# 阅读申明

1.本站收集的数据手册和产品资料都来自互联网，版权归原作者所有。如读者和版权方有任何异议请及时告之，我们将妥善解决。
2.本站提供的中文数据手册是英文数据手册的中文翻译，其目的是协助用户阅读，该译文无法自动跟随原稿更新，同时也可能存在翻译上的不当。建议读者以英文原稿为参考以便获得更精准的信息。
3.本站提供的产品资料，来自厂商的技术支持或者使用者的心得体会等，其内容可能存在描叙上的差异，建议读者做出适当判断。
4.如需与我们联系，请发邮件到marketing@iczoom.com，主题请标有“数据手册”字样。

# Read Statement

1. The datasheets and other product information on the site are all from network reference or other public materials, and the copyright belongs to the original author and original published source. If readers and copyright owners have any objections, please contact us and we will deal with it in a timely manner.

2. The Chinese datasheets provided on the website is a Chinese translation of the English datasheets. Its purpose is for reader’ s learning exchange only and do not involve commercial purposes. The translation cannot be automatically updated with the original manuscript, and there may also be improper translations. Readers are advised to use the English manuscript as a reference for more accurate information.

3. All product information provided on the website refer to solutions from manufacturers’ technical support or users the contents may have differences in description, and readers are advised to take the original article as the standard.

4. If you have any questions, please contact us at marketing@iczoom.com and mark the subject with “Datasheets”.

- **Compliant With PCI Hot-Plug Specification, Revision 1.0**
- **Supports up to Four Independently Controlled Hot-Plug Slots**
- **Provides Register Accessing Through Both Generic Parallel Bus and Two-Wire Serial Interface**
- **Provides Interrupt and Event Status/Enable State Compliant With ACPI Specification 1.0**
- **Provides an Automatic Bus Connection Sequencing Feature**
- **Supports 66-MHz PCI Clock Frequency**
- **Features Two Attention Indicators With Variable LED Blinking Rates per Slot**
- **Provides an Easy Scheme to Cascade the HPC3130A for Compact PCI Applications**
- **Provides Card Detection Mechanism Independent of PCI Present Signals for Advanced Card Protection**
- **Provides Path to Guarantee Idle State During PCI Bus Connections**
- **Fabricated in Advanced Low-Power CMOS Process**
- **Features a CBT Switch† Control Feature for $\overline{\text{REQ64}}$ Implementation**
- **Package Options:**
  - **120-pin QFP Package**
  - **128-pin LQFP Package**
  - **144-pin LQFP Package**

**Table of Contents**



Please be aware that an important notice concerning availability, standard warranty, and use in critical applications of Texas Instruments semiconductor products and disclaimers thereto appears at the end of this data sheet.

† Contact Texas Instrument's Bus Interface product group for information related to CBT switches.

## description

The Texas Instruments HPC3130A is a peripheral component interconnect (PCI) hot-plug controller, compliant with *PCI Hot-Plug Specification, Revision 1.0*. This device supports hot insertion/removal of up to four hot-plug slots on a PCI bus, provides a 64-bit data path in any of the four hot-plug slots, and supports 66-MHz systems for two slots.

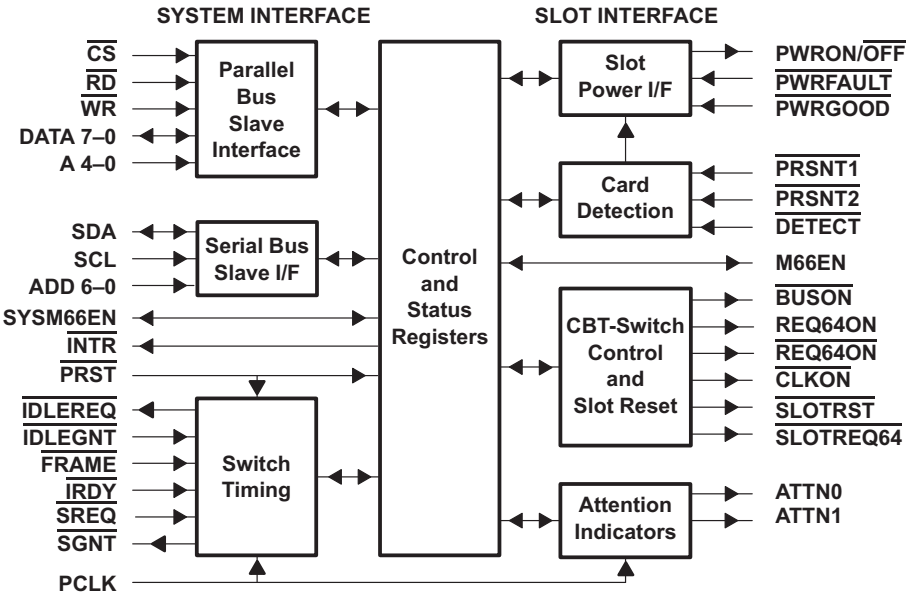
The primary function of the HPC3130A is to allow noninterfering hot-plug slot connection/disconnection with the other PCI devices on the bus. The HPC3130A provides automatic bus connection sequencing and supports a protocol for connection during bus idle conditions. It also supports an interrupt pin to report hot-plug slot events. The interrupt event status and enable state are compliant with the *Advanced Configuration and Power Interface (ACPI) Specification*.

Internal registers may be accessed through either a two-signal serial interface or a generic parallel bus. The serial interface slave decoding circuit supports up to eight different controllers or other serial bus devices with the same system base. Decoding through the parallel interface supports multiple controllers with external chip-select logic. Two double-words of configuration and control registers are provided per slot. As a result, the HPC3130A decodes an address range of 32 bytes.

An advanced complementary metal-oxide semiconductor (CMOS) process provides low system power consumption while operating at PCI clock rates up to 66 MHz.

