grants the bus to the Primary processor by PBREQ#/PBGNT# arbitration exchange. The Primary processor then flushes both of its internal caches and runs a cache flush acknowledge special cycle by asserting BE4#, to indicate to the external system that the cache line entries have been invalidated. The Dual processor halts all code execution while the processor is flushing its caches, and does not begin executing code until it recognizes the flush acknowledge special cycle. Refer to Chapter 19, “Bus Functional Description.”

When Sampled/Driven

As outputs, the byte enables are driven in the same clock as ADS#. The byte enables are driven with the same timing as the address bus (A31–A3). The byte enables remain valid from the clock in which ADS# is asserted until the earlier of the last BRDY# or the clock after NA#. The byte enables do not float with AHOLD.

The four least significant byte-enable bits are sampled for APIC ID at the falling edge of RESET. These pins should be valid and stable two clocks before and after the falling edge of RESET.

Note: Asserting the APIC ID is not specified for the rising edge of RESET. In a FRC system, the BE3#–BE0# pins must not be driven for the two clocks following the rising edge of RESET. The system design should drive these signals on the third clock or later.

There are strong pull down resistors on the byte enable pins internally that make it impractical to use pullup circuits to drive the APIC ID (on BE3#–BE0#) or enter Lock Step operation (with BE4#) at the falling edge of RESET. When not using the internal defaults on these pins, the value of the external pullup resistors would have to be 50 Ohms or less. For this reason it is suggested to use active drivers on these lines that would drive the byte enable pins during the falling edge of RESET. Passive pullups should be avoided.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
A31–A3	A31–A3 and BE7#–BE0# together define the physical area of memory or I/O accessed.
ADS#	BE7#–BE0# are driven with ADS#.
APICEN	When APICEN is sampled active, BE3#–BE0# are used to sample the APIC ID.
BOFF#	BE7#–BE0# are floated one clock after BOFF# is asserted.
D63–D0	BE7#–BE0# indicate which data bytes are being requested or driven by the processor.
DP7–DP0	Even data parity is checked/driven only on the data bytes enabled by BE7#–BE0#.
HLDA	BE7#–BE0# are floated when HLDA is asserted.
RESET	During reset the BE3#–BE0# pins are sampled to determine the APIC ID. Following RESET, they function as byte-enable outputs.

18.1.10 BF2–BF0

BF2–BF0	Bus-to-core frequency ratio
	Used to configure processor bus-to-core frequency ratio.
	Asynchronous Input
	Only the Low-power Embedded Pentium Processor with MMX technology has a BF2 pin.

Signal Description

The BF*n* pins determine whether the processor operates at a 1/2, 2/3, 2/5, 2/7 or 1/4 I/O bus-to-core frequency ratio. Since some bus-to-core ratios are not supported, these pins should *always* be connected to the proper level.

Note: External pulldowns of 500 Ohms or less must be used between the pins and ground to effectively override the default (internal) pullups, while external pullups of 2.2 KOhms or less should be used to override the default pulldowns on BF1–BF0.

Each embedded Pentium processor is specified to operate within a single bus-to-core ratio and a specific minimum to maximum bus frequency range (corresponding to a minimum to maximum core frequency range). Operation in other bus-to-core ratios or outside the specified operating frequency range is not supported. Tables 18-1 through 18-3 summarize these specifications.

Table 18-1. Bus-to-Core Frequency Ratios for the Embedded Pentium® Processor (at 100/133/166 MHz)

BF1	BF0	Embedded Pentium® Processor Bus/Core Ratio	Max Bus/Core Frequency (MHz)	Min Bus/Core Frequency (MHz)
0	0	2/5	66/166	33/83
1	0	1/2	66/133	33/66
1	1	2/3†	66/100	33/50

† This is the default bus fraction for the embedded Pentium processor (at 100/133/166 MHz). If the BF pins are left floating, the processor will be configured for the 2/3 bus to core frequency ratio.

Table 18-2. Bus-to-Core Frequency Ratios for the Embedded Pentium® Processor with MMX™ Technology

BF1	BF0	Embedded Pentium Processor with MMX™ Technology Bus/Core Ratio	Max Bus/Core Frequency (MHz)	Min Bus/Core Frequency (MHz)
1	1	2/7	66/233	33/117
0	1	1/3	66/200	33/100
1	0	1/2†	N/A	N/A

† This is the default bus-to-core ratio for the Pentium processor with MMX technology. If the BF pins are left floating, the processor will be configured for the 1/2 bus-to-core frequency ratio, which is unsupported. *Do not float the BFn pins at RESET.*

Table 18-3. Bus-to-Core Frequency Ratios for the Low-Power Embedded Pentium® Processor with MMX™ Technology

BF2	BF1	BF0	Low-Power Embedded Pentium Processor with MMX™ Technology Bus/Core Ratio	Max Bus/Core Frequency (MHz)	Min Bus/Core Frequency (MHz)
0	0	0	2/5	66/166	
1	0	0	1/4	66/266	

If BF1–BF0 are left unconnected on the embedded Pentium processor with MMX technology, the bus-to-core ratio defaults to 1/2. If BF1–BF0 are left unconnected on the embedded Pentium processor the bus-to-core ratio defaults to 2/3. If BF2–BF0 are left unconnected on the low-power embedded Pentium processor with MMX technology, the bus-to-core ratio defaults to 2/5. This ratio is not supported by the low-power embedded Pentium processor with MMX technology; do *not* float the BF*n* pins when using the low-power embedded Pentium processor with MMX technology.

When Sampled/Driven

The BF*n* pins are sampled at RESET and cannot be changed until another non-warm (1 ms) assertion of RESET. BF*n* must not change values while RESET is active.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
RESET	BF2–BF0 are sampled at the falling edge of RESET.

18.1.11 BOFF#

BOFF#	Backoff
	The back off input is used to force the processor off the bus in the next clock.
	Synchronous Input

Signal Description

In response to BOFF#, the processor aborts all outstanding bus cycles that have not yet completed and floats the processor bus in the next clock. The processor floats all pins normally floated during bus hold. Note that since the bus is floated in the clock after BOFF# is asserted, an acknowledge is not necessary (HLDA is not asserted in response to BOFF#).

The processor remains in bus hold until BOFF# is negated, at which time the processor restarts any aborted bus cycle(s) in their entirety by driving out the address and status and asserting ADS#.

This pin can be used to resolve a deadlock situation between two bus masters.

Any data with BRDY# returned to the processor while BOFF# is asserted is ignored.

BOFF# has higher priority than BRDY#. If both BOFF# and BRDY# occur in the same clock, BOFF# takes effect.

BOFF# also has precedence over BUSCHK#. When BOFF# and BUSCHK# are both asserted during a bus cycle, BUSCHK# is ignored.

When Sampled

BOFF# is sampled on every rising clock edge, including when RESET and INIT are asserted.

When a read cycle is running on the bus and an internal snoop of that read cycle hits a modified line in the data cache, causing the system to assert BOFF#, the sequence of bus cycles is as follows: Upon negation of BOFF#, the processor drives out a writeback resulting from the internal snoop hit. After completion of the writeback, the processor then restarts the original read cycle. Thus, like external snoop writebacks, internal snoop writebacks may also be reordered in front of cycles that encounter a BOFF#. Also note that, although the original read encountered both an external BOFF# and an internal snoop hit to an M-state line, it is restarted only once.

This circumstance can occur during accesses to the page tables/directories and during prefetch cycles (these accesses cause a bus cycle to be generated before the internal snoop to the data cache is performed).

Relation to Other Signals

Pin Symbol	Relation to Other Signals
A3–A31 ADS# AP BE7#–BE3# CACHE# D/C# D63–D0 DP7–DP0 LOCK# M/IO# PCD PWT SCYC W/R#	These signals float in response to BOFF#.
BRDY#	If BRDY# and BOFF# are asserted simultaneously, BOFF# takes priority and BRDY# is ignored.
EADS#	EADS# is recognized when BOFF# is asserted.
HLDA	The same pins are floated when HLDA or BOFF# is asserted.
BUSCHK#	If BUSCHK# and BOFF# are both asserted during a bus cycle, BOFF# takes priority and BUSCHK# is forgotten.
NA#	If NA# and BOFF# are asserted simultaneously, BOFF# takes priority and NA# is ignored.

18.1.12 BP3–BP0

BP3–BP0	Breakpoint signals
	BP3–BP0 externally indicate a breakpoint match.
	Synchronous Output

Signal Description

The Breakpoint pins (BP3–BP0) correspond to the debug registers, DR3–DR0. These pins externally indicate a breakpoint match when the debug registers are programmed to test for breakpoint matches. BP1 and BP0 are multiplexed with the performance monitoring pins (PM1 and PM0). The PB1 and PB0 bits in the debug mode control register determine if the pins are configured as breakpoint or performance monitoring pins. The pins come out of RESET configured for performance monitoring.

Because of the fractional-speed bus, each assertion of a processor BP pin indicates that one or more BP matches occurred. The maximum number of matches per assertion is two when using the 2/3 or 1/2 bus-to-core ratios. Similarly, the maximum number of matches per assertion is three when using the 2/5 or 1/3 bus-to-core ratios.

When Sampled/Driven

The BP3–BP0 pins are driven in every clock and are not floated during bus HOLD of BOFF#.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
PM1–PM0	BP1 and BP0 share pins with PM1 and PM0, respectively.

18.1.13 BRDY#

BRDY#	Burst Ready
	Transfer complete indication.
	Synchronous Input

Signal Description

The Burst Ready input indicates that the external system has presented valid data on the data pins in response to a read, or that the external system has accepted the processor data in response to a write request.

Each cycle generated by the processor is either a single transfer read (or write) or a burst cache line fill (or writeback). For single data transfer cycles, one BRDY# is expected to be returned to the processor. When this BRDY# is returned, the cycle is complete. For burst transfers, four data transfers are expected by the processor. The cycle is ended when the fourth BRDY# is returned.

When Sampled

This signal is sampled in the T2, T12 and T2P bus states.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
BOFF#	If BOFF# and BRDY# are asserted simultaneously, BOFF# takes priority and BRDY# is ignored.
BUSCHK#	BUSCHK# is sampled with BRDY#.
CACHE#	In conjunction with the KEN# input, CACHE# determines whether the bus cycle consists of 1 or 4 transfers.
D63–D0	During reads, the D63–D0 pins are sampled by the processor with BRDY#. During writes, BRDY# indicates that the system has accepted D63–D0.
DP7–DP0	During reads, the DP7–DP0 pins are sampled by the processor with BRDY#. During writes, BRDY# indicates that the system has accepted DP7–DP0.
EWBE#	EWBE# is sampled with each BRDY# of a write cycle.
KEN#	KEN# is sampled and latched by the processor with the earlier of the first BRDY# or NA#. Also, in conjunction with the CACHE# input, KEN# determines whether the bus cycle will consist of 1 or 4 transfers (assertions of BRDY#).
LOCK#	LOCK# is deasserted after the last BRDY# of the locked sequence.
PCHK#	PCHK# indicates the results of the parity check two clocks after BRDY# is returned for reads.
PEN#	PEN# is sampled with BRDY# for read cycles.
WB/WT#	WB/WT# is sampled and latched by the processor with the earlier of the first BRDY# or NA#.

18.1.14 BRDYC#

BRDYC#	Burst Ready
	Transfer complete indication.
	Synchronous Input

Signal Description

This signal is identical to the BRDY# input. This signal can be used to relieve tight board timings by easing the load on the Burst Ready signal.

In addition to its normal functionality, BRDYC# is sampled with BUSCHK# at RESET to select the buffer strength for some pins. BRDYC# has an internal pullup resistor. To override the default settings for the buffer strengths, this pin should be driven and not permanently strapped to ground because this would interfere with the normal operation of this pin. The driver should have an internal resistance of 1 KOhms or less. This is a function only of BRDYC#. The BRDY# signal is not sampled to select buffer sizes.

When Sampled/Driven

Refer to the BRDY# signal description.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
BRDY#	BRDYC# is identical to the BRDY# input.
RESET	BRDYC# and BUSCHK# are sampled at RESET to select the buffer strength for some pins.

