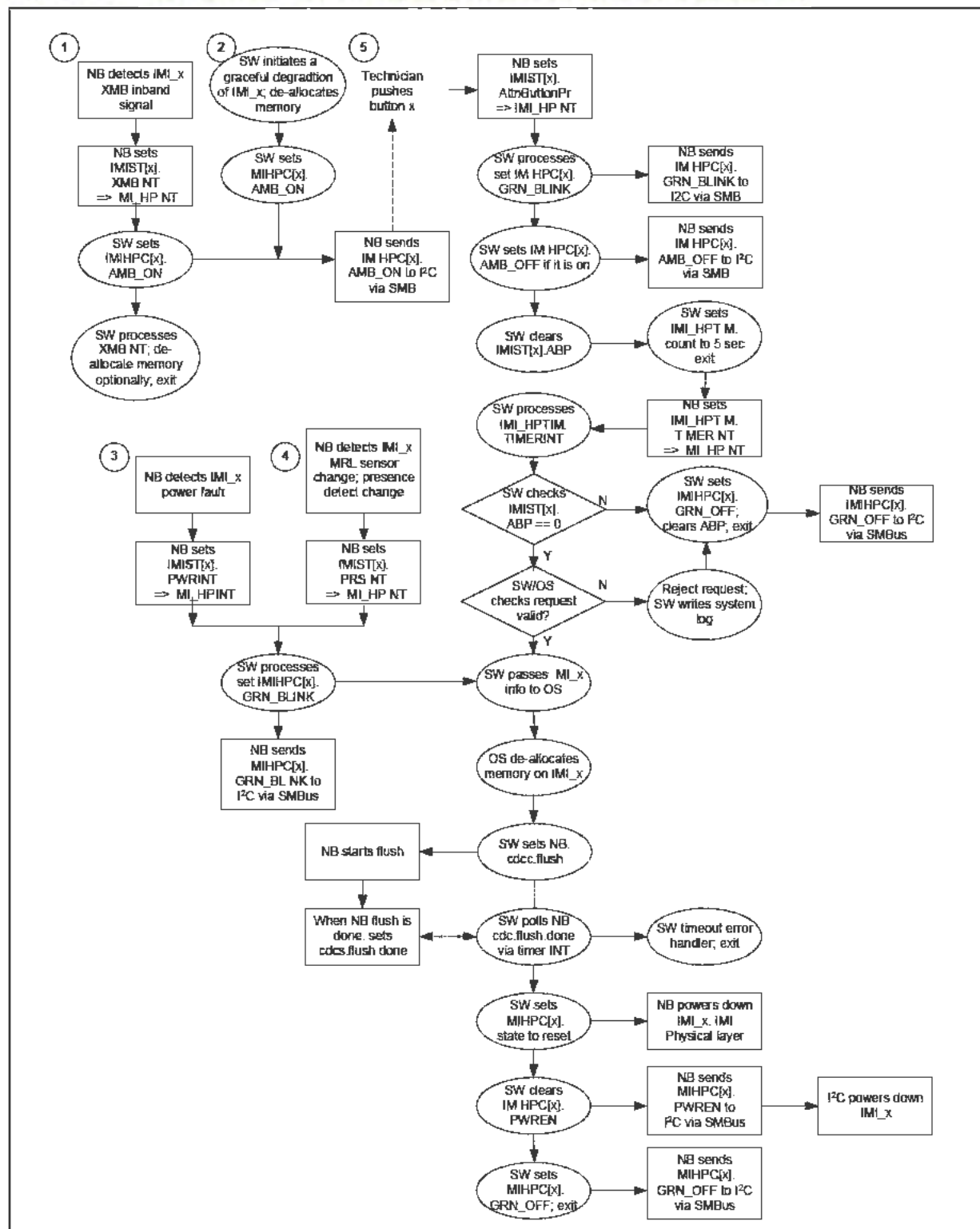


Figure 6-4. Hot Removal Flow for Capacity Reduction (Graceful Degradation)



### 6.3.12.2.2 Mirroring Mode

Figure 6-5 illustrates a hot removal flow in mirroring mode. IMI\_x is the IMI to be hot removed (where x could be A, B, C or D) while IMI\_y is the mirroring partner of IMI\_x. Both IMI\_x and IMI\_y are at redundant state (IMIST[x].state = redundant and IMIST[y].state = redundant) entering the flow. There are five entry points in this flow: 1, 2, 3, 4 and 5. Each represents an independent entry point.

At point 1, the NB detects a fatal error from IMI\_x (not fatal in mirroring mode) to fail over mode for the mirroring pair (IMI\_x, IMI\_y). The NB sets IMIST[x].FAILINT that triggers a IMI\_HPINT, software sets the amber attention LED via IMIHPC[x].ATNLED (to indicate an operational problem) and processes this interrupt and exits.

At point 2, the software initiates a hot removal by setting IMIHPC[x].state to “Disabled” state (to remove IMI\_x) and sets the indication LED. A technician can initiate the removal process via pushing the attention button at point 5.

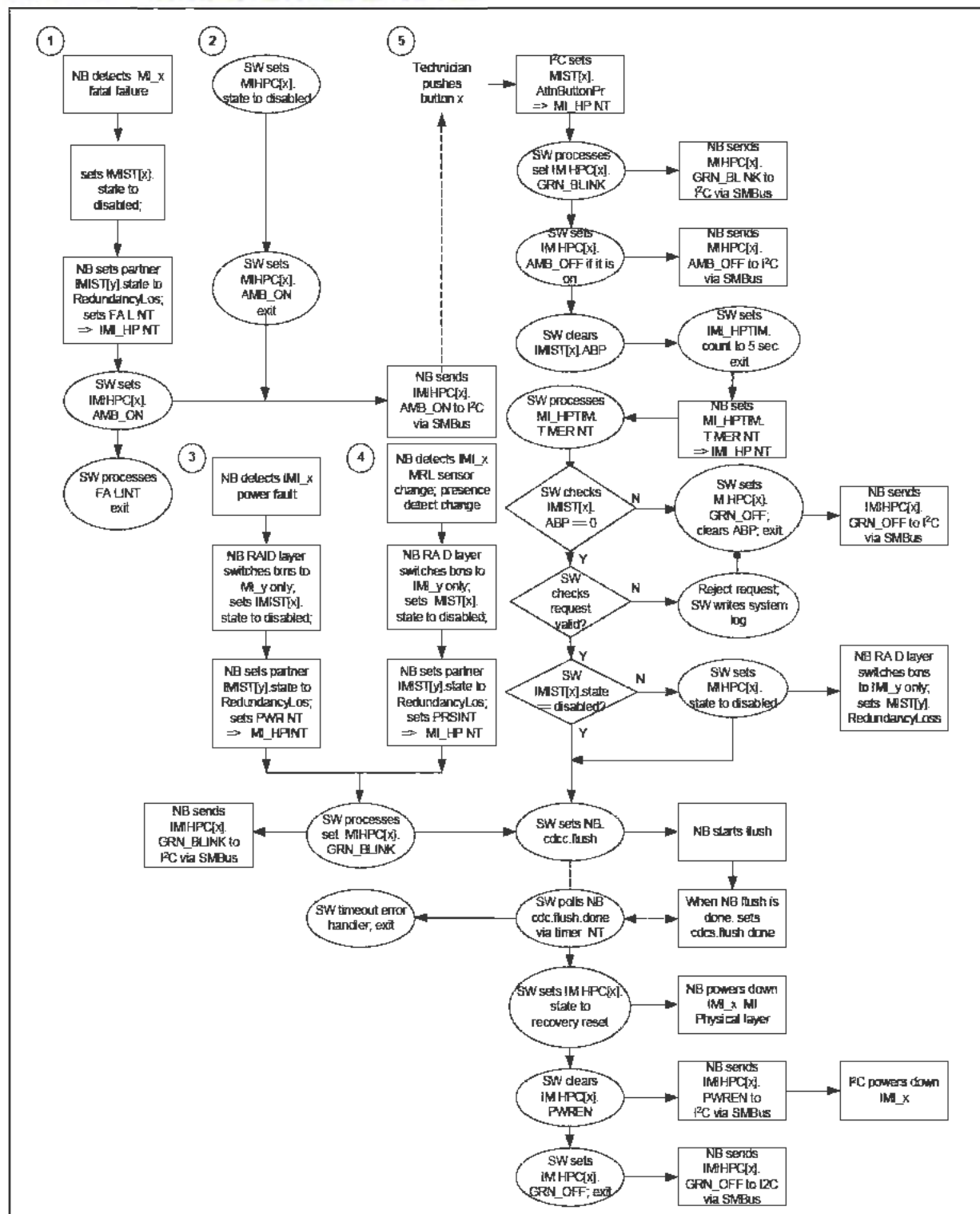
Prompted by the attention LED, the user/technician pushes the attention button following the standard usage model. The “attention button pressed” event triggers an IMI\_HPINT interrupt. The servicing software sets the green power LED to blinking state via IMIHPC[x].PWRLED and turns off the attention LED. It also clears the IMIST[x].ATNINT, and sets the IMI\_HPTIM timer to interrupt after five seconds and then exits. The NB will set IMI\_HPTIM.TIMER after the programmed five seconds period and triggers an IMI\_HPINT. If the button is pushed again during this five second period, the user indicates intention NOT to remove this IMI per standard Hot-Plug usage model. In that case, software turns off IMIHP[x].PWRLED, clears the IMIST[x].ATNINT, processes the event and exits.

If the user’s intention is confirmed after the five second period, the software and OS will validate the removal request against system policy (e.g. physical security policy).

Once the request is validated or NB detects power fault (entry point 3 with proper software handling), or the MRL sensor change (entry point 4 with proper software handling), software will flush the NB CDC via setting CDCC.FLUSH bit which starts a NB CDC flushing process. When the NB completes the flushing process, it sets CDCS.FLUSH.DONE bit. Software can use the IMI\_HPTIM to poll this completion bit periodically. The recommended waiting time is 25 ms.

After the CDC flush is completed, software should set IMIHPC[x].state to “recovery reset” state. The NB will put the IMI into reset state. Software then turns off power to the IMI/XMB via IMIHPC[x].PWRCTL. The Power LED is then turned off via IMIHPC[x].PWRLED by software. At this point, the IMI port is in reset and power to the XMB slot is turned off. Since the power LED is off, the user can remove the XMB per standard usage model.

Figure 6-5. Hot Removal for Mirroring Mode



### 6.3.12.2.3 RAID Mode

Figure 6-6 illustrates a hot removal flow in RAID mode. It is similar to the Mirroring mode flow in Figure 6-5 on page 6-294 with the main differences in NB RAID layer processing and software processing of redundant partner IMI's. Shadowed areas in the flowchart indicate the differences.

The IMI\_x is the IMI to be hot removed where x could be A, B, C or D while IMI\_w, IMI\_y, and IMI\_z are the RAID partners of IMI\_x. All IMI's are at redundant state (IMIST[w/x/y/z].state = redundant) entering the flow. There are five entry points in this flow: 1, 2, 3, 4, and 5. Each represents an independent entry point.

At point 1, NB detects a fatal error from IMI\_x (not fatal in RAID mode) and NB RAID layer switches to fail over mode in RAID mode. NB sets IMIST[x].FAILINT, which triggers a IMI\_HPINT, software sets the amber attention LED via IMIHPC[x].ATNLED to indicate an operational problem, processes this interrupt and exits. At point 2, the software initiates a hot removal by setting IMIHPC[x].state to disabled state to remove IMI\_x and set the indication LED. A technician can initiate the removal process by pushing the attention button at point 5.

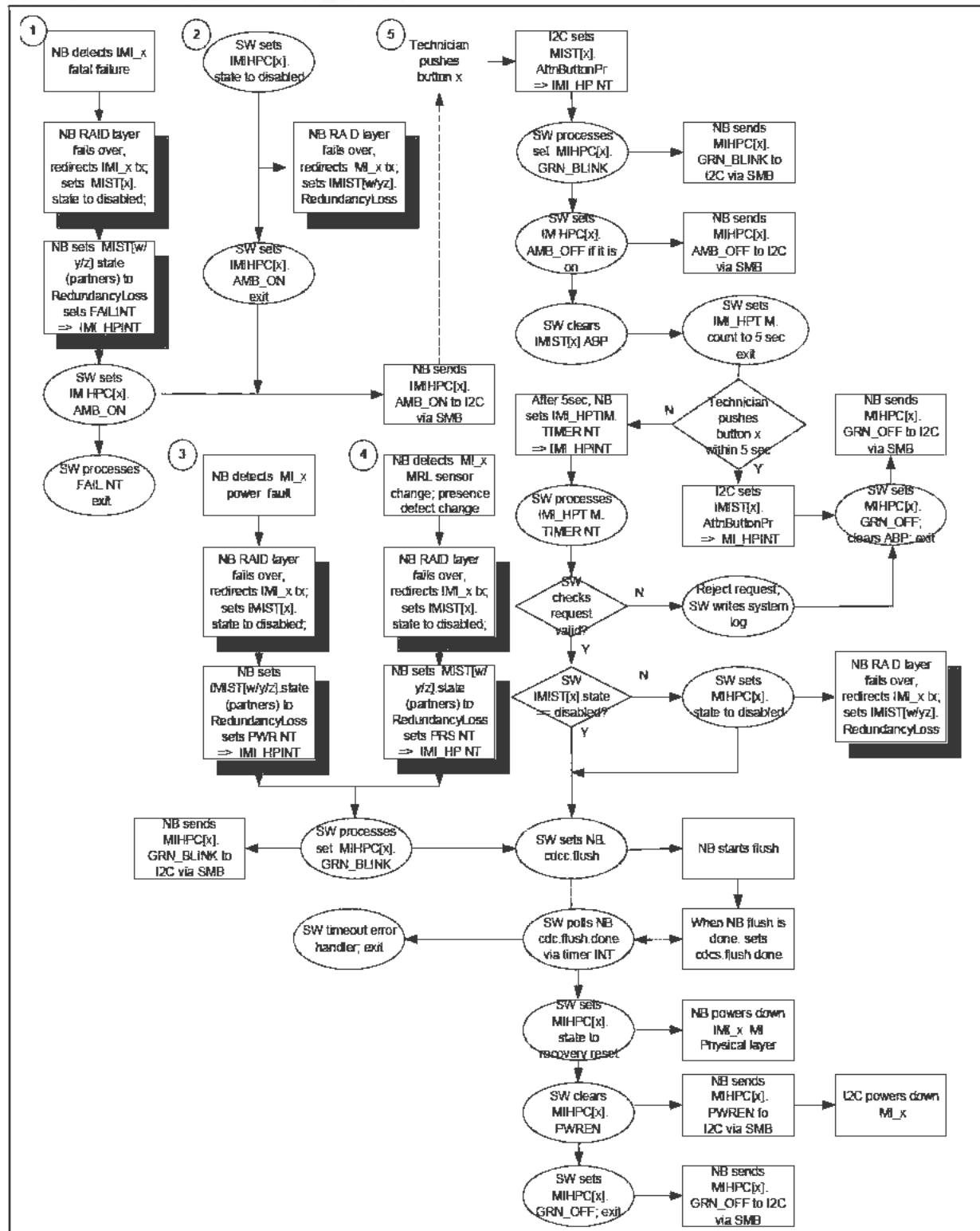
Prompted by the attention LED, the user/technician pushes the attention button, following the standard usage model. The attention button pressed event triggers IMI\_HPINT interrupt. The servicing software sets the green power LED to blinking state via IMIHPC[x].PWRLED and turns off the attention LED. It also clears the IMIST[x].ATNINT and sets IMI\_HPTIM timer to interrupt after five seconds, then exits. NB will set IMI\_HPTIM.TIMER after the programmed five-second period and triggers a IMI\_HPINT. If the button is pushed again during this five-second period, the user indicates his intention NOT to remove this IMI per standard Hot-Plug usage model. In that case, software turns off IMIHP[x].PWRLED, clears the IMIST[x].ATNINT and processes the event and exits.

If the user's intention is confirmed after the five-second period, the software and OS will validate the removal request against system policy (e.g. physical security policy).

Once the request is validated or NB detects power fault (entry point C with NB RAID layer switching and proper software handling) or MRL sensor change (entry point D with NB RAID layer switching and proper software handling), software will flush NB CDC via setting CDCC.FLUSH bit, which starts a NB CDC flushing process. When NB completes the flushing process, it sets CDCS.FLUSH.DONE bit. Software can use the IMI\_HPTIM to poll this completion bit periodically. The recommended waiting period is 25 ms; the final value will be tuned with hardware implementation.

After the CDC flush is completed, software should set IMIHPC[x].state to recovery reset state. NB will put the IMI into reset state. Software then turns off power to the IMI/XMB via IMIHPC[x].PWRCTL. Power LED is then turned off via IMIHPC[x].PWRLED by software. At this point, the IMI port is in reset and power to the XMB slot is turned off. Since the power LED is off, the user can remove the XMB per the standard usage model.

Figure 6-6. Hot Remove in RAID Mode



### 6.3.13 Memory Hot Addition

Software/User initiates hot addition for preventative maintenance or add-on upgrade purposes. NB could support at least two approaches:

- The user inserts an XMB card, closes the MRL switch and then pushes the attention button on a XMB card to indicate hot addition intention.
- User requests hot addition through system software user interface.

Hot addition flows in the chapter illustrate the attention button mechanism only.

#### 6.3.13.1 Normal Mode

Figure 6-7 illustrates a hot addition flow in normal mode. The IMI\_x is the IMI to be hot added (where x could be A,B, C or D). IMI\_x is at reset state (IMIST[x].state = reset) entering the flow.

At point A, the user/technician inserts a XMB card into slot x and closes the MRL switch. The NB sets IMIST[x].PRSINT, and triggers an IMI\_HPINT interrupt. Servicing software processes this interrupt and records the possible addition of IMI\_x. The user then pushes the attention button following the standard usage model. The “attention button pressed” event triggers the IMI\_HPINT interrupt. The servicing software sets the green power LED to “blinking state” via IMIHPC[x].PWRLD. It also clears the IMIST[x].ATNINT and sets IMI\_HPTIM timer to interrupt after five seconds and then exits. The NB will set IMI\_HPTIM.TIMER after the programmed five second period and triggers an IMI\_HPINT. If the button is pushed again during this five second period, the user indicates intention NOT to add this IMI per standard Hot-Plug usage model. In that case, software turns off the IMIHP[x].PWRLD, clears the IMIST[x].ATNINT, processes the event and exits.

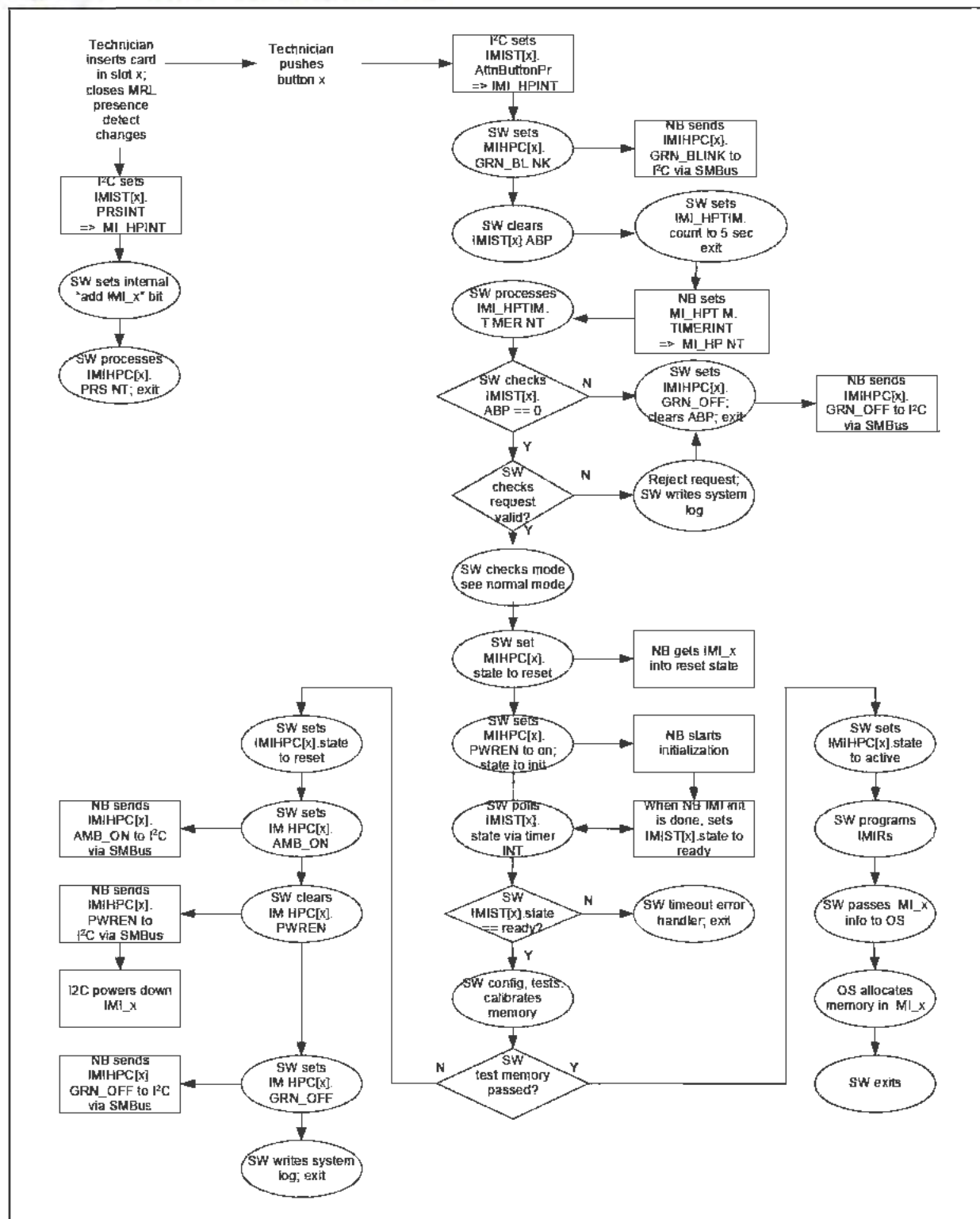
If the user’s intention is confirmed after the five second period, the software and OS will validate the addition request against system policy (e.g. physical security policy).

Once the request is validated, software sets the IMI state to reset for a clean start. Software then advances the IMI state to initialization, and the NB will start the initialization process. When the NB completes the initialization process, it sets the IMIST[x].STATE to “ready” state. Software can use the IMI\_HPTIM to poll this completion bit periodically. The recommended waiting time is 25 ms.

Once the IMI is in “ready” state, software performs memory configuration, memory testing and memory calibration. If the XMB memory passes the testing and calibration, then software sets the IMIHPC[x].STATE to active and programs IMIR’s. The OS will allocate the system memory in this IMI/XMB. This hot addition is completed successfully.

If the XMB memory does not pass the testing and calibration, software sets the IMI state back to reset and turns on the amber attention LED to report operational problem. Software then turns off the XMB slot power via IMIHPC[x].PWREN and turns off green power LED. Finally, software writes to the system log and exits.

Figure 6-7. Hot Addition in Normal Mode



### 6.3.13.2 Mirroring Mode

Figure 6-8 illustrates a hot addition flow in mirroring mode. The IMI\_x is the IMI to be hot added (where x could be 0, 1, 2, or 3), while IMI\_y is the mirroring partner of IMI\_x. IMI\_x is at recovery reset state and IMI\_y is at redundancy loss state (IMIST[x].state = recovery reset and IMIST[y].state = redundancy loss) entering the flow.

At point A, the user/technician inserts a XMB card into slot x and closes the MRL switch. The NB sets the IMIST[x].PRSINT and triggers an IMI\_HPINT interrupt. Servicing software processes this interrupt and records the possible addition of the IMI\_x. The user then pushes the attention button following the standard usage model. The “attention button” pressed event triggers IMI\_HPINT interrupt. The servicing software sets the green power LED to “blinking” state via IMIHPC[x].PWRLED, it also clears the IMIST[x].ATNINT, sets IMI\_HPTIM timer to interrupt after five seconds and then exits. The NB will set IMI\_HPTIM.TIMER after the programmed five second period and triggers an IMI\_HPINT. If the button is pushed again during this five second period, the user indicates intention NOT to add this IMI per standard Hot-Plug usage model. In that case, software turns off the IMIHP[x].PWRLED, clears the IMIST[x].ATNINT, processes the event and exits.

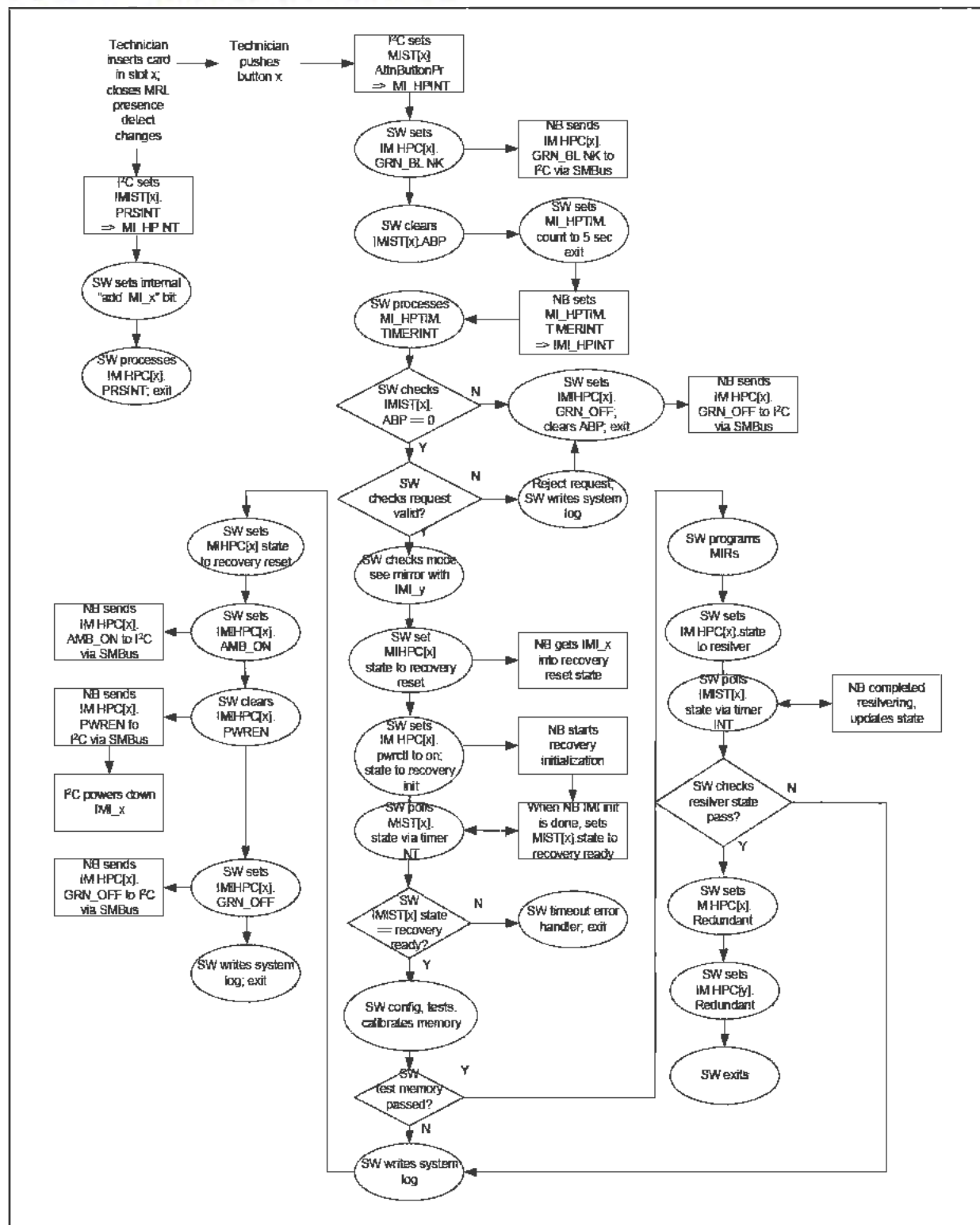
If the user’s intention is confirmed after the five second period, the software and the OS will validate the addition request against system policy (e.g. physical security policy).

Once the request is validated, software sets the IMI state to “recovery” reset for a clean start. Software then advances the IMI state to recovery initialization and the NB will start the recovery initialization process. When the NB completes the recovery initialization process, it sets the IMIST[x].STATE to recovery ready state. Software can use the IMI\_HPTIM to poll this completion bit periodically.

Once the IMI is in the “recovery ready” state, software performs memory configuration, memory testing and memory calibration. If the XMB memory passes the testing and calibration, software then programs the IMIR’s and sets the IMIHPC[x].STATE to “resilver” state. If the re-silvering process is successful, NB will set the IMI state to “recovered” state (IMIST[x].STATE = Recovered). Software can use the IMI\_HPTIM to poll this completion bit periodically. The recommended waiting time is 25 ms. Software then sets IMIHPC[x].STATE to “redundant” and IMIHPC[y].STATE to “redundant”. This hot addition is completed successfully.

If the re-silvering fails as indicated by the NB setting IMIST[x].STATE to “recovery failed” or the XMB memory does not pass the testing and calibration, software writes to system log and sets the IMI state back to “recovery reset” and turns on the amber attention LED to report operational problem. Software then turns off the XMB slot power via IMIHPC[x].PWREN and turns off green power LED. Finally, software writes to the system log and exits.

Figure 6-8. Hot Addition in Mirroring Mode



### 6.3.13.3 RAID Mode

Figure 6-9 illustrates a hot addition flow in RAID mode. It is similar to the mirroring mode flow in Figure 6-8 on page 6-300 with main differences in NB RAID layer processing and software processing of redundant IMI partners. Shadowed areas in the flowchart indicate the differences.

The IMI\_x is the IMI to be hot added where x could be A, B, C or D, while IMI\_w, IMI\_y, and IMI\_z are the RAID partners of IMI\_x. IMI\_x is at recovery reset state and IMI\_w, IMI\_y and IMI\_z are at redundancy loss state (IMIST[x].state = recovery reset, IMIST[w].state = IMIST[y].state = IMIST[z].state = redundancy loss) entering the flow.

At point 1, the user/technician inserts a XMB card into slot x and closes the MRL switch. NB sets IMIST[x].PRSINT and it triggers a IMI\_HPINT interrupt. Servicing software processes this interrupt and records the possible addition of IMI\_x. The user then pushes the attention button following the standard usage model. The attention button pressed event triggers IMI\_HPINT interrupt. The servicing software sets the green power LED to blinking state via IMIHPC[x].PWRLD. It also clears the IMIST[x].ATNINT and sets IMI\_HPTIM timer to interrupt after five seconds and then exits. NB will set IMI\_HPTIM.TIMER after the programmed five-second period and triggers a IMI\_HPINT. If the button is pushed again during this five-second period, the user indicates his intention NOT to add this IMI per standard Hot-Plug usage model. In that case, software turns off IMIHPC[x].PWRLD, clears the IMIST[x].ATNINT and processes the event and exits.

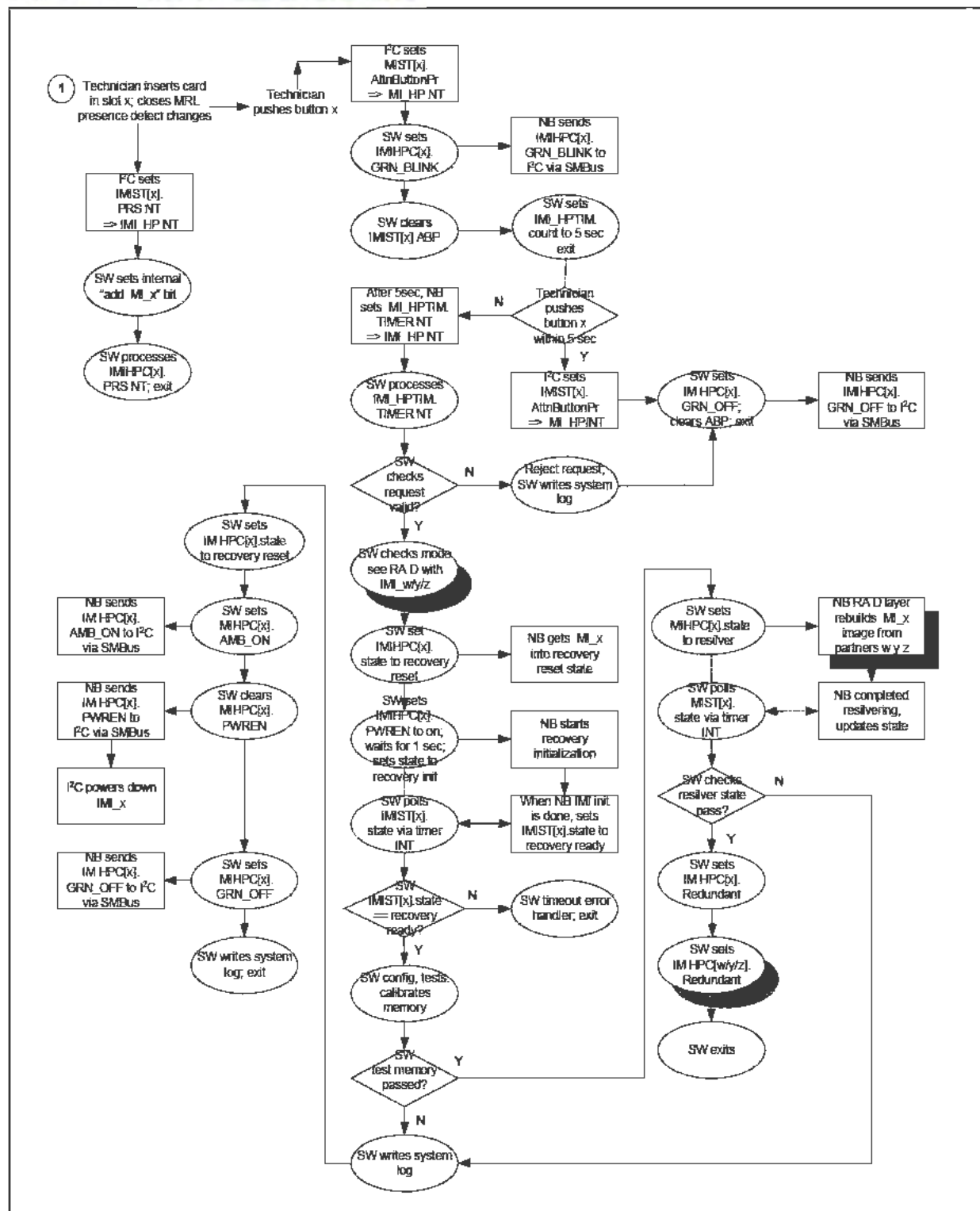
If the user's intention is confirmed after the five-second period, the software and OS will validate the addition request against system policy (e.g. physical security policy).

Once the request is validated, software sets the IMI state to recovery reset for a clean start. Software then advances the IMI state to recovery initialization and NB will start the recovery initialization process. When NB completes the recovery initialization process, it sets the IMIST[x].STATE to recovery ready state. Software can use the IMI\_HPTIM to poll this completion bit periodically. The recommended waiting period is 25 ms; the final value will be tuned with hardware implementation.

Once the IMI is in recovery ready state, software performs memory configuration, memory testing and memory calibration. If the XMB memory passes the testing and calibration, software then programs IMIR's and sets the IMIHPC[x].STATE to resilver state. NB RAID layer will re-build the correct image of IMI\_x from its RAID partners: IMI\_w, IMI\_y and IMI\_z. If the re-silvering process is successful, NB will set the IMI state to recovered (IMIST[x].STATE = recovered). Software can use the IMI\_HPTIM to poll this completion bit periodically. The recommended waiting period is 25 ms.; the final value will be tuned with hardware implementation. Software then sets all IMIHPC[x].STATE, IMIHPC[w].STATE, IMIHPC[y].STATE and IMIHPC[z].STATE to redundant. This hot addition is completed successfully.

If the re-silvering failed as indicated by NB setting IMIST[x].STATE to recovery failed, or if XMB memory does not pass the testing and calibration, software writes to the system log, sets the IMI state back to recovery reset and turns on the amber attention LED to report an operational problem. Software then turns off the XMB slot power via IMIHPC[x].PWREN and turns off the green power LED. Finally, software writes to the system log and exits.

Figure 6-9. Hot Addition in RAID Mode



## 6.4 PCI Express Interface

PCI Express is the next generation I/O interface replacing PCI-X. It offers a very high bandwidth to pin interface for general-purpose adapters interfacing a wide variety of I/O devices. The *PCI Express Base Specification*, Revision 1.0 provides the details of the PCI Express protocol.

### 6.4.1 Support

The following table describes the options and limitations supported by the NB-to-PCI Express ports.

**Table 6-19. Options and Limitations**

| Parameter        | Support                                                                                                                                           |
|------------------|---------------------------------------------------------------------------------------------------------------------------------------------------|
| Max payload      | 256B                                                                                                                                              |
| Hot-Plug         | Serial port to support pins.                                                                                                                      |
| Virtual Channels | NB only supports VC0.                                                                                                                             |
| Isochrony        | NB does not support isochrony.                                                                                                                    |
| Ordering         | NB only supports strict PCI ordering.                                                                                                             |
| No Snoop         | NB will not snoop processor caches for transactions with the No Snoop attribute.                                                                  |
| Power Management | The NB cannot be powered down, but will forward messages, generate PME_Turn_Off and collect PME_TO_ACKs. It will provide a Capabilities structure |
| Poisoning        | NB will poison data that it cannot correct.                                                                                                       |

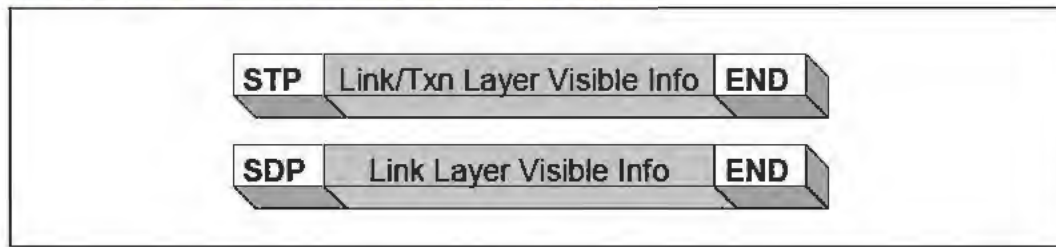
### 6.4.2 Physical Layer

The PCI Express physical layer implements high-speed serial signalling techniques including:

- Differential signalling (1.6 V peak-to-peak)
- 2.5 GHz data rate (up to 2 GB/s/direction peak bandwidth for an x8 link)
- 8b/10b encoding for embedded clocking and packet framing
- Unidirectional data path in each direction supporting full duplex operation
- Random idle packets and spread-spectrum clocking for reduced EMI
- Loop-back mode for testability
- Lane reversal
- Polarity Inversion

Figure 6-10 illustrates the scope of the physical layer on a PCI Express packet. There are two types of packets: Link layer packets and Transaction Layer Packets. The physical layer is responsible for framing these packets with STP/END symbols (Transaction Layer Packets) and SDP/END symbols (Data Link Layer packets). The grayed out segment is not decoded by the Physical layer.