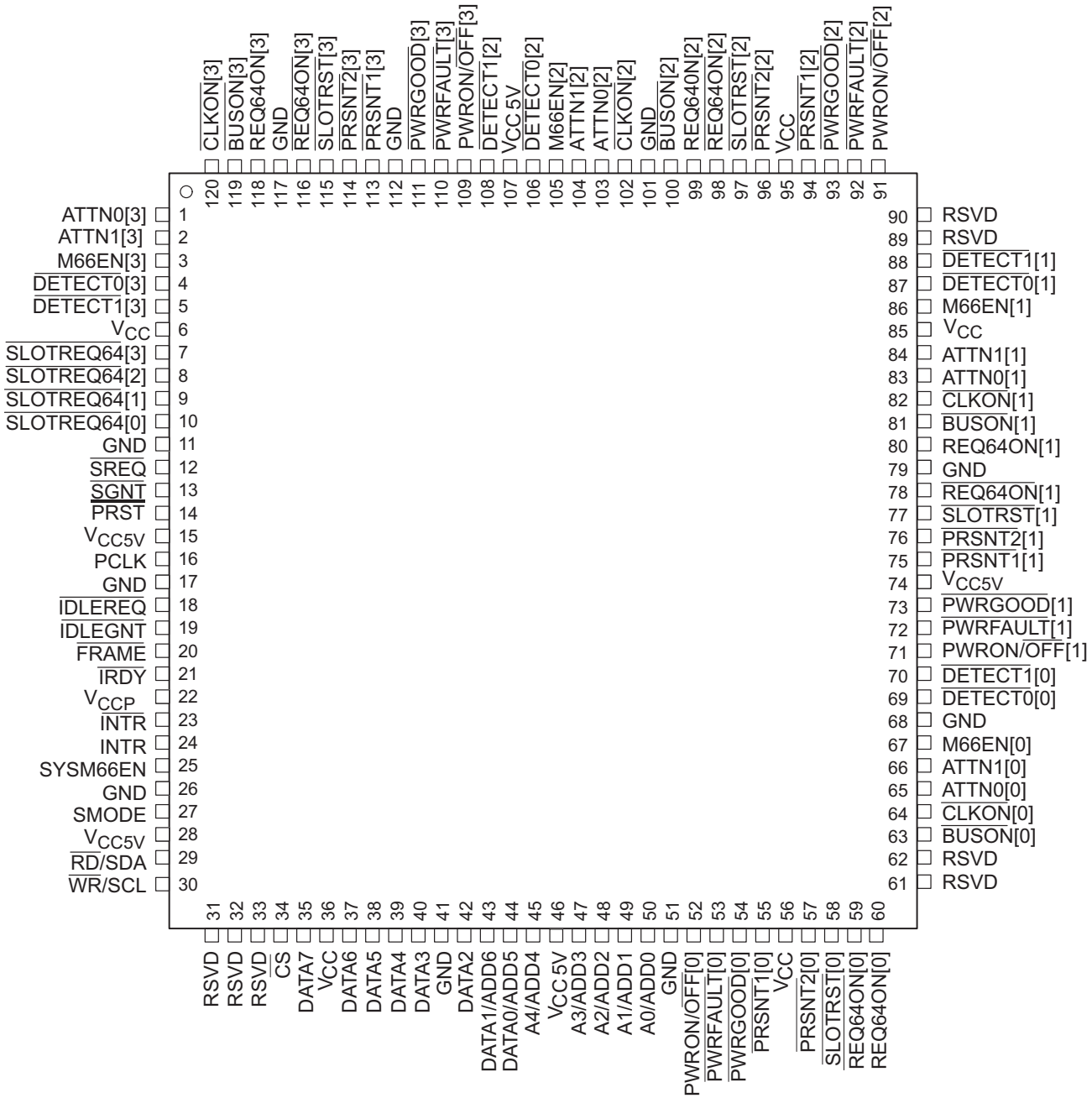
## functional block diagram

A simplified block diagram of the HPC3130A is provided below. The block diagram illustrates the HPC3130A functionality on a per slot basis. The SMODE chip input, not shown, is used for terminal multiplexing of the serial and parallel bus slave interfaces.



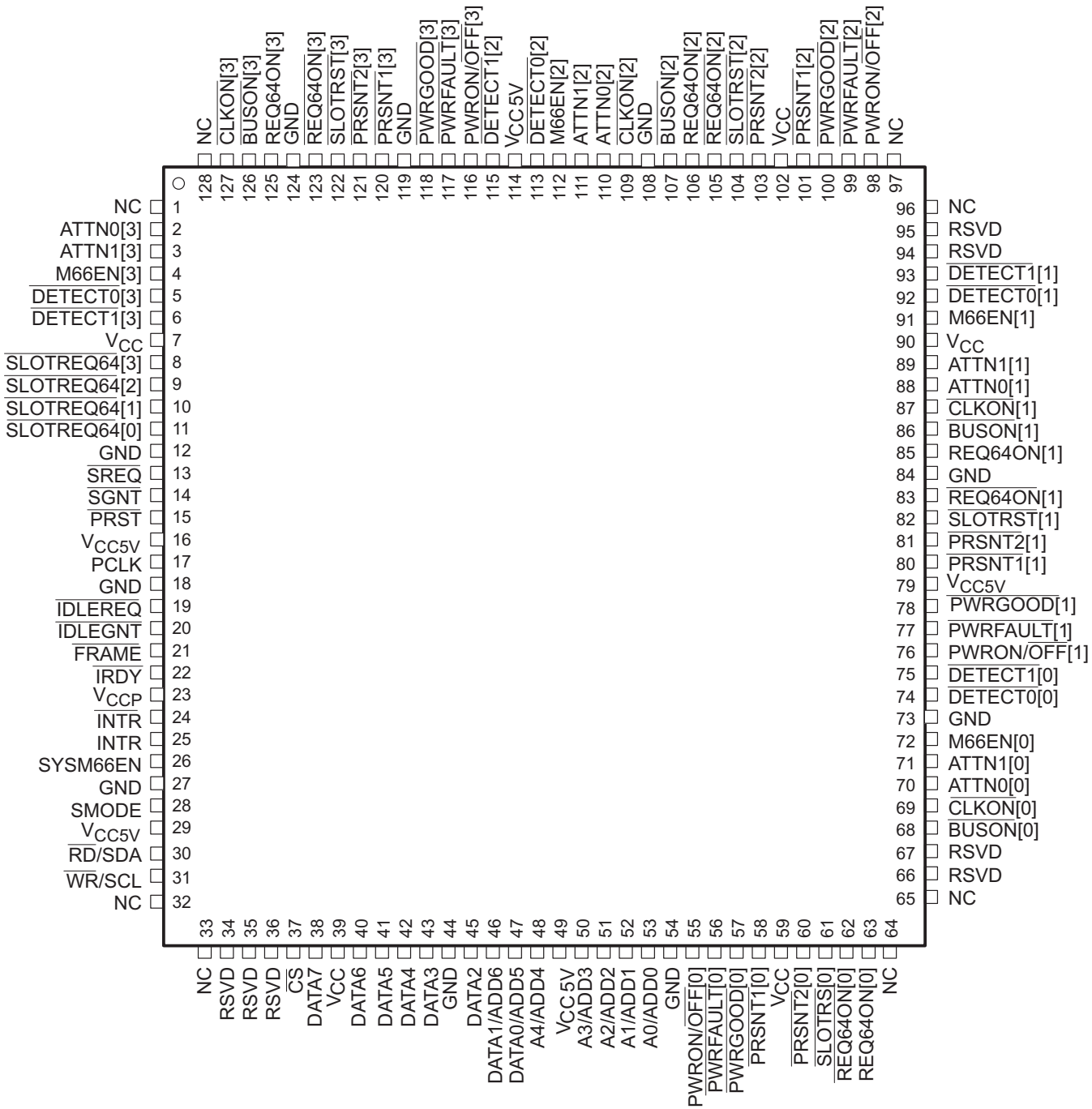
## terminal assignments (120-terminal package)

**PBM QUAD FLAT PACKAGE**
**TOP VIEW**



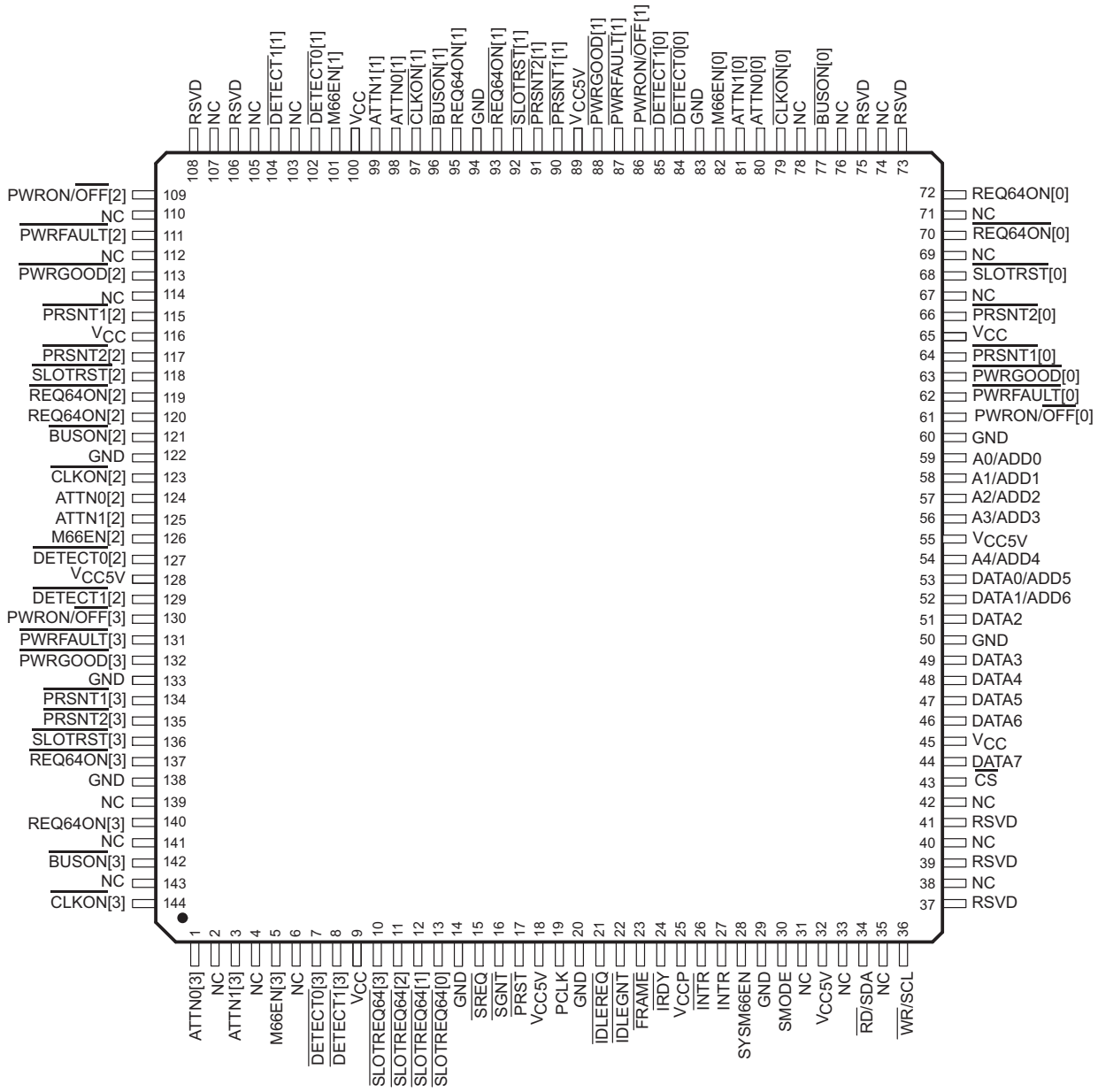
## terminal assignments (128-terminal package)