18.1.15 BREQ

BREQ	Bus Request
	Indicates externally when a bus cycle is pending internally.
	Output

Signal Description

The processor asserts the BREQ output whenever a bus cycle is pending internally. BREQ is always asserted in the first clock of a bus cycle with ADS#. Furthermore, if the processor is not currently driving the bus (due to AHOLD, HOLD, or BOFF#), BREQ is asserted in the same clock that ADS# would have been asserted if the processor were driving the bus. After the first clock of the bus cycle, BREQ may change state. Every assertion of BREQ is not guaranteed to have a corresponding assertion of ADS#.

External logic can use the BREQ signal to arbitrate between multiple processors. This signal is always driven regardless of the state of AHOLD, HOLD or BOFF#.

When Driven

BREQ is always driven by the processor, and is not floated during bus HOLD or BOFF#.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
ADS#	BREQ is always asserted in the clock in which ADS# is asserted.

18.1.16 BUSCHK#

BUSCHK#	Bus Check
	Allows the system to signal an unsuccessful completion of a bus cycle.
	Synchronous Input

Signal Description

The Bus Check input pin allows the system to signal an unsuccessful completion of a bus cycle. When this pin is sampled active, the processor latches the address and control signals of the failing cycle in the machine check registers. When the MCE bit in CR4 is also set, the processor vectors to the machine check exception upon completion of the current instruction.

If BUSCHK# is asserted in the middle of a cycle, the system must return all expected BRDY# signals to the processor. BUSCHK# is remembered by the processor if asserted during a bus cycle. The processor decides after the last BRDY# whether to take the machine check exception or not.

BOFF# has precedence over BUSCHK#. When BOFF# and BUSCHK# are both asserted during a bus cycle, BUSCHK# is ignored.

In addition to its normal functionality, BUSCHK# is sampled with BRDYC# at RESET to select the buffer strength for some pins. BUSCHK# has an internal pullup resistor. To override the default settings for the buffer strengths, this pin should be driven and not permanently strapped to ground, because this interferes with the normal operation of this pin. The driver should have an internal resistance of 1 KOhms or less.

When Sampled

BUSCHK# is sampled when BRDY# is returned to the processor.

Note: The embedded Pentium processor can remember only one machine check exception at a time. This exception is recognized on an instruction boundary. If BUSCHK# is sampled active while servicing the machine check exception for a previous BUSCHK#, it is remembered by the processor until the original machine check exception is completed. Then, the processor services the machine check exception for the second BUSCHK#. Note that only one BUSCHK# is remembered by the processor while the machine exception for the previous one is being serviced.

When the BUSCHK# is sampled active by the processor, the cycle address and cycle type information for the failing bus cycle is latched upon assertion of the last BRDY# of the bus cycle. The information is latched into the Machine Check Address (MCA) and Machine Check Type (MCT) registers respectively. However, if the BUSCHK# input is not deasserted before the first

BRDY# of the next bus cycle, and the machine check exception for the first bus cycle has not occurred, then new information is latched into the MCA and MCT registers, over-writing the previous information at the completion of this new bus cycle. Therefore, in order for the MCA and MCT registers to report the correct information for the failing bus cycle when the machine check exception for this cycle is taken at the next instruction boundary, the system must deassert the BUSCHK# input immediately after the completion of the failing bus cycle (i.e., before the first BRDY# of the next bus cycle is returned).

Relation to Other Signals

Pin Symbol	Relation to Other Signals
BOFF#	If BOFF# and BUSCHK# are both asserted during a bus cycle, the BOFF# signal causes the BUSCHK# to be forgotten.
BRDY#	BUSCHK# is sampled with BRDY#.
BRDYC#	BUSCHK# is sampled with BRDYC# at RESET to select the buffer strength for some pins.
RESET	BUSCHK# and BRDYC# are sampled at RESET to select the buffer strength for some pins.

18.1.17 CACHE#

CACHE#	Cacheability
	External indication of internal cacheability.
	Synchronous Input/Output

Signal Description

The Cacheability output is a cycle definition pin. For processor initiated cycles, this pin indicates internal cacheability of the cycle (if a read), and indicates a burst writeback (if a write). CACHE# is asserted for cycles coming from the cache (writebacks) and for cycles that will go into the cache if KEN# is asserted (linefills). More specifically, CACHE# is asserted for cacheable reads, cacheable code fetches, and writebacks. It is driven inactive for non-cacheable reads, TLB replacements, locked cycles (except writeback cycles from an external snoop that interrupt a locked read/modify/write sequence), I/O cycles, special cycles and writethroughs.

For read cycles, the CACHE# pin indicates whether the processor allows the cycle to be cached. When CACHE# is asserted for a read cycle, the cycle is turned into a cache line fill if KEN# is returned active to the processor. When this pin is driven inactive during a read cycle, processor does not cache the returned data, regardless of the state of the KEN#.

If this pin is asserted for a write cycle, it indicates that the cycle is a burst writeback cycle. Writethroughs cause a non-burst write cycle to be driven to the bus. The processor does not support write allocations (cache line fills as a result of a write miss).

When operating in dual processing mode, the embedded Pentium processors use this signal for private snooping.

When Sampled/Driven

CACHE# is driven to its valid level in the same clock as the assertion of ADS# and remains valid until the earlier of the last BRDY# or the clock after NA#.

This signal becomes an Input/Output when two embedded Pentium processors are operating together in dual processing mode.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
ADS#	CACHE# is driven to its valid level with ADS#.
BOFF#	CACHE# floats one clock after BOFF# is asserted.
BRDY#	In conjunction with the KEN# input, CACHE# determines whether the bus cycle will consist of 1 or 4 transfers (assertions of BRDY#).
HLDA	CACHE# floats when HLDA is asserted.
KEN#	KEN# and CACHE# are used together to determine if a read will be turned into a linefill.

18.1.18 CLK

CLK	Clock
	Fundamental timing source for the embedded Pentium processor.
	Input

Signal Description

The Clock input provides the fundamental timing source for the embedded Pentium processor. Its frequency is proportional to the internal operating frequency of the processor (as selected by the BF1–BF0 pins) and requires a TTL level signal. All external timing parameters except TDI, TDO, TMS, and TRST# are specified with respect to the rising edge of CLK.

Note: The CLK signal on the embedded Pentium processor with MMX technology is 3.3 V tolerant. On the embedded Pentium processor, the CLK input is 5.0 V tolerant.

When Sampled

CLK is used as a reference for sampling other signals. It is recommended that CLK begin toggling within 150 ms after V_{CC} reaches its proper operating level. This recommendation is made to ensure long term reliability of the device. V_{CC} specifications and clock duty cycle, stability, and frequency specifications must be met for 1 ms before the negation of RESET. If at any time during normal operation one of these specifications is violated, the power on RESET sequence must be repeated. This requirement is made to ensure proper operation of the phase locked loop circuitry on the clock input within the processor.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
All except TCK TDI TDO TMS TRST#	External timing parameters are measured from the rising edge of CLK for all signals except TDI, TDO, TMS, TCK, and TRST#.

18.1.19 CPUTYP

CPUTYP	Processor Type Definition Pin
	Used to configure the embedded Pentium processor as a Dual processor.
	Asynchronous Input

Signal Description

The CPUTYP pin is used to determine whether the embedded Pentium processor functions as a Primary or Dual processor. CPUTYP must be strapped to either V_{CC} or V_{SS}. When CPUTYP is strapped to V_{CC}, the embedded Pentium processor functions as a Dual processor. When CPUTYP is strapped to V_{SS} (or left unconnected), the embedded Pentium processor functions as a Primary processor. In a single socket system design, CPUTYP pin must be strapped to V_{SS}.

When Sampled/Driven

CPUTYP is sampled at RESET and cannot be changed until another non-warm (1 ms) assertion of RESET. CPUTYP must meet a 1 ms setup time to the falling edge of RESET. It is recommended that CPUTYP be strapped to V_{CC} or V_{SS}.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
A20M#	When CPUTYP is strapped to V_{CC}, the processor ignores the A20M# input.
BE4#–BE0#	The BE3#–BE0# input values are sampled during RESET to determine the APIC ID. The Dual processor uses BE4# to indicate to the Primary processor that it has completed its cache flush operation. Refer to the BE4#–BE0# pin description.
D/P#	D/P# is driven by the processor only when the CPUTYP signal is strapped to V_{SS}.
DPEN#	When CPUTYP is strapped to V_{CC}, DPEN# is driven active to indicate that the second socket is occupied.
FERR#	When CPUTYP is strapped to V_{CC}, the FERR# output is undefined.
FLUSH#	When operating in dual processing mode, the FLUSH# inputs become Synchronous to the processor clock.
IGNNE#	When CPUTYP is strapped to V_{CC}, the processor ignores the IGNNE# input.
RESET	CPUTYP is sampled at the falling edge of RESET. When operating in dual processing mode, the RESET inputs become synchronous to the processor clock.

Note: It is common practice to put either a pullup or pulldown resistor on a net. If a pullup resistor is connected to the CPUTYP pin in order to operate in a Dual Processing mode, the value of this resistor must be 100 Ohms or less to override the internal pulldown. In the absence of an external pullup, the internal pulldown sufficiently pulls down the CPUTYP pin; therefore the pin can be left floating.

18.1.20 D/C#

D/C#	Data/Code
	Distinguishes a data access from a code access.
	Synchronous Input/Output

Signal Description

The Data/Code signal is one of the primary bus cycle definition pins. D/C# distinguishes between data (D/C# = 1) and code/special cycles (D/C# = 0).

When operating in dual processing mode, the embedded Pentium processor uses this signal for private snooping.

When Sampled/Driven

The D/C# pin is driven valid in the same clock as ADS# and the cycle address. It remains valid from the clock in which ADS# is asserted until the earlier of the last BRDY# or the clock after NA#.

This signal becomes an Input/Output when two embedded Pentium processors are operating together in Dual Processing Mode.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
ADS#	D/C# is driven with ADS#.
BOFF#	D/C# floats one clock after BOFF# is asserted.
HLDA	D/C# floats when HLDA is asserted.

18.1.21 D63–D0

D63–D0	Data Lines
	Forms the 64-bit data bus.
	Input/Output

Signal Description

The bidirectional lines D63–D0 form the 64 data bus lines for the embedded Pentium processor. Lines D7–D0 define the least significant byte of the data bus; lines D63–D56 define the most significant byte of the data bus.

When Sampled/Driven

When the processor is driving the data lines (during writes), they are driven during the T2, T12, or T2P clocks for that cycle.

During reads, the processor samples the data bus when BRDY# is returned.

D63–D0 are floated during T1, TD, and Ti states.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
BE7#–BE0#	BE7#–BE0# indicate which data bytes are being requested or driven by the processor.
BOFF#	D63–D0 float one clock after BOFF# is asserted.
BRDY#	BRDY# indicates that the data bus transfer is complete.
DP7–DP0	Even data parity is driven/sampled with the data bus on DP7–DP0.
HLDA	D63–D0 float when HLDA is asserted.
PCHK#	The status of the data bus parity check is driven on PCHK#.
PEN#	Even data parity with D63–D0 should be returned to the processor on the DP pin. If a data parity error occurs, and PEN# is enabled, the cycle is latched and a machine check exception is taken if CR4.MCE = 1.

18.1.22 D/P#

D/P#	Dual Processor / Primary Processor
	Indicates whether the Dual processor or the Primary processor is driving the bus.
	Synchronous Output

Signal Description

The D/P# pin is driven low when the Primary processor is driving the bus. Otherwise, the Primary processor drives this pin high to indicate that the Dual processor owns the bus. The D/P# pin can be sampled for the current cycle with ADS#. This pin is defined only on the Primary processor. In a single socket system design, D/P# pin should be left NC.

When Sampled/Driven

The D/P# pin is always driven by the Primary processor and should be sampled with ADS# of the current cycle.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
ADS#	D/P# is valid for the current cycle with ADS# (like a status pin).
CPUTYP	D/P# is driven by the processor when the CPUTYP signal is strapped to V_{SS} (or left unconnected).
SMIACT#	When operating in dual processing mode, D/P# qualifies the SMIACT# SMM indicator.

18.1.23 DP7–DP0

DP7–DP0	Data Parity
	Bidirectional data parity pins for the data bus.
	Input/Output

Signal Description

These are the bidirectional Data Parity pins for the processor. There is one parity pin for each byte of the data bus. For example, DP7 applies to D63–D56 and DP0 applies to D7–D0.

As outputs, the data parity pins are driven by the processor with even parity information for writes in the same clock as write data. Even parity means that there are an even number of HIGH logic values on the eight corresponding data bus pins and the parity pin.