Figure 6-10. PCI Express Packet Visibility by Physical Layer

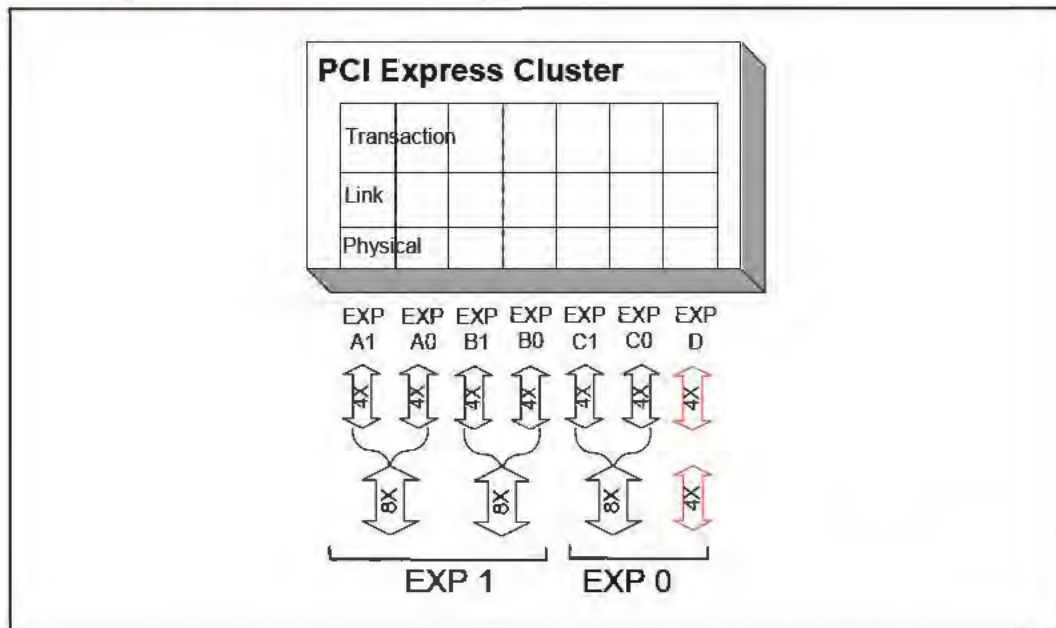


Each lane of a PCI Express port consists of 4 signals: 1 differential pair per direction. (For description of the PCI Express signals refer to Table 3-5 “Intel® E8501 Chipset North Bridge (NB) Signals (PCI Express)” on page 3-41).

#### 6.4.2.1 Supported Widths and Port Partitioning

Some of the x4 PCI Express ports can be combined to form x8 ports. Ports A0&A1, B0&B1 and C0&C1 can be combined to form x8s. Port D cannot be combined with other ports. Width configuration occurs independently per port through PCI Express training.

Figure 6-11. PCI Express Interface x8 Partitioning



The PCI Express training state machine is capable of automatically determining the port partitioning.

Table 6-20. Width Strapping Options for Port 1 (EXP\_{ A1/A0/B1/B0})

| EXP1_WIDTH[2:0] | Port A1                                         | Port A0 | Port B1 | Port B0 |
|-----------------|-------------------------------------------------|---------|---------|---------|
| 000             | x4                                              | x4      | x4      | x4      |
| 001             | x4                                              | x4      | x8      |         |
| 010             | x8                                              |         | x4      | x4      |
| 011             | x8                                              |         | x8      |         |
| 10X             | Reserved                                        |         |         |         |
| 110             | Reserved                                        |         |         |         |
| 111             | All port widths determined by link negotiation. |         |         |         |

Table 6-20 illustrates the strapping options for ports C0 and C1.

Table 6-21. Width Strapping Options for Port 0 (Exp\_{ C1/C0/D})

| EXP0_WIDTH[1:0] | Port C1                                         | Port C0 | Port D |
|-----------------|-------------------------------------------------|---------|--------|
| 00              | x4                                              | x4      | x4     |
| 01              | Reserved                                        |         |        |
| 10              | x8                                              |         |        |
| 11              | All port widths determined by link negotiation. |         |        |

The *PCI Express Base Specification*, Revision 1.0 requires only that a port be capable of negotiating and operating at the native width and x1. However, in order to support a x4 device plugged into a x8 slot, the NB attempts all supported widths narrower than the native width. For example, a port strapped at x8, will first attempt negotiation at x8. If that attempt fails, an attempt is made at x4, then x2 and then x1. When settling on a narrower width and the straps are used to override the auto-negotiation partitioning, the remaining links are unused.

### 6.4.3 Training State Machine

The *PCI Express Base Specification*, Revision 1.0 defines the training state machine required on each port. This state machine is responsible for establishing:

- Negotiated width of the port
- Negotiated frequency of the port
- Presence of a PCI Express component

The *PCI Express Base Specification*, Revision 1.0a defines a training state machine that is incompatible with the state machine defined in specification 1.0. Thus, the NB has strapping pins forcing it to behave with either of the two specifications. The EXP[1:0]\_SEPC10A pins control this behavior.

Table 6-22. Training State Machine Mode

| Pin          | Value | Function                                                      |
|--------------|-------|---------------------------------------------------------------|
| EXP0_SEPC10A | 0     | PCI Express ports D, C0, C1 will use 1.0 specification.       |
|              | 1     | PCI Express ports D, C0, C1 will use 1.0a specification.      |
| EXP1_SPEC10A | 0     | PCI Express ports B0, B1, A0, A1 will use 1.0 specification.  |
|              | 1     | PCI Express ports B0, B1, A0, A1 will use 1.0a specification. |

### 6.4.4 Link Layer

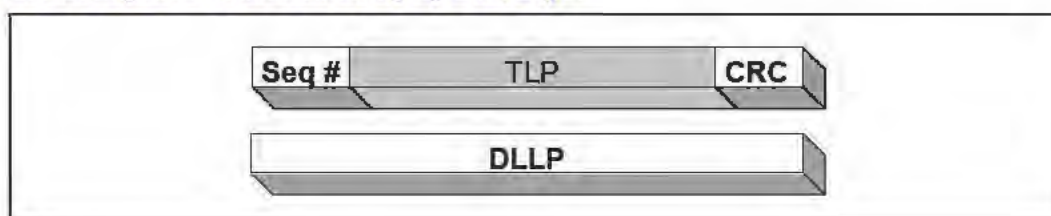
The Data Link Layer of the PCI Express protocol is primarily responsible for data integrity. This is accomplished with the following elements:

- Sequence number assignment for each packet
- ACK/NAK protocol to ensure successful transmission of every packet
- CRC protection of packets
- Time-out mechanism to detect “lost” packets
- Credit exchange

Figure 6-12 illustrates the scope of the link layer on a PCI Express packet. There are two types of packets: Data Link Layer Packets (DLLP) and Transaction Layer Packets (TLP). Data Link Layer Packets are sent between the Link Layers of each PCI Express device and do not proceed to the Transaction Layer.

For Transaction Layer Packets (TLP), the Link Layer is responsible for prepending sequence numbers and appending 32-bit CRC. The grayed out segment is not decoded by the Data Link Layer.

Figure 6-12. PCI Express Packet Visibility by Link Layer



#### 6.4.4.1 Data Link Layer Packets (DLLP)

Refer to *PCI Express Base Specification*, Revision 1.0 for an explicit definition of all the fields in a Data Link Layer packet.

DLLP's are used to ACK or NAK packets as they are sent from the transmitter to the receiver. They are sent by the receivers of the packet to indicate to the transmitter that a packet was successfully received (ACK) or not (NAK). DLLP's are also used to exchange credit information between the transmitter and receiver.

DLLP's are protected with 16b CRC. If the CRC of a DLLP is received indicates an error, the DLLP is dropped. This is safe because the PCI Express protocol supports dropping these packets and the next DLLP allows the transmitter to process successfully.

#### 6.4.4.2 ACK/NAK

The Data Link layer is responsible for ensuring that packets are successfully transmitted between PCI Express agent. PCI Express implements an ACK/NAK protocol to accomplish this. Every packet is decoded by the physical layer (8b/10b) and forwarded to the link layer. The CRC code appended to the packet is then checked. If this comparison fails, the packet is "retried".

If the comparison is successful, an ACK is issued back to the transmitter and the packet is forwarded for decoding by the receiver's Transaction layer. Typically, as each packet is successfully received by the Data Link layer, the receiver issues an ACK. However, the PCI Express protocol allows that ACK's can be combined.

#### 6.4.4.3 Link Level Retry

The *PCI Express Base Specification*, Revision 1.0 lists all the conditions where a packet gets NAK'ed. One example is on a CRC error. The Link layer in the receiver is responsible for calculating 32b CRC (using the polynomial defined in *PCI Express Base Specification*, Revision 1.0) for incoming packets and comparing the calculated CRC with the received CRC. If they do not match, then the packet is retried by NAK'ing the packet with a NAK DLLP specifying the sequence number of the last good packet. Subsequent packets are dropped until the reattempted packet is observed again.

When the transmitter receives the NAK, it is responsible for retransmitting the packet specified with the Sequence number in the DLLP + 1. Furthermore, any packets sent after the last good packet will also be resent since the receiver has dropped any packets after the corrupt packet.

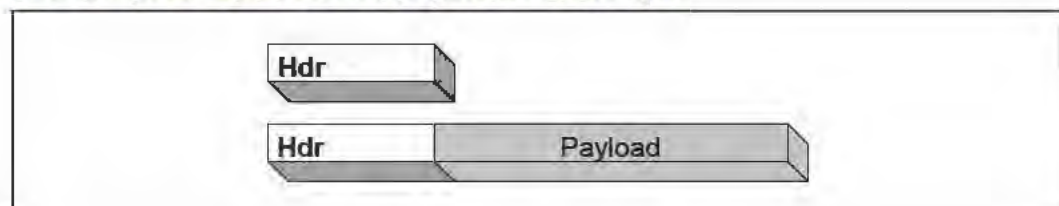
### 6.4.5 Transaction Layer

The PCI Express Transaction Layer is responsible for sending read and write operations between components. This is the PCI Express layer which actually moves software visible data between components. The transaction layer provides the mechanisms for:

- Software configuration of components
- Communication between the FSB and different I/O technologies
- Communication between the memory and different I/O technologies

Figure 6-13 illustrates the scope of the transaction layer on a PCI Express packet. Some transaction layer packets have only a header (e.g. read request). Some transaction layer packets have a header followed by data (e.g. write requests and read completions).

Figure 6-13. PCI Express Packet Visibility by Transaction Layer



### 6.4.5.1 Supported Transactions

Table 6-23 lists all the transactions supported by the NB which are expected to be received from the PCI Express interface. Similarly, Table 6-25 lists all the transactions to be expected by an attached PCI Express component. Refer to the *PCI Express Base Specification*, Revision 1.0 for the specific protocol requirements of this interface.

**Table 6-23. Incoming PCI Express Requests**

| PCI Express Transaction | Address Space or Message       | NB Response                                                                                                                                                                                        |
|-------------------------|--------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Inbound Write Requests  | Memory                         | Forward to Main Memory or PCI Express or Hub Interface port depending on address.                                                                                                                  |
|                         | I/O                            | Forward to peer PCI Express or Hub Interface port.                                                                                                                                                 |
|                         | Configuration (Type0 or Type1) | Forward to peer PCI Express or Hub Interface port if enabled.                                                                                                                                      |
| Inbound Read Requests   | Memory                         | Forward to Main Memory, or PCI Express or Hub Interface.                                                                                                                                           |
|                         | I/O                            | Forward to peer PCI Express or Hub Interface port.                                                                                                                                                 |
|                         | Configuration (Type0 or Type1) | Forward to peer PCI Express or Hub Interface port if enabled.                                                                                                                                      |
| Inbound Message         | ASSERT_INTA                    | Inband interrupt assertion/de-assertion emulating PCI interrupts.                                                                                                                                  |
|                         | DEASSERT_INTA                  |                                                                                                                                                                                                    |
|                         | ASSERT_INTB                    | Inband interrupt assertion/de-assertion emulating PCI interrupts.                                                                                                                                  |
|                         | DEASSERT_INTB                  |                                                                                                                                                                                                    |
|                         | ASSERT_INTC                    | Inband interrupt assertion/de-assertion emulating PCI interrupts.                                                                                                                                  |
|                         | DEASSERT_INTC                  |                                                                                                                                                                                                    |
|                         | ASSERT_INTD                    | Inband interrupt assertion/de-assertion emulating PCI interrupts.                                                                                                                                  |
|                         | DEASSERT_INTD                  |                                                                                                                                                                                                    |
|                         | ERR_COR                        | Propagate as an interrupt to system.                                                                                                                                                               |
|                         | ERR_UNC                        | Propagate as an interrupt to system.                                                                                                                                                               |
|                         | ERR_FATAL                      | Propagate as an interrupt to system.                                                                                                                                                               |
|                         | PM_PME                         | Propagate as a general purpose event to the system via the PME_OUT pin.                                                                                                                            |
|                         | PM_TO_ACK                      | Set when the NB receives this message from all enabled PCI Express ports.                                                                                                                          |
|                         | PM_ENTER_L1 (DLLP)             | Block subsequent TLP issue and wait for all pending TLPs to ACK. Then, send PM_REQUEST_ACK.                                                                                                        |
|                         | ATTENTION_BUTTON_PRESSED       | Terminate the message and set the EXP_SLOTSTS. ATTENTION_BUTTON_PRESSED.<br>If EXPCTRL.ATTENTION_BUTTON_PRESSED Enable and HOT-PLUG INTERRUPT ENABLE bits are set, assert EXP_HPINT.               |
|                         | ASSERT_GPE                     | Assert the EXP_HPINT output signal unless already asserted. This message should be considered a wired-or input with the other EXP_HPINT inputs.<br>NOTE: This is an Intel vendor-specific message. |
|                         | DEASSERT_GPE                   | This message should be considered a wired-or input with the other EXP_HPINT inputs.<br>NOTE: This is an Intel vendor-specific message.                                                             |
|                         | Others                         | Set IO12 error, drop transaction and return credit.                                                                                                                                                |

Table 6-24. Incoming PCI Express Completions

| PCI Express Transaction         | Address Space or Message          | NB Response                                                                         |
|---------------------------------|-----------------------------------|-------------------------------------------------------------------------------------|
| Completions for Outbound Writes | I/O or Configuration <sup>a</sup> | Forward to the FSB, PCI Express or Hub Interface from which the request originated. |
| Completions for Outbound Reads  | Memory, I/O or Configuration      | Forward to the FSB, PCI Express or Hub Interface from which the request originated. |

**NOTES:**

- a. Outbound Memory writes are posted and have no completions

Table 6-25. Outgoing PCI Express Requests

| PCI Express Transaction | Address Space or Message  | Reason for Issue                                                                                                                                                         |
|-------------------------|---------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Outbound Write Requests | Memory                    | FSB or peer memory-mapped I/O write targeting PCI Express device.                                                                                                        |
|                         | I/O                       | Processor legacy I/O write targeting PCI Express device.                                                                                                                 |
|                         | Configuration             | Processor or peer configuration write targeting PCI Express device.                                                                                                      |
| Outbound Read Requests  | Memory                    | Processor or peer memory-mapped I/O read targeting PCI Express device.                                                                                                   |
|                         | I/O                       | Processor or peer I/O read targeting PCI Express device.                                                                                                                 |
|                         | Configuration             | Processor or peer configuration read targeting PCI Express device.                                                                                                       |
| Outbound Messages       | EOI (Intel-specific)      | End-of-interrupt cycle received on FSB, NB broadcasts this message to all active PCI Express ports. Devices supporting edge triggered interrupts will ignore this cycle. |
|                         | Unlock                    | When a locked read or write transaction was previously issued to a PCI bridge, "Unlock" releases the PCI lock.                                                           |
|                         | PM_TURN_OFF               | EXP_GCTRL.PME_TURN_OFF bit was set. This message is broadcast to all enabled PCI Express ports.                                                                          |
|                         | PM_REQUEST_ACK (DLLP)     | Received PM_ENTER_L1. This message is continuously issued until link is idle.                                                                                            |
|                         | Attention_Indicator_On    | EXP_SLOTCTRL."Attention Indicator Control" has been set to On.                                                                                                           |
|                         | Attention_Indicator_Off   | EXP_SLOTCTRL."Attention Indicator Control" has been set to Off.                                                                                                          |
|                         | Attention_Indicator_Blink | EXP_SLOTCTRL."Attention Indicator Control" has been set to Blink.                                                                                                        |
|                         | Power_Indicator_On        | EXP_SLOTCTRL."Power Indicator Control" has been set to On.                                                                                                               |
|                         | Power_Indicator_Off       | EXP_SLOTCTRL."Power Indicator Control" has been set to Off.                                                                                                              |
|                         | Power_Indicator_Blink     | EXP_SLOTCTRL."Power Indicator Control" has been set to Blink.                                                                                                            |
|                         | Set_Slot_Power_Limit      | Whenever the EXP_SLOTCAP register is written, The Slot Power Limit Scale and Slot Power Limit Value are sent in the payload.                                             |

Table 6-26. Outgoing PCI Express Completions

| PCI Express Transaction      | Address Space or Message       | Reason for Issue                                                              |
|------------------------------|--------------------------------|-------------------------------------------------------------------------------|
| Completion for Inbound Read  | Memory                         | Response for an inbound read to main memory or a peer I/O device.             |
|                              | I/O                            | Response for an inbound read to a peer I/O device.                            |
|                              | Configuration (Type0 or Type1) | Response for an inbound read to a peer I/O device.                            |
| Completion for Inbound Write | Memory                         | Response for an inbound non-posted write to main memory or a peer I/O device. |
|                              | I/O                            | Response for an inbound non-posted write to a peer I/O device.                |
|                              | Configuration (Type0 or Type1) | Response for an inbound non-posted write to a peer I/O device.                |

### 6.4.6 Interrupt Handling

On PCI Express, interrupts are represented with either MSI or inbound interrupt messages (ASSERT\_INTx/DEASSERT\_INTx).

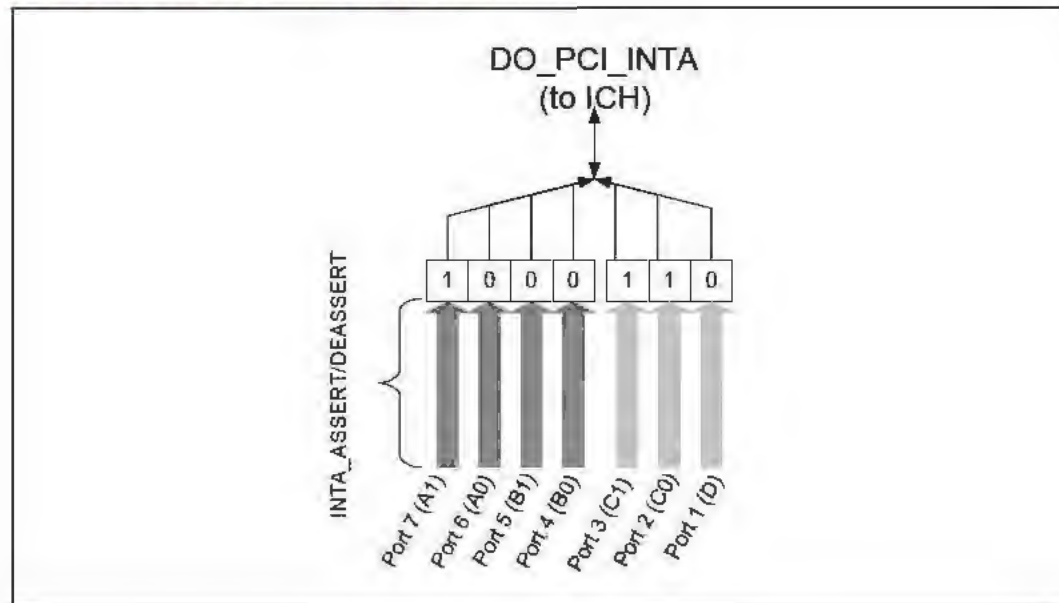
Each PCI Express port of the NB is responsible for tracking assert/deassert messages for each of the four interrupts (INTA, INTB, INTC, INTD) and representing them with four output “virtual wires” to the ICH5’s interrupt controller. These virtual wires are represented with DO\_PCI\_INTA, DO\_PCI\_INTB, DO\_PCI\_INTC, and DO\_PCI\_INTD messages to the ICH5.

Figure 6-14 illustrates how the PCI Express interrupts are routed to the ICH5. The example shown represents Interrupt A and this logic is replicated for the four interrupts. None of the bits depicted are software-visible.

When a PCI Express assert message is received for a specific interrupt, another assert message will not arrive until after a de-assert message has arrived for that interrupt first.

When MSI interrupts are used, the NB treats these writes as any other inbound write. The difference is that MSI writes are detected as a write to addresses in the range FEE0\_0000 - FEDF\_FFFF. If the write falls within this range, the NB issues the write to both FSB’s where it will be claimed by the targeted processor.

Figure 6-14. Legacy Interrupt Routing (INTA Example)



## 6.4.7 Ordering Rules

This section describes the NB ordering rules for transactions progressing through the PCI Express unit.

### 6.4.7.1 Inbound Transaction Ordering Rules

Inbound transactions originate from PCI Express and target main memory. In general, the PCI Express cluster holds inbound transactions in FIFO order. There are exceptions to this order under certain situations. For example, PCI Express requires that read completions are allowed to pass read requests. This forces any read completions to bypass any reads which might be back pressured in the queue. The PCI Express ports have no ordering relationship to each other (aside from the peer-to-peer restrictions below).

Sequential, non-posted requests are not required to be completed in the order they were requested. However, if a non-posted request requires multiple sub-completions (typically due to splitting a memory read into cacheline requests), then those sub-completions must be delivered in order.

Inbound writes cannot be posted beyond the PCI Express domain and outbound writes may only be posted after the write is acknowledged by the destination PCI Express cluster. The posting of writes relies on the fact that the system maintains a certain ordering relationship. Since the NB cannot post inbound writes beyond the PCI Express cluster, the NB must wait for snoop responses before issuing subsequent, order-dependent transactions.

### 6.4.7.2 Inbound Ordering Requirements

In general, there are no ordering requirements between transactions issued on the different PCI Express interfaces. However, the below rules apply to inbound transactions issued on the same interface.

The following rules must be ensured for inbound transactions:

- RULE 1:** Outbound non-posted read and write completions must be allowed to progress past stalled inbound non-posted requests.
- RULE 2:** Inbound posted write requests must be allowed to progress past stalled inbound non-posted requests.
- RULE 3:** Inbound posted write requests, inbound read requests, outbound non-posted read and write completions cannot pass enqueued inbound posted write requests.
- RULE 4:** The Producer - Consumer model, prevents read requests, write requests, and non-posted read or write completions from passing write requests.
- RULE 5:** Outbound non-posted read or write completions must push ahead *all* prior inbound posted write transactions from that PCI Express port.
- RULE 6:** To optimize performance, inbound, coherent, posted writes will issue ownership requests (RFO) without waiting for prior ownership requests to complete.
- RULE 7:** Inbound messages follow the same ordering rules as inbound posted writes.

Inbound messages are listed in [Table 6-23](#). Similarly to inbound posted writes, reads should push these commands ahead.

The above rules apply whether the transaction is coherent or non-coherent. Some regions of memory space are considered non-coherent (e.g. the Don't Snoop attribute is set). The PIQ will order all transactions regardless of its destination.

### 6.4.7.3 Outbound Transaction Ordering Rules

Outbound transactions through the NB are memory, I/O or configuration read/write transactions originating on a processor interface destined for a PCI Express device. Subsequent outbound transactions with different destinations have no ordering requirements between them. Multiple transactions destined for the same PCI Express port are ordered according to the ordering rules specified in *PCI Express Base Specification*, Revision 1.0.

#### 6.4.7.4 Outbound Ordering Requirements

There are no ordering requirements between outbound transactions targeting different PCI Express interfaces. For deadlock avoidance, the following rules must be ensured for outbound transactions within the same PCI Express interface:

- RULE 1:** Inbound non-posted completions must be allowed to progress past stalled outbound non-posted requests.
- RULE 2:** Outbound posted write requests must be allowed to progress past stalled outbound non-posted requests.
- RULE 3:** Outbound non-posted requests, outbound messages, outbound write requests, and inbound completions cannot pass enqueued outbound posted write requests.
- RULE 4:** The Producer - Consumer model prevents read requests, write requests, and read completions from passing write requests.
- RULE 5:** Posted outbound messages must follow the same ordering rules as outbound posted writes.
- RULE 6:** If a non-posted inbound request requires multiple snb-completions, then those sub-completions must be delivered in linearly addressing order.

#### 6.4.7.5 NB Ordering Implementation

The following table summarizes the rules enforced on transactions from a given PCI Express port by the NB.

Table 6-27. NB Ordering Implementation

| Transaction         | Will the transaction pass a stalled Posted Request? | Will the transaction pass a stalled Non-Posted Request? | Will the transaction pass a stalled completion? |
|---------------------|-----------------------------------------------------|---------------------------------------------------------|-------------------------------------------------|
| Posted requests     | never                                               | always                                                  | always                                          |
| Non-Posted Requests | never                                               | sometimes                                               | always                                          |
| Completions         | never                                               | always                                                  | never                                           |

#### 6.4.7.6 Interrupt Ordering Rules

With MSI, SAPIC and XAPIC, interrupts are simply inbound non-coherent writes to the processor. With legacy interrupts, the interrupts are ASSERT and DEASSERT messages (also following posted write ordering rules). This enforces that the interrupt will not be observed until all prior inbound writes are flushed to their destinations. The NB does not, however, guarantee that the interrupt will be observed by the processor before subsequent writes are visible to a processor.

## 6.4.8 Hot-Plug

Each PCI Express port supports the optional Hot-Plug capability described in *PCI Express Base Specification*, Revision 1.0. The PCI Express Hot-Plug model implies a Hot-Plug controller per port which is identified to software as a capability of the P2P Bridge configuration space.

PCI Express Hot-Plug support requires that the NB supports a set of Hot-Plug messages (listed in Table 6-23 and Table 6-25) to manage the states between the Hot-Plug controller and the device.

The PCI Express form factor has an impact to the level of support required of the NB. For example, some of the Hot-Plug messages are required only if the LED indicators reside on the actual card and are accessed through the endpoint device (SIOM). The NB supports all of the Hot-Plug messages so that the platform is not constrained to any particular form factor.

### Hot-Plug Controller:

PCI Express Hot-Plug requires that the NB implements a Hot-Plug controller for every Hot-Pluggable interface. The Hot-Plug controller is a capability of the bridge configuration space and the register set is accessible through the standard PCI capability mechanism defined in *PCI Express Base Specification*, Revision 1.0.

### 6.4.8.1 PCI Express Add-in Card Hot-Plug Model

The subsections below describe the signals required for PCI Express Add-in Card Hot-Plug. This model is such that the LEDs and remove button reside on the chassis. The NB provides an SMBus interface to a set of external programmable I/O devices that provide the Hot-Plug pins for any PCI Express ports that support Hot-Plug. This interface is described in Section 6.6.

### 6.4.8.2 Hot-Plug Pins

Table 6-28 describes the Hot-Plug signals used for Hot-Plug. For registers that control or provide the status of these pins please refer to Section 4.16.8, “GLOBAL\_FERR: Global First Error Register (D16, F2)” on page 196.

Table 6-28. Hot-Plug Pins

| Signal Name | Description                                                                                                                                                                                                     |
|-------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ATNLED#     | This indicator is connected to the Attention LED on the baseboard.                                                                                                                                              |
| PWRLED#     | This indicator is connected to the Power LED on the baseboard.                                                                                                                                                  |
| BUTTON#     | Input signal per slot which indicates that the user wishes to hot remove or hot add a PCI Express card.                                                                                                         |
| PRSNT#      | Input signal that indicates if a “Hot-Plugged” PCI Express device is currently plugged into the slot.                                                                                                           |
| PWRFLT#     | Input signal from the power controller to indicate that a power fault has occurred.                                                                                                                             |
| PWREN       | Output signal allowing software to enable or disable a PCI Express slot.                                                                                                                                        |
| EXP_HPINT   | Hot-Plug interrupt output signal. There is only one EXP_HPINT signal for the NB. Both parallel interfaces drive the same output signal.                                                                         |
| MRL#        | Manual (mechanical) Retention Latch input indicates that a retention latch is enabled or disabled on the board for this slot. A retention latch is used on the platform to mechanically hold the card in place. |

### 6.4.8.3 SIOM Hot-Plug Model

The SIOM Hot-Plug model is such that the LEDs and remove button reside on the endpoint module. Because of this model, the NB is responsible for issuing in-band messages to the endpoint and the endpoint is responsible for controlling the button, LEDs, etc.

The messages used for the SIOM Hot-Plug model are listed in Table 6-23 and Table 6-25 and describe the behavior of the button and LEDs.

## 6.5 Hub Interface 1.5 - (Compatibility Interface)

The HI1.5 is an 8-bit port responsible for accepting and sending packets between the NB and the ICH5 component.

### 6.5.1 Physical Layer

The compatibility interface supports parallel termination. Parallel termination is required for the routing distances required for Intel® E8501 chipset chipset-based platforms. Hub Interface uses a point-to-point bus topology using a combination of common clock framing signals and source-synchronous data transfer. The 8-bit data bus rate is 266 MHz (quad-pumped) using strobe signals. This interface is protected with one parity bit per 32 bits of data.

### 6.5.2 Transaction Layer

Table 6-29 lists all the transactions supported by the NB which are expected to be received from the ICH5. Similarly, Table 6-31 lists all the transactions to be expected by the ICH5. Refer to the *I/O Controller Hub5 (ICH5), I/O Controller Hub5-M Datasheet* for the specific protocol requirements of this interface.

**Table 6-29. Incoming Hub Interface Request Cycles**

| Hub Interface Transaction | Transaction Type              | NB Response                                                                                                                                                                                                                                                      |
|---------------------------|-------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Inbound Write Requests    | Memory                        | Forward to Main Memory or PCI Express port depending on address.                                                                                                                                                                                                 |
| Inbound Read Requests     | Memory                        | Forward to Main Memory or PCI Express.                                                                                                                                                                                                                           |
| Inbound Special Cycles    | Go C0                         | Ignore. No further action is taken.                                                                                                                                                                                                                              |
|                           | Go C3                         | Immediately respond with an "ACK C3" special cycle. No further action is taken.                                                                                                                                                                                  |
|                           | Go C3A                        | Immediately respond with an "ACK C3A" special cycle. No further action is taken.                                                                                                                                                                                 |
|                           | APIC Interrupt xmit requested | Since this chipset does not support the serial APIC bus, this special cycle is simply responded to with an "APIC Interrupt xmit granted" special cycle.                                                                                                          |
|                           | SRC                           | Special Reserved Cycle. This command is used for supporting parallel termination. This command is sent when the I/O bridge wants to do an RCOMP calculation. The NB stays off the bus for the duration indicated by the length but otherwise ignores this cycle. |

Table 6-30. Incoming Hub Interface Completions

| Hub Interface Transaction  | Transaction Type | NB Response                                           |
|----------------------------|------------------|-------------------------------------------------------|
| Outbound Write Completions | I/O              | Forward to the interface that originated the request. |
|                            | Configuration    |                                                       |
| Outbound Read Completions  | Memory           | Forward to the interface that originated the request. |
|                            | I/O              |                                                       |
|                            | Configuration    |                                                       |

Table 6-31. Outgoing Hub Interface Requests Cycles

| Hub Interface Transaction | Transaction Type                                         | Reason for Issue                                                                                                                                                               |
|---------------------------|----------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Outbound Write Requests   | Memory                                                   | FSB or peer memory-mapped I/O write targeting ICH5.                                                                                                                            |
|                           | I/O                                                      | FSB legacy I/O write targeting ICH5.                                                                                                                                           |
|                           | Configuration                                            | Configuration write targeting ICH5.                                                                                                                                            |
| Outbound Read Requests    | Memory                                                   | FSB or PCI Express memory-mapped I/O read targeting ICH5.                                                                                                                      |
|                           | I/O                                                      | FSB or PCI Express I/O read targeting ICH5.                                                                                                                                    |
|                           | Configuration                                            | Configuration read targeting ICH5.                                                                                                                                             |
| Outbound Special Cycles   | APIC Interrupt xmit granted                              | The NB will issue this command in response to an "APIC Interrupt xmit requested" command.                                                                                      |
|                           | Unlock                                                   | When a locked read or write transaction was previously issued to the ICH5, "Unlock" releases the PCI lock.                                                                     |
|                           | Shutdown                                                 | Forwarded from the "Shutdown" FSB special cycle. Treated as an outbound write with respect to ordering.                                                                        |
|                           | Halt                                                     | Forwarded from the "Halt" FSB special cycle. Treated as an outbound write with respect to ordering.                                                                            |
|                           | Stop Grant Ack                                           | Forwarded from the "StopGrant Ack" FSB special cycle. Treated as an outbound write with respect to ordering.                                                                   |
|                           | ACK C3                                                   | The NB will issue this command in response to a "Go C3" command.                                                                                                               |
|                           | ACK C3A                                                  | The NB will issue this command in response to a "Go C3A" command.                                                                                                              |
|                           | NOP                                                      | This command is used on the compatibility port and occurs during the reset sequence. Completion is required.                                                                   |
|                           | CPU_RST_DONE                                             | This command is used on the compatibility port and occurs during the reset sequence. Completion is required. The NB de-asserts CPURST# after the completion is received.       |
|                           | PHLDA On                                                 | The PHOLD protocol has been established and the ISA device can proceed without concern of back-pressure.                                                                       |
|                           | Interrupt ACK                                            | Issued by the NB when the FSB command INTA is received. Returns data from the ICH5 and must maintain same ordering rules as an outbound read.                                  |
|                           | DO_PCI_INTA<br>DO_PCI_INTB<br>DO_PCI_INTC<br>DO_PCI_INTD | Issued by the NB when a PCI Express interface receives a legacy interrupt message. These messages are issued for the assertion and deassertion of the virtual interrupt wires. |

Table 6-32. Outgoing Hub Interface Completions

| Hub Interface Transaction | Transaction Type | Reason for Issue                                            |
|---------------------------|------------------|-------------------------------------------------------------|
| Inbound Read Completions  | Memory           | Response for an inbound read to main memory or PCI Express. |

### 6.5.2.1 32/64 Bit Addressing

For inbound and outbound writes and reads, the NB supports 64-bit address format. If an outbound transaction's address is a 32-bit address, the NB will issue the transaction with a 32-bit addressing format on PCI Express. The ICH5 does not support outbound reads or writes with 64-bit addressing and the ICH5 will Master Abort these transactions.

### 6.5.3 Ordering Rules

Ordering for the HI1.5 interface is identical to that for the PCI Express interfaces. For details, refer to Section 6.4.7, "Ordering Rules" on page 311. However, HI1.5 has a stronger requirement for inbound completions of different requests. HI1.5 is strongly ordered for delayed transactions. For example, assume that a series of inbound reads are received by the NB as Rd1, Rd2, Rd3. The NB is required to complete those reads in the same order as was requested (RdCmp1, RdCmp2, RdCmp3).

### 6.5.4 Inbound Transactions

The NB will accept the transactions listed in Table 6-29. This section describes handling that is specific to the NB for transactions that target the NB or main memory.

**Note:** The inbound queue must be deep enough to ensure continuous inbound read requests are not stalled waiting for the first memory read to complete, even under heavily loaded conditions.

#### 6.5.4.1 Read Completion Policy

For inbound read requests, the NB is allowed to split completions into smaller portions if necessary. Typically, this could happen on cacheline boundaries (64B). The maximum size of a read completion is specified by HICTL.MAX\_DATA. If the interface is idle, the NB will return a completion for that read starting at the initial address up to the next cacheline boundary.

If the HI1.5 interface is busy (e.g. another outbound or inbound packet), the NB will opportunistically combine subsequent inbound read completions up to HICTL.MAX\_DATA or until the initial request length is satisfied.

### 6.5.4.2 Inbound Write Transactions

**Note:** This section applies to all write transactions targeting DRAM.

Inbound coherent write transactions actually comprise two operations:

1. Request for ownership
2. Cache line write (mark to Modified state)

The compatibility interface will enqueue each inbound write as a single atomic instruction. As the compatibility interface enqueues the write, it will also bypass all queues by sending a request-for-ownership command (RFO) directly to the FSB's requesting line ownership. The RFO commands are allowed to be issued in any order.

If the NB owns the line after all inbound ordering rules have been met, the write command proceeds and the line is modified. If the line is not owned by the NB after all inbound ordering rules have been met, the write is temporarily stalled until ownership is acquired and requests will continue to fill the inbound queue. When the request for ownership completes, the write command is forwarded where the line is marked in the Modified state.

**Note:** Because of the H1.5 ordering rules, transactions after an inbound write must wait until after the write is globally visible. The inbound queue must be deep enough to ensure that continuous inbound traffic is not stalled waiting for the above write sequence, even under heavily loaded conditions.

### 6.5.4.3 Inbound Write Combining

The NB performs write combining opportunistically in the on-chip CDC. If a write is issued to memory and the NB owns the line, the data buffer will overwrite the buffer with the new data (full or partial data).

### 6.5.4.4 PHOLD Support

The NB supports the PHOLD protocol. This protocol is used for legacy ISA devices which did not allow for the possibility of being both a master and a slave device simultaneously. Example devices that use the PHOLD protocol are legacy floppy drives and parallel port DMA.

## 6.5.5 Outbound Transactions

The NB will generate the outbound transactions listed in [Table 6-31](#). This section describes handling that is specific to the NB for transactions that target the ICH5.

### 6.5.5.1 Outbound Non-posted Transactions

Non-posted transactions that the NB propagates includes Interrupt Acknowledge, memory reads, I/O reads and writes, and configuration reads and writes. When a non-posted transaction is issued from the NB, the ICH5 device will respond with a completion.

The ICH5 supports only one outstanding non-posted transaction comprising transactions issued by the processors or a peer PCI Express device.

### 6.5.5.2 Stalled Non-posted Requests

Non-posted requests are non-blocking transactions. In other words, while a non-posted request is pending, subsequent transactions are required to bypass the transactions which are waiting for a completion.

### 6.5.5.3 Outbound Posted Transactions

Once a posted request (e.g. memory write) is issued from H11.5, the request is considered to be complete and the transaction is removed from the outbound queue. For posted requests, the acknowledge has already been sent to the initiating interface (FSB or alternate PCI Express inbound queue) when the write was enqueued in the outbound PCI Express unit, so proper ordering is guaranteed.

### 6.5.5.4 Outbound Write Combining

The NB does not combine sequential outbound writes. This optimization is left to the processor write combining buffers.

### 6.5.5.5 Lock Support

For legacy PCI functionality, the NB supports bus locks through an explicit sequence of events. The NB can receive a locked transaction sequence (Read-Write or Read-Read-Write-Write) on a FSB directed to the ICH5. The compatibility interface cluster must support the following capabilities:

- Retry all transactions on the Hub Interface
- Generate a locked read request to the Hub Interface target
- Unlock a locked Hub Interface port

## 6.5.6 Peer-to-Peer Support

Peer-to-peer support is defined as transactions which initiate on one I/O interface and target another without going through main memory. This support applies to both PCI Express and HI. The NB supports peer-to-peer transactions for memory and I/O transactions. The compatibility interface can be the destination of a peer-to-peer write or read. The compatibility interface can be the source of a posted peer-to-peer write. Non-posted requests may pre-fetch into MMIO (with potential side effects). Peer-to-peer transactions are not observed on any interface except the target and destination (e.g. no FSB snoops).