**PBK LOW-PROFILE QUAD FLAT PACKAGE**
**TOP VIEW**



## terminal assignments (144-terminal package)

**PGE LOW-PROFILE QUAD FLAT PACKAGE**
**TOP VIEW**



## signal names by terminal name

**Table 1. Signals Sorted Alphabetically by Terminal Name (120-Terminal Package)**

| TERMINAL NAME | NO. | TERMINAL NAME | NO. | TERMINAL NAME | NO. |
|---|---|---|---|---|---|
| A0/ADD0 | 50 | GND | 17 | $\overline{\text{RD}}$/SDA | 29 |
| A1/ADD1 | 49 | GND | 26 | $\overline{\text{REQ64ON}}$[0] | 59 |
| A2/ADD2 | 48 | GND | 41 | $\overline{\text{REQ64ON}}$[1] | 78 |
| A3/ADD3 | 47 | GND | 51 | $\overline{\text{REQ64ON}}$[2] | 98 |
| A4/ADD4 | 45 | GND | 68 | $\overline{\text{REQ64ON}}$[3] | 116 |
| ATTN0[0] | 65 | GND | 79 | REQ64ON[0] | 60 |
| ATTN0[1] | 83 | GND | 101 | REQ64ON[1] | 80 |
| ATTN0[2] | 103 | GND | 112 | REQ64ON[2] | 99 |
| ATTN0[3] | 1 | GND | 117 | REQ64ON[3] | 118 |
| ATTN1[0] | 66 | $\overline{\text{IDLEGNT}}$ | 19 | RSVD | 31 |
| ATTN1[1] | 84 | $\overline{\text{IDLEREQ}}$ | 18 | RSVD | 32 |
| ATTN1[2] | 104 | INTR | 24 | RSVD | 33 |
| ATTN1[3] | 2 | $\overline{\text{INTR}}$ | 23 | RSVD | 61 |
| $\overline{\text{BUSON}}$[0] | 63 | $\overline{\text{IRDY}}$ | 21 | RSVD | 62 |
| $\overline{\text{BUSON}}$[1] | 81 | M66EN[0] | 67 | RSVD | 89 |
| $\overline{\text{BUSON}}$[2] | 100 | M66EN[1] | 86 | RSVD | 90 |
| $\overline{\text{BUSON}}$[3] | 119 | M66EN[2] | 105 | $\overline{\text{SGNT}}$ | 13 |
| $\overline{\text{CLKON}}$[0] | 64 | M66EN[3] | 3 | $\overline{\text{SLOTREQ64}}$[0] | 10 |
| $\overline{\text{CLKON}}$[1] | 82 | PCLK | 16 | $\overline{\text{SLOTREQ64}}$[1] | 9 |
| $\overline{\text{CLKON}}$[2] | 102 | $\overline{\text{PRSNT1}}$[0] | 55 | $\overline{\text{SLOTREQ64}}$[2] | 8 |
| $\overline{\text{CLKON}}$[3] | 120 | $\overline{\text{PRSNT1}}$[1] | 75 | $\overline{\text{SLOTREQ64}}$[3] | 7 |
| $\overline{\text{CS}}$ | 34 | $\overline{\text{PRSNT1}}$[2] | 94 | $\overline{\text{SLOTRST}}$[0] | 58 |
| DATA0/ADD5 | 44 | $\overline{\text{PRSNT1}}$[3] | 113 | $\overline{\text{SLOTRST}}$[1] | 77 |
| DATA1/ADD6 | 43 | $\overline{\text{PRSNT2}}$[0] | 57 | $\overline{\text{SLOTRST}}$[2] | 97 |
| DATA2 | 42 | $\overline{\text{PRSNT2}}$[1] | 76 | $\overline{\text{SLOTRST}}$[3] | 115 |
| DATA3 | 40 | $\overline{\text{PRSNT2}}$[2] | 96 | SMODE | 27 |
| DATA4 | 39 | $\overline{\text{PRSNT2}}$[3] | 114 | $\overline{\text{SREQ}}$ | 12 |
| DATA5 | 38 | $\overline{\text{PRST}}$ | 14 | SYSM66EN | 25 |
| DATA6 | 37 | $\overline{\text{PWRFAULT}}$[0] | 53 | $V_{CC}$ | 6 |
| DATA7 | 35 | $\overline{\text{PWRFAULT}}$[1] | 72 | $V_{CC}$ | 36 |
| $\overline{\text{DETECT0}}$[0] | 69 | $\overline{\text{PWRFAULT}}$[2] | 92 | $V_{CC}$ | 56 |
| $\overline{\text{DETECT0}}$[1] | 87 | $\overline{\text{PWRFAULT}}$[3] | 110 | $V_{CC}$ | 85 |
| $\overline{\text{DETECT0}}$[2] | 106 | $\overline{\text{PWRGOOD}}$[0] | 54 | $V_{CC}$ | 95 |
| $\overline{\text{DETECT0}}$[3] | 4 | $\overline{\text{PWRGOOD}}$[1] | 73 | $V_{CC5V}$ | 15 |
| $\overline{\text{DETECT1}}$[0] | 70 | $\overline{\text{PWRGOOD}}$[2] | 93 | $V_{CC5V}$ | 28 |
| $\overline{\text{DETECT1}}$[1] | 88 | $\overline{\text{PWRGOOD}}$[3] | 111 | $V_{CC5V}$ | 46 |
| $\overline{\text{DETECT1}}$[2] | 108 | PWRON/$\overline{\text{OFF}}$[0] | 52 | $V_{CC5V}$ | 74 |
| $\overline{\text{DETECT1}}$[3] | 5 | PWRON/$\overline{\text{OFF}}$[1] | 71 | $V_{CC5V}$ | 107 |
| $\overline{\text{FRAME}}$ | 20 | PWRON/$\overline{\text{OFF}}$[2] | 91 | $V_{CCP}$ | 22 |
| GND | 11 | PWRON/$\overline{\text{OFF}}$[3] | 109 | $\overline{\text{WR}}$/SCL | 30 |