As inputs, even parity information must be driven back to the processor on these pins in the same clock as the data to ensure that the correct parity check status is indicated by the processor.

The value read on the data parity pins does not affect program execution unless PEN# is also asserted. If PEN# is not asserted, the value returned on the DP pins is used only to determine even parity and drive the PCHK# output with the proper value. If PEN# is asserted when a parity error occurs, the cycle address and type are latched in the MCA and MCT registers. If in addition, the MCE bit in CR4 is set, a machine check exception is taken.

It is the responsibility of the system to take appropriate actions if a parity error occurs. If parity checks are not implemented in the system, the DP7–DP0 and PEN# pins should be tied to V_{CC} through a pullup resistor and the PCHK# pin may be ignored.

When Sampled/Driven

As outputs, the data parity pins are driven by the processor with even parity information in the same clock as write data. The parity remains valid until sampled by the assertion of BRDY# by the system.

As inputs, even parity information must be driven back to the processor on these pins in the same clock as the data to ensure that the correct parity check status is indicated by the processor. The data parity pins must be at a valid logic level and meet setup and hold times when sampled.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
BE7#–BE0#	Even data parity is checked/driven only on the data bytes enabled by BE7#–BE0#.
BOFF#	DP7–DP0 are floated one clock after BOFF# is asserted.
BRDY#	DP7–DP0 are sampled with BRDY# for reads.
D63–D0	The DP7–DP0 pins are used to create even parity with D63–D0 on a byte by byte basis. DP7–DP0 are driven with D63–D0 for writes.
HLDA	DP7–DP0 are floated when HLDA is asserted.
PCHK#	The status of the data parity check is driven on the PCHK# output.
PEN#	The DP7–DP0 pins are used to create even parity with D63–D0. If even parity is not detected, and PEN# is enabled, the cycle address and type are latched. If in addition CR4.MCE = 1, the machine check exception is taken.

18.1.24 DPEN#

DPEN#	Second Socket Occupied
	Configuration signal which indicates that the second socket in a dual socket system is occupied.
	Synchronous Input (to the processor) Synchronous Output (from the processor, when configured as a Dual processor)

Signal Description

DPEN# is driven during RESET by the processor when configured as a Dual processor to indicate to the Primary processor in the first socket that there is a Dual processor present in the system.

This pin has an internal pullup resistor and is sampled at the falling edge of RESET. When using an active circuit to override the internal pullup resistor, the driver should have an internal effective pulldown resistance of 1 KOhms or less.

When Sampled/Driven

DPEN# is driven during RESET by the Dual processor, and sampled at the falling edge of RESET by the Primary processor. This pin becomes PICD0 following the falling edge of RESET. This pin should be valid and stable two clocks before and after the falling edge of RESET.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
CPUTYP	When CPUTYP is strapped to V_{CC}, DPEN# is driven active to indicate that the second socket is occupied.
RESET	DPEN# is valid during the falling edge of RESET.
PICD0	DPEN# shares a pin with PICD0.

18.1.25 EADS#

EADS#	External Address Strobe
	Signals the processor to run an inquire cycle with the address on the bus.
	Synchronous Input

Signal Description

The EADS# input indicates that a valid external address has been driven onto the processor address pins to be used for an inquire cycle. The address driven to the processor when EADS# is sampled asserted is checked with the current cache contents. The HIT# and HITM# signals are driven to indicate the result of the comparison. When the INV pin is returned active (high) to the processor in the same clock as EADS# is sampled asserted, an inquire hit will result in that line being invalidated. When the INV pin is returned inactive (low), an inquire hit will result in that line being marked Shared (S).

When Sampled

To guarantee recognition, EADS# should be asserted two clocks after an assertion of AHOLD or BOFF#, or one clock after an assertion of HLDA. In addition, the processor ignores an assertion of EADS# if the processor is driving the address bus, or if HITM# is active, or in the clock after ADS# or EADS# is asserted.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
A31–A5	The inquire cycle address must be valid on A31–A5 when EADS# is sampled asserted.
A4–A3	These signals must be at a valid logic level when EADS# is sampled asserted.
AHOLD	EADS# is recognized while AHOLD is asserted.
AP	AP is sampled when EADS# is sampled asserted.
APCHK#	APCHK# is driven to its valid level two clocks after EADS# is sampled asserted.
BOFF#	EADS# is recognized while BOFF# is asserted.
HIT#	HIT# is driven to its valid level two clocks after EADS# is sampled asserted.
HITM#	HITM# is driven to its valid level two clocks after EADS# is sampled asserted.
HLDA	EADS# is recognized while HLDA is asserted.
INV	INV is sampled with EADS# to determine the final state of the cache line in the case of an inquire hit.

18.1.26 EWBE#

EWBE#	External Write Buffer Empty
	Provides the option of strong write ordering to the memory system.
	Synchronous Input

Signal Description

The External write Buffer Empty input, when inactive (high), indicates that a writethrough cycle is pending in the external system. When the processor generates a write (memory or I/O), and EWBE# is sampled inactive, the processor holds off all subsequent writes to all E or M-state lines until all writethrough cycles have completed, as indicated by EWBE# being active. In addition, if the processor has a write pending in a write buffer, the processor also holds off all subsequent writes to E- or M-state lines. This insures that writes are visible from outside the processor in the same order as they were generated by software.

When the processor serializes instruction execution through the use of a serializing instruction, it waits for the EWBE# pin to go active before fetching and executing the next instruction.

After the OUT or OUTS instructions are executed, the processor ensures that EWBE# has been sampled active before beginning to execute the next instruction. Note that the instruction may be prefetched if EWBE# is not active, but it does not execute until EWBE# is sampled active.

When Sampled

EWBE# is sampled with each BRDY# of a write cycle. If sampled deasserted, the processor repeatedly samples EWBE# in each clock until it is asserted. Once sampled asserted, the processor ignores EWBE# until the next BRDY# of a write cycle.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
BRDY#	EWBE# is sampled with each BRDY# of a write cycle.
SMIACT#	SMIACT# is not asserted until EWBE# is asserted.

18.1.27 FERR#

FERR#	Floating-Point Error
	The floating-point error output is driven active when an unmasked floating-point error occurs.
	Synchronous Output

Signal Description

The Floating-Point Error output is driven active when an unmasked floating-point error occurs. FERR# is similar to the ERROR# pin on the Intel387 math coprocessor. FERR# is included for compatibility with systems using DOS type floating-point error reporting.

In some cases, FERR# is asserted when the next floating-point instruction is encountered and in other cases it is asserted before the next floating-point instruction is encountered depending upon the execution state of the instruction causing the exception.

The following class of floating-point exceptions drive FERR# at the time the exception occurs (i.e., before encountering the next floating-point instruction):

1. Stack fault, all invalid operation exceptions and denormal exceptions on: all transcendental instructions, FSCALE, FXTRACT, FPREM, FPREM(1), FBLD, FLD_extended, FRNDINT, and stack fault and invalid operation exceptions on Floating-Point arithmetic instructions with an integer operand (FIADD/FIMUL/FISUB/FIDIV, etc.).
2. All real stores (FST/FSTP), Floating-Point integer stores (FIST/FISTP) and BCD store (FBSTP) (true for all exception on stores except Precision Exception).

The following class of floating-point exceptions drive FERR# only after encountering the next floating-point instruction. Note that the embedded Pentium processor with MMX technology reports a pending floating-point exception (assert FERR#) upon encountering the next floating-point or MMX instruction.

1. Numeric underflow, overflow and precision exception on: Transcendental instructions, FSCALE, FXTRACT, FPREM, FPREM(1), FRNDINT, and Precision Exception on all types of stores to memory.
2. All exceptions on basic arithmetic instructions (FADD/FSUB/FMUL/FDIV/ FSQRT/FCOM/FUCOM...)

FERR# is deasserted when the FCLEX, FINIT, FSTENV, or FSAVE instructions are executed. In the event of a pending unmasked floating-point exception the FNINIT, FNCLEX, FNSTENV, FNSAVE, FNSTSW, FNSTCW, FNENI, FNDISI, and FNSETPM instructions assert the FERR# pin. Shortly after the assertion of the pin, an interrupt window is opened during which the processor samples and services interrupts, if any. If no interrupts are sampled within this window, the processor then executes these instructions with the pending unmasked exception. However, for the FNCLEX, FNINIT, FNSTENV, and FNSAVE instructions, the FERR# pin is deasserted to enable the execution of these instructions. For details please refer to the *Intel Architecture Software Developer's Manual*, Volume 1 (Chapter 7 and Appendix D).

This signal is undefined when the embedded Pentium processor is configured as a Dual processor.

When Sampled/Driven

FERR# is driven in every clock and is not floated during bus HOLD or BOFF#. The FERR# signal is glitch free.

The embedded Pentium processor, when configured as a Dual processor, does not drive this signal to valid levels.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
CPUTYP	When CPUTYP is strapped to V_{CC}, the FERR# output is undefined.

18.1.28 FLUSH#

FLUSH#	Cache Flush
	Writes all modified lines in the data cache back and flushes the code and data caches.
	Asynchronous Input (Normal, Uni-processor mode) Synchronous Input (Dual processor mode)

Signal Description

When asserted, the Cache Flush input forces the processor to writeback all modified lines in the data cache and invalidate both internal caches. A Flush Acknowledge special cycle is generated by the processor, indicating completion of the invalidation and writeback.

FLUSH# is implemented in the processor as an interrupt, so it is recognized on instruction boundaries. External interrupts are ignored while FLUSH# is being serviced. Once FLUSH# is sampled active, it is ignored until the flush acknowledge special cycle is driven.

If FLUSH# is sampled low when RESET transitions from high to low, three-state test mode is entered.

The processor, when operating with a second processor in dual processing mode, incorporates a mechanism to present an atomic cache flush operation to the system. The Dual processor performs the cache flush operation first, then grants the bus to the Primary processor. The Primary processor flushes its internal caches, and then runs the cache flush special cycle. This could cause the total flush latency of two embedded Pentium processors in dual processor mode to be up to twice that of the embedded Pentium processor in uni-processor mode.

The flush latency of the embedded Pentium processor with MMX technology may be up to twice that of the embedded Pentium processor due to the implementation of larger on-chip caches.

When Sampled/Driven

FLUSH# is sampled on every rising clock edge. FLUSH# is falling edge sensitive and is recognized on instruction boundaries. Recognition of FLUSH# is guaranteed in a specific clock if it is asserted synchronously and meets the setup and hold times. If it meets setup and hold times, FLUSH# need only be asserted for one clock. To guarantee recognition if FLUSH# is asserted

asynchronously, it must have been deasserted for a minimum of two clocks before being returned active to the embedded Pentium processor and remain asserted for a minimum pulse width of two clocks.

If the processor is in the HALT or Shutdown state, FLUSH# is still recognized. The processor returns to the HALT or Shutdown state after servicing the FLUSH#.

If FLUSH# is sampled low when RESET transitions from high to low, three-state test mode is entered. If RESET is negated synchronously, FLUSH# must be at its valid level and meet setup and hold times on the clock before the falling edge of RESET. If RESET is negated asynchronously, FLUSH# must be at its valid level two clocks before and after RESET transitions from high to low.

When operating in a dual processing system, FLUSH# must be sampled synchronously to the rising CLK edge to ensure both processors recognize an active FLUSH# signal in the same clock.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
ADS# and cycle definition pins.	Writeback cycles are driven as a result of FLUSH# assertion. The Flush Special Cycle is driven as a result of FLUSH# assertion.
RESET	If FLUSH# is sampled low when RESET transitions from high to low, three-state test mode is entered.
CPUTYP	When operating in dual processing mode, the FLUSH# inputs become synchronous to the processor clock.

18.1.29 FRCMC#

FRCMC#	Functional Redundancy Checking Master/Checker Configuration
	Determines whether the processor is configured as a Master or Checker.
	Asynchronous Input

Note: Functional Redundancy Checking is not supported on the embedded Pentium processor with MMX technology. The FRCMC# pin is defined only for the embedded Pentium processor. This pin should be left as a "NC" or tied to V_{CC3} via an external pullup resistor on the embedded Pentium processor with MMX technology.