Peer-to-peer traffic between the PCI Express ports and the ICH5 must sustain the full write bandwidth. For reads, full bandwidth is required for large request sizes. Inbound coherent transactions and peer-to-peer transactions must maintain ordering rules between each other. Peer-to-peer transactions follow inbound ordering rules until it reaches the head of the inbound queue. Once the transaction reaches the head of the inbound queue, the NB routes the transaction to the next available slot in the outbound queue where PCI ordering is maintained until the end of the transaction. The NB does not support peer-to-peer where the source and destination is the same PCI Express Interface.

## 6.6 RAS

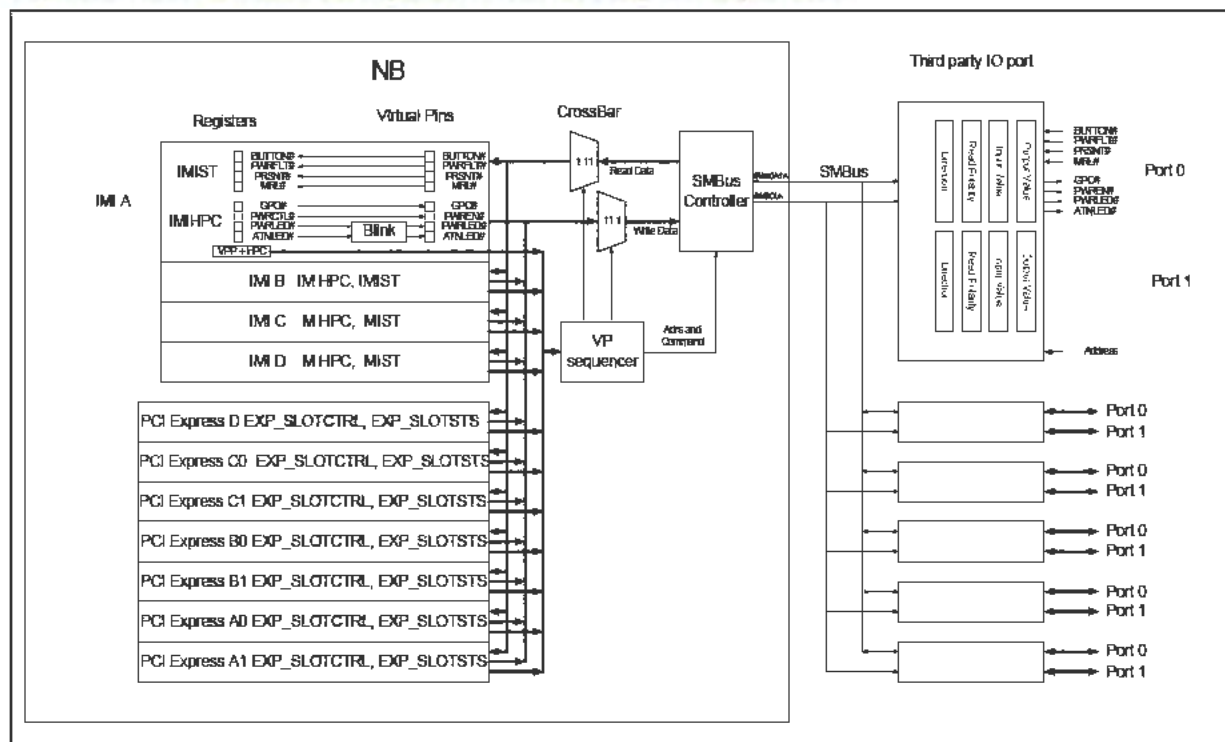
### 6.6.1 Virtual Pins for Hot-Plug

The NB supports 8 Hot-Plug pins on 11 interfaces. To allow platforms to incur cost proportional to the number of interfaces that may be Hot-Plugged, the NB masters a 100 KHz Hot-Plug SMBus interface (pins GP\_SMBCLK, GP\_SMBDATA) that connects to a variable number of serial to parallel I/O ports<sup>1</sup>. Figure 6-15 shows the NB connected to the maximum number of I/O expanders and a conceptual diagram that shows how values in Hot-Plug registers are translated to pins on the I/O expanders.

The NB only supports SMBus devices with registers mapped as per Table 6-33. These components have 16 I/O's, divided into two 8-bit ports that can be configured as inputs or outputs.

The NB has a crossbar which associates each PCI Express and IMI port with one of these 8-bit ports. The mapping is defined by a Virtual Pin Port register field (IMI\_HPC.VPP for IMI's and EXP\_CTRL.VPP for PCI Express) for each PCI Express or IMI. The VPP register holds the SMBus address and Port (0 or 1) of the I/O Port associated with the PCI Express or IMI. A[2:0] pins on each I/O Port connected to the NB must be strapped uniquely.

Figure 6-15. Maximum Configuration for Hot-Plug Pin Expanders



1. Contact your vendor of choice for the I/O ports.

**Table 6-33. I/O Port Registers Supported by Intel® E8501 Chipset North Bridge (NB)**

| Register | Name                      | Intel® E8501 chipset north bridge (NB) Usage |
|----------|---------------------------|----------------------------------------------|
| 0        | Input Port 0              | Continuously Reads Input Values              |
| 1        | Input Port 1              |                                              |
| 2        | Output Port 0             | Continuously Writes Output Values            |
| 3        | Output Port 1             |                                              |
| 4        | Polarity Inversion Port 0 | Never written by NB                          |
| 5        | Polarity Inversion Port 1 |                                              |
| 6        | Configuration Port 0      | Direction set as per Table 6-34              |
| 7        | Configuration Port 1      |                                              |

### 6.6.1.1 Operation

When the NB comes out of reset, the I/O ports are inactive. The NB is not aware of how many I/O Ports are connected to it, what their addresses are, nor what PCI Express or IMI ports are Hot-Pluggable. The NB does not master any commands on the SMBus until a Hot-Plug Capable bit is set.

An IMI port is considered Hot-Plug Capable when IMIHPC.HPC bit is set. A PCI Express port is considered Hot-Plug Capable when EXP\_SLOTCAP.HPC bit and EXP\_CTRL.EN\_VPP are set. When a port is considered Hot-Plug Capable, the NB initializes the associated VPP with Direction and Logic Level configuration as shown in Table 6-34. The polarity is left at the default values and never written. When writing to VPP registers outside the NB, addresses are used as described in Table 6-35 for writes to the VPP registers connected to the I<sup>2</sup>C port.

When the NB is not responding to a change in whether a port is Hot-Plug Capable, it performs input register reads and output register writes to all the Hot-Plug Capable VPP's. This sequence repeats indefinitely until there is a change in whether a port is Hot-Plug Capable. To minimize the completion time of this sequence and its complexity, both ports are always read and written. For the maximum number of 6 I/O ports, and assuming no clock stretching, this sequence can take up to 51 ms. If new Hot-Plug capability bits are not being set, this is the maximum timing uncertainty in sampling or driving these signals.

**Table 6-34. Hot-Plug Signals on a Virtual Pin Port (VPP) (Sheet 1 of 2)**

| Bit | Present for |     | Direction | Logic Level | Polarity | Signal  |
|-----|-------------|-----|-----------|-------------|----------|---------|
|     | PCI Express | IMI |           |             |          |         |
| 0   | Yes         | Yes | Output    | 1           | true     | ATNLED# |
| 1   | Yes         | Yes | Output    | 1           | true     | PWRLED# |
| 2   | Yes         | Yes | Input     | 1           | true     | BUTTON# |
| 3   | Yes         | Yes | Input     | 1           | true     | PWRFLT# |
| 4   | Yes         | Yes | Input     | 1           | true     | PRSENT# |

Table 6-34. Hot-Plug Signals on a Virtual Pin Port (VPP) (Sheet 2 of 2)

| Bit | Present for |     | Direction | Logic Level | Polarity | Signal  |
|-----|-------------|-----|-----------|-------------|----------|---------|
|     | PCI Express | IMI |           |             |          |         |
| 5   | Yes         | Yes | Output    | 1           | true     | PWREN#  |
| 6   | Yes         | Yes | Input     | 1           | true     | MRL#    |
| 7   | No          | Yes | Input     | 1           | true     | PWRGOOD |

Table 6-35 describes the sequence generated for a write to an I/O port. Both 8-bit ports are always written. If a VPP is valid for the 8-bit port, the Output values are updated as per the IMIHPC or the EXP\_SLOTCTRL registers for the associated IMI or the PCI Express Hot-Plug registers. The Input values affect the IMIST and PCI Express Hot-Plug registers. If no VPP for the 8-bit port is valid, 0s are written.

Table 6-35. VPP Write Command

| Bits | NB Drives       | I/O Port Drives | Comment                                                                               |
|------|-----------------|-----------------|---------------------------------------------------------------------------------------|
| 1    | Start           |                 | SDL (Serial Data Line) falling followed by SCL (Serial Clock Line) falling            |
| 7    | Address[6:0]    |                 | [6:3] = 0100<br>[2:0] = VPP[3:1]                                                      |
| 1    | 0               |                 | indicates write                                                                       |
| 1    |                 | ACK             | If NACK is received, NB completes with stop and sets Error T9                         |
| 8    | Register Number |                 | Register Address see Table 6-33<br>[2:1] = 01 for Output, 11 for Direction<br>[0] = 0 |
| 1    |                 | ACK             | If NACK is received, NB completes with stop and sets Error T9                         |
| 8    | Data            |                 | One bit for each bit of port 0 as per Table 6-34                                      |
| 1    |                 | ACK             | If NACK is received, NB completes with stop and sets Error T9                         |
| 8    | Data            |                 | One bit for each bit of port 1 as per Table 6-34                                      |
| 1    |                 | ACK             | If NACK is received, NB completes with stop and sets Error T9                         |
| 1    | Stop            |                 |                                                                                       |

The NB issues Read Commands to update NB virtual inputs from the I/O Port. The I/O port requires that a command be sent to sample the inputs, then another command is issued to return the data. The NB always reads inputs from both 8-bit ports. If the VPP is valid, the NB updates the associated MIHPS (for IMI) or the EXP\_SLOTSTS (for PCI Express) register according to the values of BUTTON#, PWRFLT#, and PRSNT# read from the value register in the I/O Port. Results from invalid VPP's are discarded. Table 6-36 defines the read command format.

If a VPP register is set to a non-existent address, it will not acknowledge the command and the NB will set the T9 error. The NB will continue to address the existing devices.

**Table 6-36. VPP Read Command**

| Bits | NB Drives           | I/O Port Drives | Comment                                                                                                                             |
|------|---------------------|-----------------|-------------------------------------------------------------------------------------------------------------------------------------|
| 1    | Start               |                 | SDL falling followed by SCL falling                                                                                                 |
| 7    | Address[6:0]        |                 | [6:3] = 0100<br>[2:0] = VPP[3:1]                                                                                                    |
| 1    | 0                   |                 | Indicates Write                                                                                                                     |
| 1    |                     | ACK             | If NACK is received, NB completes with stop and sets Error T9                                                                       |
| 8    | RegisterNumber[2:0] |                 | Register Address<br>[2:1] = 00<br>[0] = 0                                                                                           |
| 1    |                     | ACK             | If NACK is received, NB completes with stop and sets Error T9                                                                       |
| 1    | Start               |                 | SDL falling followed by SCL falling                                                                                                 |
| 7    | Address[6:0]        |                 | [6:3] = 0100<br>[2:0] = VPP[3:1]                                                                                                    |
| 1    | 1                   |                 | Indicates Read                                                                                                                      |
| 1    |                     | ACK             | If NACK is received, NB completes with stop and sets Error T9                                                                       |
| 8    |                     | Data            | One bit for each bit of port 1 as per Table 6-34. The NB always reads from both ports. Results for invalid VPP's are discarded      |
| 1    | ACK                 |                 |                                                                                                                                     |
| 8    |                     | Data            | One bit for each bit of port 1 as shown in Table 6-34. The NB always reads from both ports. Results for invalid VPP's are discarded |
| 1    | NACK                |                 |                                                                                                                                     |
| 1    | Stop                |                 |                                                                                                                                     |

### 6.6.1.2 Limitations

This interface is dedicated to Hot-Plug and does not support aspects of SMBus that are not required for this application.

**Table 6-37. Unsupported Aspects of SMBus**

| Unsupported Aspect                                  | Comment                                                                                                                                   |
|-----------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------|
| Command sequences not defined in this specification | Such as Block reads, Process Calls, Host Notification, etc.                                                                               |
| Arbitration                                         | NB is the only master.                                                                                                                    |
| PEC                                                 | The targeted I/O Ports do not support it.                                                                                                 |
| Bus Enumeration                                     | BIOS is required to configure VPP's with this information. Variation in the number of devices is not supported for a particular platform. |
| Clock Stretching                                    | NB will drive 100 KHz and I/O Ports can support higher frequencies, so they will never stretch the clock.                                 |
| NACK Retry                                          | SMBus forbids devices from NACK'ing if they are busy                                                                                      |

## 6.7 Error Reporting

The NB recognizes a variety of errors (or exception conditions) and is the collection point for many of these errors. Some errors are internally detected and others are signaled to it from the external interfaces (PCI Express and IMI). All detected errors eventually cause the NB to do one of the following:

- Assert MCERR on both FSB's
- Assert Error pins (ERR[2:0])

The error reporting for the NB is comprised of the status registers (FERR and NERR), the error log registers (recoverable and non-recoverable), and the error pins (ERR[2:0]). The individual errors can be programmed to be masked or to be signaled by ERR[2:0] or by MCERR.

### 6.7.1 Error Types

The errors can be classified into three types: (1) Correctable errors, (2) Uncorrectable Errors, (3) Hardware Fatal errors. The discussion here assumes that for all these error types, hardware provides the capability to detect the error. Errors that cannot be detected by hardware cause silent data corruption, which is an undesirable event.

- **Correctable Errors** are corrected by the hardware and software is completely oblivious to this event. Examples of such errors include single bit ECC errors and successful link level retry. Such events may be logged and reported by the system for a post-mortem by the firmware or system management software.
- **Uncorrectable Errors** are not corrected by the hardware or software. OS or other software layers may be able to recover from such errors, but it may not always be possible. Such errors are typically contained in nature, the system state is intact, and the process and the system is restartable. Errors in this category include multi-bit data error, transaction time out, and so on.
- **Fatal Errors** may compromise system integrity and continued operation may not be possible. Errors in this category include protocol errors.

### 6.7.2 Error Mask and Signaling Mapping Register

Each error is defined to be a recoverable or non-recoverable error. This determines which FERRST, NERRST, and log register get used in reporting the error. In addition, the error also has an Error Mask and Signaling registers. The purpose of the error mask is to mask / allow the observance of the error condition. Each local block has an error mask for the errors detected in that particular block.

Each unmasked error can generally be programmatically routed to the following error signals:

- MCERR
- ERR[2], ERR[1], or ERR[0]
- ICHRST

**Note:** It is up to software to enable only one of the error signaling possibilities (MCERR, ERR[2], ERR[1] & ERR[0], ICHRST signals). Software should not map PCI Express errors to the MCERR signal.

### 6.7.3 Error Status and Log Registers

Error status registers are provided: FERR (first error register), and NERR (Next error register). First fatal and/or first non-fatal errors are flagged in the FERR register; subsequent errors are indicated in the NERR register.

The contents of FERR and NERR are “sticky” across a reset (while PWRGOOD remains asserted). This provides the ability for firmware to perform diagnostics across reboots. Note that only the contents of FERR affects the update of the any error log registers.

The FERR/NERR structure is hierarchical. The Global FERR/NERR registers for the Intel® E8501 chipset are located in the NB the list of FERR/NERR registers are contained in Table 6-38. “-FERR/NERR and Log Registers”.

Once a first error for a type (non-fatal or recoverable) of error has been flagged (and logged), the log registers for that error type remain fixed until either 1) any errors in the local FERR register for which the log is valid are cleared or 2) a power-on reset.

**Table 6-38. -FERR/NERR and Log Registers**

| FERR/NERR Register | Location                        | Log Register                                    |
|--------------------|---------------------------------|-------------------------------------------------|
| GLOBAL_FERR        | NB                              | N/A                                             |
| GLOBAL_NERR        |                                 | N/A                                             |
| IMI_FERR           | NB Independent Memory Interface | NRECIMI<br>RECIMI                               |
| IMI_NERR           |                                 | N/A                                             |
| FSB{A/B}_FERR      | NB FSB{A/B} Interface Blocks    | NRECFSB{A/B}<br>RECFSB{A/B}                     |
| FSB{A/B}_NERR      |                                 | N/A                                             |
| INT_FERR           | NB                              | NRECTNB<br>RECTNB<br>RECINT_LOG[4:0]<br>INTLOGC |
| INT_NERR           |                                 | N/A                                             |
| HIFERR             | NB Hub Interface Block          | NRECHI<br>RECHI                                 |
| HINERR             |                                 | N/A                                             |
| EXP_FERR           | NB PCI Express Interface Blocks | EXP_DEVSTS<br>HDRLOG[3:0]                       |
| EXP_NERR           |                                 | EXP_UNITERR                                     |

## 6.7.4 Error Signaling

Associated with each pair of FERR/NERR registers are three open drain error pins, one for each error type: fatal, uncorrectable and correctable (ERR#[2:0]). If not masked (EMASK register), these pins will reflect the error status for each type of error in the two error status registers. The NB can signal errors via the following pins:

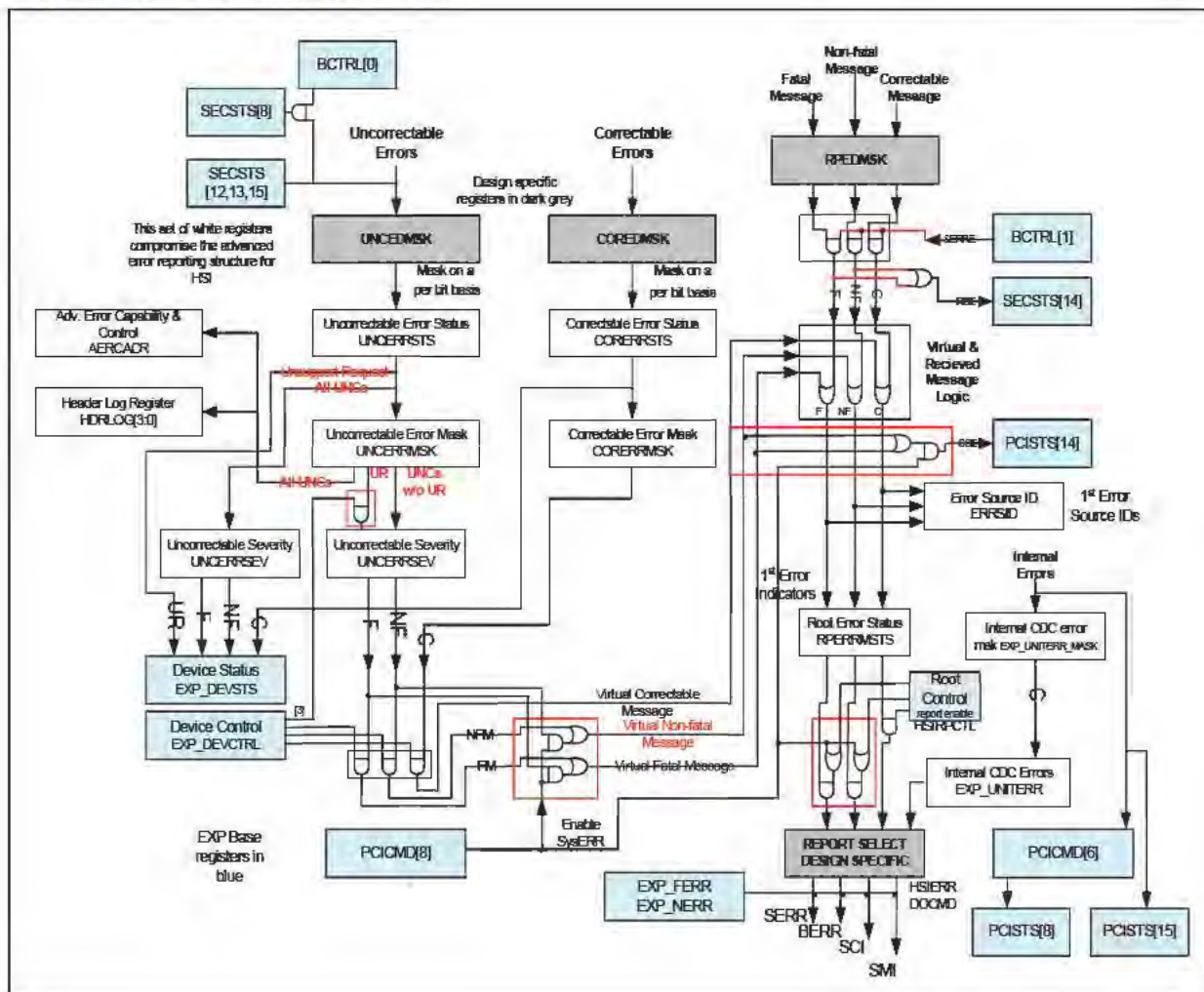
- Err[0] - Correctable Errors (Non-Fatal or Recoverable Error)
- Err[1] - Uncorrectable Errors (Non-Fatal or Recoverable Error)
- Err[2] - Fatal Error
- MCERR- Adhering to the FSB rules for driving MCERR.

The error pins are I/O signals that are synchronous to the 166 MHz bus clock.

## 6.7.5 Error Integration into Intel® E8501 Chipset North Bridge (NB) Error Model

Refer to [Figure 6-5](#) for how the PCI Express errors map into the NB error registers and for how the flow of error detection and logging is handled inside the NB.

Figure 6-16. Error Integration Model



## 6.7.6 Error Detection

In the NB chipset-based system, the task of error detection, isolation and recovery of correctable errors is performed by hardware and is distributed among the NB components. Uncorrectable errors are isolated, logged and reported to system firmware / OS for error handling. All errors, including correctable errors, can be reported to the firmware / OS for logging.

All header and data transmission over the PCI Express interface is protected with a CRC. The CRC code is capable of detecting all single bit and many multi-bit errors, as well as all error patterns with burst length of "n" (where "n" is the CRC size). This covers the error pattern due to a single wire failure on the interface. Independent Memory Interface inbound uses CRC on the link layer and ECC on inbound header and data transfers (and CRC on outbound data). IMI outbound uses CRC on all packets.

Table 6-39 provides the list of errors detected by the Intel® E8501 chipset north bridge (NB). Errors are listed by the unit / interface. Some unit / interface may provide additional error logging registers. The last column of the table, defines which component detects the error and the bit position of the register set. For FSB, IMI and NB the errors have the same bit position in the First Error, Next Error, Emask, and Error Select Registers. The I/O Errors for the Hub Interface also apply to this convention but the PCI Express errors **do not**. The PCI Express errors adhere to the PCI Express definition and location of errors.

Please note that only NB errors are listed here. For the list of XMB errors, readers should refer to the *Intel® E8501 Chipset eXternal Memory Bridge (XMB) Datasheet*.

**Table 6-39. Errors Detected by the NB (Sheet 1 of 6)**

| Error#            | Error Name                                         | Error type | Log Register(s)       | Cause / Action                                                                                                                                                                                                                   | Definition                                                                                                                                                  | Comp. Reg. Bit Position |
|-------------------|----------------------------------------------------|------------|-----------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------|
| <b>FSB Errors</b> |                                                    |            |                       |                                                                                                                                                                                                                                  |                                                                                                                                                             |                         |
| F0                | Multi-bit ECC Error (Do not Include Poisoned Data) | Non-Fatal  | RECFSB{A/B}           | Received a multi-bit ECC error. Poison Data and forward to the appropriate interface.                                                                                                                                            | The NB monitors the data/dep signals on the FSB. Set when the NB detects an uncorrectable error during the data transfer in which an F0 error is signalled. | NB-0                    |
| F1                | Correctable ECC Error                              | Non-Fatal  | RECFSB{A/B}           | Correct by Hardware, Cause CMCI.                                                                                                                                                                                                 | The NB monitors the data/dep signals on the FSB. Set when the NB detects an correctable error during the data transfer an F1 error is signalled.            | NB-1                    |
| F2                | Detected BINIT from a processor                    | Non-Fatal  | N/A                   | Do not propagate to other bus, reset arb. unit, and programmatically reset platform.                                                                                                                                             | The NB detected that a processor issued a B-INIT.                                                                                                           | NB-2                    |
| F3                | Detected MCERR from a processor                    | Non-Fatal  | N/A                   | If (receive an MCERR) forward the MCERR to the other bus, adhering to the MCERR protocol.                                                                                                                                        | The NB detected that a processor issued an MCERR.                                                                                                           | NB-3                    |
| F4                | Request/Address Parity Error                       | Fatal      | NRECFSB{A/B}_LOG[2:0] | Complete transaction on FSB, drop transaction, provide a response. Capture Address/Control information.                                                                                                                          | The NB monitors the address and request parity signals on the FSB. A parity discrepancy over these fields during a valid request.                           | NB-4                    |
| F5                | Address Strobe Glitch                              | Fatal      | NRECFSB{A/B}_LOG[2:0] | Take the received address and attempt to complete the transaction, this may cause the NB to Malfunction. Log which FSB the error occurred on. Let request continue with what it decodes to, correct operation is not guaranteed. | The NB detects that an address strobe toggled when ADS# wasn't valid.                                                                                       | NB-5                    |

Table 6-39. Errors Detected by the NB (Sheet 2 of 6)

| Error#                           | Error Name                                                                                                                   | Error type | Log Register(s)       | Cause / Action                                                                                                           | Definition                                                                                                             | Comp. Reg. Bit Position |
|----------------------------------|------------------------------------------------------------------------------------------------------------------------------|------------|-----------------------|--------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------|-------------------------|
| F6                               | Data Strobe Glitch                                                                                                           | Fatal      | NRECFSB{A/B}_LOG[2:0] | Take the received data and attempt to complete the transaction, this may cause the NB to Malfunction.                    | The NB detects that one (or more) of the complimentary FSB data strobe pairs did not toggle in an alternating fashion. | NB-6                    |
| F7                               | FSB Protocol Error                                                                                                           | Fatal      | NRECFSB{A/B}_LOG[2:0] | Alias to a known response.                                                                                               | The NB detected an unsupported FSB response encoding.                                                                  | NB-7                    |
| F8                               | Unsupported FSB Transaction                                                                                                  | Fatal      | NRECFSB{A/B}_LOG0     | Unsupported transaction or unsupported data sizes. Transactions alias to a similar transaction to complete.              | The NB detected an FSB Unsupported transaction or unsupported data sizes.                                              | NB-8                    |
| F9                               | Data Parity Error (for 64-bit Intel® Xeon® processor MP with 1MB L2 cache & Dual-Core Intel® Xeon® processor 7000 sequence ) | Fatal      | NRECFSB               | Parity Error was detected on the Data, DINV, DSTB when there was no correctable ECC Error.                               | The NB detected a Parity Error on the Data, DINV, DSTB when there was no correctable ECC Error.                        | N/A                     |
| F10                              | FSB Hang                                                                                                                     | Fatal      | NRECFSB{A/B}_LOG0     | An access issued on the FSB has timed out.                                                                               | The NB detected that a timely FSB transaction completion did not occur.                                                | NB -10                  |
| <b>IMI.NB Errors<sup>a</sup></b> |                                                                                                                              |            |                       |                                                                                                                          |                                                                                                                        |                         |
| IMI0                             | Uncorrectable Error in read return packet before retry limit exceeded                                                        | Non-Fatal  | RECIMI, REDPIMIH      | Capture the Syndrome, Locator, and Address, If possible, scrub location using redundant resources. Re-Issue IMI Request. | The NB detected an uncorrectable error on an IMI transaction before the retry limit was exceeded.                      | NB-0                    |
| IMI1                             | Inbound Correctable ECC Error                                                                                                | Non-Fatal  | RECIMI, REDPIMIH      | Capture the Syndrome, Locator, and Address, Correct.                                                                     | The NB detected an Correctable Error on the 1st request for a IMI transaction.                                         | NB-1                    |
| IMI2                             | Command Time-Out                                                                                                             | Non-Fatal  | RECIMI                | Re-issue Request Over M Link, log CDC and IMI Transaction IDs.                                                           | The NB failed to receive a timely response to a request.                                                               | NB-2                    |
| IMI3                             | Inbound CRC Error                                                                                                            | Non-Fatal  | N/A                   | NB will drop response, and rely on retry to correct.                                                                     | The NB received a link CRC error.                                                                                      | NB-3                    |
| IMI4                             | Memory Write Data Poisoned.                                                                                                  | Non-Fatal  | N/A                   | The NB has poisoned the flit that contains the error and written the data to memory. (See XMI10)                         | The NB has poisoned the flit that contains the error and written the data to memory. (See IMI.XMB10)                   | NB-4                    |
| IMI5                             | Inbound Link Layer Control Error                                                                                             | Non-Fatal  | N/A                   | N/A                                                                                                                      | The NB detected a link layer control error on the Inbound link.                                                        | NB-5                    |
| IMI6                             | Abort                                                                                                                        | Non-Fatal  | RECIMI                | N/A                                                                                                                      | The NB detected a protocol abort from the XMB.                                                                         | NB-6                    |
| IMI7                             | SMBus Hot-Plug Error                                                                                                         | Non-Fatal  | N/A                   | N/A                                                                                                                      | The NB detected an error on the Hot-Plug SMBus port.                                                                   | NB-7                    |

Table 6-39. Errors Detected by the NB (Sheet 3 of 6)

| Error#              | Error Name                                              | Error type | Log Register(s) | Cause / Action                                                                                                                          | Definition                                                                                                                                                | Comp. Reg. Bit Position |
|---------------------|---------------------------------------------------------|------------|-----------------|-----------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------|
| IMI8                | Outstanding request to Failed IMI                       | Non-Fatal  | RECIMI          | In RAID/Mirroring config, if requests are pending when IMI enters failed state. Attempt to complete requests using redundant resources. | The NB in RAID/Mirroring config, if requests are pending when IMI enters failed state.                                                                    | NB-8                    |
| IMI9                | Failed on last retry. Co-exist with IMI13               | Non-Fatal  | RECIMI          | In RAID/Mirroring config, attempt to complete request using redundant resources.                                                        | The NB in RAID/Mirroring config and not in Redundancy lost state and not in Resilver state received an uncorrectable data error on the IMI retry request. | NB-9                    |
| IMI10               | Config write data poisoned.                             | Non-Fatal  | N/A             | The NB has poisoned the flit that contains the error and written the config to the XMB. (See XMI18).                                    | The NB sent a Config memory write with poisoned data. (See XMI18)                                                                                         | NB-10                   |
| IMI11               | NB received a correctable error message from the XMB    | Non-Fatal  | N/A             | NB will signal that XMB recorded an error.                                                                                              | The NB received a IMI virtual wire that the XMB has detected a "correctable" error event.                                                                 | NB-11                   |
| IMI12               | NB received an uncorrectable error message from the XMB | Non-Fatal  | N/A             | NB will signal that XMB recorded an error.                                                                                              | The NB received a IMI virtual wire that the XMB has detected an "uncorrectable" error event.                                                              | NB-12                   |
| IMI13               | Failed on last retry. Co-exist with IMI9                | Non-Fatal  | RECIMI          | If (RAID or Mirror) Capture failed IMI Request.                                                                                         | The NB in RAID/Mirroring config, receives a timeout on the retry of transaction                                                                           | NB-13                   |
| IMI14               | NB received a fatal error message from the XMB          | Fatal      | N/A             | NB will signal that XMB recorded an error.                                                                                              | The NB received an IMI virtual wire "signal" that the XMB has detected a "fatal" error event.                                                             | NB-14                   |
| IMI15               | <b>Reserved</b>                                         | N/A        | N/A             | N/A                                                                                                                                     | N/A                                                                                                                                                       | N/A                     |
| <b>H11.5 Errors</b> |                                                         |            |                 |                                                                                                                                         |                                                                                                                                                           |                         |
| IO0                 | HI - Target Abort                                       | Non-Fatal  | RECHI           | Received from the ICH5.                                                                                                                 | The NB receives a Target-Abort for any cycle for which it was the master from the ICH5.                                                                   | NB-0                    |
| IO1                 | HI - Cmd Parity Error                                   | Non-Fatal  | RECHI           | Received from the ICH5.                                                                                                                 | The NB detected a parity error in the command portion of the packet received from the ICH5.                                                               | NB-1                    |
| IO2                 | HI - Data Phase Parity Error                            | Non-Fatal  | RECHI           | Received from the ICH5.                                                                                                                 | The NB detected a parity error in the data portion of the packet received from the ICH5.                                                                  | NB-2                    |
| IO3                 | HI - Parity Error on Outbound Data                      | Non-Fatal  | RECHI           | N/A                                                                                                                                     | The NB detected an internal parity error on outbound data.                                                                                                | NB-3                    |
| IO4                 | HI - Outbound poison                                    | Non-Fatal  | N/A             | N/A                                                                                                                                     | The NB detected a parity error in the address/header from the ICH5.                                                                                       | NB-4                    |

Table 6-39. Errors Detected by the NB (Sheet 4 of 6)

| Error#                    | Error Name                                 | Error type | Log Register(s) | Cause / Action                 | Definition                                                                                                   | Comp. Reg. Bit Position |
|---------------------------|--------------------------------------------|------------|-----------------|--------------------------------|--------------------------------------------------------------------------------------------------------------|-------------------------|
| IO5                       | HI - Port has made an illegal access       | Fatal      | RECHI           | Received from the ICH5         | The NB detected unsupported request from the ICH5.                                                           | N/A                     |
| IO6                       | HI - Parity Error on Header / Address      | Fatal      | RECHI           | Received from the ICH5         | The NB detected a parity error in the address portion of the packet received from the ICH5.                  | N/A                     |
| IO6b                      | HI - Received STOP during SRC              | Fatal      | N/A             | Received from the ICH5         | ICH5 illegally drove STOP                                                                                    | N/A                     |
| <b>PCI Express Errors</b> |                                            |            |                 |                                |                                                                                                              |                         |
| IO7                       | PCI Express - Receiver Error               | Corr       | N/A             | N/A                            | Received bad 8b/10b encoding error.                                                                          | NB - I/O Unit           |
| IO8                       | PCI Express - Bad TLP Error                | Corr       | N/A             | N/A                            | Received bad CRC or a bad sequence number in a transport layer packet.                                       | NB - I/O Unit           |
| IO9                       | PCI Express - BAD DLLP                     | Corr       | N/A             | N/A                            | Log Header of Error'ed Packet.                                                                               | NB - I/O Unit           |
| IO10                      | PCI Express - Replay_Num Rollover          | Corr       | N/A             | N/A                            | Replay maximum count for the Retry Buffer has been exceeded.                                                 | NB - I/O Unit           |
| IO11                      | PCI Express - Replay Timer Timeout         | Corr       | N/A             | N/A                            | Replay timer timed out waiting for an ACK or NAK DLLP.                                                       | NB - I/O Unit           |
| IO12                      | PCI Express - Received Unsupported Request | UnCorr     | HDRLOG[3:0]     | Log Header of Packet.          | Received an unsupported request, similar to master abort.                                                    | NB - I/O Unit           |
| IO13                      | PCI Express - Training Error               | N/A        | N/A             | N/A                            | The NB never reports this error.                                                                             | NB - I/O Unit           |
| IO14                      | PCI Express - Poisoned TLP                 | UnCorr     | HDRLOG[3:0]     | Log Header of Error'ed Packet. | Received a poisoned transaction layer packet from the southbridge.                                           | NB - I/O Unit           |
| IO15                      | PCI Express - Flow Control Protocol Error  | Fatal      | N/A             | N/A                            | NB has detected a PCI-Express Flow Control Protocol Error.                                                   | NB - I/O Unit           |
| IO16                      | PCI-Express - Completion Timeout           | UnCorr     | N/A             | N/A                            | Pending transaction was ACKed in the data link layer but not within the time limit.                          | NB - I/O Unit           |
| IO17                      | PCI Express - Completer Abort              | UnCorr     | HDRLOG[3:0]     | Log Header of Error'ed Packet. | Received return CA status for horrible error on the component. This is equivalent to a target abort on PCI.  | NB - I/O Unit           |
| IO18                      | PCI Express - Unexpected Completion Error  | UnCorr     | HDRLOG[3:0]     | Log Header of Error'ed Packet. | Received a Completion RequestorID that matches the requestor but the Tag does not match any pending entries. | NB - I/O Unit           |
| IO19                      | PCI Express - Malformed TLP                | Fatal      | HDRLOG[3:0]     | Do not update content          | Received a transaction layer packet that does not follow the TLP formation rules.                            | NB - I/O Unit           |

Table 6-39. Errors Detected by the NB (Sheet 5 of 6)

| Error#           | Error Name                                                           | Error type | Log Register(s) | Cause / Action                                                                                  | Definition                                                                                                                                       | Comp. Reg. Bit Position |
|------------------|----------------------------------------------------------------------|------------|-----------------|-------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------|
| IO20             | PCI Express - Receive Buffer Overflow Error                          | Fatal      | N/A             | Correct by Hardware                                                                             | Receiver gets more data or transactions than credits allow.                                                                                      | NB - I/O Unit           |
| IO21             | PCI Express - Data Link Layer Protocol Error                         | Fatal      | N/A             | N/A                                                                                             | The NB detected a DLLP error.                                                                                                                    | NB - I/O Unit           |
| IO22             | N/A                                                                  | N/A        | N/A             | N/A                                                                                             | N/A                                                                                                                                              | N/A                     |
| IO23             | N/A                                                                  | N/A        | N/A             | N/A                                                                                             | N/A                                                                                                                                              | N/A                     |
| IO24             | PCI Express - Received Correctable Error Message                     | Corr       | N/A             | N/A                                                                                             | NB received a correctable error message from the south bridge.                                                                                   | N/A                     |
| IO25             | PCI Express - Received NonFatal Error Message                        | Non-Fatal  | N/A             | N/A                                                                                             | NB received a non-fatal error message from the south bridge.                                                                                     | N/A                     |
| IO26             | PCI Express - Received Fatal Error Message                           | Fatal      | N/A             | N/A                                                                                             | NB received a fatal error message from the south bridge.                                                                                         | NB - I/O Unit           |
| <b>NB Errors</b> |                                                                      |            |                 |                                                                                                 |                                                                                                                                                  |                         |
| T1               | Single-bit Data ECC Error from CDC                                   | Non-Fatal  | REDTNB          | Corrected by Hardware, log CDC entry.                                                           | The NB detected in an internal block a correctable ECC error (This error was not generated by receiving bad data from an external interface).    | NB-0                    |
| T2               | Uncorrectable Data ECC Error from CDC (Do not Include Poisoned Data) | Non-Fatal  | NREDTNB         | Detected by Hardware, log CDC entry.                                                            | The NB detected in an internal block an uncorrectable ECC error. (This error was not generated by receiving bad data from an external interface) | NB-7                    |
| T3               | Multi-Tag Hit from CDC                                               | Fatal      |                 |                                                                                                 | The NB does not detect this error.                                                                                                               | NB-5                    |
| T4               | Coherency Violation Error                                            | Fatal      | NRECTNB         | Log CDC entry                                                                                   | The NB detected a cache coherency protocol error.                                                                                                | NB-6                    |
| T5               | Address Map Error                                                    | Non-Fatal  | REDTNB          |                                                                                                 | The NB received a request for an address that does not exist in the system.                                                                      | NB-1                    |
| T6               | Poison passed to another poison capable domain.                      | Non-Fatal  | REDTNB          |                                                                                                 | The NB created or detected poisoned data and gave that data to another poison capable domain inside NB.                                          | NB-2                    |
| T7               | Uncorrectable Data Error during a Re-silvering Process               | Non-Fatal  | RECINT_LOG[4:0] | Logged by RAID/mirror logic in response to PM19. Capture failed request. Poison data in memory. | The NB detected an uncorrectable error on the primary image and secondary method to recover the data is not available.                           | NB-3                    |
| T8               | Poison leaving a poison capable domain                               | Fatal      | NRECTNB         |                                                                                                 | The NB has non-poison capable domains and poison capable domains. When poison leaves a poison capable domain, this error is detected.            | NB-4                    |

Table 6-39. Errors Detected by the NB (Sheet 6 of 6)

| Error# | Error Name                                             | Error type | Log Register(s) | Cause / Action | Definition                                                                                                    | Comp. Reg. Bit Position |
|--------|--------------------------------------------------------|------------|-----------------|----------------|---------------------------------------------------------------------------------------------------------------|-------------------------|
| T9     | SMBus Virtual Pin Interface Error                      | Fatal      | N/A             |                | The NB detected an error on the SMBus virtual pin interface.                                                  | NB - I/O Unit           |
| T10    | I/O Unit Detected Failure                              | Fatal      | N/A             |                | The NB I/O Unit detected an internal failure. Uncorrectable Data Error Detected Address/Control Parity Error. | NB - I/O Unit           |
| T11    | I/O Unit Detected a correctable ECC error from CDC.    | Non-Fatal  | N/A             |                |                                                                                                               | NB - I/O Unit           |
| T12    | I/O Unit Detected an uncorrectable ECC error from CDC. | Fatal      | N/A             |                |                                                                                                               | NB - I/O Unit           |

**NOTES:**

- a. For IMI errors seen from the XMB side, readers should refer to chapter 6 of *Intel® E8501 Chipset eXternal Memory Bridge (XMB) Datasheet*

## 6.8 Interrupts

The NB chipset supports both the xAPIC and traditional 8259 methods of interrupt delivery. I/O interrupts, and inter-processor interrupts (IPIs) appear as write or interrupt transactions in the system and are delivered to the target processor via the FSB. This chipset does not support the three-wire sideband bus (the APIC bus) that is used by Pentium® and Pentium® Pro processors.

xAPIC interrupts that are generated from I/O will need to go through an I/O(x)APIC device unless they support Message Signalled Interrupts (MSI). In this document, I/O(x)APIC is an interrupt controller. This functionality will be embedded into the Intel 6700PXH 64-bit PCI Hub and ICH5 components of the chipset.