## signal names by terminal number

**Table 2. Signals Sorted Numerically by Terminal Number (120-Terminal Package)**

| NO. | TERMINAL NAME | NO. | TERMINAL NAME | NO. | TERMINAL NAME |
|---|---|---|---|---|---|
| 1 | ATTN0[3] | 41 | GND | 81 | $\overline{\text{BUSON}}$[1] |
| 2 | ATTN1[3] | 42 | DATA2 | 82 | $\overline{\text{CLKON}}$[1] |
| 3 | M66EN[3] | 43 | DATA1/ADD6 | 83 | ATTN0[1] |
| 4 | $\overline{\text{DETECT0}}$[3] | 44 | DATA0/ADD5 | 84 | ATTN1[1] |
| 5 | $\overline{\text{DETECT1}}$[3] | 45 | A4/ADD4 | 85 | $V_{CC}$ |
| 6 | $V_{CC}$ | 46 | $V_{CC5V}$ | 86 | M66EN[1] |
| 7 | $\overline{\text{SLOTREQ64}}$[3] | 47 | A3/ADD3 | 87 | $\overline{\text{DETECT0}}$[1] |
| 8 | $\overline{\text{SLOTREQ64}}$[2] | 48 | A2/ADD2 | 88 | $\overline{\text{DETECT1}}$[1] |
| 9 | $\overline{\text{SLOTREQ64}}$[1] | 49 | A1/ADD1 | 89 | RSVD |
| 10 | $\overline{\text{SLOTREQ64}}$[0] | 50 | A0/ADD0 | 90 | RSVD |
| 11 | GND | 51 | GND | 91 | PWRON/$\overline{\text{OFF}}$[2] |
| 12 | $\overline{\text{SREQ}}$ | 52 | PWRON/$\overline{\text{OFF}}$[0] | 92 | $\overline{\text{PWRFAULT}}$[2] |
| 13 | $\overline{\text{SGNT}}$ | 53 | PWRFAULT[0] | 93 | $\overline{\text{PWRGOOD}}$[2] |
| 14 | $\overline{\text{PRST}}$ | 54 | $\overline{\text{PWRGOOD}}$[0] | 94 | $\overline{\text{PRSNT1}}$[2] |
| 15 | $V_{CC5V}$ | 55 | $\overline{\text{PRSNT1}}$[0] | 95 | $V_{CC}$ |
| 16 | PCLK | 56 | $V_{CC}$ | 96 | $\overline{\text{PRSNT2}}$[2] |
| 17 | GND | 57 | $\overline{\text{PRSNT2}}$[0] | 97 | $\overline{\text{SLOTRST}}$[2] |
| 18 | $\overline{\text{IDLEREQ}}$ | 58 | $\overline{\text{SLOTRST}}$[0] | 98 | $\overline{\text{REQ64ON}}$[2] |
| 19 | $\overline{\text{IDLEGNT}}$ | 59 | $\overline{\text{REQ64ON}}$[0] | 99 | REQ64ON[2] |
| 20 | $\overline{\text{FRAME}}$ | 60 | REQ64ON[0] | 100 | $\overline{\text{BUSON}}$[2] |
| 21 | $\overline{\text{IRDY}}$ | 61 | RSVD | 101 | GND |
| 22 | $V_{CCP}$ | 62 | RSVD | 102 | $\overline{\text{CLKON}}$[2] |
| 23 | $\overline{\text{INTR}}$ | 63 | $\overline{\text{BUSON}}$[0] | 103 | ATTN0[2] |
| 24 | INTR | 64 | $\overline{\text{CLKON}}$[0] | 104 | ATTN1[2] |
| 25 | SYSM66EN | 65 | ATTN0[0] | 105 | M66EN[2] |
| 26 | GND | 66 | ATTN1[0] | 106 | $\overline{\text{DETECT0}}$[2] |
| 27 | SMODE | 67 | M66EN[0] | 107 | $V_{CC5V}$ |
| 28 | $V_{CC5V}$ | 68 | GND | 108 | $\overline{\text{DETECT1}}$[2] |
| 29 | $\overline{\text{RD}}$/SDA | 69 | $\overline{\text{DETECT0}}$[0] | 109 | PWRON/$\overline{\text{OFF}}$[3] |
| 30 | $\overline{\text{WR}}$/SCL | 70 | $\overline{\text{DETECT1}}$[0] | 110 | $\overline{\text{PWRFAULT}}$[3] |
| 31 | RSVD | 71 | PWRON/$\overline{\text{OFF}}$[1] | 111 | $\overline{\text{PWRGOOD}}$[3] |
| 32 | RSVD | 72 | $\overline{\text{PWRFAULT}}$[1] | 112 | GND |
| 33 | RSVD | 73 | $\overline{\text{PWRGOOD}}$[1] | 113 | $\overline{\text{PRSNT1}}$[3] |
| 34 | $\overline{\text{CS}}$ | 74 | $V_{CC5V}$ | 114 | $\overline{\text{PRSNT2}}$[3] |
| 35 | DATA7 | 75 | $\overline{\text{PRSNT1}}$[1] | 115 | $\overline{\text{SLOTRST}}$[3] |
| 36 | $V_{CC}$ | 76 | $\overline{\text{PRSNT2}}$[1] | 116 | $\overline{\text{REQ64ON}}$[3] |
| 37 | DATA6 | 77 | $\overline{\text{SLOTRST}}$[1] | 117 | GND |
| 38 | DATA5 | 78 | $\overline{\text{REQ64ON}}$[1] | 118 | REQ64ON[3] |
| 39 | DATA4 | 79 | GND | 119 | $\overline{\text{BUSON}}$[3] |
| 40 | DATA3 | 80 | REQ64ON[1] | 120 | $\overline{\text{CLKON}}$[3] |

## signal names by terminal name

**Table 3. Signals Sorted Alphabetically by Terminal Name (128-Terminal Package)**

| TERMINAL NAME | NO. | TERMINAL NAME | NO. | TERMINAL NAME | NO. | TERMINAL NAME | NO. |
|---|---|---|---|---|---|---|---|
| A0/ADD0 | 53 | $\overline{\text{DETECT0}}$[2] | 113 | NC | 97 | REQ64ON[3] | 125 |
| A1/ADD1 | 52 | $\overline{\text{DETECT0}}$[3] | 5 | NC | 128 | RSVD | 34 |
| A2/ADD2 | 51 | $\overline{\text{DETECT1}}$[0] | 75 | PCLK | 17 | RSVD | 35 |
| A3/ADD3 | 50 | $\overline{\text{DETECT1}}$[1] | 93 | $\overline{\text{PRSNT1}}$[0] | 58 | RSVD | 36 |
| A4/ADD4 | 48 | $\overline{\text{DETECT1}}$[2] | 115 | $\overline{\text{PRSNT1}}$[1] | 80 | RSVD | 66 |
| ATTN0[0] | 70 | $\overline{\text{DETECT1}}$[3] | 6 | $\overline{\text{PRSNT1}}$[2] | 101 | RSVD | 67 |
| ATTN0[1] | 88 | $\overline{\text{FRAME}}$ | 21 | $\overline{\text{PRSNT1}}$[3] | 120 | RSVD | 94 |
| ATTN0[2] | 110 | GND | 12 | $\overline{\text{PRSNT2}}$[0] | 60 | RSVD | 95 |
| ATTN0[3] | 2 | GND | 18 | $\overline{\text{PRSNT2}}$[1] | 81 | $\overline{\text{SGNT}}$ | 14 |
| ATTN1[0] | 71 | GND | 27 | $\overline{\text{PRSNT2}}$[2] | 103 | $\overline{\text{SLOTREQ64}}$[0] | 11 |
| ATTN1[1] | 89 | GND | 44 | $\overline{\text{PRSNT2}}$[3] | 121 | $\overline{\text{SLOTREQ64}}$[1] | 10 |
| ATTN1[2] | 111 | GND | 54 | $\overline{\text{PRST}}$ | 15 | $\overline{\text{SLOTREQ64}}$[2] | 9 |
| ATTN1[3] | 3 | GND | 73 | $\overline{\text{PWRFAULT}}$[0] | 56 | $\overline{\text{SLOTREQ64}}$[3] | 8 |
| $\overline{\text{BUSON}}$[0] | 68 | GND | 84 | $\overline{\text{PWRFAULT}}$[1] | 77 | $\overline{\text{SLOTRST}}$[0] | 61 |
| $\overline{\text{BUSON}}$[1] | 86 | GND | 108 | $\overline{\text{PWRFAULT}}$[2] | 99 | $\overline{\text{SLOTRST}}$[1] | 82 |
| $\overline{\text{BUSON}}$[2] | 107 | GND | 119 | $\overline{\text{PWRFAULT}}$[3] | 117 | $\overline{\text{SLOTRST}}$[2] | 104 |
| $\overline{\text{BUSON}}$[3] | 126 | GND | 124 | $\overline{\text{PWRGOOD}}$[0] | 57 | $\overline{\text{SLOTRST}}$[3] | 122 |
| $\overline{\text{CLKON}}$[0] | 69 | $\overline{\text{IDLEGNT}}$ | 20 | $\overline{\text{PWRGOOD}}$[1] | 78 | SMODE | 28 |
| $\overline{\text{CLKON}}$[1] | 87 | $\overline{\text{IDLEREQ}}$ | 19 | $\overline{\text{PWRGOOD}}$[2] | 100 | $\overline{\text{SREQ}}$ | 13 |
| $\overline{\text{CLKON}}$[2] | 109 | INTR | 25 | $\overline{\text{PWRGOOD}}$[3] | 118 | SYSM66EN | 26 |
| $\overline{\text{CLKON}}$[3] | 127 | $\overline{\text{INTR}}$ | 24 | PWRON/$\overline{\text{OFF}}$[0] | 55 | $V_{CC}$ | 7 |
| $\overline{\text{CS}}$ | 37 | $\overline{\text{IRDY}}$ | 22 | PWRON/$\overline{\text{OFF}}$[1] | 76 | $V_{CC}$ | 39 |
| DATA0/ADD5 | 47 | M66EN[0] | 72 | PWRON/$\overline{\text{OFF}}$[2] | 98 | $V_{CC}$ | 59 |
| DATA1/ADD6 | 46 | M66EN[1] | 91 | PWRON/$\overline{\text{OFF}}$[3] | 116 | $V_{CC}$ | 90 |
| DATA2 | 45 | M66EN[2] | 112 | $\overline{\text{RD}}$/SDA | 30 | $V_{CC}$ | 102 |
| DATA3 | 43 | M66EN[3] | 4 | $\overline{\text{REQ64ON}}$[0] | 62 | $V_{CC5V}$ | 16 |
| DATA4 | 42 | NC | 1 | $\overline{\text{REQ64ON}}$[1] | 83 | $V_{CC5V}$ | 29 |
| DATA5 | 41 | NC | 32 | $\overline{\text{REQ64ON}}$[2] | 105 | $V_{CC5V}$ | 49 |
| DATA6 | 40 | NC | 33 | $\overline{\text{REQ64ON}}$[3] | 123 | $V_{CC5V}$ | 79 |
| DATA7 | 38 | NC | 64 | REQ64ON[0] | 63 | $V_{CC5V}$ | 114 |
| $\overline{\text{DETECT0}}$[0] | 74 | NC | 65 | REQ64ON[1] | 85 | $V_{CCP}$ | 23 |
| $\overline{\text{DETECT0}}$[1] | 92 | NC | 96 | REQ64ON[2] | 106 | $\overline{\text{WR}}$/SCL | 31 |