Signal Description

The Functional Redundancy Checking Master/Checker Configuration input is sampled in every clock that RESET is asserted to determine whether the processor is configured in master mode (FRCMC# high) or checker mode (FRCMC# low). When configured as a master, the processor drives its output pins as required by the bus protocol. When configured as a checker, the processor three-states all outputs (except IERR# and TDO) and samples the output pins that would normally be driven in master mode. If the sampled value differs from the value computed internally, the Checker processor asserts IERR# to indicate an error.

Note that the final configuration as a master or checker is set after RESET and may not be changed other than by a subsequent RESET. FRCMC# is sampled in every clock that RESET is asserted to prevent bus contention before the final mode of the processor is determined.

When Sampled

This pin is sampled in any clock in which RESET is asserted. FRCMC# is sampled in the clock before RESET transitions from high to low to determine the final mode of the processor. If RESET is negated synchronously, FRCMC# must be at its valid level and meet setup and hold times on the clock before the falling edge of RESET. If RESET is negated asynchronously, FRCMC# must be at its valid level two clocks before and after RESET transitions from high to low.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
IERR#	IERR# is asserted by the Checker processor in the event of an FRC error.
RESET	FRCMC# is sampled when RESET is asserted to determine if the processor is in Master or Checker mode.

18.1.30 HIT#

HIT#	Inquire Cycle Hit/Miss
	Externally indicates whether an inquire cycle resulted in a hit or miss.
	Synchronous Input/Output

Signal Description

The HIT# output is driven to reflect the outcome of an inquire cycle. If an inquire cycle hits a valid line (M, E, or S) in either the processor data or instruction cache, HIT# is asserted two clocks after EADS# has been sampled asserted by the processor. If the inquire cycle misses the processor cache, HIT# is negated two clocks after EADS# is sampled asserted. This pin changes its value only as a result of an inquire cycle and retains its value between cycles.

When operating in dual processing mode, the embedded Pentium processor uses this signal for private snooping.

When Sampled/Driven

HIT# reflects the hit or miss outcome of the inquire cycle two clocks after EADS# is sampled asserted. After RESET, this pin is driven high. It changes it value only as a result of an inquire cycle. This pin is always driven. It is not floated during bus HOLD or BOFF#.

This signal becomes an Input/Output when two embedded Pentium processors are operating together in dual processing mode.

Relation to Other Signals

Pin Symbol	Relative to Other Signals
A31–A5	HIT# is driven to indicate whether the inquire address driven on A31–A5 is valid in an internal cache.
EADS#	HIT# is driven two clocks after EADS# is sampled asserted to indicate the outcome of the inquire cycle.
HITM#	HITM# is never asserted without HIT# also being asserted.

18.1.31 HITM#

HITM#	Inquire Cycle Hit/Miss to a Modified Line
	Externally indicates whether an inquire cycle hit a modified line in the data cache.
	Synchronous Input/Output

Signal Description

The HITM# output is driven to reflect the outcome of an inquire cycle. If an inquire cycle hits a modified line in the embedded Pentium processor data cache, HITM# is asserted two clocks after EADS# has been sampled asserted by the processor and a writeback cycle is scheduled to be driven to the bus. If the inquire cycle misses the processor cache, HITM# is negated two clocks after EADS# is sampled asserted.

HITM# can be used to inhibit another bus master from accessing the data until the line is completely written back.

HITM# is asserted two clocks after an inquire cycle hits a modified line in the processor cache. ADS# for the writeback cycle is asserted no earlier than two clocks after the assertion of HITM#. ADS# for the writeback cycle is driven even if AHOLD for the inquire cycle is not yet deasserted. ADS# for a writeback of an external snoop cycle is the only ADS# that is driven while AHOLD is asserted.

When operating in dual processing mode, the embedded Pentium processor uses this signal for private snooping.

When Sampled/Driven

HITM# is driven two clocks after EADS# is sampled asserted to reflect the outcome of the inquire cycle. HITM# remains asserted until two clocks after the last BRDY# of writeback is returned. This pin is always driven. It is not floated during bus HOLD or BOFF#.

This signal becomes an input/output when two embedded Pentium processors are operating together in dual processing mode.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
A31–A5	HITM# is driven to indicate whether the inquire address driven on A31–A5 is in the modified state in the data cache.
EADS#	HITM# is driven two clocks after EADS# is sampled asserted.
HIT#	HITM# is never asserted without HIT# also being asserted.

18.1.32 HLDA

HLDA	Bus Hold Acknowledge
	External indication that the processor outputs are floated.
	Synchronous Input/Output

Signal Description

The Bus Hold Acknowledge output goes active in response to a hold request presented on the HOLD pin. HLDA indicates that the processor has given the bus to another local bus master. Internal instruction execution continues from the internal caches during bus HOLD/HLDA.

When leaving bus hold, HLDA is driven inactive and the processor resumes driving the bus. A pending bus cycle is driven in the same clock in which HLDA is deasserted by the processor and one clock after HLDA is deasserted by the processor.

The operation of HLDA is not affected by the assertion of BOFF#. If HOLD is asserted while BOFF# is asserted, HLDA is asserted two clocks later. If HOLD goes inactive while BOFF# is asserted, HLDA is deasserted two clocks later.

When operating in dual processing mode, the embedded Pentium processor uses this signal for private snooping.

When Sampled/Driven

When the embedded Pentium processor bus is idle, HLDA is driven high two clocks after HOLD is asserted, otherwise, HLDA is driven high two clocks after the last BRDY# of the current cycle is returned. It is driven active in the same clock that the embedded Pentium processor floats its bus. When leaving bus hold, HLDA is driven inactive two clocks after HOLD is deasserted and the embedded Pentium processor resumes driving the bus. The HLDA signal is glitch free.

This signal becomes an input/output when two embedded Pentium processors are operating together in dual processing mode.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
A31–A3 ADS# AP BE7#–BE3# CACHE# D/C# D63–D0 DP7–DP0 LOCK# M/IO# PCD PWT SCYC W/R#	These signals float in response to HLDA.
BOFF#	The same pins are floated when HLDA or BOFF# is asserted.
EADS#	EADS# is recognized while HLDA is asserted.
HOLD	The assertion of HOLD causes HLDA to be asserted when all outstanding cycles are complete.

18.1.33 HOLD

HOLD	Bus Hold
	The bus hold request input allows another bus master complete control of the processor bus.
	Synchronous Input

Signal Description

The Bus Hold request input allows another bus master complete control of the embedded Pentium processor bus. In response to HOLD, after completing all outstanding bus cycles the embedded Pentium processor floats most of its output and input/output pins and asserts HLDA. The embedded Pentium processor maintains its bus in this state until HOLD is deasserted. Cycles that are locked together are not interrupted by bus HOLD. HOLD is recognized during RESET.

When Sampled

HOLD is sampled on every rising clock edge including during RESET and INIT.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
A31–A3 ADS# AP BE7#–BE3# CACHE# D/C# D63–D0 DP7–DP0 LOCK# M/IO# PCD PWT SCYC W/R#	These are the signals floated in response to HOLD.
HLDA	HLDA is asserted when the processor relinquishes the bus in response to the HOLD request.

18.1.34 IERR#

IERR#	Internal or Functional Redundancy Check Error. (Functional Redundancy Checking is not supported on the embedded Pentium® processor with MMX™ technology or the low-power embedded Pentium processor with MMX technology).
	Alerts the system of internal parity errors and functional redundancy errors.
	Output

Signal Description

The Internal Error output is used to alert the system of two types of errors, internal parity errors and functional redundancy errors.

If a parity error occurs on a read from an internal array (reads during normal instruction execution, reads during a flush operation, reads during BIST and testability cycles, and reads during inquire cycles), the embedded Pentium processor asserts the IERR# pin for one clock and then shuts down. Shutdown occurs provided the processor is not prevented from doing so by the error.

If the embedded Pentium processor is configured as a checker (by FRCMC# being sampled low while RESET is asserted) and a mismatch occurs between the value sampled on the pins and the value computed internally, the embedded Pentium processor asserts IERR# two clocks after the mismatched value is returned. Shutdown is not entered as a result of a function redundancy error.

It is the responsibility of the system to take appropriate action if an internal parity or FRC error occurs.

When Driven

IERR# is driven in every clock. While RESET is active IERR# is driven high. After RESET is deasserted, IERR# is not asserted due to an FRC mismatch until after the first clock of the first bus cycle. Note however that IERR# may be asserted due to an internal parity error before the first bus cycle. IERR# is asserted for one clock for each detected FRC or internal parity error, two clocks after the error is detected. IERR# is asserted for each detected mismatch, so IERR# may be asserted for more than one consecutive clock.

IERR# is not floated with HOLD or BOFF#. IERR# is a glitch free signal.

When paging is turned on, an additional parity check occurs to page 0 for all TLB misses. If this access is a valid entry in the cache and this entry also has a parity error, then IERR# is asserted and shutdown occurs even though the pipeline is frozen to service the TLB miss.

During a TLB miss, a cache lookup occurs (to the data cache for a data TLB miss, or the code cache for a code TLB miss) to a default page 0 physical address until the correct page translation becomes available. At this time, if a valid cache entry is found at the page 0 address, then parity is checked on the data read out of the cache. However, the data is not used until after the correct page address becomes available. If this valid line contains a true parity error, then the error is reported. This does not cause an unexpected parity error. It can cause a parity error and shutdown at a time when the data is not being used because the pipeline is frozen to service the TLB miss. However, it still remains that a true parity error must exist within the cache in order for IERR# assertion and shutdown to occur. For more details on TLB, refer to Section 3.7 of the *Intel Architecture Software Developer's Manual*, Volume 1.

Relation to Other Signals

Pin Symbol	Relative to Other Signals
FRCMC#	If the processor is configured as a Checker, IERR# is asserted in the event of an FRC error.

18.1.35 IGNNE#

IGNNE#	Ignore Numeric Exception
	Determines whether or not numeric exceptions should be ignored.
	Asynchronous Input

Signal Description

This is the Ignore Numeric Exception input. This pin has no effect when the NE bit in CR0 is set to 1. When the CR0.NE bit is 0, this pin functions as follows:

When the IGNNE# pin is asserted, the embedded Pentium processor ignores any pending unmasked numeric exception and continues executing floating-point instructions for the entire duration that this pin is asserted.

When IGNNE# is not asserted and a pending unmasked numeric exception exists, (SW.ES = 1), the embedded Pentium processor behaves as follows:

On encountering a floating-point instruction that is one of FNINIT, FNCLEX, FNSTENV, FNSAVE, FNSTSW, FNSTCW, FNENI, FNDISI, or FNSETPM, the embedded Pentium processor asserts the FERR# pin. Subsequently, the processor opens an interrupt sampling window. The interrupts are checked and serviced during this window. If no interrupts are sampled within this window, the processor then executes these instructions in spite of the pending unmasked exception. For further details please refer to the *Intel Architecture s Software Developer's Manual,* Volume1 (Chapter 7 and Appendix D).

On encountering any floating-point instruction other than FINIT, FCLEX, FSTENV, FSAVE, FSTSW, FSTCW, FENI, FDISI, or FSETPM, the embedded Pentium processor stops execution and waits for an external interrupt.

The embedded Pentium processor, when configured as a Dual processor, ignores the IGNNE# input.

When Sampled/Driven

IGNNE# is sampled on every rising clock edge. Recognition of IGNNE# is guaranteed in a specific clock if it is asserted synchronously and meets setup and hold times. To guarantee recognition if IGNNE# is asserted asynchronously, it must have been deasserted for a minimum of two clocks before being returned active to the embedded Pentium processor and remain asserted for a minimum pulse width of two clocks.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
CPUTYP	When strapped to V_{CC}, the processor ignores the IGNNE# input.

18.1.36 INIT

INIT	Initialization
	Forces the processor to begin execution in a known state without flushing the caches or affecting the floating-point state.
	Asynchronous Input

Signal Description

The Initialization input forces the embedded Pentium processor to begin execution in a known state. The processor state after INIT is the same as the state after RESET except that the internal caches, write buffers, model specific registers, and floating-point registers retain the values they had prior to INIT. The embedded Pentium processor starts execution at physical address FFFFFFF0H.

INIT can be used to help performance for DOS extenders written for the 80286. INIT provides a method to switch from protected to real mode while maintaining the contents of the internal caches and floating-point state. INIT may not be used instead of RESET after power-up.