The legacy 8259 functionality is embedded in the ICH5 component. The NB chipset will support inband 8259 interrupt messages from PCI Express devices for boot. The chipset also supports the processor generated “interrupt acknowledge” (for legacy 8259 interrupts), and “end-of-interrupt” transactions (xAPIC).

Routing and delivery of interrupt messages and special transactions are described in this chapter.

### 6.8.1 xAPIC Interrupt Message Delivery

For Message Delivery and Format, refer to the Section 2.5, “Interrupt Delivery” on page 31 of this document.

## 6.8.2 xAPIC Destination Modes

The destination mode refers to how the processor interprets the destination field of the interrupt message. There are two types of destination modes:

1. **Physical Destination Mode**
2. **Logical Destination Mode**

The destination mode is selected by A[2] in PCI Express and Ab[5] on the FSB.

### 6.8.2.1 Physical Destination Mode (xAPIC)

In physical mode, the APIC ID is 8 bits, supporting up to 255 agents. Each processor has a Local APIC ID Register where the lower 5 bits are initialized by hardware (Cluster ID=ID[4:3], Bus Agent ID=ID[2:1], thread ID=ID[0]). The upper 3 bits default to 0s at system reset. These values can be modified by software. The Cluster ID is set by address bits A[12:11] during reset. By default, the NB will drive A[12:11] to '00' for Front Side Bus A, and '01' for Front Side Bus B. The value driven on bit A[12] during reset can be modified through the POC register on the NB.

The NB will not rely on the cluster ID or any other fields in the APIC ID to route interrupts. The NB will ensure the interrupt is seen on both busses and the processor with the matching APIC ID will claim the interrupt.

Physical destination mode interrupts can be directed, broadcast, or redirected. An xAPIC message with a destination field of all 1s denotes a broadcast to all.

In a directed physical mode message the agent claims the interrupt if the upper 8 bits of the destination field (DID field) matches the Local APIC ID of the processor or the interrupt is a broadcast interrupt.

### 6.8.2.2 Logical Destination Mode (xAPIC)

In logical destination mode, destinations are specified using an 8-bit logical ID field. Each processor contains a register called the Logical Destination Register (LDR) that holds this 8-bit logical ID. Interpretation of the LDR is determined by the contents of the processor's Destination Format Register (DFR). The DFR establishes if the processor is in flat or cluster mode. Logical destination mode interrupts can be directed (fixed delivery), redirectable (lowest priority delivery), or broadcast. The LDR is initialized to flat mode (0) at reset and is programmed by firmware (and can be changed by system software). The NB also has an equivalent bit in the XTPR0 register to indicate flat or cluster mode.

Interpretation of the destination field of the xAPIC message and the logical ID of the processor is different whether the processor is in flat or cluster mode. The 2 models are flat and cluster:

**Flat Model** - The 8-bit logical ID is compared to the 8-bit destination field of the incoming interrupt message. If there is a bit-wise match, then the local xAPIC is selected as a destination of the interrupt. Each bit position in the destination field corresponds to an individual Local xAPIC Unit. The flat model supports up to 4 agents in the system. An xAPIC message where the DID (destination field) is all 1's is a broadcast interrupt.

**Cluster Model** - The 8-bit logical ID is divided into two fields, where the upper four bits contain a cluster ID, and the lower 4-bits identifies the agent within the cluster using the same encoding scheme as the flat model. Up to 8 logical clusters (2 agents each) can be supported. An xAPIC message where the lower 4-bits of the DID is all 1's is a broadcast interrupt within a cluster. Interrupt redirection can only be applied to processors *within* the cluster.

Interrupt messages that originate from I/O(x)APIC devices or from processing nodes must be routed and delivered to the target agents in the system. In general xAPIC messages are delivered to both FSB's because there is no reliable way to determine the destination node of the message for the destination field. Interrupts originating from I/O can be generated from a PCI agent using MSI interrupts, or by an interrupt controller on a bridge chip such as the Intel 6700PXH 64-bit PCI Hub or ICH5. Table 6-40 shows the routing rules used for routing xAPIC messages in a NB platform. This table is valid for both broadcast and non-broadcast interrupts.

**Table 6-40. NB xAPIC Interrupt Message Routing and Delivery**

| Source     | Type                                            | Routing                                                                       |
|------------|-------------------------------------------------|-------------------------------------------------------------------------------|
| I/O        | physical or logical directed                    | Deliver to both FSB's as an interrupt transaction.                            |
| Processor  | physical or logical directed                    | Deliver to other FSB as an interrupt transaction.                             |
| Any Source | logical, redirectable<br>physical, redirectable | Redirection is performed by the NB and is delivered to both front side buses. |

### 6.8.3 EOI

For Intel® Xeon® Processor Family (XPF) platforms using xAPIC, the EOI is a specially encoded FSB transaction with the interrupt vector attached. Since the EOI is not directed, the NB will broadcast the EOI transaction to all I/O(x)APICs.

**Note:** Since the ICH5 does not support the EOI special cycle, the Intel® E8501 chipset will translate the EOI into a directed write to the EOI register in the ICH5.

### 6.8.4 I/O Interrupts

For I/O interrupts from the Intel 6700PXH 64-bit PCI Hub and ICH5 components receive interrupts with either dedicated interrupt pins or with writes to the integrated redirection table. The I/OxAPIC controller integrated within these components turns these interrupts into writes destined for the FSB with a specific address.

Interrupts triggered from an I/O device can be triggered with either a dedicated interrupt pin or through an inbound write message from the PCI bus (MSI). Note that if the interrupt is triggered by a dedicated pin, the I/OxAPIC controller in the I/O bridge (Intel 6700PXH 64-bit PCI Hub or ICH5) turns this into an inbound write. On the FSB, the interrupt is converted to an interrupt request. Other than a special interrupt encoding, the FSB interrupt follows the same format as discussed in Section 2.5, "Interrupt Delivery" on page 31. Therefore, to all components other than the Intel 6700PXH 64-bit PCI Hub, ICH5, and the processors, an interrupt is an inbound write following the format mentioned in Section 2.5, the NB will not write combine or cache the APIC address space.

I/O(x)APICs can be configured through two mechanisms. The traditional mechanism is the hard coded FEC0\_0000h to FECF\_FFFFh range is used to communicate with the IOAPIC controllers in the Intel 6700PXH 64-bit PCI Hub or ICH5.

The second method is to use the standard MMIO range to communicate to the Intel 6700PXH 64-bit PCI Hub. To accomplish this, the PXH.MBAR must be programmed within the PCI Express device MMIO region.

### 6.8.5 Ordering

Handling interrupts as inbound writes has inherent advantages. First, there is no need for the additional APIC bus resulting in extra pins and board routing concerns. Second, with an out-of-band APIC bus, there are ordering concerns. Any interrupt needs to be ordered correctly and all prior inbound writes must get flushed ahead of the interrupt. The *PCI Local Bus Specification*, Revision 2.2 attempts to address this by requiring all interrupt routines to first read the PCI interrupt register. Since PCI read completions are required to push all writes ahead of it, then all writes prior to the interrupt are guaranteed to be flushed. This, However assumes that all drivers perform this read.

### 6.8.6 Hardware IRQ IOxAPIC Interrupts

Dedicated pin interrupts may be edge or level-triggered. They are routed to IRQ pins on IOxAPIC device such as the Intel 6700PXH 64-bit PCI Hub and ICH5. The IOxAPIC device will convert the interrupt into either an xAPIC or 8259 interrupt.

For level-triggered interrupts, the I/OxAPIC will generate an interrupt message when any of the interrupt lines coming into it become asserted. The processor will handle the interrupt and eventually write to the initiating device that the interrupt is complete. The device will de-assert the interrupt line to the I/OxAPIC. After the interrupt has been serviced, the processor sends an EOI command to inform the I/OxAPIC that the interrupt has been serviced. Since the EOI is not directed, the NB will broadcast the EOI transaction to all I/O(x)APIC's. If the original I/O(x)APIC sees the interrupt is still asserted, it knows there's another interrupt (shared interrupts) and will send another interrupt message.

For edge-triggered interrupts, the flow is the same except that there is no EOI message indicating that the interrupt is complete. Since the interrupt is issued whenever an edge is detected, EOIs are not necessary.

While not recommended, agents can share interrupts to better utilize each interrupt (implying level-triggered interrupts). Due to ordering constraints, agents can not use an interrupt controller that resides on a different PCI bus. Therefore only agents on the same PCI bus can share interrupts (or the driver MUST follow the PCI requirement that interrupt routines must first read the PCI interrupt register).

The NB supports the INTA (interrupt acknowledge) special bus cycle for legacy 8259 support. These are routed to the compatibility ICH5 in the system. The INTA will return data that provides the interrupt vector.

### 6.8.7 Message Signalled Interrupts (MSI)

A second mechanism for devices to send interrupts is to issue the Message Signalled Interrupt (MSI) introduced in the *PCI Local Bus Specification*, Revision 2.2 . This appears as a 1 DWORD write on the PCI/PCI-X/PCI Express bus.

With PCI devices, there are 2 types of MSIs. One type is where a PCI device issues the inbound write to the interrupt range. The other type of MSI is where a PCI device issues an inbound write to the upstream APIC controller (for example, in the Intel 6700PXH 64-bit PCI Hub) where the APIC controller converts it into an inbound write to the interrupt range. The second type of MSI can be used in the event the OS doesn't support MSIs, but the BIOS does. For either way, the interrupt will appear as an inbound write to the NB over the PCI Express ports.

MSI is expected to be supported by the operating systems about the same time frame as Intel® E8501 chipset availability. A NB chipset-based platform will also feature a backup interrupt mechanism in the event that there is a short period of time when MSI is not available. This is described in the next section.

### 6.8.8 Non-MSI Interrupts - “Fake MSI”

For interrupts coming through the Intel 6700PXH 64-bit PCI Hub or ICH5 components, their APIC controller will convert interrupts into inbound writes, so inbound interrupts will appear in the same format as an MSI.

For interrupts that are not coming through an APIC controller (for example Dual Northway, a dual port 1 GbE) it is still required that the interrupt appear as an MSI-like interrupt. If the OS does not yet support MSI, the PCI Express device can be programmed by the BIOS to issue inbound MSI interrupts to an IOxAPIC in the system. The safest IOxAPIC to choose would be the ICH5 since it is always present in a system. The NB does not support the PCI Express “Assert\_Int” and “Deassert\_Int” packets for xAPIC interrupts.

In this method, PCI Express devices are programmed to enable MSI functionality, and given a write path directly to the pin assertion register in a selected IOxAPIC already present in the platform. The IOxAPIC will generate an interrupt message in response, thus providing equivalent functionality to a virtual (edge-triggered) wire between the PCI Express endpoint and the IOxAPIC.

All PCI Express devices are strictly required to support MSI. When MSI is enabled, PCI Express devices generate a memory transaction with an address equal to the IOxAPIC\_MEM\_BAR + 20 and a 32-bit data equal to the interrupt vector number corresponding to the device. This information is stored in the device's MSI address and data registers, and would be initialized by the system firmware (BIOS) prior to booting a non-MSI aware operating system (With the theory that an MSI aware O/S would then over-write the registers to provide interrupt message delivery directly from the endpoint to the CPU complex).

The PCI Express memory write transaction propagates to the NB and is redirected down the appropriate PCI Express port following the NB IOAPIC address mapping definition. The IOAPIC memory space ranges are fixed and cannot be relocated by the OS. The assert message is indistinguishable from a memory write transaction, and is forwarded to the destination IOxAPIC (which will then create an upstream APIC interrupt message in the form of an inbound memory write). The write nature of the message “pushes” all applicable pre-interrupt traffic through to the NB core, and the NB core architecture guarantees that the subsequent APIC message cannot pass any posted data already within the NB.

### 6.8.9 Inter Processor Interrupts (IPIs)

The IPIs will be covered in the future revision of this document.

## 6.8.10 Chipset Generated Interrupts

The Intel® E8501 chipset can trigger interrupts for chipset errors and for PCI Express and IMI Port Hot-Plug. For these events, the chipset can be programmed to assert pins that the system can route to an APIC controller. The following is a preliminary list of interrupts that can be generated:

1. Chipset Error: Chipset asserts appropriate ERR pin, depending on severity. This can be routed by the system to generate an interrupt at an interrupt controller. (NB pins ERR[2:0], MCERR, ICH Reset). For details on Chipset error types and reporting, please see [Section 6.7.1](#).
2. IMI Hot-Plug: Chipset asserts **IMI\_HPINT** on behalf of an IMI Hot-Plug event. This can be routed by the system to generate an interrupt. This document assumes that **IMI\_HPINT** will be connected either to a system interrupt, SMI interrupt, or SCI (ACPI) general purpose event.
3. PCI Express Error: Chipset asserts appropriate ERR pin, depending on severity. This can be routed by the system to generate an interrupt.
  - The NB can receive error indications from the PCI Express ports. These are in the form of inbound ERR\_COR/UNC/FATAL messages. The NB will assert the appropriate ERR signal just like any internal NB error.
4. PCI Express Hot-Plug: The NB asserts EXP\_HPINT on behalf of a PCI Express Hot-Plug event.
  - NB generated Hot-Plug event such as PresDet change, Attn button, MRL sensor changed, power fault, etc. Each of these events has a corresponding bit in the PCI Express Hot-Plug registers (Attention Button, Power Indicator, Power Controller, Presence Detect, MRL Sensor, Port Capabilities/Slot registers). This will generate an interrupt via the EXP\_HPINT pin, or it will generate an MSI.
    - **EXP\_HPINT**: When software has completed servicing the Hot-Plug event, it will clear the appropriate bit in the PCI Express Hot-Plug register(s), at which point, the NB can deassert EXP\_HPINT.
    - **MSI**: If the MSICTRL.MSIEN bit (MSI enabled) is set, the NB will generate an MSI in response to a PCI Express Hot-Plug event. More details will be added in the subsequent revision of this document.

- PCI Express Hot-Plug event from downstream. This could be either an MSI or a GPE message.
    - **MSI:** Handled like a normal MSI interrupt.
    - **GPE Message:** Upon receipt of a Assert\_GPE message from PCI Express, NB will assert the EXP\_HPINT signal to generate this interrupt. To generate an SCI (ACPI), this signal will be routed to the ICH5 appropriate GPIO pin to match the GPE0\_EN register settings. When the Hot-Plug event has been serviced, the NB will receive a Deassert\_GPE message. At this point the NB can deassert EXP\_HPINT. There needs to be a tracking bit per PCI Express port to keep track of Assert/Deassert\_GPE pairs. These tracking bits should be ORed together to determine whether to drive the EXP\_HPINT pin. If one or more of the tracking bits are set (meaning we have received one or more Assert\_GPE messages), EXP\_HPINT must be set until all the matching Deassert\_GPE messages are received.
    - **Sideband Signals:** Some systems may choose to connect the interrupt via sideband signals directly to the ICH5. No action is required from the NB.
5. PCI Express PM: The NB asserts PME\_OUT pin when a power management event is detected. PCI Express sends a PME message.
- Upon receipt of the PME message, the NB will assert the PME\_OUT signal to generate this interrupt if enabled through bit 3 in ‘EXP\_RTCTRL[7:1]: PCI Express Root Control Register (D1-7, F0)’. To generate an SCI (ACPI), this signal will be routed to the ICH5 appropriate GPIO pin to match the GPE0\_EN settings. When software has completed servicing the power management event, it will clear the PCI Express RTSTS.PMESTS bit (by writing 1), at which point the NB can deassert PME\_OUT.
6. PCI Hot-Plug: The NB will receive an Assert/Deassert GPE message from the PCI Express port when a PCI Hot-Plug event is happening. Assert/Deassert GPE messages should be treated the same as Assert/Deassert GPE messages for PCI Express Hot-Plug. (Keep track of Assert/Deassert GPE messages from each port and assert EXP\_HPINT appropriately).

Table 6-41 summarizes the different types of chipset generated interrupts that were discussed. Although the interrupt and SW mechanism is flexible and can be changed depending on how the system is hooked up, for reference this table also describes what SW mechanism is expected to be used.

Table 6-41. Chipset Generated Interrupts

| Source                                                         | Signalling Mechanism                  | NB Signal Method                 | Expected SW Mechanism             |
|----------------------------------------------------------------|---------------------------------------|----------------------------------|-----------------------------------|
| Chipset Error                                                  | NB Chipset registers                  | ERR[2:0], MCERR, ICHReset        | Any                               |
| IMI HP                                                         | NB Chipset registers                  | IMI_INT                          | Any (SCI, SMI, system (MSI/APIC)) |
| PCI Express Error                                              | PCI Express ERR_COR/UNC/FATAL message | ERR[2:0], MCERR, ICHReset        | Any                               |
| PCI Express HP (PresDet chg, atin button, etc.)                | NB Chipset registers                  | EXP_HPINT or MSI                 | SCI->ACPI or MSI                  |
| PCI Express HP form downstream device (Native)                 | MSI                                   | MSI interrupt <sup>a</sup> (FSB) | MSI                               |
| PCI Express HP form downstream device (Intel Part, Non-Native) | PCI Express Assert/Deassert GPE       | EXP_HPINT                        | SCI->ACPI                         |
| PCI Express HP form downstream device (Intel Part, Non-Native) | Sideband signals directly to ICH5     | N/A                              | SCI->ACPI                         |
| Downstream PCI HP                                              | PCI Express Assert/Deassert GPE       | EXP_HPINT                        | SCI->ACPI                         |
| Power Management Event (PME)                                   | PCI Express PM_PME message            | PME_OUT                          | SCI->ACPI                         |

**NOTES:**

a. Technically, this is not chipset generated, but is included for completeness.

## 6.8.11 Legacy/8259 Interrupts

The 8259 interrupt controller is supported in NB platforms. 8259 interrupt request is delivered using the interrupt group sideband signals LINT[1:0] (a.k.a. NMI/INTR) or through an I/O xAPIC using the message based interrupt delivery mechanism with the delivery mode set to ExtINT (b111). There can be only one active 8259 controller in the system.

The mechanism in which a PCI Express device requests an 8259 interrupt is a PCI Express inband message. (ASSERT\_INTA/B/C/D, DEASSERT\_INTA/B/C/D).

The target processor for the interrupt uses the interrupt acknowledge transaction to obtain the interrupt vector from the 8259 controller. The NB forwards the interrupt acknowledge to the ICH5 where the active 8259 controller resides.

The NB will support PCI Express devices that generate 8259 interrupts (for example, during boot). 8259 interrupts from PCI Express devices will be sent in-band to the NB which will forward these interrupts to the ICH5.

The NB will have a mechanism to track inband 8259 interrupts from each PCI Express and assert virtual interrupt signals to the 8259 through the inband DO\_PCI\_INT HL special cycle. This is done by a tracking bit per interrupt (A, B, C, D) in each PCI Express which are combined (ORed) into virtual signals that are sent to the ICH5. Each interrupt signal (A, B, C, D) from each PCI Express is ORed together to form virtual INT A, B, C, and D signals to the ICH5.

(DO\_PCI\_INTA/B/C/D (assertion encoding)). When all the tracking bits for a given interrupt (A, B, C, or D) are cleared from all PCI Express ports, the virtual signal A, B, C, or D is deasserted via the inband DO\_PCI\_INT (deassertion encoding).

For PCI Express hierarchies, interrupts will be consolidated at each level. For example, a PCI Express switch connected to a NB PCI Express port will only send a maximum of 4 interrupts at a time, regardless of how many interrupts are issued downstream.

SMI (System Management Interrupt) interrupts are initiated by the SMI# signal in the platform. On accepting a System Management Interrupt, the processor saves the current state and enters SMM.

## 6.9 Reset

### 6.9.1 Introduction

This section describes NB-specific aspects of hardware reset. Subsequent I/O initialization procedures requiring configuration register dialogue are not covered in this section.

NB expects to receive RSTIN# from the ICH5 chip. NB uses this signal to reset the internal state of NB, and additionally, forwards this reset to the processor(s) in the system via the H\_A\_RST# and H\_B\_RST# signals. NB controls the reset of the XMB chips connected to NB via software using the IMI\_{A/B/C/D}\_RST# signals which are expected to be connected from NB to XMB.

A system is expected to connect the ICHRST# pin of NB to the PLD controlling the system reset logic that generates the PCIRST# signal into the ICH5 chip.

NB will keep H\_{A/B}\_RST# asserted on an FSB which indicates no end agent is present via the H\_{A/B}\_PRSENT# pin; the system should keep the core and cache VR disabled in this case to avoid any CPU thermal issues.

### 6.9.2 Types of Reset

- Power-up
- Power Good
- PWROK
  - The PWROK reset spawns a hard reset.
- Hard
  - The hard reset spawns IMI, PCI Express, and Processor resets.
- IMI
- Warm
- PCI Express
- JTAG
- SMBus

### 6.9.3 Triggers of Reset

- Energize power supplies
- PWRGOOD de-assertion
- RESTIN# assertion
- NB.SYRE.SYSRST (refer to [Section 4.14.8](#))
- IMI.ICHRST\_inband\_signal
- NB.SYRE.ROR (refer to [Section 4.14.8](#))
- NB.IMIHPC.NEXTSTATE change (Refer to [Section 4.13.10](#))  
IMI initialization time-out is covered by STATE change.
- NB.BCTRL.SRESET (Refer to [Section 4.11.27](#))
- SMBus protocol

### 6.9.4 Trigger to Type Association

This section describes which triggers initiate which types:

- Energize power supplies: Power-up
- PWRGOOD de-assertion: Power good
- RESTIN# assertion: Hard
- NB.SYRE.ROR: Warm
- NB.IMIHPC.NEXTSTATE set to “Reset” or “RecoveryReset”: IMI
- NB.BCTRL.SRESET: PCI Express
- TRST# assertion or TCK/TMS protocol: JTAG
- SMBus protocol: SMBus

### 6.9.5 Logic Domain Behavior

During reset, each of the logic domains is treated as follows:

- Unaffected by reset:
  - PLL's
- Indirectly affected by reset:
  - Arrays  
Initialized by BIST engines
  - Analog I/O compensation  
Only triggered by power-up
- {Boundary scan chains, JTAG protocol engine}:
  - Asynchronous PWRGOOD de-assertion, PWRGOOD de-asserted, asynchronous TRST# assertion, ITP\_TRST# asserted, or synchronous TCK/TMS protocol navigation to reset state: Reset

- SMBus protocol engine:
  - Asynchronous PWRGOOD de-assertion, PWRGOOD de-asserted, or synchronous SMBus reset protocol: Reset
- Sticky configuration bits:
 

Configuration bits with the “ST” attribute are sticky

  - Asynchronous PWRGOOD de-assertion, PWRGOOD de-asserted: Defaults
  - (Synchronized RESTIN# assertion or synchronized RESTIN# asserted) while PWRGOOD asserted: No-Change

Sticky configuration bits must be protected from spurious activity at RESTIN# transitions and while RESTIN# is asserted during hard reset.
- IMI (NB):
  - Asynchronous PWRGOOD de-assertion, PWRGOOD de-asserted, one cycle after synchronized RESTIN# assertion, IMISC.STATE set to “Reset” or “RecoveryReset” set: reset and tri-state outputs.
  - Synchronized RESTIN# de-assertion, IMISC.STATE set to “Reset” or “RecoveryReset” cleared and synchronized RESTIN# de-asserted: Initialize then begin normal operation.
- RESTIN# pad:
  - Synchronized through two HCLKIN-ed stages

This is the synchronized RESTIN#.
- ICHRST# pad:
  - asynchronous PWRGOOD de-assertion, PWRGOOD de-asserted: de-asserted
  - ICHRST# cycle counter not extinguished: Asserted
- PCI Express:
  - Asynchronous PWRGOOD de-assertion, PWRGOOD de-asserted: Tri-state and Reset
  - PWRGOOD asserted, one cycle after synchronized RESTIN# assertion, BCTRL.SRESET set: link down
  - RESTIN# de-assertion, BCTRL.SRESET cleared: initialize, train, link up
- IMI\_{A/B/C/D}\_RST# pad:
  - Controlled by BIOS
  - Asynchronous PWRGOOD de-assertion, PWRGOOD de-asserted, one cycle after synchronized RESTIN# assertion, IMISC.STATE set to “Reset” or “RecoveryReset” set: Asserted
  - Synchronized RESTIN# de-assertion, IMISC.STATE set to “Reset” or “RecoveryReset” cleared and synchronized RESTIN# de-asserted: De-asserted.
- H\_{A/B}\_RST# pad:
  - Asynchronous PWRGOOD de-assertion, PWRGOOD de-asserted, one cycle after synchronized RESTIN# assertion, SYRE.ROR asserted and bus locks cleared and IOQ’s flushed and data buffers flushed and IMI.REFRESH detected: assert
- SYRE.ROR configuration bit
  - asynchronous PWRGOOD de-assertion, PWRGOOD de-asserted, one cycle after synchronized RESTIN# assertion, 1 HCLKIN after processor RESET# pad asserted: clear

- Power-On Configuration:
  - Minimum of 1 HCLKIN after processor RESET# pad asserted: Assert
  - 2-19 HCLKIN after processor RESET# pad de-asserted: De-assert
- Hub Interface:
  - Asynchronous PWRGOOD de-assertion, PWRGOOD de-asserted, one cycle after synchronized RESTIN# assertion: Reset and Tri-stated
  - Synchronized RESTIN# de-asserted: Send NOP -> wait for NOP\_ACK -> send CPU\_RESET\_DONE

## 6.9.6 PWRGOOD De-asserted

Trigger = PWRGOOD de-assertion: Asynchronous

Power and master clocks remain stable.

- De-assert ICHRST# asynchronously.
- Assert H\_{A/B}\_RST# and IMI\_{A/B/C/D}\_RST# asynchronously.
- De-assert other front side bus I/O asynchronously.
- Tri-state other I/O.
- Toggle PLL outputs.
- Reset I/O Ports.
- Resolve multiple drivers to a non-destructive state.

## 6.9.7 PWRGOOD Assertion

Trigger = PWRGOOD assertion: Synchronized

Voltages are within specifications. Master clocks are stable. The TCK signal may be in any state.

- Sample straps.
- Enable Power-On-Configuration (POC) to the front side bus.
- Un-tri-state I/O.
- Hold PCI Express link down.

## 6.9.8 NB ICHRST# Sequence

Triggers = {H\_{A/B}\_RST#, ICHRST#, IMI\_{A/B/C/D}\_RST# Reset Signal and NB.SYRE.SYSRST Set}: Synchronous

- Assert ICHRST# for approximately 100  $\mu$ s asynchronously to CLK66
- De-assert ICHRST# asynchronously to CLK66.  
This de-assertion must wait until 100  $\mu$ s elapses, even though RESTIN# will be asserted before 100  $\mu$ s elapses.
- Clear SYRE.SYSRST when ICHRST# is asserted.

## 6.9.9 Hard Reset Asserted

Trigger = RESTIN# assertion: Synchronized

PWRGOOD is asserted.

- Protect sticky configuration bits.  
Sticky configuration bits must not log spurious errors resulting from the reset process. These processes include de-activated buses, initialization sequences, and spurious inputs on internal interfaces.
- Synchronously assert internal asynchronous flip-flop initialization inputs.  
Private JTAG chains may be reset.  
Exceptions: synchronous test-mode flip-flops, sticky configuration bits, JTAG controller and boundary-scan logic, and analog I/O compensation.
- Assert H\_{A/B}\_RST#.
- Do not assert IMI\_{A/B/C/D}\_RST# - software causes this to assert.
- Initialize non-PCI Express I/O port operation.
- Tri-state IMI I/O after 16 HCLKIN.
- Reset PCI Express protocol.
- Take PCI Express link down.
- Drive processor Power-On-Configuration (POC).  
POC is driven from the SYRE.CPUBIST, POC, and POC\_AUX{A/B} configuration registers. The default values of these configuration registers do not require any processor request signals to be asserted when PWRGOOD is first asserted.
- De-assert non-POC front side bus signals.

## 6.9.10 Hard Reset De-assertion

Trigger = RESTIN# de-assertion: synchronized

PWRGOOD is asserted.

- Allow normal operation of sticky configuration bits.
- Initiate physical IMI initialization.
- Initiate each remaining IMI initialization sequence: handshake, link-width negotiation, and deskew-and-frame.
- Send NOP message to Hub Interface. Wait for NOP\_ACK reply from Hub Interface. Send CPU\_RESET\_DONE message to Hub Interface.
- Initialize PCI Express link.
- Engage PCI Express link training.
- Bring PCI Express link up.
- After approximately 1 ms, de-assert processor RESET#.
- After processor RESET# de-assertion + 2 HCLKIN, disable POC.

### 6.9.11 PCI Express Reset Asserted

Trigger = BCTRL.SRESET set: Synchronous

- Initialize PCI Express protocol.
- Take PCI Express link down.

### 6.9.12 NB PCI Express Reset De-asserted

Trigger = BCTRL.SRESET cleared: synchronous

- Initialize PCI Express link.
- Engage PCI Express link training.
- Bring PCI Express link up.

### 6.9.13 NB IMI Reset Assertion

Trigger = IMIHPC.NEXTSTATE set to “Reset” or “RecoveryReset”: synchronous

- Assert IMI\_{A/B/C/D}\_RST#.
- Tri-state IMI I/O (not including IMI\_{A/B/C/D}\_RST#).

### 6.9.14 NB IMI Reset De-assertion

Trigger = IMIHPC.NEXTSTATE set to “Init” or “RecoveryInit”: Synchronous

- De-assert XMB IMI\_RST#.
- IMI initiates physical initialization.
- After completion of IMI physical initialization, each IMI initiates each remaining initialization sequence: handshake, link-width negotiation, and deskew-and-frame.

### 6.9.15 Warm Reset Sequence

Trigger = SYRE.ROR set or SYRE.CPURST set: Synchronous

- If SYRE.ROR is set, wait for an inband IMI.REFRESH signal from IMI designated by SYRE.REFPML.
- Assert H\_{A/B}\_RST#.

#### 6.9.15.1 Processor Reset Sequence

This is the entry point for a processor reset spawned by a “hard” reset. This is also a continuation of the warm reset sequence.

Trigger = Processor RESET# assertion: Synchronous

- After processor RESET# assertion + minimum of 1 HLCKIN, enable POC.  
POC is driven from the SYRE.CPUBIST, POC, and POC\_AUX[1:0] configuration registers.

- Logically mask front side bus inputs.
- Clear SYRE.ROR and SYRE.CPURST.
- After minimum of 1 ms, de-assert processor RESET#.
- After observing de-assertion of processor RESET#, resume front side bus protocol.
- After processor RESET# de-assertion + minimum of 2 HCLKIN, disable POC.

### 6.9.16 JTAG Reset Assertion

Triggers = {TRST# assertion: asynchronous; TMS asserted for 5 TCK rising edges: Synchronous}

- Initialize JTAG finite-state-machine.
- Initialize boundary-scan chain.

### 6.9.17 JTAG Reset De-assertion

Trigger = ITP\_TRST# de-assertion: Asynchronous

- Release JTAG port and to operate normally.
- Release boundary-scan chain to operate normally.

### 6.9.18 SMBus Reset Sequence

Trigger = SMBus protocol: Synchronous

- Reset SMBus interface

## 6.10 System Management

The NB chipset supports three methods of providing system management to the platform:

- **Processor Access** - All registers of the NB Chipset are available via a FSB access.
- **SMBus Access** - Via an SMBus access to the NB, a register access can be generated to the NB, any XMB, PCI Express device or ICH5. In addition, an SMBus access can go directly to the XMB. It should be noted that unlike processor accesses, the SMBus interface does have some restrictions, namely:
  - SMBus cannot generate a Memory access at the NB or XMB
- **JTAG Access** - The same limitation and restrictions that apply to the SMBus interface also apply to the JTAG interface of the NB components.

The NB chipset provides a fully functional SMBus (or JTAG) target interface, which gives direct access to all internal chipset configuration registers. SMBus/JTAG access provides for a highly flexible platform management architecture when used in conjunction with a baseboard management controller (BMC) with an integrated network interface controller (NIC) function.

## 6.10.1 SMBus Access

System Management software in a NB platform can initiate system management access to the configuration registers. This can be done through SMBus accesses.

The mechanism for the Server Management (SM) software to access configuration registers is a slave port that is close to SMBus Specification, Revision 2.0, but not fully compliant. Some Intel® E8501 chipset components contain this slave port and allow accesses to their configuration registers. The product specific details are compatible with the Intel 6700PXH 64-bit PCI Hub SMBus configuration access mechanism. The NB/XMB do not require the Intel 6700PXH 64-bit PCI Hub memory read/write because all registers (including the Intel 6700PXH 64-bit PCI Hub registers) can be accessed through the configuration mechanism.

SMBus operations are made up of two major steps:

- Writing information to registers within each component
- Reading configuration registers from each component.

The following sections describe the protocol for an SMBus master to access a NB component's internal configuration registers. For the bus protocol, timings, and waveforms, refer to the SMBus Specification, Revision 2.0.

Each component in the NB platform must have a unique address. NB component addresses are defined in Table 6-42.

**Table 6-42. SMBus Addresses for the Intel® E8501 Chipset Platform**

| Intel® E8501 Chipset Component         | SMBus Address (bits [7:1]) |
|----------------------------------------|----------------------------|
| Intel® E8501 Chipset North Bridge (NB) | 0001100                    |
| eXternal Memory Bridge D (XMB D)       | 0001011                    |
| eXternal Memory Bridge C (XMB C)       | 0001010                    |
| eXternal Memory Bridge B (XMB B)       | 0001001                    |
| eXternal Memory Bridge A (XMB A)       | 0001000                    |
| Devices south of the NB                | 0000000 - 0000111          |

**NOTES:**

1. Bit 0 is not applicable to the address, since it is a R/W bit.
2. The NB does not implement the SMBALERT# pin. The fact that NB uses 0001100 means no slave-only device that requires use of the SMBALERT# signal and the subsequent access to the 0001100 address can be used in the system.

### 6.10.1.1 Supported SMBus Commands

NB components SMBus Rev. 2.0 slave ports support the following six SMBus commands:

- |               |              |              |
|---------------|--------------|--------------|
| • Block Write | • Word Write | • Byte Write |
| • Block Read  | • Word Read  | • Byte Read  |

Sequencing these commands will initiate internal accesses to the component's configuration registers.

Each configuration read or write first consists of an SMBus write sequence which initializes the Bus Number, Device Number, etc. The term sequence is used, since these variables may be written with a single block write or multiple word or byte writes. Once these parameters are initialized, the SMBus master can initiate a read sequence (which perform a configuration read) or a write sequence (which performs a configuration write).

Each SMBus transaction has an 8-bit command driven by the master. The format for this command is illustrated in Table 6-43 below.

**Table 6-43. SMBus Command Encoding**

| 7     | 6   | 5    | 4      | 3:2                                                                                            | 1:0                                                                 |
|-------|-----|------|--------|------------------------------------------------------------------------------------------------|---------------------------------------------------------------------|
| Begin | End | Rsvd | PEC_en | Internal Command:<br>00 - Read DWord<br>01 - Write Byte<br>10 - Write Word<br>11 - Write DWord | SMBus Command:<br>00 - Byte<br>01 - Word<br>10 - Block<br>11 - Rsvd |

The *Begin* bit indicates the first transaction of a read or write sequence.

The *End* bit indicates the last transaction of a read or write sequence.

The *PEC\_en* bit enables the 8-bit PEC generation and checking logic.

The *Internal Command* field specifies the internal command to be issued by the SMBus slave logic. Note that the Internal Command must remain consistent (i.e. not change) during a sequence that accesses a configuration register. Operation cannot be guaranteed if it is not consistent when the command setup sequence is done.

The *SMBus Command* field specifies the SMBus command to be issued on the bus. This field is used as an indication of the length of transfer so the slave knows when to expect the PEC packet (if enabled).

Reserved bits should be written to zero to preserve future compatibility.

### 6.10.1.2 Configuration Register Read Protocol

Configuration reads are accomplished through an SMBus write(s) and later followed by an SMBus read. The write sequence is used to initialize the Bus Number, Device, Function, and Register Number for the configuration access. The writing of this information can be accomplished through any combination of the supported SMBus write commands (Block, Word or Byte). The *Internal Command* field for each write should specify Read DWord.

After all the information is set up, the last write (*End* bit is set) initiates an internal configuration read. If the data is not available before the slave interface acknowledges this last write command (ACK), the slave will “clock stretch” until the data returns to the SMBus interface unit. If an error occurs during the internal access, the last write command will receive a NACK. A status field indicates abnormal termination and contains status information such as target abort, master abort, and time-outs. The status field encoding is defined in Table 6-44.