## signal names by terminal number

**Table 4. Signals Sorted Numerically by Terminal Number (128-Terminal Package)**

| NO. | TERMINAL NAME | NO. | TERMINAL NAME | NO. | TERMINAL NAME | NO. | TERMINAL NAME |
|---|---|---|---|---|---|---|---|
| 1 | NC | 33 | NC | 65 | NC | 97 | NC |
| 2 | ATTN0[3] | 34 | RSVD | 66 | RSVD | 98 | PWRON/$\overline{\text{OFF}}$[2] |
| 3 | ATTN1[3] | 35 | RSVD | 67 | RSVD | 99 | $\overline{\text{PWRFAULT}}$[2] |
| 4 | M66EN[3] | 36 | RSVD | 68 | $\overline{\text{BUSON}}$[0] | 100 | $\overline{\text{PWRGOOD}}$[2] |
| 5 | $\overline{\text{DETECT0}}$[3] | 37 | $\overline{\text{CS}}$ | 69 | $\overline{\text{CLKON}}$[0] | 101 | $\overline{\text{PRSNT1}}$[2] |
| 6 | $\overline{\text{DETECT1}}$[3] | 38 | DATA7 | 70 | ATTN0[0] | 102 | $V_{CC}$ |
| 7 | $V_{CC}$ | 39 | $V_{CC}$ | 71 | ATTN1[0] | 103 | $\overline{\text{PRSNT2}}$[2] |
| 8 | $\overline{\text{SLOTREQ64}}$[3] | 40 | DATA6 | 72 | M66EN[0] | 104 | $\overline{\text{SLOTRST}}$[2] |
| 9 | $\overline{\text{SLOTREQ64}}$[2 | 41 | DATA5 | 73 | GND | 105 | $\overline{\text{REQ64ON}}$[2] |
| 10 | $\overline{\text{SLOTREQ64}}$[1] | 42 | DATA4 | 74 | $\overline{\text{DETECT0}}$[0] | 106 | REQ64ON[2] |
| 11 | $\overline{\text{SLOTREQ64}}$[0] | 43 | DATA3 | 75 | $\overline{\text{DETECT1}}$[0] | 107 | $\overline{\text{BUSON}}$[2] |
| 12 | GND | 44 | GND | 76 | PWRON/$\overline{\text{OFF}}$[1] | 108 | GND |
| 13 | $\overline{\text{SREQ}}$ | 45 | DATA2 | 77 | $\overline{\text{PWRFAULT}}$[1] | 109 | $\overline{\text{CLKON}}$[2] |
| 14 | $\overline{\text{SGNT}}$ | 46 | DATA1/ADD6 | 78 | $\overline{\text{PWRGOOD}}$[1] | 110 | ATTN0[2] |
| 15 | $\overline{\text{PRST}}$ | 47 | DATA0/ADD5 | 79 | $V_{CC5V}$ | 111 | ATTN1[2] |
| 16 | $V_{CC5V}$ | 48 | A4/ADD4 | 80 | $\overline{\text{PRSNT1}}$[1] | 112 | M66EN[2] |
| 17 | PCLK | 49 | $V_{CC5V}$ | 81 | $\overline{\text{PRSNT2}}$[1] | 113 | $\overline{\text{DETECT0}}$[2] |
| 18 | GND | 50 | A3/ADD3 | 82 | $\overline{\text{SLOTRST}}$[1] | 114 | $V_{CC5V}$ |
| 19 | $\overline{\text{IDLEREQ}}$ | 51 | A2/ADD2 | 83 | $\overline{\text{REQ64ON}}$[1] | 115 | $\overline{\text{DETECT1}}$[2] |
| 20 | $\overline{\text{IDLEGNT}}$ | 52 | A1/ADD1 | 84 | GND | 116 | PWRON/$\overline{\text{OFF}}$[3] |
| 21 | $\overline{\text{FRAME}}$ | 53 | A0/ADD0 | 85 | REQ64ON[1] | 117 | $\overline{\text{PWRFAULT}}$[3] |
| 22 | $\overline{\text{IRDY}}$ | 54 | GND | 86 | $\overline{\text{BUSON}}$[1] | 118 | $\overline{\text{PWRGOOD}}$[3] |
| 23 | $V_{CCP}$ | 55 | PWRON/$\overline{\text{OFF}}$[0] | 87 | $\overline{\text{CLKON}}$[1] | 119 | GND |
| 24 | $\overline{\text{INTR}}$ | 56 | $\overline{\text{PWRFAULT}}$[0] | 88 | ATTN0[1] | 120 | $\overline{\text{PRSNT1}}$[3] |
| 25 | INTR | 57 | $\overline{\text{PWRGOOD}}$[0] | 89 | ATTN1[1] | 121 | $\overline{\text{PRSNT2}}$[3 |
| 26 | SYSM66EN | 58 | $\overline{\text{PRSNT1}}$[0] | 90 | $V_{CC}$ | 122 | $\overline{\text{SLOTRST}}$[3] |
| 27 | GND | 59 | $V_{CC}$ | 91 | M66EN[1] | 123 | $\overline{\text{REQ64ON}}$[3] |
| 28 | SMODE | 60 | $\overline{\text{PRSNT2}}$[0] | 92 | $\overline{\text{DETECT0}}$[1] | 124 | GND |
| 29 | $V_{CC5V}$ | 61 | $\overline{\text{SLOTRST}}$[0] | 93 | $\overline{\text{DETECT1}}$[1] | 125 | REQ64ON[3] |
| 30 | $\overline{\text{RD}}$/SDA | 62 | $\overline{\text{REQ64ON}}$[0] | 94 | RSVD | 126 | $\overline{\text{BUSON}}$[3] |
| 31 | $\overline{\text{WR}}$/SCL | 63 | REQ64ON[0] | 95 | RSVD | 127 | $\overline{\text{CLKON}}$[3] |
| 32 | NC | 64 | NC | 96 | NC | 128 | NC |