Once INIT is sampled active, the INIT sequence begins on the next instruction boundary (unless a higher priority interrupt is requested before the next instruction boundary). The INIT sequence continues to completion and then normal processor execution resumes, independent of the deassertion of INIT. ADS# is asserted to drive bus cycles even if INIT is not deasserted.

If INIT is sampled high when RESET transitions from high to low, the embedded Pentium processor performs built-in self test (BIST) prior to the start of program execution.

When Sampled

INIT is sampled on every rising clock edge. INIT is an edge sensitive interrupt. Recognition of INIT is guaranteed in a specific clock if it is asserted synchronously and meets the setup and hold times. To guarantee recognition if INIT is asserted asynchronously, it must have been deasserted for a minimum of two clocks before being returned active to the embedded Pentium processor and remain asserted for a minimum pulse width of two clocks. INIT must remain active for three clocks prior to the BRDY# of an I/O write cycle to guarantee that the embedded Pentium processor recognizes and processes INIT right after an I/O write instruction.

If INIT is sampled high when RESET transitions from high to low the embedded Pentium processor performs built-in self test. If RESET is driven synchronously, INIT must be at its valid level the clock before the falling edge of RESET. If RESET is driven asynchronously, INIT must be at its valid level two clocks before and after RESET transitions from high to low.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
RESET	If INIT is sampled high when RESET transitions from high to low, BIST will be performed.

18.1.37 INTR

INTR	External Interrupt
	Indicates that an external interrupt has been generated.
	Asynchronous Input

Signal Description

The INTR input indicates that an external interrupt has been generated. The interrupt is maskable by the IF bit in the EFLAGS register. If the IF bit is set, the embedded Pentium processor will vector to an interrupt handler after the current instruction execution is completed. Upon recognizing the interrupt request, the embedded Pentium processor will generate two locked interrupt acknowledge bus cycles in response to the INTR pin going active. INTR must remain active until the first interrupt acknowledge cycle is completed to assure that the interrupt is recognized.

When the local APIC is hardware disabled, this pin is the INTR input for the processor. It bypasses the local APIC in that case.

When the local APIC is hardware enabled, this pin becomes the programmable interrupt LINT0. It can be programmed in software in any of the interrupt modes. Since this pin is the INTR input when the APIC is disabled, it is logical to program the vector table entry for this pin as ExtINT (i.e., through local mode). In this mode, the interrupt signal is passed on to the processor through the local APIC. The processor generates the interrupt acknowledge, INTA, cycle in response to this interrupt and receives the vector on the processor data bus.

When Sampled/Driven

INTR is sampled on every rising clock edge. INTR is an asynchronous input, but recognition of INTR is guaranteed in a specific clock if it is asserted synchronously and meets the setup and hold times. To guarantee recognition if INTR is asserted asynchronously it must have been deasserted for a minimum of two clocks before being returned active to the embedded Pentium processor.

Note: This applies only when using the APIC in the through local (virtual wire) mode. Once INTR has been asserted (by a rising edge), it must not be asserted again until after the end of the first resulting interrupt acknowledge cycle. Otherwise, the new interrupt may not be recognized. The end of an interrupt acknowledge cycle is defined by the end of the system's BRDY# response to the processor cycle. Note that the APIC through local mode was designed to match the protocol of an 8259A PIC, and an 8259A will always satisfy this requirement.

To ensure INTR is not recognized inadvertantly a second time, deassert INTR no later than the BRDY# of the second INTA cycle and no earlier than the BRDY# of the first INTA cycle.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
ADS# and cycle definition pins	An interrupt acknowledge cycle is driven as a result of the INTR pin assertion.
APICEN	When the APICEN configuration input is sampled inactive, this input becomes the INTR interrupt.
LINT0	INTR shares a pin with LINT0.
LOCK#	LOCK# is asserted for interrupt acknowledge cycles.

18.1.38 INV

INV	Invalidation Request
	Determines final state of a cache line as a result of an inquire hit.
	Synchronous Input

Signal Description

The INV input is driven to the embedded Pentium processor during an inquire cycle to determine the final cache line state (S or I) in case of an inquire cycle hit. If INV is returned active (high) to the embedded Pentium processor in the same clock as EADS# is sampled asserted, an inquire hit will result in that line being invalidated. If the INV pin is returned inactive (low), an inquire hit will result in that line being marked Shared (S). If the inquire cycle is a miss in the cache, the INV input has no effect.

If an inquire cycle hits a modified line in the data cache, the line will be written back regardless of the state of INV.

When Sampled

The INV input is sampled with the EADS# of the inquire cycle.

Relation to Other Signals

Pin Symbol	Relative to Other Signals
A31–A5	INV determines if the inquire address driven to the processor on A31–A5 should be invalidated or marked as shared if it is valid in an internal cache.
EADS#	INV is sampled with EADS#.

18.1.39 KEN#

KEN#	Cache Enable
	Indicates to the processor whether or not the system can support a cache line fill for the current cycle.
	Synchronous Input

Signal Description

KEN# is the cache enable input. It is used to determine whether the current cycle is cacheable or not and consequently is used to determine cycle length.

When the embedded Pentium processor generates a read cycle that can be cached (CACHE# asserted) and KEN# is active, the cycle will be transformed into a burst cache linefill. During a cache line fill the byte enable outputs should be ignored and valid data must be returned on all 64 data lines. The embedded Pentium processor will expect 32 bytes of valid data to be returned in four BRDY# transfers.

If KEN# is not sampled active, a linefill will not be performed (regardless of the state of CACHE#) and the cycle will be a single transfer read.

Once KEN# is sampled active for a cycle, the cacheability cannot be changed. If a cycle is restarted for any reason after the cacheability of the cycle has been determined, the same cacheability attribute on KEN# must be returned to the processor when the cycle is redriven.

When Sampled

KEN# is sampled once in a cycle to determine cacheability. It is sampled and latched with the earlier of the first BRDY# or NA# of a cycle, however it must meet setup and hold times on every clock edge.

Relation to Other Signals

Pin Symbol	Relative to Other Signals
BRDY#	KEN# is sampled with the earlier of the first BRDY# or NA# for that cycle. Also, in conjunction with the CACHE# input, KEN# determines whether the bus cycle will consist of 1 or 4 transfers (assertions of BRDY#).
CACHE#	KEN# determines cacheability only if the CACHE# pin is asserted.
NA#	KEN# is sampled with the earlier of the first BRDY# or NA# for that cycle.
W/R#	KEN# determines cacheability only if W/R# indicates a read.

18.1.40 LINT1–LINT0

LINT1–LINT0	Local Interrupts 1 and 0
	APIC Programmable Interrupts.
	Asynchronous Inputs

Signal Description

When the local APIC is hardware enabled, these pins become the programmable interrupts (LINT1–LINT0). They can be programmed in software in any of the interrupt modes. Since these pins are the INTR and NMI inputs when the APIC is disabled, it is logical to program the vector table entry for them as ExtINT (i.e. through local mode) and NMI, respectively. In this mode, the interrupt signals are passed on to the processor through the local APIC.

When the local APIC is hardware disabled, these pins are the INTR and NMI inputs for the processor. They bypass the APIC in that case.

When Sampled

LINT1–LINT0 are sampled on every rising clock edge. LINT1–LINT0 are asynchronous inputs, but recognition of LINT1–LINT0 are guaranteed in a specific clock if they are asserted synchronously and meets the setup and hold times. To guarantee recognition if LINT1–LINT0 are asserted asynchronously they must have been deasserted for a minimum of two clocks before being returned active to the embedded Pentium processor.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
APICEN	When the APICEN configuration input is sampled inactive, these inputs become the INTR and NMI interrupts.
INTR	INTR shares a pin with LINT0.
NMI	NMI shares a pin with LINT1.

18.1.41 LOCK#

LOCK#	Bus Lock
	Indicates to the system that the current sequence of bus cycles should not be interrupted.
	Synchronous Input/Output

Signal Description

The bus lock output indicates that the embedded Pentium processor is running a read-modify-write cycle where the external bus must not be relinquished between the read and write cycles. Read-modify-write cycles of this type are used to implement memory based semaphores. Interrupt Acknowledge cycles are also locked.

If a cycle is split due to a misaligned memory operand, two reads followed by two writes may be locked together. When LOCK# is asserted, the current bus master should be allowed exclusive access to the system bus.

The embedded Pentium processor will not allow a bus hold when LOCK# is asserted, but address holds (AHOLD) and BOFF# are allowed. LOCK# is floated during bus hold.

All locked cycles will be driven to the external bus. If a locked address hits a valid location in one of the internal caches, the cache location is invalidated (if the line is in the modified state, it is written back before it is invalidated). Locked read cycles will not be transformed into cache line fill cycles regardless of the state of KEN#.

LOCK# is guaranteed to be deasserted for at least one clock between back to back locked cycles.

When operating in dual processing mode, the embedded Pentium processor uses this signal for private snooping.

When Sampled/Driven

LOCK# goes active with the ADS# of the first locked bus cycle and goes inactive after the BRDY# is returned for the last locked bus cycle. The LOCK# signal is glitch free.

This signal becomes an input/output when two embedded Pentium processors are operating together in dual processing mode.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
ADS#	LOCK# is driven with the ADS# of the first locked cycle.
BOFF#	LOCK# floats one clock after BOFF# is asserted.
BRDY#	LOCK# is deasserted after the last BRDY# of the locked sequence.
HLDA	LOCK# floats when HLDA is asserted.
NA#	ADS# is not asserted to pipeline an additional cycle if LOCK# is asserted, regardless of the state of NA#.
INTR	LOCK# is asserted for interrupt acknowledge cycles.
SCYC	SCYC is driven active if the locked cycle is misaligned.

18.1.42 M/IO#

M/IO#	Memory Input/Output
	Distinguishes a memory access from an I/O access.
	Synchronous Input/Output

Signal Description

The Memory/Input-Output signal is one of the primary bus cycle definition pins. M/IO# distinguishes between memory (M/IO# =1) and I/O (M/IO# =0) cycles.

When operating in dual processing mode, the embedded Pentium processor uses this signal for private snooping.

When Sampled/Driven

M/IO# is driven valid in the same clock as ADS# and the cycle address. It remains valid from the clock in which ADS# is asserted until the earlier of the last BRDY# or the clock after NA#.

This signal becomes an input/output when two embedded Pentium processors are operating together in dual processing mode.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
ADS#	M/IO# is driven to its valid state with ADS#.
BOFF#	M/IO# floats one clock after BOFF# is asserted.
HLDA	M/IO# floats when HLDA is asserted.

18.1.43 NA#

NA#	Next Address
	Indicates that external memory is prepared for a pipelined cycle.
	Synchronous Input

Signal Description

The Next Address input, when active, indicates that external memory is ready to accept a new bus cycle although all data transfers for the current cycle have not yet completed. This is referred to as bus cycle pipelining.

The embedded Pentium processor will drive out a pending cycle in response to NA# no sooner than two clocks after NA# is asserted. The embedded Pentium processor supports up to 2 outstanding bus cycles. ADS# is not asserted to pipeline an additional cycle if LOCK# is asserted, or during a writeback cycle. In addition, ADS# will not be asserted to pipeline a locked cycle or a writeback cycle into the current cycle.

NA# is latched internally, so once it is sampled active during a cycle, it need not be held active to be recognized. The KEN#, and WB/WT# inputs for the current cycle are sampled with the first NA#, if NA# is asserted before the first BRDY# of the current cycle.

When Sampled

NA# is sampled in all T2, TD and T2P clocks.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
ADS#	If NA# is sampled asserted and an internal bus request is pending, the processor drives out the next bus cycle and asserts ADS#.
KEN#	KEN# is sampled with the earlier of the first BRDY# or NA# for that cycle.
WB/WT#	WB/WT# is sampled with the earlier of the first BRDY# or NA# for that cycle.
LOCK#	ADS# is not asserted to pipeline an additional cycle if LOCK# is asserted, regardless of the state of NA#.
BOFF#	If NA# and BOFF# are asserted simultaneously, BOFF# takes priority and NA# is ignored.

18.1.44 NMI

NMI	Non-Maskable Interrupt
	Indicates that an external non-maskable interrupt has been generated.
	Asynchronous Input

Signal Description

The Non-Maskable interrupt request input indicates that an external non-maskable interrupt has been generated. Asserting NMI causes an interrupt with an internally supplied vector value of 2. External interrupt acknowledge cycles are not generated.