**Table 6-44. Status Field Encoding for SMBus Reads (Sheet 1 of 2)**

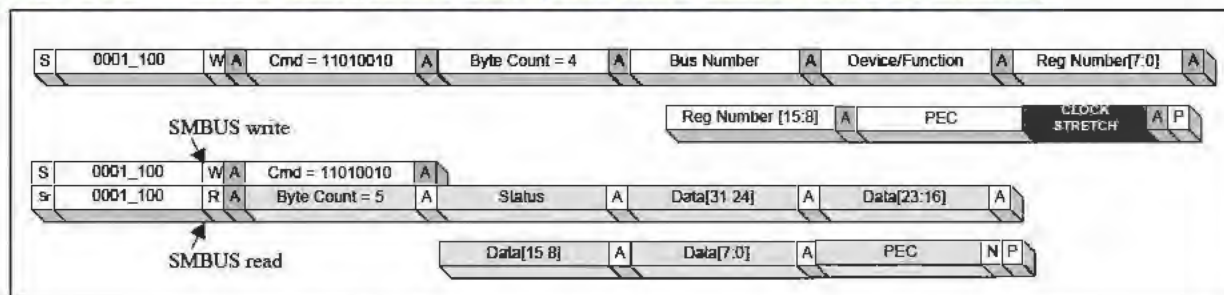
| Bit | Description           |
|-----|-----------------------|
| 7:6 | Reserved              |
| 5   | Internal Master Abort |

Table 6-44. Status Field Encoding for SMBus Reads (Sheet 2 of 2)

| Bit | Description           |
|-----|-----------------------|
| 4   | Internal Target Abort |
| 3:1 | Reserved              |
| 0   | Successful            |

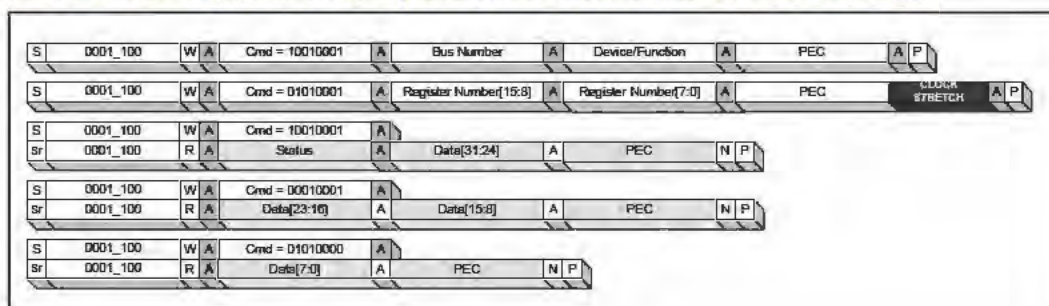
Examples of configuration reads are illustrated below. All of these examples use the NB address and have PEC (Packet Error Code) enabled. Refer to Table 6-42, “SMBus Addresses for the Intel® E8501 Chipset Platform” for other Intel® E8501 chipset component addresses. If the master does not support PEC, then bit 4 of the command would be cleared and there would not be a PEC phase. For the definition of the diagram conventions below, refer to the *SMBus Specification*, Revision 2.0. For SMBus read transactions, the last byte of data (or the PEC byte if enabled) is NACK'ed by the master to indicate the end of the transaction. For diagram compactness, “Register Number[]” is also sometimes referred to as “Reg Number” or “Reg Num”.

Figure 6-17. SMBus Configuration Read (Block Write / Block Read, PEC Enabled)



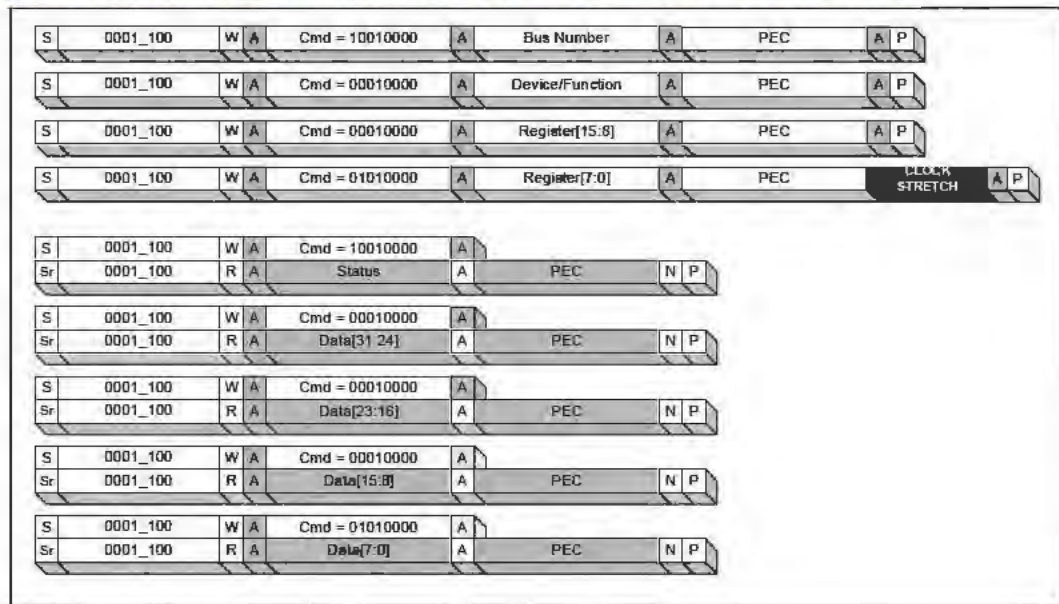
This is an example using word reads. The final data is a byte read.

Figure 6-18. SMBus Configuration Read (Word Writes / Word Reads, PEC Enabled)



The following example uses byte reads.

Figure 6-19. SMBus Configuration Read (Write Bytes / Read Bytes, PEC Enabled)



### 6.10.1.3 Configuration Register Write Protocol

Configuration writes are accomplished through a series of SMBus writes. As with configuration reads, a write sequence is first used to initialize the Bus Number, Device, Function, and Register Number for the configuration access. The writing of this information can be accomplished through any combination of the supported SMBus write commands (Block, Word or Byte).

**Note:** On SMBus, there is no concept of byte enables. Therefore, the Register Number written to the slave is assumed to be aligned to the length of the Internal Command. In other words, for a Write Byte internal command, the Register Number specifies the byte address. For a Write DWord internal command, the two least-significant bits of the Register Number are ignored. This is different from PCI, where the byte enables are used to indicate the byte of interest.

After all the information is set up, the SMBus master initiates one or more writes, which sets up the data to be written. The final write (*End* bit is set) initiates an internal configuration write. The slave interface could potentially clock stretch the last data write until the write completes without error. If an error occurred, the SMBus interface NACK's the last write operation just before the stop bit.

Examples of configuration writes are illustrated below. All of these examples use the NB address. Refer to Table 6-42, "SMBus Addresses for the Intel® E8501 Chipset Platform" for other Intel® E8501 chipset addresses. For the definition of the diagram conventions below, refer to the *SMBus Specification*, Revision 2.0.

Figure 6-20. SMBus Configuration Write (Block Write, PEC Enabled)

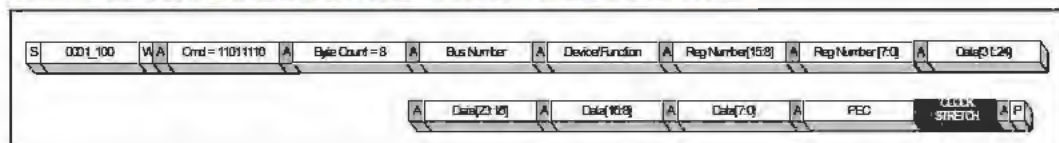


Figure 6-21. SMBus Configuration Write (Word Writes, PEC Enabled)

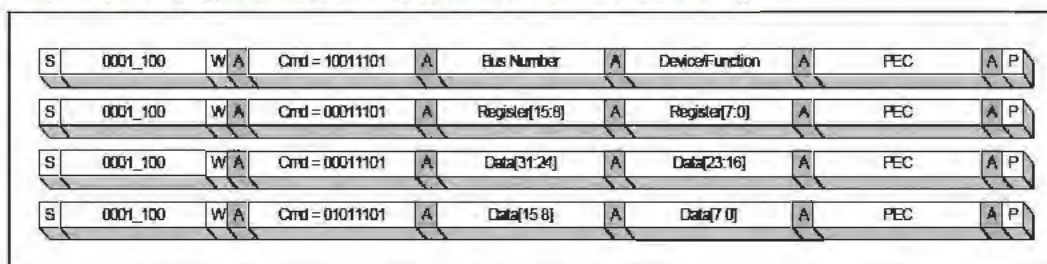
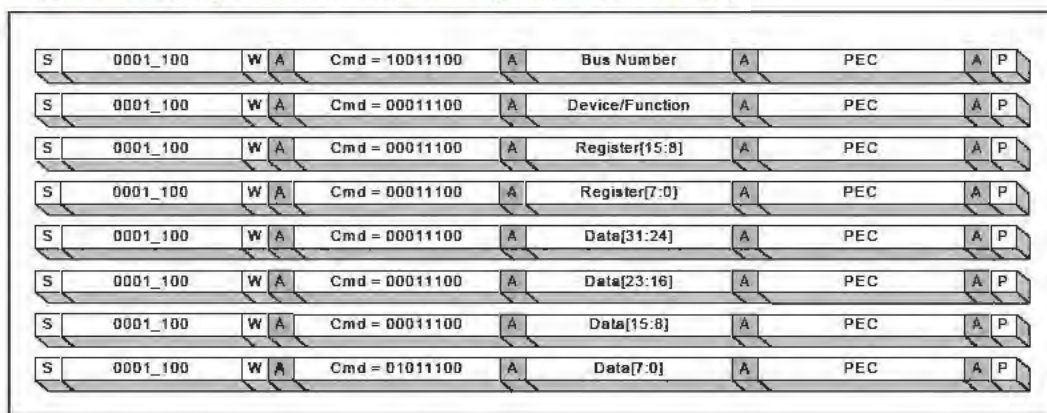


Figure 6-22. SMBus Configuration Write (Write Bytes, PEC Enabled)



#### 6.10.1.4 SMBus Error Handling

The SMBus slave interface handles two types of errors: Internal and PEC. For example, internal errors can occur when the NB issues a configuration read on the PCI Express port Scalability Port and that read terminates in error. These errors manifest as a Not-Acknowledge (NACK) for the read command (*End* bit is set). If an internal error occurs during a configuration write, the final write command receives a NACK just before the stop bit. If the master receives a NACK, the entire configuration transaction should be reattempted.

If the master supports packet error checking (PEC) and the `PEC_en` bit in the command is set, then the PEC byte is checked in the slave interface. If the check indicates a failure, then the slave will NACK the PEC packet.

#### 6.10.1.5 SMBus Interface Reset

The slave interface state machine can be reset by the master in two ways:

- The master holds SCL low for 25ms cumulative. “Cumulative” in this case means that all the “low time” for SCL is counted between the Start and Stop bit. If this totals 25 ms before reaching the Stop bit, the interface is reset.
- The master holds SCL continuously high for 50 ms.

**Note:** Since the configuration registers are affected by the reset pin, SMBus masters will not be able to access the internal registers while the system is reset.

## 6.10.2 JTAG Access

NB chipset provides a JTAG configuration access mechanism that allows a user to access any register in the system. Table 6-45 defines the mapping of registers to JTAG bit positions.

Table 6-45. JTAG Configuration Register Access

| Bit   | Description                                                                                                                                                                    |
|-------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 71:40 | <b>Data</b>                                                                                                                                                                    |
| 39:32 | <b>Register Address[7:0]</b>                                                                                                                                                   |
| 31:29 | <b>Function[2:0]</b>                                                                                                                                                           |
| 28:24 | <b>Device ID[4:0]</b>                                                                                                                                                          |
| 23:16 | <b>Bus number[7:0]</b>                                                                                                                                                         |
| 15:12 | <b>Extended Register Address[11:8]</b>                                                                                                                                         |
| 11:8  | <b>Reserved</b>                                                                                                                                                                |
| 7:4   | <b>Status</b>                                                                                                                                                                  |
| 3:0   | <b>Command:</b><br>0xxx = NOP used in polling the chain to determine if the unit is busy.<br>1001 = write byte<br>1010 = write word<br>1011 = write dword<br>1100 = read dword |

### 6.10.2.1 JTAG Accesses to Registers on Other Components

The Intel® E8501 chipset north bridge (NB) provides functionality that allows accessing registers in the system through the NB JTAG port. The NB will translate these accesses into configuration accesses to the other components. For XMB, ICH5, and PCI Express components, the accesses will be the same as processor initiated register accesses.

§





## 7 *Ballout/Pinout and Package Information*

---

This chapter specifies the ballout/pinout and mechanical specifications for the Intel® E8501 chipset north bridge (NB), and provides references to documents that contain similar information for the remaining core components of the Intel® E8501 chipset, specifically:

**XMB** Intel® E8501 Chipset eXternal Memory Bridge (XMB)

**ICH5** Intel® 82801EB I/O Controller Hub 5 (ICH5)

This information is intended to help with component placement and board routing.

### 7.1 *Intel® E8501 Chipset North Bridge (NB) Ballout/Pinout*

The NB ballout includes the front-side bus (FSB) interfaces, the Independent Memory Interfaces (IMI), the PCI Express interfaces, the Hub 1.5 interface, and the various power, ground, and reference pins.

For detailed information about the XMB ballout and pin list, refer to the *Intel® E8501 Chipset eXternal Memory Bridge (XMB) Datasheet*.

For detailed information about the ICH5 ballout and pin list, refer to the *Intel® 82801EB I/O Controller Hub 5 (ICH5) and Intel® 82801ER I/O Controller Hub 5 (ICH5R) Datasheet*. The ICH5 Datasheet is available at: <http://developer.intel.com/design/chipsets/datashts>.

Figure 7-1. NB Ballout (Top View)

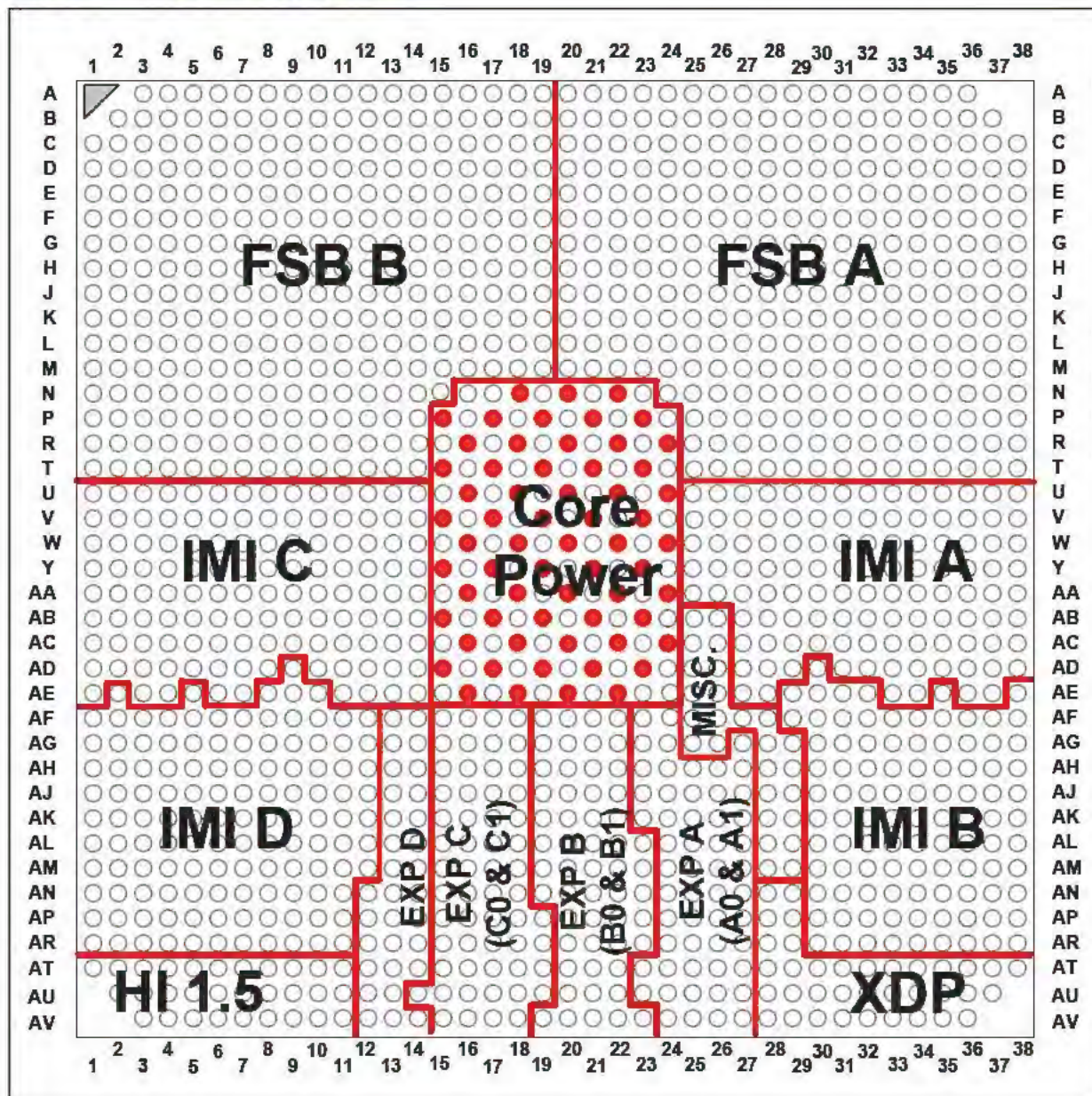


Figure 7-2. NB Ballout with Signal Names (Top View - Left Region)

|    | 1             | 2             | 3          | 4             | 5             | 6            | 7             | 8             | 9             | 10                 | 11                   | 12            | 13           |    |
|----|---------------|---------------|------------|---------------|---------------|--------------|---------------|---------------|---------------|--------------------|----------------------|---------------|--------------|----|
| A  |               |               | VSS        | VSS           | H B D44#      | H B D43#     | VSS           | H B D35#      | H B D39#      | VSS                | H B D32#             | H B D15#      | VSS          | A  |
| B  |               | VSS           | H B D62#   | H B DEP4#     | H B D47#      | VSS          | H B D812#     | H B D39#      | VTT           | H B D37#           | H B D34#             | VSS           | H B DSTBP 0# | B  |
| C  | VSS           | H B D56#      | H B D813#  | H B DEP5#     | VSS           | H B D42#     | H B D40#      | VSS           | H B DSTBP 2#  | H B DSTBN 2#       | VSS                  | H B D14#      | H B D10#     | C  |
| D  | H B D59#      | H B D57#      | H B D61#   | VSS           | H B D46#      | H B D45#     | VTT           | H B D41#      | H B D33#      | VSS                | H B D36#             | H B D12#      | VTT          | D  |
| E  | H B D63#      | H B D58#      | VSS        | H B D59#      | H B D53#      | VSS          | H B D52#      | H B D50#      | VSS           | H B DSTBP 1#       | H B DSTBN 1#         | VSS           | H B D25#     | E  |
| F  | H B DEP6#     | VSS           | H B D62#   | H B D54#      | VTT           | H B D51#     | H B D48#      | VSS           | H B DEP3#     | H B D31#           | VTT                  | H B D32#      | H B D28#     | F  |
| G  | VSS           | H B D7#       | H B DEP7#  | VSS           | H B DSTBP 3#  | H B DSTBN 3# | VSS           | H B D49#      | Reserved      | VSS                | H B DEP2#            | H B D24#      | VSS          | G  |
| H  | H B D4#       | H B D5#       | VTT        | H B A37#      | H B VREF0     | VSS          | H B VREF1     | Reserved      | VTT           | Reserved           | Reserved             | VSS           | H B D6#      | H  |
| J  | H B D1#       | VSS           | H B D0#    | H B REQ4#     | VSS           | H B A3#      | H B REQ1#     | VSS           | H B A18#      | H B A19#           | VSS                  | H B D5#       | H B D0#      | J  |
| K  | VTT           | H B BPR#      | H B D2#    | VSS           | H B REQ2#     | H B REQ3#    | VTT           | H B A24#      | H B ADSTB 1#  | VSS                | H B A39#             | Reserved      | VTT          | K  |
| L  | H B HIT#      | H B DEFER #   | VSS        | H B A7#       | H B A4#       | VSS          | H B REQ0#     | H B A2#       | VSS           | H B A25#           | H B A17#             | VSS           | H B HITM#    | L  |
| M  | H B R51#      | VSS           | H B R52#   | H B A6#       | VTT           | H B A5#      | H B A8#       | VSS           | H B A29#      | H B A23#           | VTT                  | H B BINIT#    | H B AP0#     | M  |
| N  | VSS           | H B BMR#      | H B R50#   | VSS           | H B A9#       | H B ADSTB 0# | VSS           | H B A21#      | H B A30#      | VSS                | H B A22#             | H B AP1#      | VSS          | N  |
| P  | H B TRDY#     | H B SREQ#     | VTT        | H B A36#      | H B A11#      | VSS          | H B A15#      | H B A34#      | VTT           | H B A20#           | H B A26#             | VSS           | H B MCER R#  | P  |
| R  | H B ADS#      | VSS           | H B DRDY#  | H B A16#      | VSS           | H B A12#     | H B A13#      | VSS           | H B A27#      | H B A35#           | VSS                  | H B RSP#      | Reserved     | R  |
| T  | VTT           | H B LOCK#     | H B DBSY#  | VSS           | H B A10#      | H B A14#     | VTT           | H B A33#      | H B A31#      | VSS                | H B A32#             | H B A38#      | VTT          | T  |
| U  | IMI C RXN0    | IMI C ICOM PI | VSS        | IMI C RXN1    | IMI C ICOM PO | VSS          | IMI C RXN8    | Reserved      | VSS           | IMI C RXN5         | IMI C FRA ME         | VSS           | IMI C VCC BG | U  |
| V  | IMI C RXP0    | VSS           | IMI C TXN0 | IMI C RXP1    | P1V5          | IMI C TXN1   | IMI C RXP8    | VSS           | IMI C TXN8    | IMI C RXP3         | P1V5                 | IMI C TXN9    | IMI C VSSB G | V  |
| W  | VSS           | IMI C RXN2    | IMI C TXP0 | VSS           | IMI C RXN3    | IMI C TXP1   | VSS           | IMI C RXN1 0  | IMI C TXP6    | VSS                | IMI C RXN1 1         | IMI C TXP9    | VSS          | W  |
| Y  | IMI C TXN2    | IMI C RXP2    | P1V5       | IMI C TXN3    | IMI C RXP3    | VSS          | IMI C RXN1 2  | IMI C RXP1 0  | P1V5          | IMI C RXN1 3       | IMI C RXP1 1         | VSS           | Reserved     | Y  |
| AA | IMI C TXP2    | VSS           | IMI C RXN4 | IMI C TXP3    | VSS           | IMI C RXN5   | IMI C RXP1 2  | VSS           | IMI C RXN1 4  | IMI C RXP1 3       | VSS                  | IMI C RXN1 5  | Reserved     | AA |
| AB | P1V5          | IMI C TXN4    | IMI C RXP4 | VSS           | IMI C TXN5    | IMI C RXP5   | P1V5          | IMI C RXN1 6  | IMI C RXP1 4  | VSS                | IMI C RXN1 7         | IMI C RXP1 5  | P1V5         | AB |
| AC | IMI C RXN6    | IMI C TXP4    | VSS        | IMI C RXN7    | IMI C TXP5    | VSS          | IMI C LINK N0 | IMI C RXP1 6  | VSS           | IMI C LINK N1      | IMI C RXP1 7         | VSS           | IMI D VCC BG | AC |
| AD | IMI C RXP6    | VSS           | IMI C TXN6 | IMI C RXP7    | P1V5          | IMI C TXN7   | IMI C LINK P0 | VSS           | IMI D RXN1 0  | IMI C LINK P1      | P1V5                 | IMI C LINK N2 | IMI D VSSB G | AD |
| AE | VSS           | IMI D RXN0    | IMI C TXP6 | VSS           | IMI D RXN1    | IMI C TXP7   | VSS           | IMI D RXN1 1  | IMI D RXP1 0  | VSS                | IMI C RST#           | IMI C LINK P2 | VSS          | AE |
| AF | IMI D TXN0    | IMI D RXP0    | P1V5       | IMI D TXN1    | IMI D RXP1    | VSS          | IMI D RXN1 2  | IMI D RXP1 1  | P1V5          | ICHRT#             | IMI D FRA ME         | VSS           | Reserved     | AF |
| AG | IMI D TXP0    | VSS           | IMI D RXN2 | IMI D TXP1    | VSS           | IMI D RXN3   | IMI D RXP1 2  | VSS           | IMI D RXN1 3  | PME_OUTEXPL WIDTH2 | VSS                  | Reserved      | Reserved     | AG |
| AH | P1V5          | IMI D TXN2    | IMI D RXP2 | VSS           | IMI D TXN3    | IMI D RXP3   | P1V5          | IMI D RXN1 4  | IMI D RXP1 3  | VSS                | EXP_HPINTEXP1 WIDTH1 | IMI HPINT     | P1V5         | AH |
| AJ | IMI D RXN4    | IMI D TXP2    | VSS        | IMI D RXN5    | IMI D TXP3    | VSS          | IMI D RXN1 5  | IMI D RXP1 4  | VSS           | V3REF1             | Reserved             | VSS           | EXP D RXP 0  | AJ |
| AK | IMI D RXP4    | VSS           | IMI D TXN4 | IMI D RXP5    | P1V5          | IMI D TXN5   | IMI D RXP1 5  | VSS           | IMI D RXN1 6  | Reserved           | P1V5                 | EXP1 SPEC 10A | EXP D RXN 0  | AK |
| AL | VSS           | IMI D RXN6    | IMI D TXP4 | VSS           | IMI D RXN7    | IMI D TXP5   | VSS           | IMI D RXN1 7  | IMI D RXP1 6  | VSS                | Reserved             | Reserved      | VSS          | AL |
| AM | IMI D TXN6    | IMI D RXP6    | P1V5       | IMI D TXN7    | IMI D RXP7    | VSS          | IMI D LINK N0 | IMI D RXP1 7  | P1V5          | Reserved           | Reserved             | VSS           | EXP D RXP 1  | AM |
| AN | IMI D TXP6    | VSS           | IMI D RXN8 | IMI D TXP7    | VSS           | IMI D RXN9   | IMI D LINK P0 | VSS           | IMI D LINK N1 | Reserved           | VSS                  | EXP D TXP 2   | EXP D RXN 1  | AN |
| AP | P1V5          | IMI D TXN8    | IMI D RXP8 | VSS           | IMI D TXN9    | IMI D RXP9   | P1V5          | IMI D LINK N2 | IMI D LINK P1 | VSS                | Reserved             | EXP D TXN 2   | P1V5         | AP |
| AR | IMI D ICOM PI | IMI D TXP8    | VSS        | IMI D ICOM PO | IMI D TXP9    | VSS          | IMI D RST#    | IMI D LINK P2 | VSS           | Reserved           | Reserved             | VSS           | Reserved     | AR |
| AT | VSS           | HI11          | HI0        | HI3           | P1V5          | HI10         | HI5           | VSS           | HIRCOMP       | Reserved           | P1V5                 | EXP D TXP 3   | EXP D TXN 3  | AT |
| AU |               | VSS           | HI1        | HI2           | HI8           | HI9          | VSS           | HI7           | HIVSWING      | VSS                | Reserved             | Reserved      | VSS          | AU |
| AV |               |               | VSS        | HI STBS       | HI STBF       | VSS          | HI4           | HI6           | P1V5          | HIVREF             | Reserved             | VSS           | EXP D RXP 3  | AV |
|    | 1             | 2             | 3          | 4             | 5             | 6            | 7             | 8             | 9             | 10                 | 11                   | 12            | 13           |    |

Figure 7-3. NB Ballout with Signal Names (Top View - Center Region)

|    | 14           | 15           | 16           | 17           | 18           | 19           | 20           | 21           | 22           | 23           | 24           | 25           |    |
|----|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|----|
| A  | H B D11#     | H B D9#      | VSS          | H B D8#      | H B D4#      | VSS          | H A D46#     | H A D45#     | VSS          | H A D41#     | H A D39#     | VSS          | A  |
| B  | H B D5TB#    | VTT          | H B D6#      | H B D0#      | VSS          | H B DEP1#    | H A DEP4#    | VTT          | H A D43#     | H A D82#     | VSS          | H A D37#     | B  |
| C  | VSS          | H B D5#      | H B D3#      | VSS          | H B D1#      | H B DEP0#    | VSS          | H A D44#     | H A D42#     | VSS          | H A D5TB#    | H A D5TB#    | C  |
| D  | H B D13#     | H B D2#      | VSS          | H B D7#      | H B D810#    | VTT          | H A DEP5#    | H A D47#     | VSS          | H A D40#     | H A D38#     | VTT          | D  |
| E  | H B D26#     | VSS          | H B D22#     | H B D21#     | VSS          | H B D17#     | H A DEP7#    | VSS          | H A D59#     | H A D57#     | VSS          | H A D62#     | E  |
| F  | VSS          | H B D27#     | H B D23#     | VTT          | H B D18#     | H B D16#     | VSS          | H A D60#     | H A D58#     | VTT          | H A D61#     | H A D54#     | F  |
| G  | H B D29#     | H B D811#    | VSS          | H B D20#     | H B D19#     | VSS          | H A DEP6#    | H A D63#     | VSS          | H A D56#     | H A D83#     | VSS          | G  |
| H  | H B DP1#     | VTT          | H B DP0#     | H B VREF2    | VSS          | Reserved     | Reserved     | VTT          | H A VREF1    | H A DP0#     | VSS          | H A DP1#     | H  |
| J  | VSS          | H B DP2#     | H B DP3#     | VSS          | TESTLO4      | P1V5         | VSS          | HCLKIN#      | HCLKINP      | VSS          | H A DP2#     | H A BPM5#    | J  |
| K  | EXP1 WIDT#   | H B D3#      | VSS          | H VSSA       | H VCCA       | VTT          | VSSA         | VCCA         | VSS          | H A DP3#     | H A BPM4#    | VTT          | K  |
| L  | H B BREQ1#   | VSS          | H SLOWCRE#   | H CRE#       | VSS          | Reserved     | TESTLO3      | VSS          | TESTH1       | TESTLO2      | VSS          | H A AP1#     | L  |
| M  | VSS          | H B BPM4#    | H ODTCRE#    | VTT          | TESTLO1      | Reserved     | VSS          | H A PRSNT#   | H B PRSNT#   | VTT          | TESTLO5      | H A BREQ1#   | M  |
| N  | H B RST#     | H B BPM5#    | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | H A OOD#     | VSS          | N  |
| P  | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | P1V5         | P  |
| R  | P1V5         | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | R  |
| T  | IMI C CLKP   | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | P1V5         | T  |
| U  | IMI C CLKN   | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | IMI A CLKP   | U  |
| V  | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | IMI A VCCA   | V  |
| W  | IMI_C_VSSA   | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | IMI_A_VSSA   | W  |
| Y  | IMI_C_VCCA   | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | IMI_A_VSSB   | Y  |
| AA | IMI D CLKP   | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | IMI A VCCB   | AA |
| AB | IMI D CLKN   | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | IMI B VCCA   | AB |
| AC | IMI_D_VSSA   | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | IMI_B_VSSA   | AC |
| AD | IMI_D_VCCA   | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | VSS          | IMI B CLKN   | AD |
| AE | Reserved     | VSS          | P1V5         | VSS          | P1V5         | VSS          | P1V5         | VSS          | P1V5         | VSS          | EXP0 WIDT#   | IMI B CLKP   | AE |
| AF | Reserved     | Reserved     | CLK66        | HI_VCCA      | HI_VSSA      | EXP_VSSBG    | EXP CLKP     | EXP CLKN     | EXP_VCCA     | IMI B VSSB   | IMI B VCCBG  | SMBCLK       | AF |
| AG | VSS          | EXP C0 TX P0 | Reserved     | VSS          | EXP C1 RX N3 | EXP_VCCBG    | VSS          | EXP B1 RX N0 | EXP_VSSA     | VSS          | EXP A0 TX P0 | GP SMBDATA   | AG |
| AH | EXP D TXN0   | EXP C0 TX N0 | VSS          | EXP C0 RX P0 | EXP C1 RX P3 | P1V5         | EXP B0 TX P0 | EXP B1 RX P3 | VSS          | EXP A0 RX P0 | EXP A0 TX N0 | P1V5         | AH |
| AJ | EXP D TXN1   | VSS          | EXP C0 TX P1 | EXP C0 RX N0 | VSS          | EXP B0 RX P0 | EXP B0 TX N0 | VSS          | EXP B1 TX N3 | EXP A0 RX N0 | VSS          | EXP A0 TX P1 | AJ |
| AK | VSS          | EXP C0 RX P1 | EXP C0 TX N1 | P1V5         | EXP C1 TX N3 | EXP B0 RX N0 | VSS          | EXP B0 TX P1 | EXP B1 TX P3 | P1V5         | EXP A0 RX P1 | EXP A0 TX N1 | AK |
| AL | EXP D TXN1   | EXP C0 RX N1 | VSS          | EXP C1 RX N2 | EXP C1 TX P3 | VSS          | EXP B0 RX P1 | EXP B0 TX N1 | VSS          | EXP B1 RX N2 | EXP A0 RX N1 | VSS          | AL |
| AM | EXP D TXN2   | P1V5         | EXP C0 TX P2 | EXP C1 RX P2 | VSS          | EXP B0 TX P2 | EXP B0 RX N1 | P1V5         | EXP B1 TX N2 | EXP B1 RX P2 | VSS          | EXP A0 TX P2 | AM |
| AN | VSS          | EXP C0 RX P2 | EXP C0 TX N2 | VSS          | EXP C1 TX N2 | EXP B0 TX N2 | VSS          | EXP B1 RX N1 | EXP B1 TX P2 | VSS          | EXP A0 RX P2 | EXP A0 TX N2 | AN |
| AP | EXP D RXN2   | EXP C0 RX N2 | VSS          | EXP C1 TX N1 | EXP C1 TX P2 | P1V5         | EXP B0 RX P2 | EXP B1 RX P1 | VSS          | EXP B1 TX N1 | EXP A0 RX N2 | P1V5         | AP |
| AR | EXP D RXN2   | VSS          | EXP ICOMP1   | EXP C1 TX P1 | VSS          | EXP C1 RX N1 | EXP B0 RX N2 | VSS          | EXP RCOMP    | EXP B1 TX P1 | VSS          | ITP TRST#    | AR |
| AT | VSS          | EXP C0 RX P3 | EXP C0 RX N3 | P1V5         | Reserved     | EXP C1 RX P1 | VSS          | EXP B1 TX P0 | EXP B1 TX N0 | P1V5         | EXP A0 RX P3 | EXP A0 RX N3 | AT |
| AU | EXP C0 TX P3 | EXP C0 TX N3 | VSS          | EXP C1 RX P0 | EXP C1 RX N0 | VSS          | EXP B0 RX P3 | EXP B0 RX N0 | VSS          | EXP A0 TX P3 | EXP A0 TX N3 | VSS          | AU |
| AV | EXP D RXN3   | P1V5         | EXP C1 TX P0 | EXP C1 TX N0 | VSS          | EXP B0 TX P3 | EXP B0 TX N3 | P1V5         | EXP B1 RX P0 | EXP B1 RX N0 | VSS          | EXP A1 TX P0 | AV |
|    | 14           | 15           | 16           | 17           | 18           | 19           | 20           | 21           | 22           | 23           | 24           | 25           |    |