## signal names by terminal name

**Table 5. Signals Sorted Alphabetically by Terminal Name (144-Terminal Package)**

| TERMINAL NAME | NO. | TERMINAL NAME | NO. | TERMINAL NAME | NO. | TERMINAL NAME | NO. |
|---|---|---|---|---|---|---|---|
| A0/ADD0 | 59 | $\overline{\text{DETECT1}}$[2] | 129 | NC | 78 | $\overline{\text{REQ64ON}}$[3] | 137 |
| A1/ADD1 | 58 | $\overline{\text{DETECT1}}$[3] | 8 | NC | 103 | REQ64ON[0] | 72 |
| A2/ADD2 | 57 | $\overline{\text{FRAME}}$ | 23 | NC | 105 | REQ64ON[1] | 95 |
| A3/ADD3 | 56 | GND | 14 | NC | 107 | REQ64ON[2] | 120 |
| A4/ADD4 | 54 | GND | 20 | NC | 110 | REQ64ON[3] | 140 |
| ATTN0[0] | 80 | GND | 29 | NC | 112 | RSVD | 37 |
| ATTN0[1] | 98 | GND | 50 | NC | 114 | RSVD | 39 |
| ATTN0[2] | 124 | GND | 60 | NC | 139 | RSVD | 41 |
| ATTN0[3] | 1 | GND | 83 | NC | 141 | RSVD | 73 |
| ATTN1[0] | 81 | GND | 94 | NC | 143 | RSVD | 75 |
| ATTN1[1] | 99 | GND | 122 | PCLK | 19 | RSVD | 106 |
| ATTN1[2] | 125 | GND | 133 | $\overline{\text{PRSNT1}}$[0] | 64 | RSVD | 108 |
| ATTN1[3] | 3 | GND | 138 | $\overline{\text{PRSNT1}}$[1] | 90 | $\overline{\text{SGNT}}$ | 16 |
| $\overline{\text{BUSON}}$[0] | 77 | $\overline{\text{IDLEGNT}}$ | 22 | $\overline{\text{PRSNT1}}$[2] | 115 | $\overline{\text{SLOTREQ64}}$[0] | 13 |
| $\overline{\text{BUSON}}$[1] | 96 | $\overline{\text{IDLEREQ}}$ | 21 | $\overline{\text{PRSNT1}}$[3] | 134 | $\overline{\text{SLOTREQ64}}$[1] | 12 |
| $\overline{\text{BUSON}}$[2] | 121 | INTR | 27 | $\overline{\text{PRSNT2}}$[0] | 66 | $\overline{\text{SLOTREQ64}}$[2] | 11 |
| $\overline{\text{BUSON}}$[3] | 142 | $\overline{\text{INTR}}$ | 26 | $\overline{\text{PRSNT2}}$[1] | 91 | $\overline{\text{SLOTREQ64}}$[3] | 10 |
| $\overline{\text{CLKON}}$[0] | 79 | $\overline{\text{IRDY}}$ | 24 | $\overline{\text{PRSNT2}}$[2] | 117 | $\overline{\text{SLOTRST}}$[0] | 68 |
| $\overline{\text{CLKON}}$[1] | 97 | M66EN[0] | 82 | $\overline{\text{PRSNT2}}$[3] | 135 | $\overline{\text{SLOTRST}}$[1] | 92 |
| $\overline{\text{CLKON}}$[2] | 123 | M66EN[1] | 101 | $\overline{\text{PRST}}$ | 17 | $\overline{\text{SLOTRST}}$[2] | 118 |
| $\overline{\text{CLKON}}$[3] | 144 | M66EN[2] | 126 | $\overline{\text{PWRFAULT}}$[0] | 62 | $\overline{\text{SLOTRST}}$[3] | 136 |
| $\overline{\text{CS}}$ | 43 | M66EN[3] | 5 | $\overline{\text{PWRFAULT}}$[1] | 87 | SMODE | 30 |
| DATA0/ADD5 | 53 | NC | 2 | $\overline{\text{PWRFAULT}}$[2] | 111 | $\overline{\text{SREQ}}$ | 15 |
| DATA1/ADD6 | 52 | NC | 4 | $\overline{\text{PWRFAULT}}$[3] | 131 | SYSM66EN | 28 |
| DATA2 | 51 | NC | 6 | $\overline{\text{PWRGOOD}}$[0] | 63 | $V_{CC}$ | 9 |
| DATA3 | 49 | NC | 31 | $\overline{\text{PWRGOOD}}$[1] | 88 | $V_{CC}$ | 45 |
| DATA4 | 48 | NC | 33 | $\overline{\text{PWRGOOD}}$[2] | 113 | $V_{CC}$ | 65 |
| DATA5 | 47 | NC | 35 | $\overline{\text{PWRGOOD}}$[3] | 132 | $V_{CC}$ | 100 |
| DATA6 | 46 | NC | 38 | PWRON/$\overline{\text{OFF}}$[0] | 61 | $V_{CC}$ | 116 |
| DATA7 | 44 | NC | 40 | PWRON/$\overline{\text{OFF}}$[1] | 86 | $V_{CC5V}$ | 18 |
| $\overline{\text{DETECT0}}$[0] | 84 | NC | 42 | PWRON/$\overline{\text{OFF}}$[2] | 109 | $V_{CC5V}$ | 32 |
| $\overline{\text{DETECT0}}$[1] | 102 | NC | 67 | PWRON/$\overline{\text{OFF}}$[3] | 130 | $V_{CC5V}$ | 55 |
| $\overline{\text{DETECT0}}$[2] | 127 | NC | 69 | $\overline{\text{RD}}$/SDA | 34 | $V_{CC5V}$ | 89 |
| $\overline{\text{DETECT0}}$[3] | 7 | NC | 71 | $\overline{\text{REQ64ON}}$[0] | 70 | $V_{CC5V}$ | 128 |
| $\overline{\text{DETECT1}}$[0] | 85 | NC | 74 | $\overline{\text{REQ64ON}}$[1] | 93 | $V_{CCP}$ | 25 |
| $\overline{\text{DETECT1}}$[1] | 104 | NC | 76 | $\overline{\text{REQ64ON}}$[2] | 119 | $\overline{\text{WR}}$/SCL | 36 |