When a second NMI is asserted during the execution of the NMI service routine, the second NMI will remain pending and will be recognized after IRET is executed by the NMI service routine. At most, one assertion of NMI will be held pending. If NMI is reasserted prior to the NMI service routine entry, the reassertion will be ignored.

When the local APIC is hardware enabled, this pin becomes the programmable interrupt LINT1. It can be programmed in software in any of the interrupt modes. Since this pin is the NMI input when the APIC is disabled, it is logical to program the vector table entry for this pin as NMI. In this mode, the interrupt signal is passed on to the processor through the local APIC.

When the local APIC is hardware disabled, this pin is the NMI input for the processor. It bypasses the APIC in that case.

When Sampled

NMI is sampled on every rising clock edge. NMI is rising edge sensitive. Recognition of NMI is guaranteed in a specific clock if it is asserted synchronously and meets the setup and hold times. To guarantee recognition if NMI is asserted asynchronously, it must have been deasserted for a minimum of two clocks before being returned active to the embedded Pentium processor and remain asserted for a minimum pulse width of two clocks.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
APICEN	When the APICEN configuration input is sampled inactive, this input becomes the NMI interrupt.
LINT1	NMI shares a pin with LINT1.

18.1.45 PBGNT#

PBGNT#	Dual Processor Bus Grant
	Indicates to the LRM processor that it will become the MRM in the next clock.
	Synchronous Input (to the Least Recent Master, LRM, processor) Synchronous Output (of the Most Recent Master, MRM, processor)

Signal Description

Two embedded Pentium processors, when configured as dual processors, will arbitrate for the system bus via two private arbitration pins (PBREQ# and PBGNT#). The processor that currently owns the system bus is referred to as the MRM processor. The processor that does not own the bus is referred to as the LRM processor.

PBGNT# is used by the dual processing private arbitration mechanism to indicate that bus ownership will change in the next clock. The LRM processor will request ownership of the processor bus by asserting the private arbitration request pin, PBREQ#. The processor that is currently the MRM and owns the bus, will grant the bus to the LRM as soon as any pending bus transactions have completed. The MRM will notify that the LRM can assume ownership by asserting the private arbitration grant pin, PBGNT#. The PBGNT# pin is always the output of the MRM and an input to the LRM.

Note: In a single socket system design, PBGNT# pin should be left NC. For proper operation, PBGNT# must not be loaded by the system.

When Sampled/Driven

PBGNT# is driven by the MRM processor in response to the PBREQ# signal from the LRM processor. It is asserted following the completion of the current cycle on the processor bus, or in the clock following the request if the bus is idle.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
PBREQ#	PBGNT# is asserted in response to a bus request, PBREQ#, from the LRM processor.
A31–A3, AP, BE7#–BE0#, CACHE#, D/C#, M/IO#, PCD, PWT, SCYC, W/R#	These signals are three-stated for one CLK in response to PBGNT# (when the MRM becomes the LRM).

18.1.46 PBREQ#

PBREQ#	Dual Processor Bus Request
	Indicates to the MRM processor that the LRM processor requires ownership of the bus.
	Synchronous Input (to the Most Recent Master, MRM, processor) Synchronous Output (of the Least Recent Master, LRM, processor)

Signal Description

Two embedded Pentium processors, when configured as dual processors, will arbitrate for the system bus via two private arbitration pins (PBREQ# and PBGNT#). The processor that currently owns the system bus is referred to as the MRM processor. The processor that does not own the bus is referred to as the LRM processor.

PBREQ# is used by the dual processing private arbitration mechanism to indicate that the LRM processor requests bus ownership. The processor that is currently the MRM and owns the bus, will grant the bus to the LRM as soon as any pending bus transactions have completed. The MRM will notify that the LRM can assume ownership by asserting the private arbitration grant pin, PBGNT#. The PBREQ**#** pin is always the output of the LRM and an input to the MRM.

Note: In a single socket system design, PBREQ# pin should be left NC. For proper operation, PBREQ# must not be loaded by the system.

When Sampled/Driven

PBREQ# is driven by the LRM processor, and sampled by the MRM processor.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
PBGNT#	PBGNT# is asserted in response to a bus request, PBREQ#, from the LRM processor.

18.1.47 PCD

PCD	Page Cacheability Disable
	Externally reflects the cacheability paging attribute bit in CR3, PDE, or PTE.
	Output

Signal Description

PCD is driven to externally reflect the cache disable paging attribute bit for the current cycle. PCD corresponds to bit 4 of CR3, the Page Directory Entry, or the Page Table Entry. For cycles that are not paged when paging is enabled (for example I/O cycles), PCD corresponds to bit 4 in CR3. In real mode or when paging is disabled, the PCD pin reflects the cache disable bit in control register 0 (CR0.CD).

PCD is masked by the CD (cache disable) bit in CR0. When CD=1, the embedded Pentium processor forces PCD high. When CD=0, PCD is driven with the value of the Page Table Entry/Directory.

The purpose of PCD is to provide an external cacheability indication on a page by page basis.

When Driven

The PCD pin is driven valid in the same clock as ADS# and the cycle address. It remains valid from the clock in which ADS# is asserted until the earlier of the last BRDY# or the clock after NA#.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
ADS#	PCD is driven valid with ADS#.
BOFF#	PCD floats one clock after BOFF# is asserted.
HLDA	PCD floats when HLDA is asserted.

18.1.48 PCHK#

PCHK#	Data Parity Check
	Indicates the result of a parity check on a data read.
	Synchronous Output

Signal Description

The data parity check pin indicates the result of a parity check on a data read. Data parity is checked during code reads, memory reads, and I/O reads. Data parity is not checked during the first Interrupt Acknowledge cycle. PCHK# indicates the parity status only for the bytes on which valid data is expected. Parity is checked for all data bytes for which a byte enable is asserted. In addition, during a cache linefill, parity is checked on the entire data bus regardless of the state of the byte enables.

PCHK# is driven low two clocks after BRDY# is returned if incorrect parity was returned.

Driving PCHK# is the only effect that bad data parity has on the embedded Pentium processor unless PEN# is also asserted. The data returned to the processor is not discarded.

If PEN# is asserted when a parity error occurs, the cycle address and type will be latched in the MCA and MCT registers. If in addition, the MCE bit in CR4 is set, a machine check exception will be taken.

It is the responsibility of the system to take appropriate actions if a parity error occurs. If parity checks are not implemented in the system, the PCHK# pin may be ignored, and PEN# pulled high (or CR4.MCE cleared).

When operating in dual processing mode, the PCHK# signal can be asserted either 2 OR 3 CLKs following incorrect parity being detected on the data bus. When operating in Dual Processing mode, the PCHK# pin circuit is implemented as a weak driving high output that operates similar to

an open drain output. This implementation allows connection of the two processor PCHK# pins together in a dual processing system with no ill effects. Nominally, this circuit acts like a 360 Ohm resistor tied to V_{CC}.

When Sampled/Driven

PCHK# is driven low two clocks after BRDY# is returned if incorrect parity was returned. PCHK# remains low one clock for each clock in which a parity error was detected. At all other times PCHK# is inactive (high). PCHK# is not floated during bus HOLD or BOFF#. PCHK# is a glitch free signal.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
BRDY#	PCHK# is driven to its valid level two clocks after the assertion of BRDY#.
D63–D0	The DP7–DP0 pins are used to create even parity with D63–D0. If even parity is not returned, the PCHK# pin is asserted.
DP7–DP0	Even data parity with D63–D0 should be returned on to the processor on the dual processor pin. If even parity is not returned, the PCHK# pin is asserted.

18.1.49 PHIT#

PHIT#	Private Inquire Cycle Hit/Miss Indication
	Indicates whether a private, dual processor, inquire cycle resulted in a hit or miss.
	Synchronous Input (to the Most Recent Master, MRM, processor) Synchronous Output (of the Least Recent Master, LRM, processor)

Signal Description

A private snoop interface has been added to the embedded Pentium processor for use in dual processing. The interface consists of two pins (PHIT# and PHITM#).

The LRM processor will initiate a snoop sequence for all ADS# cycles that are initiated by the MRM. The LRM processor will assert the private hit indication (PHIT#) if the data requested by the MRM matches a valid cache line in the LRM. In addition, if the data requested by the MRM matches a valid cache line in the LRM that is in the modified state, the LRM will also assert the PHITM# signal. The system snooping indication signals (HIT#, HITM#) will not change state as a result of a private snoop.

The MRM will use an assertion of the PHIT# signal as an indication that the requested data is being shared with the LRM. Independent of the WB/WT# pin, a cache line will be placed in the shared state if PHIT# is asserted. This will make all subsequent writes to that line externally visible until the state of the line becomes exclusive (E or M states). In a uni-processor system, the line may have been placed in the cache in the E state. In this situation, all subsequent writes to that line will not be visible on the bus until the state is changed to I.

PHIT# will also be driven by the LRM during external snoop operations (e.g., following EADS#) to indicate the private snoop results.

Note: In a single socket system, PHIT# pin should be left NC. For proper operation, PHIT# must not be loaded by the system.

When Sampled/Driven

PHIT# is driven by the LRM processor, and sampled by the MRM processor. It is asserted within two clocks following an assertion of ADS# or EADS#.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
A31–A5	PHIT# is driven to indicate whether the private inquire address driven on A31–A5 is valid in the LRM's on-chip cache.
ADS#	PHIT# is driven within two clocks after ADS# is sampled asserted to indicate the outcome of the private inquire cycle.
EADS#	PHIT# is driven within two clocks after EADS# is sampled asserted to indicate the outcome of the external inquire cycle.
PHITM#	PHITM# is never asserted without PHIT# also being asserted.
WB/WT#	The state of the WB/WT# pin will be ignored by the MRM if the PHIT# pin is sampled active, and the cache line placed in the shared state.

18.1.50 PHITM#

PHITM#	Private Inquire Cycle Hit/Miss to a Modified Line Indication
	Indicates whether a private, dual processor, inquire cycle resulted in a hit or miss to a Modified line.
	Synchronous Input (to the Most Recent Master, MRM, processor) Synchronous Output (of the Least Recent Master, LRM, processor)

Signal Description

A private snoop interface has been added to the embedded Pentium processor for use in dual processing. The interface consists of two pins (PHIT# and PHITM#).

The LRM processor will initiate a snoop sequence for all ADS# cycles that are initiated by the MRM. The LRM processor will assert the private hit indication (PHIT#) if the data requested by the MRM matches a valid cache line in the LRM. In addition, if the data requested by the MRM matches a valid cache line in the LRM that is in the modified state, the LRM will also assert the PHITM# signal. The system snooping indication signals (HIT#, HITM#) will not change state as a result of a private snoop.

PHITM# will also be driven by the LRM during external snoop operations (e.g. following EADS#) to indicate the private snoop results.

Note: In a single socket system, PHITM# pin should be left NC. For proper operation, PHITM# must not be loaded by the system.

When Sampled/Driven

PHITM# is driven by the LRM processor, and sampled by the MRM processor. It is asserted within two clocks following an assertion of ADS# or EADS#.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
A31–A5	PHITM# is driven to indicate whether the private inquire address driven on A31–A5 is modified in the LRM's on-chip cache.
ADS#	PHITM# is driven within two clocks after ADS# is sampled asserted to indicate the outcome of the private inquire cycle.
EADS#	PHITM# is driven within two clocks after EADS# is sampled asserted to indicate the outcome of the external inquire cycle.
PHIT#	PHITM# is never asserted without PHIT# also being asserted.

18.1.51 PICCLK

PICCLK	Processor Interrupt Controller Clock
	This pin drives the clock for the APIC serial data bus operation.
	Input

Signal Description

This pin provides the clock timings for the on-chip APIC unit of the processor. This clock input controls the frequency for the APIC operation and data transmission on the 2-wire APIC serial data bus. All the timings on APIC bus are referenced to this clock.

When hardware disabled, PICCLK must be tied high.

Note that the PICCLK signal on the embedded Pentium processor with MMX technology is 3.3V tolerant, while on the embedded Pentium processor the PICCLK input is 5.0V tolerant.

When Sampled

PICCLK is a clock signal and is used as a reference for sampling the APIC data signals.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
APICEN	PICCLK must be tied or driven high when APICEN is sampled low at the falling edge of RESET.
PICD0–PICD1	External timing parameters for the PICD0–PICD1 pins are measured with respect to this clock.