Figure 7-4. NB Ballout with Signal Names (Top View - Right Region)

|    | 26           | 27            | 28            | 29           | 30            | 31            | 32            | 33           | 34            | 35            | 36         | 37            | 38            |    |
|----|--------------|---------------|---------------|--------------|---------------|---------------|---------------|--------------|---------------|---------------|------------|---------------|---------------|----|
| A  | H A D32#     | H A D15#      | VSS           | H A D11#     | H A D10#      | VSS           | H A D6#       | H A D0#      | VSS           | H A DEP1#     | VSS        |               |               | A  |
| B  | H A D33#     | VTT           | H A DSTB0#    | H A DSTB0#   | VSS           | H A D5#       | H A D3#       | VTT          | H A D1#       | H A DEP0#     | H A D26#   | VSS           |               | B  |
| C  | VSS          | H A D36#      | H A D12#      | VSS          | H A D13#      | H A D2#       | VSS           | H A D7#      | H A D8#       | VSS           | H A D81#   | H A D23#      | VSS           | C  |
| D  | H A D35#     | H A D34#      | VSS           | H A D14#     | H A D9#       | VTT           | H A D8#       | H A D4#      | VSS           | Reserved      | H A D27#   | H A D22#      | H A D21#      | D  |
| E  | H A D55#     | VSS           | H A D51#      | H A D52#     | VSS           | H A DEP3#     | H A D31#      | VSS          | H A D20#      | H A D25#      | VSS        | H A D20#      | H A D19#      | E  |
| F  | VSS          | H A DSTB3#    | H A DSTB3#    | VTT          | H A D50#      | H A DEP2#     | VSS           | H A D24#     | H A D30#      | VTT           | H A D18#   | H A D16#      | VSS           | F  |
| G  | H A D53#     | H A D48#      | VSS           | H A D45#     | Reserved      | VSS           | H A DSTB1#    | H A DSTB1#   | VSS           | H A D25#      | H A D17#   | VSS           | H A LOCK#     | G  |
| H  | Reserved     | VTT           | Reserved      | Reserved     | VSS           | H A VREF2     | Reserved      | VTT          | H A VREF0     | H A A14#      | VSS        | H A DRDY#     | H A DBSY#     | H  |
| J  | VSS          | H A R5P#      | H A A38#      | VSS          | H A A31#      | H A A33#      | VSS           | H A A10#     | H A A13#      | VSS           | H A TRDY#  | H A ADS#      | VSS           | J  |
| K  | H A MCERR#   | H A RST#      | VSS           | H A A35#     | H A A27#      | VTT           | H A A12#      | H A A15#     | VSS           | H A A16#      | H A BREQ#  | VTT           | H A BNR#      | K  |
| L  | H A AP0#     | VSS           | H A A32#      | H A A26#     | VSS           | H A A34#      | H A ADSTB0#   | VSS          | H A A11#      | H A A36#      | VSS        | H A RS2#      | H A RS0#      | L  |
| M  | VSS          | H A BINT#     | H A A20#      | VTT          | H A A21#      | H A A30#      | VSS           | H A A8#      | H A A9#       | VTT           | H A HIT#   | H A RS1#      | VSS           | M  |
| N  | H A HIT#     | H A D3#       | VSS           | H A A22#     | H A A29#      | VSS           | H A REQ0#     | H A A6#      | VSS           | H A A6#       |            | VSS           | H A BPR#      | N  |
| P  | Reserved     | VTT           | H A A25#      | H A A23#     | VSS           | H A A28#      | H A REQ3#     | VTT          | H A A4#       | H A A7#       | VSS        | H A ID0#      | H A ID2#      | P  |
| R  | VSS          | H A ID5#      | H A A17#      | VSS          | H A ADSTB1#   | H A A24#      | VSS           | H A REQ1#    | H A REQ2#     | VSS           | H A ID4#   | H A ID1#      | VSS           | R  |
| T  | H A ID6#     | H A A39#      | VSS           | H A A19#     | H A A18#      | VTT           | H A A37#      | H A A3#      | VSS           | H A REQ4#     | H A ID5#   | VTT           | H A ID7#      | T  |
| U  | IMI A CLK#   | VSS           | IMI A LINK P2 | Reserved     | VSS           | IMI A LINK P1 | IMI A RST#    | VSS          | IMI A TXP7    | IMI A ICOM P1 | VSS        | IMI A TXP5    | IMI A ICOM P0 | U  |
| V  | VSS          | IMI A LINK P0 | IMI A LINK N2 | P1V5         | IMI A RXP1 7  | IMI A LINK N1 | VSS           | IMI A RXP7   | IMI A TXN7    | P1V5          | IMI A RXP6 | IMI A TXN6    | VSS           | V  |
| W  | Reserved     | IMI A LINK N0 | VSS           | IMI A RXP1 6 | IMI A RXN1 7  | VSS           | IMI A RXP1 5  | IMI A RXN1 5 | VSS           | IMI A TXP5    | IMI A RXN6 | VSS           | IMI A TXP4    | W  |
| Y  | Reserved     | P1V5          | IMI A RXP1 4  | IMI A RXN1 6 | VSS           | IMI A RXP1 3  | IMI A RXN1 5  | P1V5         | IMI A RXP5    | IMI A TXN5    | VSS        | IMI A RXP4    | IMI A TXN4    | Y  |
| AA | VSS          | IMI A RXP1 2  | IMI A RXN1 4  | VSS          | IMI A RXP1 1  | IMI A RXN1 3  | VSS           | IMI A TXP3   | IMI A RXN5    | VSS           | IMI A TXP2 | IMI A RXN4    | VSS           | AA |
| AB | Reserved     | IMI A RXN1 2  | VSS           | IMI A RXP1 0 | IMI A RXN1 1  | P1V5          | IMI A TXP8    | IMI A TXN3   | VSS           | IMI A RXP0    | IMI A TXN2 | P1V5          | IMI A RXP2    | AB |
| AC | Reserved     | VSS           | IMI A TXP9    | IMI A RXN1 0 | VSS           | IMI A RXP8    | IMI A TXN8    | VSS          | IMI A TXP1    | IMI A RXN3    | VSS        | IMI A TXP0    | IMI A RXN2    | AC |
| AD | VSS          | IMI A RXP9    | IMI A TXN9    | P1V5         | IMI B LINK P2 | IMI A RXN8    | VSS           | IMI A RXP1   | IMI A TXN1    | P1V5          | IMI A RXP0 | IMI A TXN0    | VSS           | AD |
| AE | GP SMBCLK    | IMI A RXN9    | VSS           | IMI B RST#   | IMI B LINK N2 | VSS           | IMI B LINK P1 | IMI A RXN1   | VSS           | IMI B TXP9    | IMI A RXN0 | VSS           | IMI B TXP8    | AE |
| AF | V3REF0       | ITP TCK       | ITP TDI       | IME          | VSS           | IMI B LINK P0 | IMI B LINK N1 | P1V5         | IMI B RXP9    | IMI B TXN9    | VSS        | IMI B RXP8    | IMI B TXN8    | AF |
| AG | VSS          | EXP A1 RX N3  | PWRGOOD       | VSS          | IMI B RXP1 7  | IMI B LINK N0 | VSS           | IMI B TXP7   | IMI B RXN9    | VSS           | IMI B TXP6 | IMI B RXN8    | VSS           | AG |
| AH | EXP A1 TX N3 | EXP A1 RX P3  | ITP TMS       | RSTIN#       | IMI B RXN1 7  | P1V5          | IMI B RXP1 6  | IMI B TXN7   | VSS           | IMI B RXP7    | IMI B TXN6 | P1V5          | IMI B RXP6    | AH |
| AJ | EXP A1 TX P3 | VSS           | Reserved      | Reserved     | VSS           | IMI B RXP1 5  | IMI B RXN1 6  | VSS          | IMI B TXP5    | IMI B RXN7    | VSS        | IMI B TXP4    | IMI B RXN6    | AJ |
| AK | VSS          | EXP A1 RX N2  | ERR0#         | P1V5         | IMI B RXP1 4  | IMI B RXN1 5  | VSS           | IMI B RXP5   | IMI B TXN5    | P1V5          | IMI B RXP4 | IMI B TXN4    | VSS           | AK |
| AL | EXP A1 TX N2 | EXP A1 RX P2  | VSS           | ERR1#        | IMI B RXN1 4  | VSS           | IMI B RXP1 3  | IMI B RXN5   | VSS           | IMI B TXP3    | IMI B RXN4 | VSS           | IMI B TXP2    | AL |
| AM | EXP A1 TX P2 | P1V5          | EXP0 SPEC 10A | ERR2#        | VSS           | IMI B RXP1 2  | IMI B RXN1 3  | P1V5         | IMI B RXP3    | IMI B TXN3    | VSS        | IMI B RXP2    | IMI B TXN2    | AM |
| AN | VSS          | EXP A1 RX N1  | XDP ODT#      | RES          | VSS           | IMI B RXP1 1  | IMI B RXN1 2  | VSS          | IMI B TXP1    | IMI B RXN3    | VSS        | IMI B TXP0    | IMI B RXN2    | AN |
| AP | EXP A1 TX N1 | EXP A1 RX P1  | VSS           | XDP SLWC#    | RES           | IMI B RXN1 1  | P1V5          | IMI B TXN1   | VSS           | IMI B RXP1    | IMI B TXN0 | P1V5          | IMI B RXP0    | AP |
| AR | EXP A1 TX P1 | VSS           | SMBDATA       | XDP CRES     | VSS           | IMI B FRA ME  | IMI B RXN1 0  | VSS          | IMI B ICOM P1 | IMI B RXN1    | VSS        | IMI B ICOM P0 | IMI B RXN0    | AR |
| AT | VSS          | EXP0_WIDTH1   | XDP D15#      | P1V5         | XDP D13#      | XDP D12#      | VSS           | XDP D10#     | XDP D9#       | VSS           | XDP DSTB#  | XDP DSTB#     | VSS           | AT |
| AU | EXP A1 RX P0 | EXP A1 RX N0  | ITP TDO       | XDP D14#     | XDP D5#       | P1V5          | XDP D11#      | XDP D2#      | P1V5          | XDP D8#       | XDP RDY#   | VSS           |               | AU |
| AV | EXP A1 TX N0 | P1V5          | XDP D7#       | XDP D6#      | VSS           | XDP D4#       | XDP D3#       | VSS          | XDP D1#       | XDP D0#       | VSS        |               |               | AV |
|    | 26           | 27            | 28            | 29           | 30            | 31            | 32            | 33           | 34            | 35            | 36         | 37            | 38            |    |

Table 7-1. NB Pin List (by Ball Number) (Sheet 1 of 13)

| Ball No. | Signal    | Ball No. | Signal      | Ball No. | Signal      |
|----------|-----------|----------|-------------|----------|-------------|
| A3       | VSS       | B6       | VSS         | C7       | H_B_D40#    |
| A4       | VSS       | B7       | H_B_DBI2#   | C8       | VSS         |
| A5       | H_B_D44#  | B8       | H_B_D38#    | C9       | H_B_DSTBP2# |
| A6       | H_B_D43#  | B9       | VTT         | C10      | H_B_DSTBN2# |
| A7       | VSS       | B10      | H_B_D37#    | C11      | VSS         |
| A8       | H_B_D35#  | B11      | H_B_D34#    | C12      | H_B_D14#    |
| A9       | H_B_D39#  | B12      | VSS         | C13      | H_B_D10#    |
| A10      | VSS       | B13      | H_B_DSTBP0# | C14      | VSS         |
| A11      | H_B_D32#  | B14      | H_B_DSTBN0# | C15      | H_B_D5#     |
| A12      | H_B_D15#  | B15      | VTT         | C16      | H_B_D3#     |
| A13      | VSS       | B16      | H_B_D6#     | C17      | VSS         |
| A14      | H_B_D11#  | B17      | H_B_D0#     | C18      | H_B_D1#     |
| A15      | H_B_D9#   | B18      | VSS         | C19      | H_B_DEP0#   |
| A16      | VSS       | B19      | H_B_DEP1#   | C20      | VSS         |
| A17      | H_B_D8#   | B20      | H_A_DEP4#   | C21      | H_A_D44#    |
| A18      | H_B_D4#   | B21      | VTT         | C22      | H_A_D42#    |
| A19      | VSS       | B22      | H_A_D43#    | C23      | VSS         |
| A20      | H_A_D46#  | B23      | H_A_DBI2#   | C24      | H_A_DSTBP2# |
| A21      | H_A_D45#  | B24      | VSS         | C25      | H_A_DSTBN2# |
| A22      | VSS       | B25      | H_A_D37#    | C26      | VSS         |
| A23      | H_A_D41#  | B26      | H_A_D33#    | C27      | H_A_D36#    |
| A24      | H_A_D39#  | B27      | VTT         | C28      | H_A_D12#    |
| A25      | VSS       | B28      | H_A_DSTBP0# | C29      | VSS         |
| A26      | H_A_D32#  | B29      | H_A_DSTBN0# | C30      | H_A_D13#    |
| A27      | H_A_D15#  | B30      | VSS         | C31      | H_A_D2#     |
| A28      | VSS       | B31      | H_A_D5#     | C32      | VSS         |
| A29      | H_A_D11#  | B32      | H_A_D3#     | C33      | H_A_D7#     |
| A30      | H_A_D10#  | B33      | VTT         | C34      | H_A_DBI0#   |
| A31      | VSS       | B34      | H_A_D1#     | C35      | VSS         |
| A32      | H_A_D6#   | B35      | H_A_DEP0#   | C36      | H_A_DBI1#   |
| A33      | H_A_D0#   | B36      | H_A_D26#    | C37      | H_A_D23#    |
| A34      | VSS       | B37      | VSS         | C38      | VSS         |
| A35      | H_A_DEP1# | C1       | VSS         | D1       | H_B_D59#    |
| A36      | VSS       | C2       | H_B_D56#    | D2       | H_B_D57#    |
| B2       | VSS       | C3       | H_B_DBI3#   | D3       | H_B_D61#    |
| B3       | H_B_D62#  | C4       | H_B_DEP5#   | D4       | VSS         |
| B4       | H_B_DEP4# | C5       | VSS         | D5       | H_B_D46#    |
| B5       | H_B_D47#  | C6       | H_B_D42#    | D6       | H_B_D45#    |

Table 7-1. NB Pin List (by Ball Number) (Sheet 2 of 13)

| Ball No. | Signal    | Ball No. | Signal      | Ball No. | Signal      |
|----------|-----------|----------|-------------|----------|-------------|
| D7       | VTT       | E7       | H_B_D52#    | F7       | H_B_D48#    |
| D8       | H_B_D41#  | E8       | H_B_D50#    | F8       | VSS         |
| D9       | H_B_D33#  | E9       | VSS         | F9       | H_B_DEP3#   |
| D10      | VSS       | E10      | H_B_DSTBP1# | F10      | H_B_D31#    |
| D11      | H_B_D36#  | E11      | H_B_DSTBN1# | F11      | VTT         |
| D12      | H_B_D12#  | E12      | VSS         | F12      | H_B_D30#    |
| D13      | VTT       | E13      | H_B_D25#    | F13      | H_B_D28#    |
| D14      | H_B_D13#  | E14      | H_B_D26#    | F14      | VSS         |
| D15      | H_B_D2#   | E15      | VSS         | F15      | H_B_D27#    |
| D16      | VSS       | E16      | H_B_D22#    | F16      | H_B_D23#    |
| D17      | H_B_D7#   | E17      | H_B_D21#    | F17      | VTT         |
| D18      | H_B_DB10# | E18      | VSS         | F18      | H_B_D18#    |
| D19      | VTT       | E19      | H_B_D17#    | F19      | H_B_D16#    |
| D20      | H_A_DEP5# | E20      | H_A_DEP7#   | F20      | VSS         |
| D21      | H_A_D47#  | E21      | VSS         | F21      | H_A_D60#    |
| D22      | VSS       | E22      | H_A_D59#    | F22      | H_A_D58#    |
| D23      | H_A_D40#  | E23      | H_A_D57#    | F23      | VTT         |
| D24      | H_A_D38#  | E24      | VSS         | F24      | H_A_D61#    |
| D25      | VTT       | E25      | H_A_D62#    | F25      | H_A_D54#    |
| D26      | H_A_D35#  | E26      | H_A_D55#    | F26      | VSS         |
| D27      | H_A_D34#  | E27      | VSS         | F27      | H_A_DSTBP3# |
| D28      | VSS       | E28      | H_A_D51#    | F28      | H_A_DSTBN3# |
| D29      | H_A_D14#  | E29      | H_A_D52#    | F29      | VTT         |
| D30      | H_A_D9#   | E30      | VSS         | F30      | H_A_D50#    |
| D31      | VTT       | E31      | H_A_DEP3#   | F31      | H_A_DEP2#   |
| D32      | H_A_D8#   | E32      | H_A_D31#    | F32      | VSS         |
| D33      | H_A_D4#   | E33      | VSS         | F33      | H_A_D24#    |
| D34      | VSS       | E34      | H_A_D28#    | F34      | H_A_D30#    |
| D35      | Reserved  | E35      | H_A_D25#    | F35      | VTT         |
| D36      | H_A_D27#  | E36      | VSS         | F36      | H_A_D18#    |
| D37      | H_A_D22#  | E37      | H_A_D20#    | F37      | H_A_D16#    |
| D38      | H_A_D21#  | E38      | H_A_D19#    | F38      | VSS         |
| E1       | H_B_D63#  | F1       | H_B_DEP6#   | G1       | VSS         |
| E2       | H_B_D58#  | F2       | VSS         | G2       | H_B_ID7#    |
| E3       | VSS       | F3       | H_B_D60#    | G3       | H_B_DEP7#   |
| E4       | H_B_D55#  | F4       | H_B_D54#    | G4       | VSS         |
| E5       | H_B_D53#  | F5       | VTT         | G5       | H_B_DSTBP3# |
| E6       | VSS       | F6       | H_B_D51#    | G6       | H_B_DSTBN3# |

Table 7-1. NB Pin List (by Ball Number) (Sheet 3 of 13)

| Ball No. | Signal      | Ball No. | Signal    | Ball No. | Signal    |
|----------|-------------|----------|-----------|----------|-----------|
| G7       | VSS         | H7       | H_B_VREF1 | J7       | H_B_REQ1# |
| G8       | H_B_D49#    | H8       | Reserved  | J8       | VSS       |
| G9       | RESERVED    | H9       | VTT       | J9       | H_B_A18#  |
| G10      | VSS         | H10      | Reserved  | J10      | H_B_A19#  |
| G11      | H_B_DEP2#   | H11      | Reserved  | J11      | VSS       |
| G12      | H_B_D24#    | H12      | VSS       | J12      | H_B_IDS#  |
| G13      | VSS         | H13      | H_B_ID6#  | J13      | H_B_OOD#  |
| G14      | H_B_D29#    | H14      | H_B_DP1#  | J14      | VSS       |
| G15      | H_B_DBI1#   | H15      | VTT       | J15      | H_B_DP2#  |
| G16      | VSS         | H16      | H_B_DP0#  | J16      | H_B_DP3#  |
| G17      | H_B_D20#    | H17      | H_B_VREF2 | J17      | VSS       |
| G18      | H_B_D19#    | H18      | VSS       | J18      | TESTLO4   |
| G19      | VSS         | H19      | Reserved  | J19      | P1V5      |
| G20      | H_A_DEP6#   | H20      | Reserved  | J20      | VSS       |
| G21      | H_A_D63#    | H21      | VTT       | J21      | HCLKINN   |
| G22      | VSS         | H22      | H_A_VREF1 | J22      | HCLKINP   |
| G23      | H_A_D56#    | H23      | H_A_DP0#  | J23      | VSS       |
| G24      | H_A_DBI3#   | H24      | VSS       | J24      | H_A_DP2#  |
| G25      | VSS         | H25      | H_A_DP1#  | J25      | H_A_BPM5# |
| G26      | H_A_D53#    | H26      | Reserved  | J26      | VSS       |
| G27      | H_A_D48#    | H27      | VTT       | J27      | H_A_RSP#  |
| G28      | VSS         | H28      | Reserved  | J28      | H_A_A38#  |
| G29      | H_A_D49#    | H29      | Reserved  | J29      | VSS       |
| G30      | Reserved    | H30      | VSS       | J30      | H_A_A31#  |
| G31      | VSS         | H31      | H_A_VREF2 | J31      | H_A_A33#  |
| G32      | H_A_DSTBP1# | H32      | Reserved  | J32      | VSS       |
| G33      | H_A_DSTBN1# | H33      | VTT       | J33      | H_A_A10#  |
| G34      | VSS         | H34      | H_A_VREF0 | J34      | H_A_A13#  |
| G35      | H_A_D29#    | H35      | H_A_A14#  | J35      | VSS       |
| G36      | H_A_D17#    | H36      | VSS       | J36      | H_A_TRDY# |
| G37      | VSS         | H37      | H_A_DRDY# | J37      | H_A_ADS#  |
| G38      | H_A_LOCK#   | H38      | H_A_DBSY# | J38      | VSS       |
| H1       | H_B_ID4#    | J1       | H_B_ID1#  | K1       | VTT       |
| H2       | H_B_ID5#    | J2       | VSS       | K2       | H_B_BPRI# |
| H3       | VTT         | J3       | H_B_ID0#  | K3       | H_B_ID2#  |
| H4       | H_B_A37#    | J4       | H_B_REQ4# | K4       | VSS       |
| H5       | H_B_VREF0   | J5       | VSS       | K5       | H_B_REQ2# |
| H6       | VSS         | J6       | H_B_A3#   | K6       | H_B_REQ3# |

Table 7-1. NB Pin List (by Ball Number) (Sheet 4 of 13)

| Ball No. | Signal      | Ball No. | Signal      | Ball No. | Signal      |
|----------|-------------|----------|-------------|----------|-------------|
| K7       | VTT         | L7       | H_B_REQ0#   | M7       | H_B_A8#     |
| K8       | H_B_A24#    | L8       | H_B_A28#    | M8       | VSS         |
| K9       | H_B_ADSTB1# | L9       | VSS         | M9       | H_B_A29#    |
| K10      | VSS         | L10      | H_B_A25#    | M10      | H_B_A23#    |
| K11      | H_B_A39#    | L11      | H_B_A17#    | M11      | VTT         |
| K12      | Reserved    | L12      | VSS         | M12      | H_B_BINIT#  |
| K13      | VTT         | L13      | H_B_HITM#   | M13      | H_B_AP0#    |
| K14      | EXP1_WIDTH0 | L14      | H_B_BREQ1#  | M14      | VSS         |
| K15      | H_B_ID3#    | L15      | VSS         | M15      | H_B_BPM4#   |
| K16      | VSS         | L16      | H_SLWCRES   | M16      | H_ODTCRES   |
| K17      | H_VSSA      | L17      | H_CRES      | M17      | VTT         |
| K18      | H_VCCA      | L18      | VSS         | M18      | TESTLO1     |
| K19      | VTT         | L19      | Reserved    | M19      | Reserved    |
| K20      | VSSA        | L20      | TESTLO3     | M20      | VSS         |
| K21      | VCCA        | L21      | VSS         | M21      | H_A_PRSENT# |
| K22      | VSS         | L22      | TESTHI1     | M22      | H_B_PRSENT# |
| K23      | H_A_DP3#    | L23      | TESTLO2     | M23      | VTT         |
| K24      | H_A_BPM4#   | L24      | VSS         | M24      | TESTLO5     |
| K25      | VTT         | L25      | H_A_AP1#    | M25      | H_A_BREQ1#  |
| K26      | H_A_MCERR#  | L26      | H_A_AP0#    | M26      | VSS         |
| K27      | H_A_RST#    | L27      | VSS         | M27      | H_A_BINIT#  |
| K28      | VSS         | L28      | H_A_A32#    | M28      | H_A_A20#    |
| K29      | H_A_A35#    | L29      | H_A_A26#    | M29      | VTT         |
| K30      | H_A_A27#    | L30      | VSS         | M30      | H_A_A21#    |
| K31      | VTT         | L31      | H_A_A34#    | M31      | H_A_A30#    |
| K32      | H_A_A12#    | L32      | H_A_ADSTB0# | M32      | VSS         |
| K33      | H_A_A15#    | L33      | VSS         | M33      | H_A_A8#     |
| K34      | VSS         | L34      | H_A_A11#    | M34      | H_A_A9#     |
| K35      | H_A_A16#    | L35      | H_A_A36#    | M35      | VTT         |
| K36      | H_A_BREQ0#  | L36      | VSS         | M36      | H_A_HIT#    |
| K37      | VTT         | L37      | H_A_RS2#    | M37      | H_A_RS1#    |
| K38      | H_A_BNR#    | L38      | H_A_RS0#    | M38      | VSS         |
| L1       | H_B_HIT#    | M1       | H_B_RS1#    | N1       | VSS         |
| L2       | H_B_DEFER#  | M2       | VSS         | N2       | H_B_BNR#    |
| L3       | VSS         | M3       | H_B_RS2#    | N3       | H_B_RS0#    |
| L4       | H_B_A7#     | M4       | H_B_A6#     | N4       | VSS         |
| L5       | H_B_A4#     | M5       | VTT         | N5       | H_B_A9#     |
| L6       | VSS         | M6       | H_B_A5#     | N6       | H_B_ADSTB0# |

Table 7-1. NB Pin List (by Ball Number) (Sheet 5 of 13)

| Ball No. | Signal     | Ball No. | Signal     | Ball No. | Signal      |
|----------|------------|----------|------------|----------|-------------|
| N7       | VSS        | P7       | H_B_A15#   | R7       | H_B_A13#    |
| N8       | H_B_A21#   | P8       | H_B_A34#   | R8       | VSS         |
| N9       | H_B_A30#   | P9       | VTT        | R9       | H_B_A27#    |
| N10      | VSS        | P10      | H_B_A20#   | R10      | H_B_A35#    |
| N11      | H_B_A22#   | P11      | H_B_A26#   | R11      | VSS         |
| N12      | H_B_AP1#   | P12      | VSS        | R12      | H_B_RSP#    |
| N13      | VSS        | P13      | H_B_MCERR# | R13      | Reserved    |
| N14      | H_B_RST#   | P14      | VSS        | R14      | P1V5        |
| N15      | H_B_BPM5#  | P15      | P1V5       | R15      | VSS         |
| N16      | VSS        | P16      | VSS        | R16      | P1V5        |
| N17      | VSS        | P17      | P1V5       | R17      | VSS         |
| N18      | P1V5       | P18      | VSS        | R18      | P1V5        |
| N19      | VSS        | P19      | P1V5       | R19      | VSS         |
| N20      | P1V5       | P20      | VSS        | R20      | P1V5        |
| N21      | VSS        | P21      | P1V5       | R21      | VSS         |
| N22      | P1V5       | P22      | VSS        | R22      | P1V5        |
| N23      | VSS        | P23      | P1V5       | R23      | VSS         |
| N24      | H_A_OOD#   | P24      | VSS        | R24      | P1V5        |
| N25      | VSS        | P25      | P1V5       | R25      | VSS         |
| N26      | H_A_HITM#  | P26      | Reserved   | R26      | VSS         |
| N27      | H_A_ID3#   | P27      | VTT        | R27      | H_A_IDS#    |
| N28      | VSS        | P28      | H_A_A25#   | R28      | H_A_A17#    |
| N29      | H_A_A22#   | P29      | H_A_A23#   | R29      | VSS         |
| N30      | H_A_A29#   | P30      | VSS        | R30      | H_A_ADSTB1# |
| N31      | VSS        | P31      | H_A_A28#   | R31      | H_A_A24#    |
| N32      | H_A_REQ0#  | P32      | H_A_REQ3#  | R32      | VSS         |
| N33      | H_A_A5#    | P33      | VTT        | R33      | H_A_REQ1#   |
| N34      | VSS        | P34      | H_A_A4#    | R34      | H_A_REQ2#   |
| N35      | H_A_A6#    | P35      | H_A_A7#    | R35      | VSS         |
| N36      | H_A_DEFER# | P36      | VSS        | R36      | H_A_ID4#    |
| N37      | VSS        | P37      | H_A_ID0#   | R37      | H_A_ID1#    |
| N38      | H_A_BPRI#  | P38      | H_A_ID2#   | R38      | VSS         |
| P1       | H_B_TRDY#  | R1       | H_B_ADS#   | T1       | VTT         |
| P2       | H_B_BREQ0# | R2       | VSS        | T2       | H_B_LOCK#   |
| P3       | VTT        | R3       | H_B_DRDY#  | T3       | H_B_DBSY#   |
| P4       | H_B_A36#   | R4       | H_B_A16#   | T4       | VSS         |
| P5       | H_B_A11#   | R5       | VSS        | T5       | H_B_A10#    |
| P6       | VSS        | R6       | H_B_A12#   | T6       | H_B_A14#    |

Table 7-1. NB Pin List (by Ball Number) (Sheet 6 of 13)

| Ball No. | Signal       | Ball No. | Signal       | Ball No. | Signal       |
|----------|--------------|----------|--------------|----------|--------------|
| T7       | VTT          | U7       | IMI_C_RXN8   | V7       | IMI_C_RXP8   |
| T8       | H_B_A33#     | U8       | Reserved     | V8       | VSS          |
| T9       | H_B_A31#     | U9       | VSS          | V9       | IMI_C_TXN8   |
| T10      | VSS          | U10      | IMI_C_RXN9   | V10      | IMI_C_RXP9   |
| T11      | H_B_A32#     | U11      | IMI_C_FRAME  | V11      | P1V5         |
| T12      | H_B_A38#     | U12      | VSS          | V12      | IMI_C_TXN9   |
| T13      | VTT          | U13      | IMI_C_VCCBG  | V13      | IMI_C_VSSBG  |
| T14      | IMI_C_CLKP   | U14      | IMI_C_CLKN   | V14      | VSS          |
| T15      | P1V5         | U15      | VSS          | V15      | P1V5         |
| T16      | VSS          | U16      | P1V5         | V16      | VSS          |
| T17      | P1V5         | U17      | VSS          | V17      | P1V5         |
| T18      | VSS          | U18      | P1V5         | V18      | VSS          |
| T19      | P1V5         | U19      | VSS          | V19      | P1V5         |
| T20      | VSS          | U20      | P1V5         | V20      | VSS          |
| T21      | P1V5         | U21      | VSS          | V21      | P1V5         |
| T22      | VSS          | U22      | P1V5         | V22      | VSS          |
| T23      | P1V5         | U23      | VSS          | V23      | P1V5         |
| T24      | VSS          | U24      | P1V5         | V24      | VSS          |
| T25      | P1V5         | U25      | IMI_A_CLKP   | V25      | IMI_A_VCCA   |
| T26      | H_A_ID6#     | U26      | IMI_A_CLKN   | V26      | VSS          |
| T27      | H_A_A39#     | U27      | VSS          | V27      | IMI_A_LINKP0 |
| T28      | VSS          | U28      | IMI_A_LINKP2 | V28      | IMI_A_LINKN2 |
| T29      | H_A_A19#     | U29      | Reserved     | V29      | P1V5         |
| T30      | H_A_A18#     | U30      | VSS          | V30      | IMI_A_RXP17  |
| T31      | VTT          | U31      | IMI_A_LINKP1 | V31      | IMI_A_LINKN1 |
| T32      | H_A_A37#     | U32      | IMI_A_RST#   | V32      | VSS          |
| T33      | H_A_A3#      | U33      | VSS          | V33      | IMI_A_RXP7   |
| T34      | VSS          | U34      | IMI_A_TXP7   | V34      | IMI_A_TXN7   |
| T35      | H_A_REQ4#    | U35      | IMI_A_ICOMPI | V35      | P1V5         |
| T36      | H_A_ID5#     | U36      | VSS          | V36      | IMI_A_RXP6   |
| T37      | VTT          | U37      | IMI_A_TXP6   | V37      | IMI_A_TXN6   |
| T38      | H_A_ID7#     | U38      | IMI_A_ICOMPO | V38      | VSS          |
| U1       | IMI_C_RXN0   | V1       | IMI_C_RXP0   | W1       | VSS          |
| U2       | IMI_C_ICOMPI | V2       | VSS          | W2       | IMI_C_RXN2   |
| U3       | VSS          | V3       | IMI_C_TXN0   | W3       | IMI_C_TXP0   |
| U4       | IMI_C_RXN1   | V4       | IMI_C_RXP1   | W4       | VSS          |
| U5       | IMI_C_ICOMPO | V5       | P1V5         | W5       | IMI_C_RXN3   |
| U6       | VSS          | V6       | IMI_C_TXN1   | W6       | IMI_C_TXP1   |

Table 7-1. NB Pin List (by Ball Number) (Sheet 7 of 13)

| Ball No. | Signal       | Ball No. | Signal      | Ball No. | Signal      |
|----------|--------------|----------|-------------|----------|-------------|
| W7       | VSS          | Y7       | IMI_C_RXN12 | AA7      | IMI_C_RXP12 |
| W8       | IMI_C_RXN10  | Y8       | IMI_C_RXP10 | AA8      | VSS         |
| W9       | IMI_C_TXP8   | Y9       | P1V5        | AA9      | IMI_C_RXN14 |
| W10      | VSS          | Y10      | IMI_C_RXN13 | AA10     | IMI_C_RXP13 |
| W11      | IMI_C_RXN11  | Y11      | IMI_C_RXP11 | AA11     | VSS         |
| W12      | IMI_C_TXP9   | Y12      | VSS         | AA12     | IMI_C_RXN15 |
| W13      | VSS          | Y13      | Reserved    | AA13     | Reserved    |
| W14      | IMI_C_VSSA   | Y14      | IMI_C_VCCA  | AA14     | IMI_D_CLKP  |
| W15      | VSS          | Y15      | P1V5        | AA15     | VSS         |
| W16      | P1V5         | Y16      | VSS         | AA16     | P1V5        |
| W17      | VSS          | Y17      | P1V5        | AA17     | VSS         |
| W18      | P1V5         | Y18      | VSS         | AA18     | P1V5        |
| W19      | VSS          | Y19      | P1V5        | AA19     | VSS         |
| W20      | P1V5         | Y20      | VSS         | AA20     | P1V5        |
| W21      | VSS          | Y21      | P1V5        | AA21     | VSS         |
| W22      | P1V5         | Y22      | VSS         | AA22     | P1V5        |
| W23      | VSS          | Y23      | P1V5        | AA23     | VSS         |
| W24      | P1V5         | Y24      | VSS         | AA24     | P1V5        |
| W25      | IMI_A_VSSA   | Y25      | IMI_A_VSSBG | AA25     | IMI_A_VCCBG |
| W26      | Reserved     | Y26      | Reserved    | AA26     | VSS         |
| W27      | IMI_A_LINKN0 | Y27      | P1V5        | AA27     | IMI_A_RXP12 |
| W28      | VSS          | Y28      | IMI_A_RXP14 | AA28     | IMI_A_RXN14 |
| W29      | IMI_A_RXP16  | Y29      | IMI_A_RXN16 | AA29     | VSS         |
| W30      | IMI_A_RXN17  | Y30      | VSS         | AA30     | IMI_A_RXP11 |
| W31      | VSS          | Y31      | IMI_A_RXP13 | AA31     | IMI_A_RXN13 |
| W32      | IMI_A_RXP15  | Y32      | IMI_A_RXN15 | AA32     | VSS         |
| W33      | IMI_A_RXN7   | Y33      | P1V5        | AA33     | IMI_A_TXP3  |
| W34      | VSS          | Y34      | IMI_A_RXP5  | AA34     | IMI_A_RXN5  |
| W35      | IMI_A_TXP5   | Y35      | IMI_A_TXN5  | AA35     | VSS         |
| W36      | IMI_A_RXN6   | Y36      | VSS         | AA36     | IMI_A_TXP2  |
| W37      | VSS          | Y37      | IMI_A_RXP4  | AA37     | IMI_A_RXN4  |
| W38      | IMI_A_TXP4   | Y38      | IMI_A_TXN4  | AA38     | VSS         |
| Y1       | IMI_C_TXN2   | AA1      | IMI_C_TXP2  | AB1      | P1V5        |
| Y2       | IMI_C_RXP2   | AA2      | VSS         | AB2      | IMI_C_TXN4  |
| Y3       | P1V5         | AA3      | IMI_C_RXN4  | AB3      | IMI_C_RXP4  |
| Y4       | IMI_C_TXN3   | AA4      | IMI_C_TXP3  | AB4      | VSS         |
| Y5       | IMI_C_RXP3   | AA5      | VSS         | AB5      | IMI_C_TXN5  |
| Y6       | VSS          | AA6      | IMI_C_RXN5  | AB6      | IMI_C_RXP5  |

Table 7-1. NB Pin List (by Ball Number) (Sheet 8 of 13)

| Ball No. | Signal      | Ball No. | Signal       | Ball No. | Signal       |
|----------|-------------|----------|--------------|----------|--------------|
| AB7      | P1V5        | AC7      | IMI_C_LINKN0 | AD7      | IMI_C_LINKP0 |
| AB8      | IMI_C_RXN16 | AC8      | IMI_C_RXP16  | AD8      | VSS          |
| AB9      | IMI_C_RXP14 | AC9      | VSS          | AD9      | IMI_D_RXN10  |
| AB10     | VSS         | AC10     | IMI_C_LINKN1 | AD10     | IMI_C_LINKP1 |
| AB11     | IMI_C_RXN17 | AC11     | IMI_C_RXP17  | AD11     | P1V5         |
| AB12     | IMI_C_RXP15 | AC12     | VSS          | AD12     | IMI_C_LINKN2 |
| AB13     | P1V5        | AC13     | IMI_D_VCCBG  | AD13     | IMI_D_VSSBG  |
| AB14     | IMI_D_CLKN  | AC14     | IMI_D_VSSA   | AD14     | IMI_D_VCCA   |
| AB15     | P1V5        | AC15     | VSS          | AD15     | P1V5         |
| AB16     | VSS         | AC16     | P1V5         | AD16     | VSS          |
| AB17     | P1V5        | AC17     | VSS          | AD17     | P1V5         |
| AB18     | VSS         | AC18     | P1V5         | AD18     | VSS          |
| AB19     | P1V5        | AC19     | VSS          | AD19     | P1V5         |
| AB20     | VSS         | AC20     | P1V5         | AD20     | VSS          |
| AB21     | P1V5        | AC21     | VSS          | AD21     | P1V5         |
| AB22     | VSS         | AC22     | P1V5         | AD22     | VSS          |
| AB23     | P1V5        | AC23     | VSS          | AD23     | P1V5         |
| AB24     | VSS         | AC24     | P1V5         | AD24     | VSS          |
| AB25     | IMI_B_VCCA  | AC25     | IMI_B_VSSA   | AD25     | IMI_B_CLKN   |
| AB26     | Reserved    | AC26     | Reserved     | AD26     | VSS          |
| AB27     | IMI_A_RXN12 | AC27     | VSS          | AD27     | IMI_A_RXP9   |
| AB28     | VSS         | AC28     | IMI_A_TXP9   | AD28     | IMI_A_TXN9   |
| AB29     | IMI_A_RXP10 | AC29     | IMI_A_RXN10  | AD29     | P1V5         |
| AB30     | IMI_A_RXN11 | AC30     | VSS          | AD30     | IMI_B_LINKP2 |
| AB31     | P1V5        | AC31     | IMI_A_RXP8   | AD31     | IMI_A_RXN8   |
| AB32     | IMI_A_TXP8  | AC32     | IMI_A_TXN8   | AD32     | VSS          |
| AB33     | IMI_A_TXN3  | AC33     | VSS          | AD33     | IMI_A_RXP1   |
| AB34     | VSS         | AC34     | IMI_A_TXP1   | AD34     | IMI_A_TXN1   |
| AB35     | IMI_A_RXP3  | AC35     | IMI_A_RXN3   | AD35     | P1V5         |
| AB36     | IMI_A_TXN2  | AC36     | VSS          | AD36     | IMI_A_RXP0   |
| AB37     | P1V5        | AC37     | IMI_A_TXP0   | AD37     | IMI_A_TXN0   |
| AB38     | IMI_A_RXP2  | AC38     | IMI_A_RXN2   | AD38     | VSS          |
| AC1      | IMI_C_RXN6  | AD1      | IMI_C_RXP6   | AE1      | VSS          |
| AC2      | IMI_C_TXP4  | AD2      | VSS          | AE2      | IMI_D_RXN0   |
| AC3      | VSS         | AD3      | IMI_C_TXN6   | AE3      | IMI_C_TXP6   |
| AC4      | IMI_C_RXN7  | AD4      | IMI_C_RXP7   | AE4      | VSS          |
| AC5      | IMI_C_TXP5  | AD5      | P1V5         | AE5      | IMI_D_RXN1   |
| AC6      | VSS         | AD6      | IMI_C_TXN7   | AE6      | IMI_C_TXP7   |

Table 7-1. NB Pin List (by Ball Number) (Sheet 9 of 13)

| Ball No. | Signal       | Ball No. | Signal       | Ball No. | Signal                  |
|----------|--------------|----------|--------------|----------|-------------------------|
| AE7      | VSS          | AF7      | IMI_D_RXN12  | AG7      | IMI_D_RXP12             |
| AE8      | IMI_D_RXN11  | AF8      | IMI_D_RXP11  | AG8      | VSS                     |
| AE9      | IMI_D_RXP10  | AF9      | P1V5         | AG9      | IMI_D_RXN13             |
| AE10     | VSS          | AF10     | ICHRST#      | AG10     | PME_OUT/<br>EXP1_WIDTH2 |
| AE11     | IMI_C_RST#   | AF11     | IMI_D_FRAME  | AG11     | VSS                     |
| AE12     | IMI_C_LINKP2 | AF12     | VSS          | AG12     | Reserved                |
| AE13     | VSS          | AF13     | Reserved     | AG13     | Reserved                |
| AE14     | Reserved     | AF14     | Reserved     | AG14     | VSS                     |
| AE15     | VSS          | AF15     | Reserved     | AG15     | EXP_C0_TXP0             |
| AE16     | P1V5         | AF16     | CLK66        | AG16     | Reserved                |
| AE17     | VSS          | AF17     | HI_VCCA      | AG17     | VSS                     |
| AE18     | P1V5         | AF18     | HI_VSSA      | AG18     | EXP_C1_RXN3             |
| AE19     | VSS          | AF19     | EXP_VSSBG    | AG19     | EXP_VCCBG               |
| AE20     | P1V5         | AF20     | EXP_CLKP     | AG20     | VSS                     |
| AE21     | VSS          | AF21     | EXP_CLKN     | AG21     | EXP_B1_RXN3             |
| AE22     | P1V5         | AF22     | EXP_VCCA     | AG22     | EXP_VSSA                |
| AE23     | VSS          | AF23     | IMI_B_VSSBG  | AG23     | VSS                     |
| AE24     | EXP0_WIDTH0  | AF24     | IMI_B_VCCBG  | AG24     | EXP_A0_TXP0             |
| AE25     | IMI_B_CLKP   | AF25     | SMBCLK       | AG25     | GP_SMBDATA              |
| AE26     | GP_SMBCLK    | AF26     | V3REF0       | AG26     | VSS                     |
| AE27     | IMI_A_RXN9   | AF27     | ITP_TCK      | AG27     | EXP_A1_RXN3             |
| AE28     | VSS          | AF28     | ITP_TDI      | AG28     | PWRGOOD                 |
| AE29     | IMI_B_RST#   | AF29     | IMI_A_FRAME  | AG29     | VSS                     |
| AE30     | IMI_B_LINKN2 | AF30     | VSS          | AG30     | IMI_B_RXP17             |
| AE31     | VSS          | AF31     | IMI_B_LINKP0 | AG31     | IMI_B_LINKN0            |
| AE32     | IMI_B_LINKP1 | AF32     | IMI_B_LINKN1 | AG32     | VSS                     |
| AE33     | IMI_A_RXN1   | AF33     | P1V5         | AG33     | IMI_B_TXP7              |
| AE34     | VSS          | AF34     | IMI_B_RXP9   | AG34     | IMI_B_RXN9              |
| AE35     | IMI_B_TXP9   | AF35     | IMI_B_TXN9   | AG35     | VSS                     |
| AE36     | IMI_A_RXN0   | AF36     | VSS          | AG36     | IMI_B_TXP6              |
| AE37     | VSS          | AF37     | IMI_B_RXP8   | AG37     | IMI_B_RXN8              |
| AE38     | IMI_B_TXP8   | AF38     | IMI_B_TXN8   | AG38     | VSS                     |
| AF1      | IMI_D_TXN0   | AG1      | IMI_D_TXP0   | AH1      | P1V5                    |
| AF2      | IMI_D_RXP0   | AG2      | VSS          | AH2      | IMI_D_TXN2              |
| AF3      | P1V5         | AG3      | IMI_D_RXN2   | AH3      | IMI_D_RXP2              |
| AF4      | IMI_D_TXN1   | AG4      | IMI_D_TXP1   | AH4      | VSS                     |
| AF5      | IMI_D_RXP1   | AG5      | VSS          | AH5      | IMI_D_TXN3              |