## signal names by terminal number

**Table 6. Signals Sorted Numerically by Terminal Number (144-Terminal Package)**

| NO. | TERMINAL NAME | NO. | TERMINAL NAME | NO. | TERMINAL NAME | NO. | TERMINAL NAME |
|---|---|---|---|---|---|---|---|
| 1 | ATTN0[3] | 37 | RSVD | 73 | RSVD | 109 | PWRON/$\overline{\text{OFF}}$[2] |
| 2 | NC | 38 | NC | 74 | NC | 110 | NC |
| 3 | ATTN1[3] | 39 | RSVD | 75 | RSVD | 111 | $\overline{\text{PWRFAULT}}$[2] |
| 4 | NC | 40 | NC | 76 | NC | 112 | NC |
| 5 | M66EN[3] | 41 | RSVD | 77 | $\overline{\text{BUSON}}$[0] | 113 | $\overline{\text{PWRGOOD}}$[2] |
| 6 | NC | 42 | NC | 78 | NC | 114 | NC |
| 7 | $\overline{\text{DETECT0}}$[3] | 43 | $\overline{\text{CS}}$ | 79 | $\overline{\text{CLKON}}$[0] | 115 | $\overline{\text{PRSNT1}}$[2] |
| 8 | $\overline{\text{DETECT1}}$[3] | 44 | DATA7 | 80 | ATTN0[0] | 116 | $V_{CC}$ |
| 9 | $V_{CC}$ | 45 | $V_{CC}$ | 81 | ATTN1[0] | 117 | $\overline{\text{PRSNT2}}$[2] |
| 10 | $\overline{\text{SLOTREQ64}}$[3] | 46 | DATA6 | 82 | M66EN[0] | 118 | $\overline{\text{SLOTRST}}$[2] |
| 11 | $\overline{\text{SLOTREQ64}}$[2] | 47 | DATA5 | 83 | GND | 119 | $\overline{\text{REQ64ON}}$[2] |
| 12 | $\overline{\text{SLOTREQ64}}$[1] | 48 | DATA4 | 84 | $\overline{\text{DETECT0}}$[0] | 120 | REQ64ON[2] |
| 13 | $\overline{\text{SLOTREQ64}}$[0] | 49 | DATA3 | 85 | $\overline{\text{DETECT1}}$[0] | 121 | $\overline{\text{BUSON}}$[2] |
| 14 | GND | 50 | GND | 86 | PWRON/$\overline{\text{OFF}}$[1] | 122 | GND |
| 15 | $\overline{\text{SREQ}}$ | 51 | DATA2 | 87 | $\overline{\text{PWRFAULT}}$[1] | 123 | $\overline{\text{CLKON}}$[2] |
| 16 | $\overline{\text{SGNT}}$ | 52 | DATA1/ADD6 | 88 | $\overline{\text{PWRGOOD}}$[1] | 124 | ATTN0[2] |
| 17 | $\overline{\text{PRST}}$ | 53 | DATA0/ADD5 | 89 | $V_{CC5V}$ | 125 | ATTN1[2] |
| 18 | $V_{CC5V}$ | 54 | A4/ADD4 | 90 | $\overline{\text{PRSNT1}}$[1] | 126 | M66EN[2] |
| 19 | PCLK | 55 | $V_{CC5V}$ | 91 | $\overline{\text{PRSNT2}}$[1] | 127 | $\overline{\text{DETECT0}}$[2] |
| 20 | GND | 56 | A3/ADD3 | 92 | $\overline{\text{SLOTRST}}$[1] | 128 | $V_{CC5V}$ |
| 21 | $\overline{\text{IDLEREQ}}$ | 57 | A2/ADD2 | 93 | $\overline{\text{REQ64ON}}$[1] | 129 | $\overline{\text{DETECT1}}$[2] |
| 22 | $\overline{\text{IDLEGNT}}$ | 58 | A1/ADD1 | 94 | GND | 130 | PWRON/$\overline{\text{OFF}}$[3] |
| 23 | $\overline{\text{FRAME}}$ | 59 | A0/ADD0 | 95 | REQ64ON[1] | 131 | $\overline{\text{PWRFAULT}}$[3] |
| 24 | $\overline{\text{IRDY}}$ | 60 | GND | 96 | $\overline{\text{BUSON}}$[1] | 132 | $\overline{\text{PWRGOOD}}$[3] |
| 25 | $V_{CCP}$ | 61 | PWRON/$\overline{\text{OFF}}$[0] | 97 | $\overline{\text{CLKON}}$[1] | 133 | GND |
| 26 | $\overline{\text{INTR}}$ | 62 | $\overline{\text{PWRFAULT}}$[0] | 98 | ATTN0[1] | 134 | $\overline{\text{PRSNT1}}$[3] |
| 27 | INTR | 63 | $\overline{\text{PWRGOOD}}$[0] | 99 | ATTN1[1] | 135 | $\overline{\text{PRSNT2}}$[3] |
| 28 | SYSM66EN | 64 | $\overline{\text{PRSNT1}}$[0] | 100 | $V_{CC}$ | 136 | $\overline{\text{SLOTRST}}$[3] |
| 29 | GND | 65 | $V_{CC}$ | 101 | M66EN[1] | 137 | $\overline{\text{REQ64ON}}$[3] |
| 30 | SMODE | 66 | $\overline{\text{PRSNT2}}$[0] | 102 | $\overline{\text{DETECT0}}$[1] | 138 | GND |
| 31 | NC | 67 | NC | 103 | NC | 139 | NC |
| 32 | $V_{CC5V}$ | 68 | $\overline{\text{SLOTRST}}$[0] | 104 | $\overline{\text{DETECT1}}$[1] | 140 | REQ64ON[3] |
| 33 | NC | 69 | NC | 105 | NC | 141 | NC |
| 34 | $\overline{\text{RD}}$/SDA | 70 | $\overline{\text{REQ64ON}}$[0] | 106 | RSVD | 142 | $\overline{\text{BUSON}}$[3] |
| 35 | NC | 71 | NC | 107 | NC | 143 | NC |
| 36 | $\overline{\text{WR}}$/SCL | 72 | REQ64ON[0] | 108 | RSVD | 144 | $\overline{\text{CLKON}}$[3] |

## Terminal Functions

This section describes the HPC3130A terminal functions. The terminals are grouped in tables by function.

### power supply terminal functions

| TERMINAL | | | | I/O | FUNCTION |
|---|---|---|---|---|---|
| NAME | NO. 120 | NO. 128 | NO. 144 | | |
| GND | 11, 17, 26, 41, 51, 68, 79, 101, 112, 117 | 12, 18, 27, 44, 54, 73, 84, 108, 119, 124 | 14, 20, 29, 50, 60, 83, 94, 122, 133, 138 | I | Device ground terminals |
| $V_{CC}$ | 6, 36, 56, 85, 95 | 7, 39, 59, 90, 102 | 9, 45, 65, 100, 116 | I | 3.3-V power supply |
| $V_{CC5V}$ | 15, 28, 46, 74, 107 | 16, 29, 49, 79, 114 | 18, 32, 55, 89, 128 | I | 5-V clamp-rail voltage supply |
| $V_{CCP}$ | 22 | 23 | 25 | I | Clamp rail voltage for PCI signaling (5V or 3.3V) |

### control bus interface

| TERMINAL | | | | I/O | FUNCTION |
|---|---|---|---|---|---|
| NAME | NO. 120 | NO. 128 | NO. 144 | | |
| A2/ADD2<br>A1/ADD1<br>A0/ADD0 | 48<br>49<br>50 | 51<br>52<br>53 | 57<br>58<br>59 | I | Parallel bus address. These terminals are address inputs in generic parallel bus cycles and are only used when the SMODE is input low. These lower address terminals select one of the eight registers for read/write access.<br>Serial bus address select. These terminals indicate the full serial bus address of the HPC3130A when the SMODE is input high. |
| A4/ADD4<br>A3/ADD3 | 45<br>47 | 48<br>50 | 54<br>56 | I | Parallel bus address. These terminals are address inputs in generic parallel bus cycles, and are only used when SMODE is input low. These upper address terminals select one of four hot-plug slots supported by the HPC3130A.<br>Serial bus address select. These terminals indicate the full serial bus address of the HPC3130A when the SMODE is input high. |
| $\overline{CS}$ | 34 | 37 | 43 | I | Chip selection. This active low input selects the HPC3130A chip as addressed in the current generic parallel bus cycle. This chip input is only valid if the SMODE is input low. Multiple HPC3130A chips may exist in a system with external logic driving this signal. |
| DATA1/ADD6<br>DATA0/ADD5 | 43<br>44 | 46<br>47 | 52<br>53 | I/O | Parallel bus data. This bus is the data bus in generic parallel bus cycles and is selected when the SMODE is input low. The data path is used during both read and write transactions to internal registers when the parallel control bus interface is implemented.<br>Serial bus address selection. These terminals indicate the full serial bus address of the HPC3130A when the SMODE is input high. |
| DATA7<br>DATA6<br>DATA5<br>DATA4<br>DATA3<br>DATA2 | 35<br>37<br>38<br>39<br>40<br>42 | 38<br>40<br>41<br>42<br>43<br>45 | 44<br>46<br>47<br>48<br>49<br>51 | I/O | Parallel bus data. This bus is the data bus in generic parallel bus cycles and is selected when the SMODE is input low. The data path is used during both read and write transactions to internal registers when the parallel control bus interface is implemented. |
| $\overline{RD}$/SDA | 29 | 30 | 34 | I/O | Read selection. This terminal indicates a register read cycle when the SMODE input is low and the $\overline{CS}$ terminal input is asserted. This is used to read an internal HPC3130A register.<br>Serial bus data. This terminal signals the serial bus data when the SMODE input is high. It is used during internal register read and write transactions. |
| $\overline{WR}$/SCL | 30 | 31 | 36 | I | Write selection. This terminal indicates a register write cycle when the SMODE input is low and the $\overline{CS}$ terminal input is asserted. This input is used to write to an internal HPC3130A register.<br>Serial bus clock. This terminal inputs serial bus clock in when the SMODE input is high. It is used during internal register read and write transactions. |