18.1.52 PICD1–PICD0

PICD1–PICD0	Processor Interrupt Controller Data
	These are the data pins for the 3-wire APIC bus.
	Synchronous Input/Output to PICCLK
	Needs external pull-up resistors.

Signal Description

The PICD1–PICD0 are bidirectional pins which comprise the data portion of the 3-wire APIC bus.

When Sampled/Driven

These signals are sampled with the rising edge of PICCLK.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
APICEN	PICD1 shares a pin with APICEN.
DPEN#	PICD0 shares a pin with DPEN#.

18.1.53 PEN#

PEN#	Parity Enable
	Indicates to the processor that the correct data parity is being returned by the system. Determines if a Machine Check Exception should be taken if a data parity error is detected.
	Synchronous Input

Signal Description

The PEN# input (along with CR4.MCE) determines whether a machine check exception will be taken as a result of a data parity error on a read cycle. If this pin is sampled active in the clock a data parity error is detected, the embedded Pentium processor will latch the address and control signals of the cycle with the parity error in the machine check registers. If, in addition, the machine check enable bit in CR4 is set to “1,” the embedded Pentium processor will vector to the machine check exception before the beginning of the next instruction. If this pin is sampled inactive, it does not prevent PCHK# from being asserted in response to a bus parity error. If systems are using PCHK#, they should be aware of this usage of PEN#.

This pin may be tied to V_{SS}.

When Sampled

This signal is sampled when BRDY# is asserted for memory and I/O read cycles and the second interrupt acknowledge cycle.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
BRDY#	PEN# is sampled with BRDY# for read cycles.
D63–D0	The DP7–DP0 pins are used to create even parity with D63–D0. If even parity is not returned, and PEN# is enabled, the cycle will be latched and an MCE will be taken if CR4.MCE = 1.
DP7–DP0	Even data parity with D63–D0 should be returned to the processor on the dual-processor pins. If even parity is not returned, and PEN# is enabled, the cycle will be latched and a MCE will be taken if CR4.MCE = 1.

18.1.54 PM1–PM0

PM1/BP1–PM0/BP0	Performance Monitoring
	PM1–PM0 externally indicate the status of the performance monitor counter.
	Output pins

Signal Description

The performance monitoring pins can be individually configured to externally indicate either that the associated performance monitoring counter has incremented or that it has overflowed. PM1 indicates the status of CTR1; PM0 indicates the status of CTR0.

BP1 and BP0 are multiplexed with the Performance Monitoring pins (PM1 and PM0). The PB1 and PB0 bits in the Debug Mode Control Register determine if the pins are configured as breakpoint or performance monitoring pins. The pins come out of reset configured for performance monitoring.

When Driven

The BP3–BP2, PM1/BP1–PM0/BP0 pins are driven in every clock and are not floated during bus HOLD or BOFF#.

The PM1/PM0 pins externally indicate the status of the performance monitoring counters on the embedded Pentium processor. These counters are undefined after RESET, and must be cleared or pre-set (using the WRMSR instruction) before they are assigned to specific events.

However, it is possible for these pins to toggle even during RESET. This may occur ONLY if the RESET pin was asserted while the embedded Pentium processor was in the process of counting a particular performance monitoring event. Since the event counters continue functioning until the CESR (Control and Event Select Register) is cleared by RESET, it is possible for the event counters to increment even during RESET. Externally, the state of the event counters would also be reflected on the PM1/PM0 pins. Any assertion of the PM1/PM0 pins during RESET should be ignored until after the start of the first bus cycle.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
BP1–BP0	PM1 and PM0 are share pins with BP1 and BP0.

18.1.55 PRDY

PRDY	Probe Ready
	For use with the Intel debug port.
	Output

Signal Description

The PRDY pin is provided for use with the Intel debug port described in the Chapter 25, “Debugging.”

When Driven

This output is always driven by the embedded Pentium processor. It is not floated during bus HOLD or BOFF#.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
R/S#	R/S# is also used with the Intel debug port.

18.1.56 PWT

PWT	Page Writethrough
	Externally reflects the writethrough paging attribute bit in CR3, PDE, or PTE.
	Output

Signal Description

PWT is driven to externally reflect the cache writethrough paging attribute bit for the current cycle. PWT corresponds to bit 3 of CR3, the Page Directory Entry, or the Page Table Entry. For cycles that are not paged when paging is enabled (for example I/O cycles), PWT corresponds to bit 3 in CR3. In real mode or when paging is disabled, the embedded Pentium processor drives PWT low.

PWT can override the effect of the WB/WT# pin. If PWT is asserted for either reads or writes, the line is saved in, or remains in, the Shared (S) state.

When Driven

The PWT pin is driven valid in the same clock as ADS# and the cycle address. It remains valid from the clock in which ADS# is asserted until the earlier of the last BRDY# or the clock after NA#.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
ADS#	PWT is driven valid with ADS#.
BOFF#	PWT floats one clock after BOFF# is asserted.
HLDA	PWT floats when HLDA is asserted.
WB/WT#	PWT is used in conjunction with the WB/WT# pin to determine the MESI state of cache lines.

18.1.57 R/S#

R/S#	Run/Stop
	For use with the Intel debug port.
	Asynchronous Input

Signal Description

The R/S# pin is provided for use with the Intel debug port described in Chapter 25, “Debugging.”

When Sampled

This pin should not be driven except in conjunction with the Intel debug port.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
PRDY	PRDY is also used with the Intel debug port.

18.1.58 RESET

RESET	Reset
	Forces the processor to begin execution at a known state.
	Asynchronous Input (Normal, Uni-processor, mode) Synchronous Input (Dual processor mode)

Signal Description

The RESET input forces the embedded Pentium processor to begin execution at a known state. All the embedded Pentium processor internal caches (code and data caches, the translation lookaside buffers, branch target buffer and segment descriptor cache) will be invalidated upon the RESET. Modified lines in the data cache are not written back. When RESET is asserted, the embedded Pentium processor will immediately abort all bus activity and perform the RESET sequence. The embedded Pentium processor starts execution at FFFFFFF0H.

When RESET transitions from high to low, FLUSH# is sampled to determine if three-state test mode is to be entered, FRCMC# is sampled to determine if the embedded Pentium processor will be configured as a master or a checker (only on the embedded Pentium processor), and INIT is sampled to determine if BIST will be run.

When Sampled/Driven

RESET is sampled on every rising clock edge. RESET must remain asserted for a minimum of 1 millisecond after V_{CC} and CLK have reached their AC/DC specifications for the “cold” or “power on” reset. During power up, RESET should be asserted while V_{CC} is approaching nominal operating voltage (the simplest way to insure this is to place a pullup resistor on RESET). RESET must remain active for at least 15 clocks while V_{CC} and CLK are within their operating limits for a “warm reset.” Recognition of RESET is guaranteed in a specific clock if it is asserted synchronously and meets the setup and hold times. To guarantee recognition if RESET is asserted asynchronously, it must have been deasserted for a minimum of two clocks before being returned active to the embedded Pentium processor.

FLUSH#, FRCMC# and INIT are sampled when RESET transitions from high to low to determine if three-state test mode or checker mode will be entered, or if BIST will be run. If RESET is driven synchronously, these signals must be at their valid level and meet setup and hold times on the clock before the falling edge of RESET. If RESET is driven asynchronously, these signals must be at their valid level two clocks before and after RESET transitions from high to low.

When operating in a dual processing system, RESET is sampled synchronously to the rising CLK edge to ensure both processors recognize the falling edge in the same clock.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
APICEN	APICEN is sampled at the falling edge of RESET.
BE3#–BE0#	During reset the BE3#–BE0# pins are sampled to determine the APIC ID. Following RESET, they function as Byte Enable outputs.
BF1–BF0	BF1–BF0 are sampled at the falling edge of RESET.
CPUTYP	CPUTYP is sampled at the falling edge of RESET.
DPEN#	DPEN# is valid during RESET.
FLUSH#	If FLUSH# is sampled low when RESET transitions from high to low, three-state test mode will be entered.
FRCMC#	FRCMC# is sampled when RESET transitions from high to low to determine if the embedded Pentium processor is in Master or Checker mode.
INIT	If INIT is sampled high when RESET transitions from high to low, BIST will be performed.

18.1.59 SCYC

SCYC	Split Cycle Indication
	Indicates that a misaligned locked transfer is on the bus.
	Synchronous Input/Output

Signal Description

The Split Cycle output is activated during misaligned locked transfers. It is asserted to indicate that more than two cycles will be locked together. This signal is defined for locked cycles only. It is undefined for cycles which are not locked.

The embedded Pentium processor defines misaligned transfers as a 16-bit or 32-bit transfer which crosses a 4-byte boundary, or a 64-bit transfer which crosses an 8-byte boundary.

When operating in dual processing mode, the embedded Pentium processor uses this signal for private snooping.

When Sampled/Driven

SCYC is only driven during the length of the locked cycle that is split. SCYC is asserted with the first ADS# of a misaligned split cycle and remains valid until the earlier of the last BRDY# of the last split cycle or the clock after NA# of the last split cycle.

This signal becomes an input/output when two embedded Pentium processors are operating together in dual processing mode.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
ADS#	SCYC is driven valid in the same clock as ADS#.
BOFF#	SCYC is floated one clock after BOFF# is asserted.
HLDA	SCYC is floated when HLDA is asserted.
LOCK#	SCYC is defined for locked cycles only.

18.1.60 SMI#

SMI#	System Management Interrupt
	Latches a System Management Interrupt request.
	Asynchronous Input
	Internal Pullup Resistor

Signal Description

The System Management Interrupt input latches a System Management Interrupt request. After SMI# is recognized on an instruction boundary, the embedded Pentium processor waits for all writes to complete and EWBE# to be asserted, then asserts the SMIACT# output. The processor will then save its register state to SMRAM space and begin to execute the SMM handler. The RSM instruction restores the registers and returns to the user program.

SMI# has greater priority than debug exceptions and external interrupts. This means that if more than one of these conditions occur at an instruction boundary, only the SMI# processing occurs, not a debug exception or external interrupt. Subsequent SMI# requests are not acknowledged while the processor is in system management mode (SMM). The first SMI# interrupt request that occurs while the processor is in SMM is latched, and serviced when the processor exits SMM with the RSM instruction. Only one SMI# will be latched by the processor while it is in SMM.

When Sampled

SMI# is sampled on every rising clock edge. SMI# is a falling edge sensitive input. Recognition of SMI# is guaranteed in a specific clock if it is asserted synchronously and meets the setup and hold times. To guarantee recognition if SMI# is asserted asynchronously, it must have been deasserted for a minimum of two clocks before being returned active to the embedded Pentium processor and remain asserted for a minimum pulse width of two clocks.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
SMIACT#	When the SMI# input is recognized, the processor asserts SMIACT#.

18.1.61 SMIACT#

SMIACT#	System Management Interrupt Active
	Indicates that the processor is operating in SMM.
	Synchronous Output

Signal Description

The System Management Interrupt Active output is asserted in response to the assertion of SMI#. It indicates that the processor is operating in System Management Mode (SMM). It will remain active (low) until the processor executes the RSM instruction to leave SMM.

When the system is operating in dual processing mode, the D/P# signal alternates between asserted and deasserted based on whether the Primary or Dual processor owns the bus (MRM). The SMIACT# pins may be tied together or be used separately to insure SMRAM access by the correct processor.

Caution: If SMIACT# is used separately, note that the SMIACT# signal is only driven by the processor when it is the MRM (so this signal must be qualified with the D/P# signal).

Connecting the SMIACT# signals on the Primary and Dual processors together is strongly recommended for operation with the Dual processor and upgradability with the Pentium OverDrive® processor.

In dual processing systems, SMIACT# may not remain low (e.g., may toggle) if both processors are not in SMM mode. The SMIACT# signal is asserted by either the Primary or Dual processor based on two conditions: the processor is in SMM mode and is the bus master (MRM). If one processor is executing in normal address space, the SMIACT# signal will go inactive when that processor is MRM. The LRM processor, even if in SMM mode, will not drive the SMIACT# signal low.

When Sampled/Driven

SMIACT# is driven active in response to the assertion of SMI# after all internally pending writes are complete and the EWBE# pin is active (low). It will remain active (low) until the processor executes the RSM instruction to leave SMM. This signal is always driven. It does not float during bus HOLD or BOFF#.