Table 7-1. NB Pin List (by Ball Number) (Sheet 10 of 13)

| Ball No. | Signal                | Ball No. | Signal      | Ball No. | Signal        |
|----------|-----------------------|----------|-------------|----------|---------------|
| AH6      | VSS                   | AG6      | IMI_D_RXN3  | AH6      | IMI_D_RXP3    |
| AH7      | P1V5                  | AJ7      | IMI_D_RXN15 | AK7      | IMI_D_RXP15   |
| AH8      | IMI_D_RXN14           | AJ8      | IMI_D_RXP14 | AK8      | VSS           |
| AH9      | IMI_D_RXP13           | AJ9      | VSS         | AK9      | IMI_D_RXN16   |
| AH10     | VSS                   | AJ10     | V3REF1      | AK10     | Reserved      |
| AH11     | EXP_HPINT/EXP1_WIDTH1 | AJ11     | Reserved    | AK11     | P1V5          |
| AH12     | IMI_HPINT             | AJ12     | VSS         | AK12     | EXP1_SPE C10A |
| AH13     | P1V5                  | AJ13     | EXP_D_RXP0  | AK13     | EXP_D_RXN0    |
| AH14     | EXP_D_TXP0            | AJ14     | EXP_D_TXN0  | AK14     | VSS           |
| AH15     | EXP_C0_TXN0           | AJ15     | VSS         | AK15     | EXP_C0_RXP1   |
| AH16     | VSS                   | AJ16     | EXP_C0_TXP1 | AK16     | EXP_C0_TXN1   |
| AH17     | EXP_C0_RXP0           | AJ17     | EXP_C0_RXN0 | AK17     | P1V5          |
| AH18     | EXP_C1_RXP3           | AJ18     | VSS         | AK18     | EXP_C1_TXN3   |
| AH19     | P1V5                  | AJ19     | EXP_B0_RXP0 | AK19     | EXP_B0_RXN0   |
| AH20     | EXP_B0_TXP0           | AJ20     | EXP_B0_TXN0 | AK20     | VSS           |
| AH21     | EXP_B1_RXP3           | AJ21     | VSS         | AK21     | EXP_B0_TXP1   |
| AH22     | VSS                   | AJ22     | EXP_B1_TXN3 | AK22     | EXP_B1_TXP3   |
| AH23     | EXP_A0_RXP0           | AJ23     | EXP_A0_RXN0 | AK23     | P1V5          |
| AH24     | EXP_A0_TXN0           | AJ24     | VSS         | AK24     | EXP_A0_RXP1   |
| AH25     | P1V5                  | AJ25     | EXP_A0_TXP1 | AK25     | EXP_A0_TXN1   |
| AH26     | EXP_A1_TXN3           | AJ26     | EXP_A1_TXP3 | AK26     | VSS           |
| AH27     | EXP_A1_RXP3           | AJ27     | VSS         | AK27     | EXP_A1_RXN2   |
| AH28     | ITP_TMS               | AJ28     | Reserved    | AK28     | ERR0#         |
| AH29     | RSTIN#                | AJ29     | Reserved    | AK29     | P1V5          |
| AH30     | IMI_B_RXN17           | AJ30     | VSS         | AK30     | IMI_B_RXP14   |
| AH31     | P1V5                  | AJ31     | IMI_B_RXP15 | AK31     | IMI_B_RXN15   |
| AH32     | IMI_B_RXP16           | AJ32     | IMI_B_RXN16 | AK32     | VSS           |
| AH33     | IMI_B_TXN7            | AJ33     | VSS         | AK33     | IMI_B_RXP5    |
| AH34     | VSS                   | AJ34     | IMI_B_TXP5  | AK34     | IMI_B_TXN5    |
| AH35     | IMI_B_RXP7            | AJ35     | IMI_B_RXN7  | AK35     | P1V5          |
| AH36     | IMI_B_TXN6            | AJ36     | VSS         | AK36     | IMI_B_RXP4    |
| AH37     | P1V5                  | AJ37     | IMI_B_TXP4  | AK37     | IMI_B_TXN4    |
| AH38     | IMI_B_RXP6            | AJ38     | IMI_B_RXN6  | AK38     | VSS           |
| AJ1      | IMI_D_RXN4            | AK1      | IMI_D_RXP4  | AL1      | VSS           |
| AJ2      | IMI_D_TXP2            | AK2      | VSS         | AL2      | IMI_D_RXN6    |
| AJ3      | VSS                   | AK3      | IMI_D_TXN4  | AL3      | IMI_D_TXP4    |
| AJ4      | IMI_D_RXN5            | AK4      | IMI_D_RXP5  | AL4      | VSS           |
| AJ5      | IMI_D_TXP3            | AK5      | P1V5        | AL5      | IMI_D_RXN7    |

Table 7-1. NB Pin List (by Ball Number) (Sheet 11 of 13)

| Ball No. | Signal      | Ball No. | Signal       | Ball No. | Signal       |
|----------|-------------|----------|--------------|----------|--------------|
| AJ6      | VSS         | AK6      | IMI_D_TXN5   | AL6      | IMI_D_TXP5   |
| AL7      | VSS         | AM7      | IMI_D_LINKN0 | AN7      | IMI_D_LINKP0 |
| AL8      | IMI_D_RXN17 | AM8      | IMI_D_RXP17  | AN8      | VSS          |
| AL9      | IMI_D_RXP16 | AM9      | P1V5         | AN9      | IMI_D_LINKN1 |
| AL10     | VSS         | AM10     | Reserved     | AN10     | Reserved     |
| AL11     | Reserved    | AM11     | Reserved     | AN11     | VSS          |
| AL12     | Reserved    | AM12     | VSS          | AN12     | EXP_D_TXP2   |
| AL13     | VSS         | AM13     | EXP_D_RXP1   | AN13     | EXP_D_RXN1   |
| AL14     | EXP_D_TXP1  | AM14     | EXP_D_TXN1   | AN14     | VSS          |
| AL15     | EXP_C0_RXN1 | AM15     | P1V5         | AN15     | EXP_C0_RXP2  |
| AL16     | VSS         | AM16     | EXP_C0_TXP2  | AN16     | EXP_C0_TXN2  |
| AL17     | EXP_C1_RXN2 | AM17     | EXP_C1_RXP2  | AN17     | VSS          |
| AL18     | EXP_C1_TXP3 | AM18     | VSS          | AN18     | EXP_C1_TXN2  |
| AL19     | VSS         | AM19     | EXP_B0_TXP2  | AN19     | EXP_B0_TXN2  |
| AL20     | EXP_B0_RXP1 | AM20     | EXP_B0_RXN1  | AN20     | VSS          |
| AL21     | EXP_B0_TXN1 | AM21     | P1V5         | AN21     | EXP_B1_RXN1  |
| AL22     | VSS         | AM22     | EXP_B1_TXN2  | AN22     | EXP_B1_TXP2  |
| AL23     | EXP_B1_RXN2 | AM23     | EXP_B1_RXP2  | AN23     | VSS          |
| AL24     | EXP_A0_RXN1 | AM24     | VSS          | AN24     | EXP_A0_RXP2  |
| AL25     | VSS         | AM25     | EXP_A0_TXP2  | AN25     | EXP_A0_TXN2  |
| AL26     | EXP_A1_TXN2 | AM26     | EXP_A1_TXP2  | AN26     | VSS          |
| AL27     | EXP_A1_RXP2 | AM27     | P1V5         | AN27     | EXP_A1_RXN1  |
| AL28     | VSS         | AM28     | EXP0_SPEC10A | AN28     | XDP_ODTCRES  |
| AL29     | ERR1#       | AM29     | ERR2#        | AN29     | VSS          |
| AL30     | IMI_B_RXN14 | AM30     | VSS          | AN30     | IMI_B_RXP11  |
| AL31     | VSS         | AM31     | IMI_B_RXP12  | AN31     | IMI_B_RXN12  |
| AL32     | IMI_B_RXP13 | AM32     | IMI_B_RXN13  | AN32     | VSS          |
| AL33     | IMI_B_RXN5  | AM33     | P1V5         | AN33     | IMI_B_TXP1   |
| AL34     | VSS         | AM34     | IMI_B_RXP3   | AN34     | IMI_B_RXN3   |
| AL35     | IMI_B_TXP3  | AM35     | IMI_B_TXN3   | AN35     | VSS          |
| AL36     | IMI_B_RXN4  | AM36     | VSS          | AN36     | IMI_B_TXP0   |
| AL37     | VSS         | AM37     | IMI_B_RXP2   | AN37     | IMI_B_RXN2   |
| AL38     | IMI_B_TXP2  | AM38     | IMI_B_TXN2   | AN38     | VSS          |
| AM1      | IMI_D_TXN6  | AN1      | IMI_D_TXP6   | AP1      | P1V5         |
| AM2      | IMI_D_RXP6  | AN2      | VSS          | AP2      | IMI_D_TXN8   |
| AM3      | P1V5        | AN3      | IMI_D_RXN8   | AP3      | IMI_D_RXP8   |
| AM4      | IMI_D_TXN7  | AN4      | IMI_D_TXP7   | AP4      | VSS          |
| AM5      | IMI_D_RXP7  | AN5      | VSS          | AP5      | IMI_D_TXN9   |

Table 7-1. NB Pin List (by Ball Number) (Sheet 12 of 13)

| Ball No. | Signal       | Ball No. | Signal       | Ball No. | Signal      |
|----------|--------------|----------|--------------|----------|-------------|
| AM6      | VSS          | AN6      | IMI_D_RXN9   | AP6      | IMI_D_RXP9  |
| AP7      | P1V5         | AR7      | IMI_D_RST#   | AT7      | HI5         |
| AP8      | IMI_D_LINKN2 | AR8      | IMI_D_LINKP2 | AT8      | VSS         |
| AP9      | IMI_D_LINKP1 | AR9      | VSS          | AT9      | HIRCOMP     |
| AP10     | VSS          | AR10     | Reserved     | AT10     | Reserved    |
| AP11     | Reserved     | AR11     | Reserved     | AT11     | P1V5        |
| AP12     | EXP_D_TXN2   | AR12     | VSS          | AT12     | EXP_D_TXP3  |
| AP13     | P1V5         | AR13     | Reserved     | AT13     | EXP_D_TXN3  |
| AP14     | EXP_D_RXP2   | AR14     | EXP_D_RXN2   | AT14     | VSS         |
| AP15     | EXP_C0_RXN2  | AR15     | VSS          | AT15     | EXP_C0_RXP3 |
| AP16     | VSS          | AR16     | EXP_ICOMPI   | AT16     | EXP_C0_RXN3 |
| AP17     | EXP_C1_TXN1  | AR17     | EXP_C1_TXP1  | AT17     | P1V5        |
| AP18     | EXP_C1_TXP2  | AR18     | VSS          | AT18     | Reserved    |
| AP19     | P1V5         | AR19     | EXP_C1_RXN1  | AT19     | EXP_C1_RXP1 |
| AP20     | EXP_B0_RXP2  | AR20     | EXP_B0_RXN2  | AT20     | VSS         |
| AP21     | EXP_B1_RXP1  | AR21     | VSS          | AT21     | EXP_B1_TXP0 |
| AP22     | VSS          | AR22     | EXP_RCOMP    | AT22     | EXP_B1_TXN0 |
| AP23     | EXP_B1_TXN1  | AR23     | EXP_B1_TXP1  | AT23     | P1V5        |
| AP24     | EXP_A0_RXN2  | AR24     | VSS          | AT24     | EXP_A0_RXP3 |
| AP25     | P1V5         | AR25     | ITP_TRST#    | AT25     | EXP_A0_RXN3 |
| AP26     | EXP_A1_TXN1  | AR26     | EXP_A1_TXP1  | AT26     | VSS         |
| AP27     | EXP_A1_RXP1  | AR27     | VSS          | AT27     | EXP0_WIDTH1 |
| AP28     | VSS          | AR28     | SMBDATA      | AT28     | XDP_D15#    |
| AP29     | XDP_SLWCRES  | AR29     | XDP_CRES     | AT29     | P1V5        |
| AP30     | IMI_B_RXN11  | AR30     | VSS          | AT30     | XDP_D13#    |
| AP31     | P1V5         | AR31     | IMI_B_FRAME  | AT31     | XDP_D12#    |
| AP32     | IMI_B_RXP10  | AR32     | IMI_B_RXN10  | AT32     | VSS         |
| AP33     | IMI_B_TXN1   | AR33     | VSS          | AT33     | XDP_D10#    |
| AP34     | VSS          | AR34     | IMI_B_ICOMPI | AT34     | XDP_D9#     |
| AP35     | IMI_B_RXP1   | AR35     | IMI_B_RXN1   | AT35     | VSS         |
| AP36     | IMI_B_TXN0   | AR36     | VSS          | AT36     | XDP_DSTBP   |
| AP37     | P1V5         | AR37     | IMI_B_ICOMPO | AT37     | XDP_DSTBN   |
| AP38     | IMI_B_RXP0   | AR38     | IMI_B_RXN0   | AT38     | VSS         |
| AR1      | IMI_D_ICOMPI | AT1      | VSS          | AU2      | VSS         |
| AR2      | IMI_D_TXP8   | AT2      | HI11         | AU3      | HI1         |
| AR3      | VSS          | AT3      | HI0          | AU4      | HI2         |
| AR4      | IMI_D_ICOMPO | AT4      | HI3          | AU5      | HI8         |
| AR5      | IMI_D_TXP9   | AT5      | P1V5         | AU6      | HI9         |

Table 7-1. NB Pin List (by Ball Number) (Sheet 13 of 13)

| Ball No. | Signal      | Ball No. | Signal      | Ball No. | Signal      |
|----------|-------------|----------|-------------|----------|-------------|
| AR6      | VSS         | AT6      | HI10        | AU7      | VSS         |
| AU8      | HI7         | AU30     | XDP_D5#     | AV17     | EXP_C1_TXN0 |
| AU9      | HIVSWING    | AU31     | P1V5        | AV18     | VSS         |
| AU10     | VSS         | AU32     | XDP_D11#    | AV19     | EXP_B0_TXP3 |
| AU11     | Reserved    | AU33     | XDP_D2#     | AV20     | EXP_B0_TXN3 |
| AU12     | Reserved    | AU34     | P1V5        | AV21     | P1V5        |
| AU13     | VSS         | AU35     | XDP_D8#     | AV22     | EXP_B1_RXP0 |
| AU14     | EXP_C0_TXP3 | AU36     | XDP_RDY#    | AV23     | EXP_B1_RXN0 |
| AU15     | EXP_C0_TXN3 | AU37     | VSS         | AV24     | VSS         |
| AU16     | VSS         | AV3      | VSS         | AV25     | EXP_A1_TXP0 |
| AU17     | EXP_C1_RXP0 | AV4      | HI_STBS     | AV26     | EXP_A1_TXN0 |
| AU18     | EXP_C1_RXN0 | AV5      | HI_STBF     | AV27     | P1V5        |
| AU19     | VSS         | AV6      | VSS         | AV28     | XDP_D7#     |
| AU20     | EXP_B0_RXP3 | AV7      | HI4         | AV29     | XDP_D6#     |
| AU21     | EXP_B0_RXN3 | AV8      | HI6         | AV30     | VSS         |
| AU22     | VSS         | AV9      | P1V5        | AV31     | XDP_D4#     |
| AU23     | EXP_A0_TXP3 | AV10     | HIVREF      | AV32     | XDP_D3#     |
| AU24     | EXP_A0_TXN3 | AV11     | Reserved    | AV33     | VSS         |
| AU25     | VSS         | AV12     | VSS         | AV34     | XDP_D1#     |
| AU26     | EXP_A1_RXP0 | AV13     | EXP_D_RXP3  | AV35     | XDP_D0#     |
| AU27     | EXP_A1_RXN0 | AV14     | EXP_D_RXN3  | AV36     | VSS         |
| AU28     | ITP_TDO     | AV15     | P1V5        |          |             |
| AU29     | XDP_D14#    | AV16     | EXP_C1_TXP0 |          |             |

Table 7-2. NB Pin List (by Signal Name) (Sheet 1 of 13)

| Ball No. | Signal      | Ball No. | Signal      | Ball No. | Signal      |
|----------|-------------|----------|-------------|----------|-------------|
| AF16     | CLK66       | AR20     | EXP_B0_RXN2 | AH15     | EXP_C0_TXN0 |
| AK28     | ERR0#       | AU21     | EXP_B0_RXN3 | AK16     | EXP_C0_TXN1 |
| AL29     | ERR1#       | AJ19     | EXP_B0_RXP0 | AN16     | EXP_C0_TXN2 |
| AM29     | ERR2#       | AL20     | EXP_B0_RXP1 | AU15     | EXP_C0_TXN3 |
| AJ23     | EXP_A0_RXN0 | AP20     | EXP_B0_RXP2 | AG15     | EXP_C0_TXP0 |
| AL24     | EXP_A0_RXN1 | AU20     | EXP_B0_RXP3 | AJ16     | EXP_C0_TXP1 |
| AP24     | EXP_A0_RXN2 | AJ20     | EXP_B0_TXN0 | AM16     | EXP_C0_TXP2 |
| AT25     | EXP_A0_RXN3 | AL21     | EXP_B0_TXN1 | AU14     | EXP_C0_TXP3 |
| AH23     | EXP_A0_RXP0 | AN19     | EXP_B0_TXN2 | AU18     | EXP_C1_RXN0 |
| AK24     | EXP_A0_RXP1 | AV20     | EXP_B0_TXN3 | AR19     | EXP_C1_RXN1 |
| AN24     | EXP_A0_RXP2 | AH20     | EXP_B0_TXP0 | AL17     | EXP_C1_RXN2 |
| AT24     | EXP_A0_RXP3 | AK21     | EXP_B0_TXP1 | AG18     | EXP_C1_RXN3 |
| AH24     | EXP_A0_TXN0 | AM19     | EXP_B0_TXP2 | AU17     | EXP_C1_RXP0 |
| AK25     | EXP_A0_TXN1 | AV19     | EXP_B0_TXP3 | AT19     | EXP_C1_RXP1 |
| AN25     | EXP_A0_TXN2 | AV23     | EXP_B1_RXN0 | AM17     | EXP_C1_RXP2 |
| AU24     | EXP_A0_TXN3 | AN21     | EXP_B1_RXN1 | AH18     | EXP_C1_RXP3 |
| AG24     | EXP_A0_TXP0 | AL23     | EXP_B1_RXN2 | AV17     | EXP_C1_TXN0 |
| AJ25     | EXP_A0_TXP1 | AG21     | EXP_B1_RXN3 | AP17     | EXP_C1_TXN1 |
| AM25     | EXP_A0_TXP2 | AV22     | EXP_B1_RXP0 | AN18     | EXP_C1_TXN2 |
| AU23     | EXP_A0_TXP3 | AP21     | EXP_B1_RXP1 | AK18     | EXP_C1_TXN3 |
| AU27     | EXP_A1_RXN0 | AM23     | EXP_B1_RXP2 | AV16     | EXP_C1_TXP0 |
| AN27     | EXP_A1_RXN1 | AH21     | EXP_B1_RXP3 | AR17     | EXP_C1_TXP1 |
| AK27     | EXP_A1_RXN2 | AT22     | EXP_B1_TXN0 | AP18     | EXP_C1_TXP2 |
| AG27     | EXP_A1_RXN3 | AP23     | EXP_B1_TXN1 | AL18     | EXP_C1_TXP3 |
| AU26     | EXP_A1_RXP0 | AM22     | EXP_B1_TXN2 | AF21     | EXP_CLKN    |
| AP27     | EXP_A1_RXP1 | AJ22     | EXP_B1_TXN3 | AF20     | EXP_CLKP    |
| AL27     | EXP_A1_RXP2 | AT21     | EXP_B1_TXP0 | AK13     | EXP_D_RXN0  |
| AH27     | EXP_A1_RXP3 | AR23     | EXP_B1_TXP1 | AN13     | EXP_D_RXN1  |
| AV26     | EXP_A1_TXN0 | AN22     | EXP_B1_TXP2 | AR14     | EXP_D_RXN2  |
| AP26     | EXP_A1_TXN1 | AK22     | EXP_B1_TXP3 | AV14     | EXP_D_RXN3  |
| AL26     | EXP_A1_TXN2 | AJ17     | EXP_C0_RXN0 | AJ13     | EXP_D_RXP0  |
| AH26     | EXP_A1_TXN3 | AL15     | EXP_C0_RXN1 | AM13     | EXP_D_RXP1  |
| AV25     | EXP_A1_TXP0 | AP15     | EXP_C0_RXN2 | AP14     | EXP_D_RXP2  |
| AR26     | EXP_A1_TXP1 | AT16     | EXP_C0_RXN3 | AV13     | EXP_D_RXP3  |
| AM26     | EXP_A1_TXP2 | AH17     | EXP_C0_RXP0 | AJ14     | EXP_D_TXN0  |
| AJ26     | EXP_A1_TXP3 | AK15     | EXP_C0_RXP1 | AM14     | EXP_D_TXN1  |
| AK19     | EXP_B0_RXN0 | AN15     | EXP_C0_RXP2 | AP12     | EXP_D_TXN2  |
| AM20     | EXP_B0_RXN1 | AT15     | EXP_C0_RXP3 | AT13     | EXP_D_TXN3  |

Table 7-2. NB Pin List (by Signal Name) (Sheet 2 of 13)

| Ball No. | Signal                | Ball No. | Signal      | Ball No. | Signal   |
|----------|-----------------------|----------|-------------|----------|----------|
| AH14     | EXP_D_TXP0            | P29      | H_A_A23#    | D30      | H_A_D9#  |
| AL14     | EXP_D_TXP1            | R31      | H_A_A24#    | A30      | H_A_D10# |
| AN12     | EXP_D_TXP2            | P28      | H_A_A25#    | A29      | H_A_D11# |
| AT12     | EXP_D_TXP3            | L29      | H_A_A26#    | C28      | H_A_D12# |
| AH11     | EXP_HPINT/EXP1_WIDTH1 | K30      | H_A_A27#    | C30      | H_A_D13# |
| AR16     | EXP_ICOMPI            | P31      | H_A_A28#    | D29      | H_A_D14# |
| AR22     | EXP_RCOMP             | N30      | H_A_A29#    | A27      | H_A_D15# |
| AF22     | EXP_VCCA              | M31      | H_A_A30#    | F37      | H_A_D16# |
| AG19     | EXP_VCCBG             | J30      | H_A_A31#    | G36      | H_A_D17# |
| AG22     | EXP_VSSA              | L28      | H_A_A32#    | F36      | H_A_D18# |
| AF19     | EXP_VSSBG             | J31      | H_A_A33#    | E38      | H_A_D19# |
| AM28     | EXP0_SPEC10A          | L31      | H_A_A34#    | E37      | H_A_D20# |
| AE24     | EXP0_WIDTH0           | K29      | H_A_A35#    | D38      | H_A_D21# |
| AT27     | EXP0_WIDTH1           | L35      | H_A_A36#    | D37      | H_A_D22# |
| AK12     | EXP1_SPE C10A         | T32      | H_A_A37#    | C37      | H_A_D23# |
| K14      | EXP1_WIDTH0           | J28      | H_A_A38#    | F33      | H_A_D24# |
| AE26     | GP_SMBCLK             | T27      | H_A_A39#    | E35      | H_A_D25# |
| AG25     | GP_SMBDATA            | J37      | H_A_ADS#    | B36      | H_A_D26# |
| T33      | H_A_A3#               | L32      | H_A_ADSTB0# | D36      | H_A_D27# |
| P34      | H_A_A4#               | R30      | H_A_ADSTB1# | E34      | H_A_D28# |
| N33      | H_A_A5#               | L26      | H_A_AP0#    | G35      | H_A_D29# |
| N35      | H_A_A6#               | L25      | H_A_AP1#    | F34      | H_A_D30# |
| P35      | H_A_A7#               | M27      | H_A_BINIT#  | E32      | H_A_D31# |
| M33      | H_A_A8#               | K38      | H_A_BNR#    | A26      | H_A_D32# |
| M34      | H_A_A9#               | K24      | H_A_BPM4#   | B26      | H_A_D33# |
| J33      | H_A_A10#              | J25      | H_A_BPM5#   | D27      | H_A_D34# |
| L34      | H_A_A11#              | N38      | H_A_BPRI#   | D26      | H_A_D35# |
| K32      | H_A_A12#              | K36      | H_A_BREQ0#  | C27      | H_A_D36# |
| J34      | H_A_A13#              | M25      | H_A_BREQ1#  | B25      | H_A_D37# |
| H35      | H_A_A14#              | A33      | H_A_D0#     | D24      | H_A_D38# |
| K33      | H_A_A15#              | B34      | H_A_D1#     | A24      | H_A_D39# |
| K35      | H_A_A16#              | C31      | H_A_D2#     | D23      | H_A_D40# |
| R28      | H_A_A17#              | B32      | H_A_D3#     | A23      | H_A_D41# |
| T30      | H_A_A18#              | D33      | H_A_D4#     | C22      | H_A_D42# |
| T29      | H_A_A19#              | B31      | H_A_D5#     | B22      | H_A_D43# |
| M28      | H_A_A20#              | A32      | H_A_D6#     | C21      | H_A_D44# |
| M30      | H_A_A21#              | C33      | H_A_D7#     | A21      | H_A_D45# |
| N29      | H_A_A22#              | D32      | H_A_D8#     | A20      | H_A_D46# |

Table 7-2. NB Pin List (by Signal Name) (Sheet 3 of 13)

| Ball No. | Signal      | Ball No. | Signal      | Ball No. | Signal      |
|----------|-------------|----------|-------------|----------|-------------|
| D21      | H_A_D47#    | C25      | H_A_DSTBN2# | M4       | H_B_A6#     |
| G27      | H_A_D48#    | F28      | H_A_DSTBN3# | L4       | H_B_A7#     |
| G29      | H_A_D49#    | B28      | H_A_DSTBP0# | M7       | H_B_A8#     |
| F30      | H_A_D50#    | G32      | H_A_DSTBP1# | N5       | H_B_A9#     |
| E28      | H_A_D51#    | C24      | H_A_DSTBP2# | T5       | H_B_A10#    |
| E29      | H_A_D52#    | F27      | H_A_DSTBP3# | P5       | H_B_A11#    |
| G26      | H_A_D53#    | M36      | H_A_HIT#    | R6       | H_B_A12#    |
| F25      | H_A_D54#    | N26      | H_A_HITM#   | R7       | H_B_A13#    |
| E26      | H_A_D55#    | P37      | H_A_ID0#    | T6       | H_B_A14#    |
| G23      | H_A_D56#    | R37      | H_A_ID1#    | P7       | H_B_A15#    |
| E23      | H_A_D57#    | P38      | H_A_ID2#    | R4       | H_B_A16#    |
| F22      | H_A_D58#    | N27      | H_A_ID3#    | L11      | H_B_A17#    |
| E22      | H_A_D59#    | R36      | H_A_ID4#    | J9       | H_B_A18#    |
| F21      | H_A_D60#    | T36      | H_A_ID5#    | J10      | H_B_A19#    |
| F24      | H_A_D61#    | T26      | H_A_ID6#    | P10      | H_B_A20#    |
| E25      | H_A_D62#    | T38      | H_A_ID7#    | N8       | H_B_A21#    |
| G21      | H_A_D63#    | R27      | H_A_IDS#    | N11      | H_B_A22#    |
| C34      | H_A_DBI0#   | G38      | H_A_LOCK#   | M10      | H_B_A23#    |
| C36      | H_A_DBI1#   | K26      | H_A_MCERR#  | K8       | H_B_A24#    |
| B23      | H_A_DBI2#   | N24      | H_A_OOD#    | L10      | H_B_A25#    |
| G24      | H_A_DBI3#   | M21      | H_A_PRSENT# | P11      | H_B_A26#    |
| H38      | H_A_DBSY#   | N32      | H_A_REQ0#   | R9       | H_B_A27#    |
| N36      | H_A_DEFER#  | R33      | H_A_REQ1#   | L8       | H_B_A28#    |
| B35      | H_A_DEP0#   | R34      | H_A_REQ2#   | M9       | H_B_A29#    |
| A35      | H_A_DEP1#   | P32      | H_A_REQ3#   | N9       | H_B_A30#    |
| F31      | H_A_DEP2#   | T35      | H_A_REQ4#   | T9       | H_B_A31#    |
| E31      | H_A_DEP3#   | L38      | H_A_RS0#    | T11      | H_B_A32#    |
| B20      | H_A_DEP4#   | M37      | H_A_RS1#    | T8       | H_B_A33#    |
| D20      | H_A_DEP5#   | L37      | H_A_RS2#    | P8       | H_B_A34#    |
| G20      | H_A_DEP6#   | J27      | H_A_RSP#    | R10      | H_B_A35#    |
| E20      | H_A_DEP7#   | K27      | H_A_RST#    | P4       | H_B_A36#    |
| H23      | H_A_DP0#    | J36      | H_A_TRDY#   | H4       | H_B_A37#    |
| H25      | H_A_DP1#    | H34      | H_A_VREF0   | T12      | H_B_A38#    |
| J24      | H_A_DP2#    | H22      | H_A_VREF1   | K11      | H_B_A39#    |
| K23      | H_A_DP3#    | H31      | H_A_VREF2   | R1       | H_B_ADS#    |
| H37      | H_A_DRDY#   | J6       | H_B_A3#     | N6       | H_B_ADSTB0# |
| B29      | H_A_DSTBN0# | L5       | H_B_A4#     | K9       | H_B_ADSTB1# |
| G33      | H_A_DSTBN1# | M6       | H_B_A5#     | M13      | H_B_AP0#    |

Table 7-2. NB Pin List (by Signal Name) (Sheet 4 of 13)

| Ball No. | Signal     | Ball No. | Signal    | Ball No. | Signal      |
|----------|------------|----------|-----------|----------|-------------|
| N12      | H_B_AP1#   | F12      | H_B_D30#  | T3       | H_B_DBSY#   |
| M12      | H_B_BINIT# | F10      | H_B_D31#  | L2       | H_B_DEFER#  |
| N2       | H_B_BNR#   | A11      | H_B_D32#  | C19      | H_B_DEP0#   |
| M15      | H_B_BPM4#  | D9       | H_B_D33#  | B19      | H_B_DEP1#   |
| N15      | H_B_BPM5#  | B11      | H_B_D34#  | G11      | H_B_DEP2#   |
| K2       | H_B_BPRI#  | A8       | H_B_D35#  | F9       | H_B_DEP3#   |
| P2       | H_B_BREQ0# | D11      | H_B_D36#  | B4       | H_B_DEP4#   |
| L14      | H_B_BREQ1# | B10      | H_B_D37#  | C4       | H_B_DEP5#   |
| B17      | H_B_D0#    | B8       | H_B_D38#  | F1       | H_B_DEP6#   |
| C18      | H_B_D1#    | A9       | H_B_D39#  | G3       | H_B_DEP7#   |
| D15      | H_B_D2#    | C7       | H_B_D40#  | H16      | H_B_DP0#    |
| C16      | H_B_D3#    | D8       | H_B_D41#  | H14      | H_B_DP1#    |
| A18      | H_B_D4#    | C6       | H_B_D42#  | J15      | H_B_DP2#    |
| C15      | H_B_D5#    | A6       | H_B_D43#  | J16      | H_B_DP3#    |
| B16      | H_B_D6#    | A5       | H_B_D44#  | R3       | H_B_DRDY#   |
| D17      | H_B_D7#    | D6       | H_B_D45#  | B14      | H_B_DSTBN0# |
| A17      | H_B_D8#    | D5       | H_B_D46#  | E11      | H_B_DSTBN1# |
| A15      | H_B_D9#    | B5       | H_B_D47#  | C10      | H_B_DSTBN2# |
| C13      | H_B_D10#   | F7       | H_B_D48#  | G6       | H_B_DSTBN3# |
| A14      | H_B_D11#   | G8       | H_B_D49#  | B13      | H_B_DSTBP0# |
| D12      | H_B_D12#   | E8       | H_B_D50#  | E10      | H_B_DSTBP1# |
| D14      | H_B_D13#   | F6       | H_B_D51#  | C9       | H_B_DSTBP2# |
| C12      | H_B_D14#   | E7       | H_B_D52#  | G5       | H_B_DSTBP3# |
| A12      | H_B_D15#   | E5       | H_B_D53#  | L1       | H_B_HIT#    |
| F19      | H_B_D16#   | F4       | H_B_D54#  | L13      | H_B_HITM#   |
| E19      | H_B_D17#   | E4       | H_B_D55#  | J3       | H_B_ID0#    |
| F18      | H_B_D18#   | C2       | H_B_D56#  | J1       | H_B_ID1#    |
| G18      | H_B_D19#   | D2       | H_B_D57#  | K3       | H_B_ID2#    |
| G17      | H_B_D20#   | E2       | H_B_D58#  | K15      | H_B_ID3#    |
| E17      | H_B_D21#   | D1       | H_B_D59#  | H1       | H_B_ID4#    |
| E16      | H_B_D22#   | F3       | H_B_D60#  | H2       | H_B_ID5#    |
| F16      | H_B_D23#   | D3       | H_B_D61#  | H13      | H_B_ID6#    |
| G12      | H_B_D24#   | B3       | H_B_D62#  | G2       | H_B_ID7#    |
| E13      | H_B_D25#   | E1       | H_B_D63#  | J12      | H_B_IDS#    |
| E14      | H_B_D26#   | D18      | H_B_DBI0# | T2       | H_B_LOCK#   |
| F15      | H_B_D27#   | G15      | H_B_DBI1# | P13      | H_B_MCERR#  |
| F13      | H_B_D28#   | B7       | H_B_DBI2# | J13      | H_B_OOD#    |
| G14      | H_B_D29#   | C3       | H_B_DBI3# | M22      | H_B_PRSENT# |

Table 7-2. NB Pin List (by Signal Name) (Sheet 5 of 13)

| Ball No. | Signal    | Ball No. | Signal       | Ball No. | Signal      |
|----------|-----------|----------|--------------|----------|-------------|
| L7       | H_B_REQ0# | AU9      | HISWING      | Y34      | IMI_A_RXP5  |
| J7       | H_B_REQ1# | AV10     | HIVREF       | V36      | IMI_A_RXP6  |
| K5       | H_B_REQ2# | AF10     | ICHRST#      | V33      | IMI_A_RXP7  |
| K6       | H_B_REQ3# | U26      | IMI_A_CLKN   | AC31     | IMI_A_RXP8  |
| J4       | H_B_REQ4# | U25      | IMI_A_CLKP   | AD27     | IMI_A_RXP9  |
| N3       | H_B_RS0#  | AF29     | IMI_A_FRAME  | AB29     | IMI_A_RXP10 |
| M1       | H_B_RS1#  | U35      | IMI_A_ICOMPI | AA30     | IMI_A_RXP11 |
| M3       | H_B_RS2#  | U38      | IMI_A_ICOMPO | AA27     | IMI_A_RXP12 |
| R12      | H_B_RSP#  | W27      | IMI_A_LINKN0 | Y31      | IMI_A_RXP13 |
| N14      | H_B_RST#  | V31      | IMI_A_LINKN1 | Y28      | IMI_A_RXP14 |
| P1       | H_B_TRDY# | V28      | IMI_A_LINKN2 | W32      | IMI_A_RXP15 |
| H5       | H_B_VREF0 | V27      | IMI_A_LINKP0 | W29      | IMI_A_RXP16 |
| H7       | H_B_VREF1 | U31      | IMI_A_LINKP1 | V30      | IMI_A_RXP17 |
| H17      | H_B_VREF2 | U28      | IMI_A_LINKP2 | AD37     | IMI_A_TXN0  |
| L17      | H_CRES    | U32      | IMI_A_RST#   | AD34     | IMI_A_TXN1  |
| M16      | H_ODTCRES | AE36     | IMI_A_RXN0   | AB36     | IMI_A_TXN2  |
| L16      | H_SLWCRES | AE33     | IMI_A_RXN1   | AB33     | IMI_A_TXN3  |
| K18      | H_VCCA    | AC38     | IMI_A_RXN2   | Y38      | IMI_A_TXN4  |
| K17      | H_VSSA    | AC35     | IMI_A_RXN3   | Y35      | IMI_A_TXN5  |
| J21      | HCLKINN   | AA37     | IMI_A_RXN4   | V37      | IMI_A_TXN6  |
| J22      | HCLKINP   | AA34     | IMI_A_RXN5   | V34      | IMI_A_TXN7  |
| AV5      | HI_STBF   | W36      | IMI_A_RXN6   | AC32     | IMI_A_TXN8  |
| AV4      | HI_STBS   | W33      | IMI_A_RXN7   | AD28     | IMI_A_TXN9  |
| AF17     | HI_VCCA   | AD31     | IMI_A_RXN8   | AC37     | IMI_A_TXP0  |
| AF18     | HI_VSSA   | AE27     | IMI_A_RXN9   | AC34     | IMI_A_TXP1  |
| AT3      | HI0       | AC29     | IMI_A_RXN10  | AA36     | IMI_A_TXP2  |
| AU3      | HI1       | AB30     | IMI_A_RXN11  | AA33     | IMI_A_TXP3  |
| AU4      | HI2       | AB27     | IMI_A_RXN12  | W38      | IMI_A_TXP4  |
| AT4      | HI3       | AA31     | IMI_A_RXN13  | W35      | IMI_A_TXP5  |
| AV7      | HI4       | AA28     | IMI_A_RXN14  | U37      | IMI_A_TXP6  |
| AT7      | HI5       | Y32      | IMI_A_RXN15  | U34      | IMI_A_TXP7  |
| AV8      | HI6       | Y29      | IMI_A_RXN16  | AB32     | IMI_A_TXP8  |
| AU8      | HI7       | W30      | IMI_A_RXN17  | AC28     | IMI_A_TXP9  |
| AU5      | HI8       | AD36     | IMI_A_RXP0   | V25      | IMI_A_VCCA  |
| AU6      | HI9       | AD33     | IMI_A_RXP1   | AA25     | IMI_A_VCCBG |
| AT6      | HI10      | AB38     | IMI_A_RXP2   | W25      | IMI_A_VSSA  |
| AT2      | HI11      | AB35     | IMI_A_RXP3   | Y25      | IMI_A_VSSBG |
| AT9      | HIRCOMP   | Y37      | IMI_A_RXP4   | AD25     | IMI_B_CLKN  |

Table 7-2. NB Pin List (by Signal Name) (Sheet 6 of 13)

| Ball No. | Signal       | Ball No. | Signal       | Ball No. | Signal       |
|----------|--------------|----------|--------------|----------|--------------|
| AE25     | IMI_B_CLKP   | AF34     | IMI_B_RXP9   | AC7      | IMI_C_LINKN0 |
| AR31     | IMI_B_FRAME  | AP32     | IMI_B_RXP10  | AC10     | IMI_C_LINKN1 |
| AR34     | IMI_B_ICOMPI | AN30     | IMI_B_RXP11  | AD12     | IMI_C_LINKN2 |
| AR37     | IMI_B_ICOMPO | AM31     | IMI_B_RXP12  | AD7      | IMI_C_LINKP0 |
| AG31     | IMI_B_LINKN0 | AL32     | IMI_B_RXP13  | AD10     | IMI_C_LINKP1 |
| AF32     | IMI_B_LINKN1 | AK30     | IMI_B_RXP14  | AE12     | IMI_C_LINKP2 |
| AE30     | IMI_B_LINKN2 | AJ31     | IMI_B_RXP15  | AE11     | IMI_C_RST#   |
| AF31     | IMI_B_LINKP0 | AH32     | IMI_B_RXP16  | U1       | IMI_C_RXN0   |
| AE32     | IMI_B_LINKP1 | AG30     | IMI_B_RXP17  | U4       | IMI_C_RXN1   |
| AD30     | IMI_B_LINKP2 | AP36     | IMI_B_TXN0   | W2       | IMI_C_RXN2   |
| AE29     | IMI_B_RST#   | AP33     | IMI_B_TXN1   | W5       | IMI_C_RXN3   |
| AR38     | IMI_B_RXN0   | AM38     | IMI_B_TXN2   | AA3      | IMI_C_RXN4   |
| AR35     | IMI_B_RXN1   | AM35     | IMI_B_TXN3   | AA6      | IMI_C_RXN5   |
| AN37     | IMI_B_RXN2   | AK37     | IMI_B_TXN4   | AC1      | IMI_C_RXN6   |
| AN34     | IMI_B_RXN3   | AK34     | IMI_B_TXN5   | AC4      | IMI_C_RXN7   |
| AL36     | IMI_B_RXN4   | AH36     | IMI_B_TXN6   | U7       | IMI_C_RXN8   |
| AL33     | IMI_B_RXN5   | AH33     | IMI_B_TXN7   | U10      | IMI_C_RXN9   |
| AJ38     | IMI_B_RXN6   | AF38     | IMI_B_TXN8   | W8       | IMI_C_RXN10  |
| AJ35     | IMI_B_RXN7   | AF35     | IMI_B_TXN9   | W11      | IMI_C_RXN11  |
| AG37     | IMI_B_RXN8   | AN36     | IMI_B_TXP0   | Y7       | IMI_C_RXN12  |
| AG34     | IMI_B_RXN9   | AN33     | IMI_B_TXP1   | Y10      | IMI_C_RXN13  |
| AR32     | IMI_B_RXN10  | AL38     | IMI_B_TXP2   | AA9      | IMI_C_RXN14  |
| AP30     | IMI_B_RXN11  | AL35     | IMI_B_TXP3   | AA12     | IMI_C_RXN15  |
| AN31     | IMI_B_RXN12  | AJ37     | IMI_B_TXP4   | AB8      | IMI_C_RXN16  |
| AM32     | IMI_B_RXN13  | AJ34     | IMI_B_TXP5   | AB11     | IMI_C_RXN17  |
| AL30     | IMI_B_RXN14  | AG36     | IMI_B_TXP6   | V1       | IMI_C_RXP0   |
| AK31     | IMI_B_RXN15  | AG33     | IMI_B_TXP7   | V4       | IMI_C_RXP1   |
| AJ32     | IMI_B_RXN16  | AE38     | IMI_B_TXP8   | Y2       | IMI_C_RXP2   |
| AH30     | IMI_B_RXN17  | AE35     | IMI_B_TXP9   | Y5       | IMI_C_RXP3   |
| AP38     | IMI_B_RXP0   | AB25     | IMI_B_VCCA   | AB3      | IMI_C_RXP4   |
| AP35     | IMI_B_RXP1   | AF24     | IMI_B_VCCBG  | AB6      | IMI_C_RXP5   |
| AM37     | IMI_B_RXP2   | AC25     | IMI_B_VSSA   | AD1      | IMI_C_RXP6   |
| AM34     | IMI_B_RXP3   | AF23     | IMI_B_VSSBG  | AD4      | IMI_C_RXP7   |
| AK36     | IMI_B_RXP4   | U14      | IMI_C_CLKN   | V7       | IMI_C_RXP8   |
| AK33     | IMI_B_RXP5   | T14      | IMI_C_CLKP   | V10      | IMI_C_RXP9   |
| AH38     | IMI_B_RXP6   | U11      | IMI_C_FRAME  | Y8       | IMI_C_RXP10  |
| AH35     | IMI_B_RXP7   | U2       | IMI_C_ICOMPI | Y11      | IMI_C_RXP11  |
| AF37     | IMI_B_RXP8   | U5       | IMI_C_ICOMPO | AA7      | IMI_C_RXP12  |

Table 7-2. NB Pin List (by Signal Name) (Sheet 7 of 13)

| Ball No. | Signal       | Ball No. | Signal       | Ball No. | Signal      |
|----------|--------------|----------|--------------|----------|-------------|
| AA10     | IMI_C_RXP13  | AP9      | IMI_D_LINKP1 | AM8      | IMI_D_RXP17 |
| AB9      | IMI_C_RXP14  | AR8      | IMI_D_LINKP2 | AF1      | IMI_D_TXN0  |
| AB12     | IMI_C_RXP15  | AR7      | IMI_D_RST#   | AF4      | IMI_D_TXN1  |
| AC8      | IMI_C_RXP16  | AE2      | IMI_D_RXN0   | AH2      | IMI_D_TXN2  |
| AC11     | IMI_C_RXP17  | AE5      | IMI_D_RXN1   | AH5      | IMI_D_TXN3  |
| V3       | IMI_C_TXN0   | AG3      | IMI_D_RXN2   | AK3      | IMI_D_TXN4  |
| V6       | IMI_C_TXN1   | AG6      | IMI_D_RXN3   | AK6      | IMI_D_TXN5  |
| Y1       | IMI_C_TXN2   | AJ1      | IMI_D_RXN4   | AM1      | IMI_D_TXN6  |
| Y4       | IMI_C_TXN3   | AJ4      | IMI_D_RXN5   | AM4      | IMI_D_TXN7  |
| AB2      | IMI_C_TXN4   | AL2      | IMI_D_RXN6   | AP2      | IMI_D_TXN8  |
| AB5      | IMI_C_TXN5   | AL5      | IMI_D_RXN7   | AP5      | IMI_D_TXN9  |
| AD3      | IMI_C_TXN6   | AN3      | IMI_D_RXN8   | AG1      | IMI_D_TXP0  |
| AD6      | IMI_C_TXN7   | AN6      | IMI_D_RXN9   | AG4      | IMI_D_TXP1  |
| V9       | IMI_C_TXN8   | AD9      | IMI_D_RXN10  | AJ2      | IMI_D_TXP2  |
| V12      | IMI_C_TXN9   | AE8      | IMI_D_RXN11  | AJ5      | IMI_D_TXP3  |
| W3       | IMI_C_TXP0   | AF7      | IMI_D_RXN12  | AL3      | IMI_D_TXP4  |
| W6       | IMI_C_TXP1   | AG9      | IMI_D_RXN13  | AL6      | IMI_D_TXP5  |
| AA1      | IMI_C_TXP2   | AH8      | IMI_D_RXN14  | AN1      | IMI_D_TXP6  |
| AA4      | IMI_C_TXP3   | AJ7      | IMI_D_RXN15  | AN4      | IMI_D_TXP7  |
| AC2      | IMI_C_TXP4   | AK9      | IMI_D_RXN16  | AR2      | IMI_D_TXP8  |
| AC5      | IMI_C_TXP5   | AL8      | IMI_D_RXN17  | AR5      | IMI_D_TXP9  |
| AE3      | IMI_C_TXP6   | AF2      | IMI_D_RXP0   | AD14     | IMI_D_VCCA  |
| AE6      | IMI_C_TXP7   | AF5      | IMI_D_RXP1   | AC13     | IMI_D_VCCBG |
| W9       | IMI_C_TXP8   | AH3      | IMI_D_RXP2   | AC14     | IMI_D_VSSA  |
| W12      | IMI_C_TXP9   | AH6      | IMI_D_RXP3   | AD13     | IMI_D_VSSBG |
| Y14      | IMI_C_VCCA   | AK1      | IMI_D_RXP4   | AH12     | IMI_HPINT   |
| U13      | IMI_C_VCCBG  | AK4      | IMI_D_RXP5   | AF27     | ITP_TCK     |
| W14      | IMI_C_VSSA   | AM2      | IMI_D_RXP6   | AF28     | ITP_TDI     |
| V13      | IMI_C_VSSBG  | AM5      | IMI_D_RXP7   | AU28     | ITP_TDO     |
| AB14     | IMI_D_CLKN   | AP3      | IMI_D_RXP8   | AH28     | ITP_TMS     |
| AA14     | IMI_D_CLKP   | AP6      | IMI_D_RXP9   | AR25     | ITP_TRST#   |
| AF11     | IMI_D_FRAME  | AE9      | IMI_D_RXP10  | J19      | P1V5        |
| AR1      | IMI_D_ICOMP1 | AF8      | IMI_D_RXP11  | N18      | P1V5        |
| AR4      | IMI_D_ICOMP0 | AG7      | IMI_D_RXP12  | N20      | P1V5        |
| AM7      | IMI_D_LINKN0 | AH9      | IMI_D_RXP13  | N22      | P1V5        |
| AN9      | IMI_D_LINKN1 | AJ8      | IMI_D_RXP14  | P15      | P1V5        |
| AP8      | IMI_D_LINKN2 | AK7      | IMI_D_RXP15  | P17      | P1V5        |
| AN7      | IMI_D_LINKP0 | AL9      | IMI_D_RXP16  | P19      | P1V5        |

Table 7-2. NB Pin List (by Signal Name) (Sheet 8 of 13)

| Ball No. | Signal | Ball No. | Signal | Ball No. | Signal |
|----------|--------|----------|--------|----------|--------|
| P21      | P1V5   | Y19      | P1V5   | AF3      | P1V5   |
| P23      | P1V5   | Y21      | P1V5   | AF9      | P1V5   |
| P25      | P1V5   | Y23      | P1V5   | AF33     | P1V5   |
| R14      | P1V5   | Y27      | P1V5   | AH1      | P1V5   |
| R16      | P1V5   | Y33      | P1V5   | AH7      | P1V5   |
| R18      | P1V5   | AA16     | P1V5   | AH13     | P1V5   |
| R20      | P1V5   | AA18     | P1V5   | AH19     | P1V5   |
| R22      | P1V5   | AA20     | P1V5   | AH25     | P1V5   |
| R24      | P1V5   | AA22     | P1V5   | AH31     | P1V5   |
| T15      | P1V5   | AA24     | P1V5   | AH37     | P1V5   |
| T17      | P1V5   | AB1      | P1V5   | AK5      | P1V5   |
| T19      | P1V5   | AB7      | P1V5   | AK11     | P1V5   |
| T21      | P1V5   | AB13     | P1V5   | AK17     | P1V5   |
| T23      | P1V5   | AB15     | P1V5   | AK23     | P1V5   |
| T25      | P1V5   | AB17     | P1V5   | AK29     | P1V5   |
| U16      | P1V5   | AB19     | P1V5   | AK35     | P1V5   |
| U18      | P1V5   | AB21     | P1V5   | AM3      | P1V5   |
| U20      | P1V5   | AB23     | P1V5   | AM9      | P1V5   |
| U22      | P1V5   | AB31     | P1V5   | AM15     | P1V5   |
| U24      | P1V5   | AB37     | P1V5   | AM21     | P1V5   |
| V5       | P1V5   | AC16     | P1V5   | AM27     | P1V5   |
| V11      | P1V5   | AC18     | P1V5   | AM33     | P1V5   |
| V15      | P1V5   | AC20     | P1V5   | AP1      | P1V5   |
| V17      | P1V5   | AC22     | P1V5   | AP7      | P1V5   |
| V19      | P1V5   | AC24     | P1V5   | AP13     | P1V5   |
| V21      | P1V5   | AD5      | P1V5   | AP19     | P1V5   |
| V23      | P1V5   | AD11     | P1V5   | AP25     | P1V5   |
| V29      | P1V5   | AD15     | P1V5   | AP31     | P1V5   |
| V35      | P1V5   | AD17     | P1V5   | AP37     | P1V5   |
| W16      | P1V5   | AD19     | P1V5   | AT5      | P1V5   |
| W18      | P1V5   | AD21     | P1V5   | AT11     | P1V5   |
| W20      | P1V5   | AD23     | P1V5   | AT17     | P1V5   |
| W22      | P1V5   | AD29     | P1V5   | AT23     | P1V5   |
| W24      | P1V5   | AD35     | P1V5   | AT29     | P1V5   |
| Y3       | P1V5   | AE16     | P1V5   | AU31     | P1V5   |
| Y9       | P1V5   | AE18     | P1V5   | AU34     | P1V5   |
| Y15      | P1V5   | AE20     | P1V5   | AV9      | P1V5   |
| Y17      | P1V5   | AE22     | P1V5   | AV15     | P1V5   |

Table 7-2. NB Pin List (by Signal Name) (Sheet 9 of 13)

| Ball No. | Signal              | Ball No. | Signal   | Ball No. | Signal |
|----------|---------------------|----------|----------|----------|--------|
| AV21     | P1V5                | AJ29     | RESERVED | A31      | VSS    |
| AV27     | P1V5                | AK10     | RESERVED | A34      | VSS    |
| AG10     | PME_OUT/EXP1_WIDTH2 | AL11     | RESERVED | A36      | VSS    |
| AG28     | PWRGOOD             | AL12     | RESERVED | B2       | VSS    |
| D35      | RESERVED            | AM10     | RESERVED | B6       | VSS    |
| G9       | RESERVED            | AM11     | RESERVED | B12      | VSS    |
| G30      | RESERVED            | AN10     | RESERVED | B18      | VSS    |
| H8       | RESERVED            | AP11     | RESERVED | B24      | VSS    |
| H10      | RESERVED            | AR10     | RESERVED | B30      | VSS    |
| H11      | RESERVED            | AR11     | RESERVED | B37      | VSS    |
| H19      | RESERVED            | AR13     | RESERVED | C1       | VSS    |
| H20      | RESERVED            | AT10     | RESERVED | C5       | VSS    |
| H26      | RESERVED            | AT18     | RESERVED | C8       | VSS    |
| H28      | RESERVED            | AU11     | RESERVED | C11      | VSS    |
| H29      | RESERVED            | AU12     | RESERVED | C14      | VSS    |
| H32      | RESERVED            | AV11     | RESERVED | C17      | VSS    |
| K12      | RESERVED            | AH29     | RSTIN#   | C20      | VSS    |
| L19      | RESERVED            | AF25     | SMBCLK   | C23      | VSS    |
| M19      | RESERVED            | AR28     | SMBDATA  | C26      | VSS    |
| P26      | RESERVED            | L22      | TESTH1   | C29      | VSS    |
| R13      | RESERVED            | M18      | TESTLO1  | C32      | VSS    |
| U8       | RESERVED            | L23      | TESTLO2  | C35      | VSS    |
| U29      | RESERVED            | L20      | TESTLO3  | C38      | VSS    |
| W26      | RESERVED            | J18      | TESTLO4  | D4       | VSS    |
| Y13      | RESERVED            | M24      | TESTLO5  | D10      | VSS    |
| Y26      | RESERVED            | AF26     | V3REF0   | D16      | VSS    |
| AA13     | RESERVED            | AJ10     | V3REF1   | D22      | VSS    |
| AB26     | RESERVED            | K21      | VCCA     | D28      | VSS    |
| AC26     | RESERVED            | A3       | VSS      | D34      | VSS    |
| AE14     | RESERVED            | A4       | VSS      | E3       | VSS    |
| AF13     | RESERVED            | A7       | VSS      | E6       | VSS    |
| AF14     | RESERVED            | A10      | VSS      | E9       | VSS    |
| AF15     | RESERVED            | A13      | VSS      | E12      | VSS    |
| AG12     | RESERVED            | A16      | VSS      | E15      | VSS    |
| AG13     | RESERVED            | A19      | VSS      | E18      | VSS    |
| AG16     | RESERVED            | A22      | VSS      | E21      | VSS    |
| AJ11     | RESERVED            | A25      | VSS      | E24      | VSS    |
| AJ28     | RESERVED            | A28      | VSS      | E27      | VSS    |

Table 7-2. NB Pin List (by Signal Name) (Sheet 10 of 13)

| Ball No. | Signal | Ball No. | Signal | Ball No. | Signal |
|----------|--------|----------|--------|----------|--------|
| E30      | VSS    | J29      | VSS    | N23      | VSS    |
| E33      | VSS    | J32      | VSS    | N25      | VSS    |
| E36      | VSS    | J35      | VSS    | N28      | VSS    |
| F2       | VSS    | J38      | VSS    | N31      | VSS    |
| F8       | VSS    | K4       | VSS    | N34      | VSS    |
| F14      | VSS    | K10      | VSS    | N37      | VSS    |
| F20      | VSS    | K16      | VSS    | P6       | VSS    |
| F26      | VSS    | K22      | VSS    | P12      | VSS    |
| F32      | VSS    | K28      | VSS    | P14      | VSS    |
| F38      | VSS    | K34      | VSS    | P16      | VSS    |
| G1       | VSS    | L3       | VSS    | P18      | VSS    |
| G4       | VSS    | L6       | VSS    | P20      | VSS    |
| G7       | VSS    | L9       | VSS    | P22      | VSS    |
| G10      | VSS    | L12      | VSS    | P24      | VSS    |
| G13      | VSS    | L15      | VSS    | P30      | VSS    |
| G16      | VSS    | L18      | VSS    | P36      | VSS    |
| G19      | VSS    | L21      | VSS    | R2       | VSS    |
| G22      | VSS    | L24      | VSS    | R5       | VSS    |
| G25      | VSS    | L27      | VSS    | R8       | VSS    |
| G28      | VSS    | L30      | VSS    | R11      | VSS    |
| G31      | VSS    | L33      | VSS    | R15      | VSS    |
| G34      | VSS    | L36      | VSS    | R17      | VSS    |
| G37      | VSS    | M2       | VSS    | R19      | VSS    |
| H6       | VSS    | M8       | VSS    | R21      | VSS    |
| H12      | VSS    | M14      | VSS    | R23      | VSS    |
| H18      | VSS    | M20      | VSS    | R25      | VSS    |
| H24      | VSS    | M26      | VSS    | R26      | VSS    |
| H30      | VSS    | M32      | VSS    | R29      | VSS    |
| H36      | VSS    | M38      | VSS    | R32      | VSS    |
| J2       | VSS    | N1       | VSS    | R35      | VSS    |
| J5       | VSS    | N4       | VSS    | R38      | VSS    |
| J8       | VSS    | N7       | VSS    | T4       | VSS    |
| J11      | VSS    | N10      | VSS    | T10      | VSS    |
| J14      | VSS    | N13      | VSS    | T16      | VSS    |
| J17      | VSS    | N16      | VSS    | T18      | VSS    |
| J20      | VSS    | N17      | VSS    | T20      | VSS    |
| J23      | VSS    | N19      | VSS    | T22      | VSS    |
| J26      | VSS    | N21      | VSS    | T24      | VSS    |

Table 7-2. NB Pin List (by Signal Name) (Sheet 11 of 13)

| Ball No. | Signal | Ball No. | Signal | Ball No. | Signal |
|----------|--------|----------|--------|----------|--------|
| T28      | VSS    | W34      | VSS    | AC15     | VSS    |
| T34      | VSS    | W37      | VSS    | AC17     | VSS    |
| U3       | VSS    | Y6       | VSS    | AC19     | VSS    |
| U6       | VSS    | Y12      | VSS    | AC21     | VSS    |
| U9       | VSS    | Y16      | VSS    | AC23     | VSS    |
| U12      | VSS    | Y18      | VSS    | AC27     | VSS    |
| U15      | VSS    | Y20      | VSS    | AC30     | VSS    |
| U17      | VSS    | Y22      | VSS    | AC33     | VSS    |
| U19      | VSS    | Y24      | VSS    | AC36     | VSS    |
| U21      | VSS    | Y30      | VSS    | AD2      | VSS    |
| U23      | VSS    | Y36      | VSS    | AD8      | VSS    |
| U27      | VSS    | AA2      | VSS    | AD16     | VSS    |
| U30      | VSS    | AA5      | VSS    | AD18     | VSS    |
| U33      | VSS    | AA8      | VSS    | AD20     | VSS    |
| U36      | VSS    | AA11     | VSS    | AD22     | VSS    |
| V2       | VSS    | AA15     | VSS    | AD24     | VSS    |
| V8       | VSS    | AA17     | VSS    | AD26     | VSS    |
| V14      | VSS    | AA19     | VSS    | AD32     | VSS    |
| V16      | VSS    | AA21     | VSS    | AD38     | VSS    |
| V18      | VSS    | AA23     | VSS    | AE1      | VSS    |
| V20      | VSS    | AA26     | VSS    | AE4      | VSS    |
| V22      | VSS    | AA29     | VSS    | AE7      | VSS    |
| V24      | VSS    | AA32     | VSS    | AE10     | VSS    |
| V26      | VSS    | AA35     | VSS    | AE13     | VSS    |
| V32      | VSS    | AA38     | VSS    | AE15     | VSS    |
| V38      | VSS    | AB4      | VSS    | AE17     | VSS    |
| W1       | VSS    | AB10     | VSS    | AE19     | VSS    |
| W4       | VSS    | AB16     | VSS    | AE21     | VSS    |
| W7       | VSS    | AB18     | VSS    | AE23     | VSS    |
| W10      | VSS    | AB20     | VSS    | AE28     | VSS    |
| W13      | VSS    | AB22     | VSS    | AE31     | VSS    |
| W15      | VSS    | AB24     | VSS    | AE34     | VSS    |
| W17      | VSS    | AB28     | VSS    | AE37     | VSS    |
| W19      | VSS    | AB34     | VSS    | AF6      | VSS    |
| W21      | VSS    | AC3      | VSS    | AF12     | VSS    |
| W23      | VSS    | AC6      | VSS    | AF30     | VSS    |
| W28      | VSS    | AC9      | VSS    | AF36     | VSS    |
| W31      | VSS    | AC12     | VSS    | AG2      | VSS    |

Table 7-2. NB Pin List (by Signal Name) (Sheet 12 of 13)

| Ball No. | Signal | Ball No. | Signal | Ball No. | Signal |
|----------|--------|----------|--------|----------|--------|
| AG5      | VSS    | AL7      | VSS    | AR9      | VSS    |
| AG8      | VSS    | AL10     | VSS    | AR12     | VSS    |
| AG11     | VSS    | AL13     | VSS    | AR15     | VSS    |
| AG14     | VSS    | AL16     | VSS    | AR18     | VSS    |
| AG17     | VSS    | AL19     | VSS    | AR21     | VSS    |
| AG20     | VSS    | AL22     | VSS    | AR24     | VSS    |
| AG23     | VSS    | AL25     | VSS    | AR27     | VSS    |
| AG26     | VSS    | AL28     | VSS    | AR30     | VSS    |
| AG29     | VSS    | AL31     | VSS    | AR33     | VSS    |
| AG32     | VSS    | AL34     | VSS    | AR36     | VSS    |
| AG35     | VSS    | AL37     | VSS    | AT1      | VSS    |
| AG38     | VSS    | AM6      | VSS    | AT8      | VSS    |
| AH4      | VSS    | AM12     | VSS    | AT14     | VSS    |
| AH10     | VSS    | AM18     | VSS    | AT20     | VSS    |
| AH16     | VSS    | AM24     | VSS    | AT26     | VSS    |
| AH22     | VSS    | AM30     | VSS    | AT32     | VSS    |
| AH34     | VSS    | AM36     | VSS    | AT35     | VSS    |
| AJ3      | VSS    | AN2      | VSS    | AT38     | VSS    |
| AJ6      | VSS    | AN5      | VSS    | AU2      | VSS    |
| AJ9      | VSS    | AN8      | VSS    | AU7      | VSS    |
| AJ12     | VSS    | AN11     | VSS    | AU10     | VSS    |
| AJ15     | VSS    | AN14     | VSS    | AU13     | VSS    |
| AJ18     | VSS    | AN17     | VSS    | AU16     | VSS    |
| AJ21     | VSS    | AN20     | VSS    | AU19     | VSS    |
| AJ24     | VSS    | AN23     | VSS    | AU22     | VSS    |
| AJ27     | VSS    | AN26     | VSS    | AU25     | VSS    |
| AJ30     | VSS    | AN29     | VSS    | AU37     | VSS    |
| AJ33     | VSS    | AN32     | VSS    | AV3      | VSS    |
| AJ36     | VSS    | AN35     | VSS    | AV6      | VSS    |
| AK2      | VSS    | AN38     | VSS    | AV12     | VSS    |
| AK8      | VSS    | AP4      | VSS    | AV18     | VSS    |
| AK14     | VSS    | AP10     | VSS    | AV24     | VSS    |
| AK20     | VSS    | AP16     | VSS    | AV30     | VSS    |
| AK26     | VSS    | AP22     | VSS    | AV33     | VSS    |
| AK32     | VSS    | AP28     | VSS    | AV36     | VSS    |
| AK38     | VSS    | AP34     | VSS    | K20      | VSSA   |
| AL1      | VSS    | AR3      | VSS    | B9       | VTT    |
| AL4      | VSS    | AR6      | VSS    | B15      | VTT    |

Table 7-2. NB Pin List (by Signal Name) (Sheet 13 of 13)

| Ball No. | Signal | Ball No. | Signal   | Ball No. | Signal      |
|----------|--------|----------|----------|----------|-------------|
| B21      | VTT    | K13      | VTT      | AV34     | XDP_D1#     |
| B27      | VTT    | K19      | VTT      | AU33     | XDP_D2#     |
| B33      | VTT    | K25      | VTT      | AV32     | XDP_D3#     |
| D7       | VTT    | K31      | VTT      | AV31     | XDP_D4#     |
| D13      | VTT    | K37      | VTT      | AU30     | XDP_D5#     |
| D19      | VTT    | M5       | VTT      | AV29     | XDP_D6#     |
| D25      | VTT    | M11      | VTT      | AV28     | XDP_D7#     |
| D31      | VTT    | M17      | VTT      | AU35     | XDP_D8#     |
| F5       | VTT    | M23      | VTT      | AT34     | XDP_D9#     |
| F11      | VTT    | M29      | VTT      | AT33     | XDP_D10#    |
| F17      | VTT    | M35      | VTT      | AU32     | XDP_D11#    |
| F23      | VTT    | P3       | VTT      | AT31     | XDP_D12#    |
| F29      | VTT    | P9       | VTT      | AT30     | XDP_D13#    |
| F35      | VTT    | P27      | VTT      | AU29     | XDP_D14#    |
| H3       | VTT    | P33      | VTT      | AT28     | XDP_D15#    |
| H9       | VTT    | T1       | VTT      | AT37     | XDP_DSTBN   |
| H15      | VTT    | T7       | VTT      | AT36     | XDP_DSTBP   |
| H21      | VTT    | T13      | VTT      | AN28     | XDP_ODTCRES |
| H27      | VTT    | T31      | VTT      | AU36     | XDP_RDY#    |
| H33      | VTT    | T37      | VTT      | AP29     | XDP_SLWCRES |
| K1       | VTT    | AR29     | XDP_CRES |          |             |
| K7       | VTT    | AV35     | XDP_D0#  |          |             |

## 7.2 Intel® E8501 Chipset North Bridge (NB) Mechanical Package Information

For the NB Mechanical Package drawing, refer to Figure 7-5 through Figure 7-7.

For detailed information about the XMB Mechanical specification, refer to the *Intel® E8501 Chipset eXternal Memory Bridge (XMB) Datasheet*.

For detailed information about the ICH5 mechanical specifications, refer to the *Intel® 82801EB I/O Controller Hub 5 (ICH5) and Intel® 82801ER I/O Controller Hub 5 (ICH5R) Datasheet*. The ICH5 Datasheet is available at: <http://developer.intel.com/design/chipsets/datashts>.

For detailed information about the Intel 6700PXH/6702PXH 64-bit PCI Hub ballout and pin list, refer to the *Intel® 6700PXH 64-bit PCI Hub and Intel® 6702PXH 64-bit PCI Hub Datasheet*.

**Figure 7-5. Intel® E8501 Chipset North Bridge (NB) Package Dimensions (Top View)**

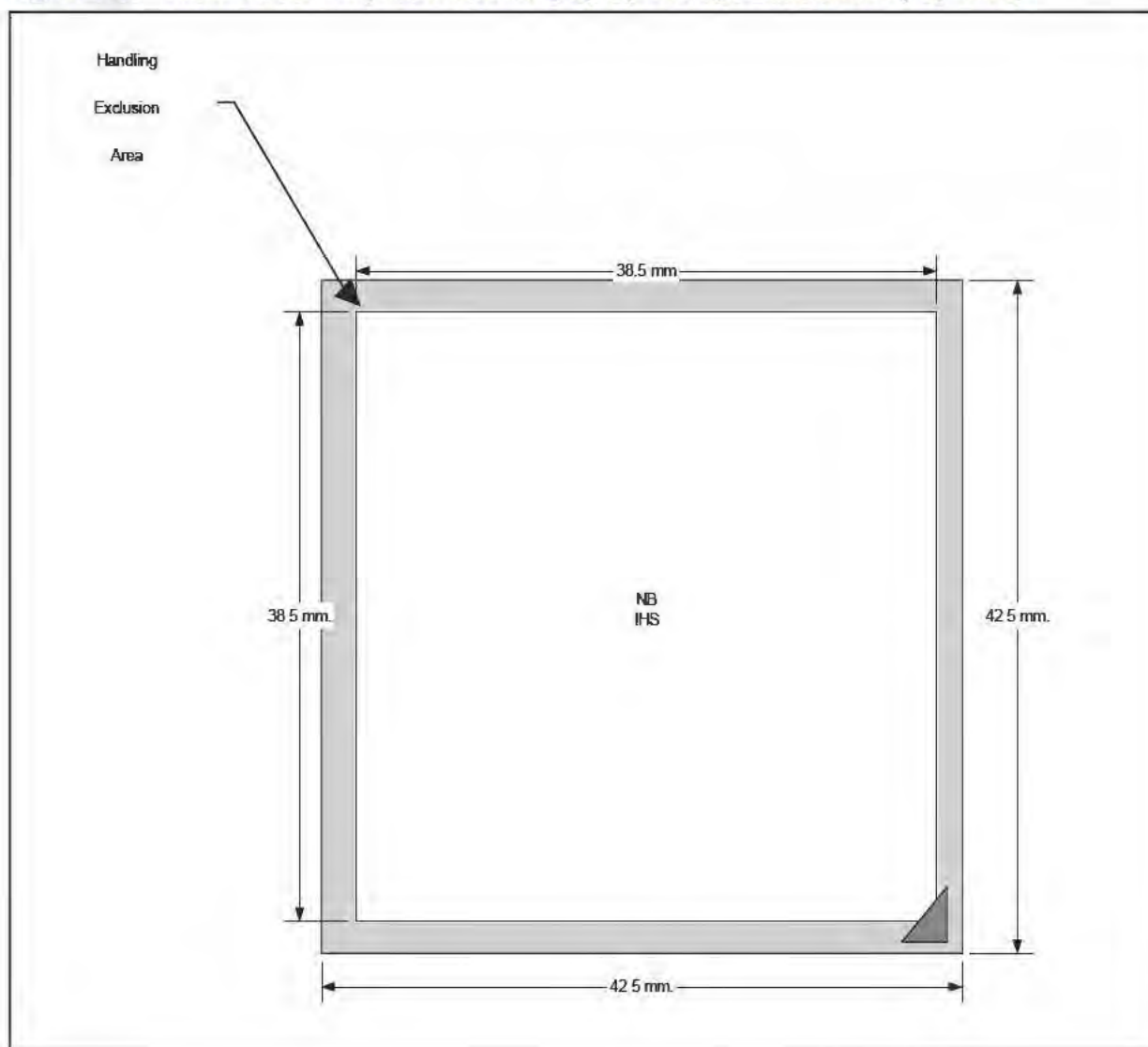


Figure 7-6. NB Package Dimensions (Side View)

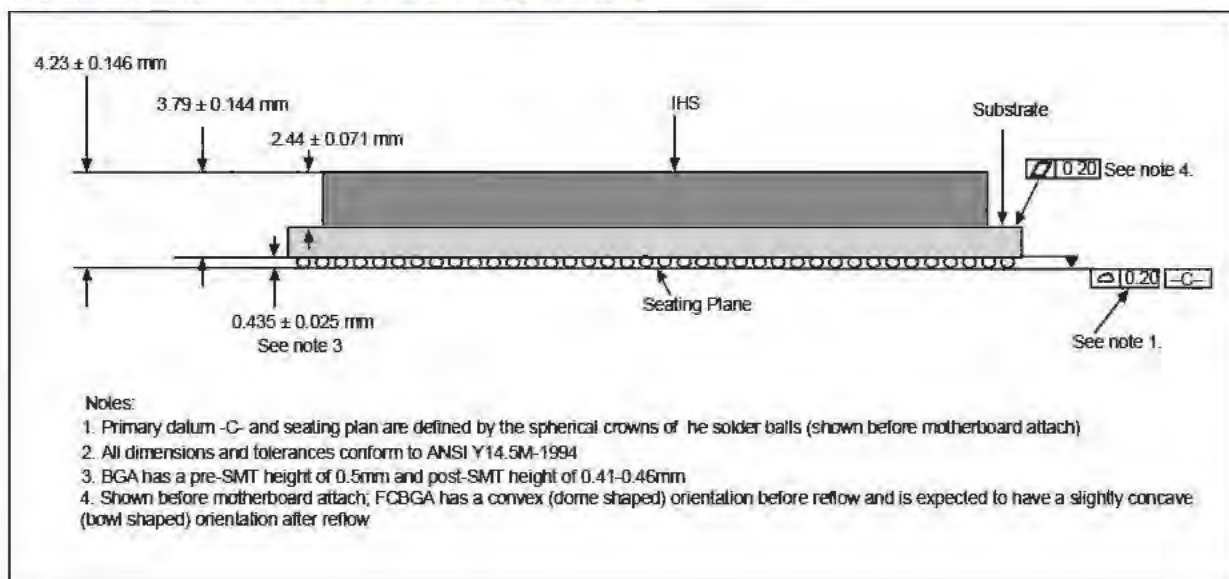
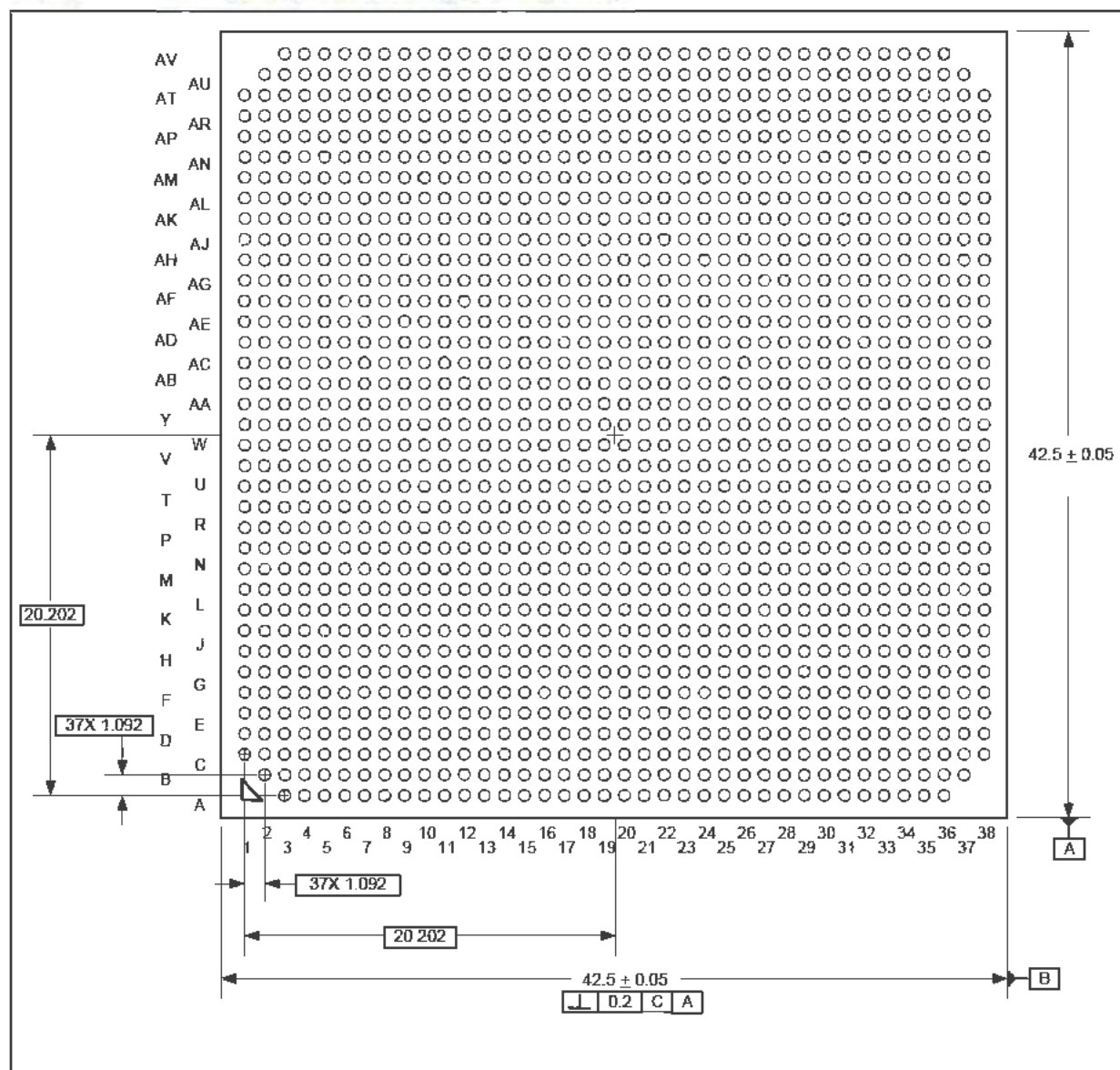


**Figure 7-7. NB Package Dimensions (Bottom View)**



27