When operating in dual processing mode, the SMIACT# output must be sampled with an active ADS# and qualified with the D/P# signal to determine which embedded Pentium processor (i.e., the Primary or Dual) is driving the SMM cycle.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
ADS#	SMIACT# should be sampled with an active ADS# during dual processing operation.
D/P#	When operating in dual processing mode, D/P# qualifies the SMIACT# SMM indicator.
EWBE#	SMIACT# is not asserted until EWBE# is active.
SMI#	SMIACT# is asserted when the SMI# is recognized.

18.1.62 STPCLK#

STPCLK#	Stop Clock
	Used to stop the internal processor clock and consume less power.
	Asynchronous Input

Signal Description

Assertion of STPCLK# causes the embedded Pentium processor to stop its internal clock and consume less power while still responding to interprocessor and external snoop requests. This low-power state is called the stop grant state. When the processor recognizes a STPCLK# interrupt, the processor will do the following:

1. Wait for all instructions being executed to complete.
2. Flush the instruction pipeline of any instructions waiting to be executed.
3. Wait for all pending bus cycles to complete and EWBE# to go active.
4. Drive a special bus cycle (stop grant bus cycle) to indicate that the clock is being stopped.
5. Enter low power mode.

The stop grant bus cycle consists of the following signal states: M/IO# = 0, D/C# = 0, W/R# = 1, Address Bus = 0000 0010H (A_4 = 1), BE7#–BE0# = 1111 1011, Data bus = undefined.

STPCLK# must be driven high (not floated) to exit the stop grant state. The rising edge of STPCLK# will tell the processor that it can return to program execution at the instruction following the interrupted instruction.

When Sampled/Driven

STPCLK# is treated as a level triggered interrupt to the embedded Pentium processor and is prioritized below all of the external interrupts. When the embedded Pentium processor recognizes the STPCLK# interrupt, the processor will stop execution on the instruction boundary following the STPCLK# assertion.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
A4, Cycle Control signals (M/IO#, D/C#, W/R#, BE7#–BE0#, D/P#)	The Stop Grant Special Bus Cycle is driven on these pins in response to an assertion of the STPCLK# signal. M/IO# = 0, D/C# = 0, W/R# = 1. Address Bus = 0000 0010H (A4 = 1), BE7#–BE0# = 1111 1011.
EWBE#	After STPCLK# has been recognized, all pending cycles must be completed and EWBE# must go active before the internal clock will be disabled.
External Interrupt signals (FLUSH#, INIT, INTR, NMI, R/S#, SMI#)	While in the Stop Grant state, the processor will latch transitions on the external interrupt signals. All of these interrupts are taken after the deassertion of STPCLK#. The processor requires that INTR be held active until the processor issues an interrupt acknowledge cycle in order to guarantee recognition.
HLDA	The processor will not respond to a STPCLK# request from a HLDA state because it cannot generate a Stop Grant cycle.

18.1.63 TCK

TCK	Test Clock Input
	Provides Boundary Scan clocking function.
	Input

Signal Description

This is the Testability Clock input that provides the clocking function for the embedded Pentium processor boundary scan in accordance with the boundary scan interface (IEEE Std 1149.1). It is used to clock state information and data into and out of the embedded Pentium processor during boundary scan. State select information and data are clocked into the embedded Pentium processor on the rising edge of TCK on TMS and TDI inputs respectively. Data is clocked out of the embedded Pentium processor on the falling edge of TCK on TDO.

When TCK is stopped in a low state, the boundary scan latches retain their state indefinitely. When boundary scan is not used, TCK should be tied high or left as a no-connect.

When Sampled

TCK is a clock signal and is used as a reference for sampling other boundary scan signals.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
TDI	Serial data is clocked into the processor on the rising edge of TCK.
TDO	Serial data is clocked out of the processor on the falling edge of TCK.
TMS	TAP controller state transitions occur on the rising edge of TCK.

18.1.64 TDI

TDI	Test Data Input
	Input to receive serial test data and instructions.
	Synchronous Input to TCK

Signal Description

This is the serial input for the Boundary Scan test logic. TAP instructions and data are shifted into the embedded Pentium processor on the TDI pin on the rising edge of TCK when the TAP controller is in the SHIFT-IR and SHIFT-DR states. During all other states, TDI is a “don’t care.”

An internal pull-up resistor is provided on TDI to ensure a known logic state if an open circuit occurs on the TDI path. Note that when “1” is continuously shifted into the instruction register, the BYPASS instruction is selected.

When Sampled

TDI is sampled on the rising edge of TCK during the SHIFT-IR and SHIFT-DR states. During all other states, TDI is a “don’t care.”

Relation to Other Signals

Pin Symbol	Relation to Other Signals
TCK	TDI is sampled on the rising edge of TCK.
TDO	In the SHIFT-IR and SHIFT-DR TAP controller states, TDO contains the output data of the register being shifted, and TDI provides the input.
TMS	TDI is sampled only in the SHIFT-IR and SHIFT DR states (controlled by TMS).

18.1.65 TDO

TDO	Test Data Output
	Outputs serial test data and instructions.
	Output

Signal Description

This is the serial output of the Boundary Scan test logic. TAP instructions and data are shifted out of the embedded Pentium processor on the TDO pin on the falling edge of TCK when the TAP controller is in the SHIFT-IR and SHIFT-DR states. During all other states, the TDO pin is driven to the high impedance state to allow connecting TDO of different devices in parallel.

When Driven

TDO is driven on the falling edge of TCK during the SHIFT-IR and SHIFT-DR TAP controller states. At all other times, TDO is driven to the high impedance state. TDO does not float during bus HOLD or BOFF#.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
TCK	TDO is driven on the falling edge of TCK.
TDI	In the SHIFT-IR and SHIFT-DR TAP controller states, TDI provides the input data to the register being shifted, and TDO provides the output.
TMS	TDO is driven only in the SHIFT-IR and SHIFT DR states (controlled by TMS).

18.1.66 TMS

TMS	Test Mode Select
	Controls TAP controller state transitions.
	Synchronous Input to TCK

Signal Description

This a Boundary Scan test logic control input. The value of this input signal sampled at the rising edge of TCK controls the sequence of TAP controller state changes.

To ensure deterministic behavior of the TAP controller, TMS is provided with an internal pullup resistor. If boundary scan is not used, TMS may be tied to V_{CC} or left unconnected.

When Sampled

TMS is sampled on every rising edge of TCK.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
TCK	TMS is sampled on every rising edge of TCK.
TDI	TDI is sampled only in the SHIFT-IR and SHIFT DR states (controlled by TMS).
TDO	TDO is driven only in the SHIFT-IR and SHIFT DR states (controlled by TMS).

18.1.67 TRST#

TRST#	Test Reset
	Allows the TAP controller to be asynchronously initialized.
	Asynchronous Input

Signal Description

This is a Boundary Scan test logic reset or initialization pin. When asserted, it allows the TAP controller to be asynchronously initialized. When asserted, TRST# will force the TAP controller into the Test Logic Reset State. When in this state, the test logic is disabled so that normal operation of the device can continue unhindered. During initialization, the embedded Pentium processor initializes the instruction register with the IDCODE instruction.

An alternate method of initializing the TAP controller is to Drive TMS high for at least 5 TCK cycles. In addition, the embedded Pentium processor implements a power on TAP controller reset function. When the embedded Pentium processor is put through its normal power on/RESET function, the TAP controller is automatically reset by the processor. The user does not have to assert the TRST# pin or drive TMS high after the falling edge of RESET.

When Sampled

TRST# is an asynchronous input.

Relation to Other Signals

None

18.1.68 V_{CC}

V_{CC}	Supply Voltage for the processor
	V_{CC} is used to supply power to the embedded Pentium processor.
	Power Input

Signal Description

The embedded Pentium processor requires 3.3 V V_{CC} inputs.

18.1.69 V_{CC2}

V_{CC2}	Core Supply Voltage
	V_{CC2} is used to supply the core of the embedded Pentium processor with MMX technology and the low-power embedded Pentium processor with MMX technology.
	Power Input

Signal Description

The embedded Pentium processor with MMX technology requires a 2.8 V V_{CC2} (core) voltage.

The low-power embedded Pentium processor with MMX technology core voltage V_{CC2} is 1.9 V for the PPGA package. The core voltage V_{CC2} for the HL-PBGA package is 1.8 V (166 MHz) or 2.0 V (266 MHz).

18.1.70 V_{CC3}

V_{CC3}	I/O Supply Voltage
	V_{CC3} is used to supply the I/O of the embedded Pentium processor with MMX technology and the low-power embedded Pentium processor with MMX technology.
	Power Input

Signal Description

The embedded Pentium processor with MMX technology requires a 3.3 V V_{CC3} (I/O) voltage. This enables compatibility with embedded Pentium processor system components.

The low-power embedded Pentium processor with MMX technology requires a 2.5 V V_{CC3} (I/O) voltage.

18.1.71 VCC2DET#

VCC2DET#	V_{CC2} Detect
	VCC2DET# can be used in flexible motherboard implementations to configure the voltage regulator output set-point appropriately for the V_{CC2} inputs of the embedded Pentium® processor with MMX™ technology. This pin can also be used to differentiate between the Pentium Processor with MMX technology and the low-power embedded Pentium processor with MMX technology
	Output

NOTE: This pin is an INC on the embedded Pentium processor.

Signal Description

The embedded Pentium processor with MMX technology requires 2.8 V on the V_{CC2} pins and 3.3 V on the V_{CC3} pins. By using the VCC2DET# signal the system can adjust the core voltage to the processor when a embedded Pentium processor with MMX technology is inserted into Socket 7.

VCC2DET# is driven active (low) to indicate that a embedded Pentium processor with MMX technology is installed in the system and can be used in flexible motherboard designs to configure the voltage regulator output set-point appropriately for the V_{CC2} inputs of the embedded Pentium processor with MMX technology.

This pin can be used to differentiate between the Pentium Processor with MMX technology and the low-power embedded Pentium processor with MMX technology. This is an Internal No Connect (INC) pin on the low-power embedded Pentium processor with MMX technology. This pin is not defined on the HL-PBGA package.

When Sampled/Driven

This pin is internally strapped to V_{SS}.

18.1.72 W/R#

W/R#	Write/Read
	Distinguishes a Write cycle from a Read cycle.
	Synchronous Input/Output

Signal Description

The Write/Read signal is one of the primary bus cycle definition pins. W/R# distinguishes between write (W/R# = 1) and read cycles (W/R# = 0).

When operating in dual processing mode, the embedded Pentium processor uses this signal for private snooping.

When Sampled/Driven

W/R# is driven valid in the same clock as ADS# and the cycle address. It remains valid from the clock in which ADS# is asserted until the earlier of the last BRDY# or the clock after NA#.

This signal becomes an input/output when two embedded Pentium processors are operating together in dual processing mode.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
ADS#	W/R# is driven to its valid state with ADS#.
BOFF#	W/R# floats one clock after BOFF# is asserted.
HLDA	W/R# floats when HLDA is asserted.
KEN#	KEN# determines cacheability only if W/R# indicates a read.

18.1.73 WB/WT#

WB/WT#	Writeback/Writethrough
	This pin allows a cache line to be defined as writeback or writethrough on a line by line basis.
	Synchronous Input

Signal Description

This pin allows a cache line to be defined as writeback or writethrough on a line by line basis. As a result, in conjunction with the PWT pin, it controls the MESI state in which the line is saved.

If WB/WT# is sampled high during a memory read cycle and the PWT pin is low, the line is saved in the Exclusive (E) state in the cache. If WB/WT# is sampled low during a memory read cycle, the line is saved in the Shared (S) state in the cache.

If WB/WT# is sampled high during a write to a shared line in the cache and the PWT pin is low, the line transitions to the E state. If WB/WT# is sampled low during a write to a shared line in the cache, the line remains in the S state.

If for either reads or writes the PWT pin is high, the line is saved in, or remains in, the Shared (S) state.

When Sampled

This pin is sampled with KEN# on the clock in which NA# or the first BRDY# is returned, however it must meet setup and hold times on every clock edge.

Relation to Other Signals

Pin Symbol	Relation to Other Signals
BRDY# NA#	WB/WT# is sampled with the earlier of the first BRDY# or NA# for that cycle.
PWT	If PWT is high, WB/WT# is a “don’t care.”