## Terminal Functions (Continued)

### system interface

| TERMINAL | | | | I/O | FUNCTION |
|---|---|---|---|---|---|
| NAME | NO. 120 | NO. 128 | NO. 144 | | |
| $\overline{\text{FRAME}}$ | 20 | 21 | 23 | I | Frame. This input and the $\overline{\text{IRDY}}$ input indicate that the PCI bus is idle. When the HPC3130A senses the PCI bus is idle after $\overline{\text{IDLEGNT}}$ is low, a hot-plug slot can be connected to the PCI bus. This input must be wired to a valid logic level if the bus idling procedure is not implemented. |
| $\overline{\text{IDLEGNT}}$ | 19 | 20 | 22 | I | Idle grant. This input indicates when the PCI bus is idled by the HOST-PCI bridge after a request is made by $\overline{\text{IDLEREQ}}$. The protocol is identical to PCI request/grant. This input must be wired to a valid logic level if the bus idling procedure is not implemented. |
| $\overline{\text{IDLEREQ}}$ | 18 | 19 | 21 | O | Idle request. This output is driven to request the HOST-PCI bridge to idle the PCI bus before connecting a hot-plug slot. The protocol is identical to PCI request/grant. A pullup resistor must be implemented on this terminal if the bus idling procedure is not implemented. |
| INTR | 24 | 25 | 27 | O | System interrupt. This output provides a system interrupt. The HPC3130A can be programmed to assert this interrupt under various conditions, which may be serviced by the hot-plug service. Furthermore, the event status/enable state is compliant with the *ACPI Specification* and, as a result, supports ACPI control methods for switching the HPC3130A. |
| $\overline{\text{INTR}}$ | 23 | 24 | 26 | O | System interrupt. This open drain output provides a system interrupt. The HPC3130A can be programmed to assert this interrupt under various conditions, which may be serviced by the hot-plug. Furthermore, the event status/enable state is compliant with the *ACPI Specification* and, as a result, supports ACPI control methods for switching the HPC3130A. |
| $\overline{\text{IRDY}}$ | 21 | 22 | 24 | I | Initiator ready. This and the $\overline{\text{FRAME}}$ input indicate that the PCI bus is idle. When the HPC3130A senses the PCI bus is idle after $\overline{\text{IDLEGNT}}$ is low, a hot-plug slot may be connected to the PCI bus. This input must be wired to a valid logic level if the bus idling procedure is not implemented. |
| PCLK | 16 | 17 | 19 | I | PCI clock input. These terminals provide the PCI clock to the HPC3130A, which uses it only for activity indicator timing, $\overline{\text{IDLEREQ}}$/$\overline{\text{IDLEGNT}}$ protocol, and connection sequencing. |
| $\overline{\text{PRST}}$ | 14 | 15 | 17 | I | PCI reset. This signal provides the PCI reset to the HPC3130A. After a PCI reset, the HPC3130A resides in a state where all slots are enabled, as in a non-hot-plug system. The HPC3130A passes PCI resets from the host to all hot-plug slots. |
| $\overline{\text{SGNT}}$ | 13 | 14 | 16 | O | Secondary grant. This output provides a scheme to cascade a secondary HPC3130A device in order to provide more than four slots. The $\overline{\text{SGNT}}$ output from the primary HPC3130A is input to the $\overline{\text{IDLEGNT}}$ terminal for the secondary HPC3130A. After the secondary HPC3130A requests the primary HPC3130A to idle the bus, the primary HPC3130A arbitrates for the bus using $\overline{\text{IDLEREQ}}$. Once $\overline{\text{IDLEGNT}}$ is asserted, the primary HPC3130A asserts its $\overline{\text{SGNT}}$ output. This indicates to the secondary HPC3130A device that it can connect to the bus. |
| SMODE | 27 | 28 | 30 | I | Serial bus mode. When this input is asserted high, the internal HPC3130A registers are accessible through the serial bus interface; otherwise, they are accessed through the generic parallel bus interface. This input selects the control bus interface. |
| $\overline{\text{SREQ}}$ | 12 | 13 | 15 | I | Secondary request. This input provides a scheme to cascade a second HPC3130A device in order to provide more than four slots. The $\overline{\text{IDLEREQ}}$ from the second HPC3130A device is input to the $\overline{\text{SREQ}}$ terminal of the primary HPC3130A. If the second HPC3130A device arbitrates for the bus by asserting its $\overline{\text{IDLEREQ}}$ output, this scheme causes the primary HPC3130A to assert its $\overline{\text{IDLEREQ}}$. If cascading is not used, this input is pulled high. |
| SYSM66EN | 25 | 26 | 28 | I/O | PCI bus frequency indicator. This signal indicates the PCI clock frequency requirements of the hot-plug slots, and must be tied to the system PCI bus M66EN signal. The output from this terminal only changes state after a PCI reset and is only required in a 66-MHz system. |

## Terminal Functions (Continued)

### slot control and status functions

| TERMINAL | | | | I/O | FUNCTION |
|---|---|---|---|---|---|
| NAME | NO. 120 | NO. 128 | NO. 144 | | |
| ATTN0[3] | 1 | 2 | 1 | O | Attention indicators. These two outputs are provided per slot as attention indicators and can be independently programmed to drive high, low, fast blink, and slow blink. The timer is based on the PCI clock frequency and the state of SYSM66EN. |
| ATTN0[2] | 103 | 110 | 124 | | |
| ATTN0[1] | 83 | 88 | 98 | | |
| ATTN0[0] | 65 | 70 | 80 | | |
| ATTN1[3] | 2 | 3 | 3 | | |
| ATTN1[2] | 104 | 111 | 125 | | |
| ATTN1[1] | 84 | 89 | 99 | | |
| ATTN1[0] | 66 | 71 | 81 | | |
| $\overline{\text{BUSON}}$[3] | 119 | 126 | 142 | O | CBT switch control for PCI bus. This output controls the CBT switch that connects the hot-plug slot to the system PCI bus. This output is only driven by the HPC3130A under programmed control. |
| $\overline{\text{BUSON}}$[2] | 100 | 107 | 121 | | |
| $\overline{\text{BUSON}}$[1] | 81 | 86 | 96 | | |
| $\overline{\text{BUSON}}$[0] | 63 | 68 | 77 | | |
| $\overline{\text{CLKON}}$[3] | 120 | 127 | 144 | O | PCI clock connection control. This output is used to control the CBT switch or clock driver that connects the hot-plug slot to the system PCI clock. This output is only driven by the HPC3130A under programmed control. |
| $\overline{\text{CLKON}}$[2] | 102 | 109 | 123 | | |
| $\overline{\text{CLKON}}$[1] | 82 | 87 | 97 | | |
| $\overline{\text{CLKON}}$[0] | 64 | 69 | 79 | | |
| $\overline{\text{DETECT0}}$[3] | 4 | 5 | 7 | I | Card detection signals. These two card detect input signals, $\overline{\text{DETECT0}}$ and $\overline{\text{DETECT1}}$, are provided as additional card detection signals to the $\overline{\text{PRSNT1}}$ and $\overline{\text{PRSNT2}}$. Since only one present input must be tied to ground to indicate a card is present per the *PCI Specification*, these optional inputs are provided for designers of a more mechanically robust system. If the protection enable bit is set to 1 in the general configuration register, the HPC3130A does not power a hot-plug slot unless $\overline{\text{DETECT0}}$ and $\overline{\text{DETECT1}}$ are input low. A design not implementing additional card detection must tie these signals to ground. When this feature is utilized, the HPC3130A guarantees that power can not be applied to an empty slot or a slot with a partially inserted card. |
| $\overline{\text{DETECT0}}$[2] | 106 | 113 | 127 | | |
| $\overline{\text{DETECT0}}$[1] | 87 | 92 | 102 | | |
| $\overline{\text{DETECT0}}$[0] | 69 | 74 | 84 | | |
| $\overline{\text{DETECT1}}$[3] | 5 | 6 | 8 | | |
| $\overline{\text{DETECT1}}$[2] | 108 | 115 | 129 | | |
| $\overline{\text{DETECT1}}$[1] | 88 | 93 | 104 | | |
| $\overline{\text{DETECT1}}$[0] | 70 | 75 | 85 | | |
| M66EN[3] | 3 | 4 | 5 | I | PCI bus frequency indicator. This signal indicates the PCI clock frequency requirements of the hot-plug slots and is only required in a 66-MHz system (two slot maximum electrical loading limits). The two slot interfaces that provide the M66EN terminals are sensed at PCI reset and are driven afterwards. |
| M66EN[2] | 105 | 112 | 126 | | |
| M66EN[1] | 86 | 91 | 101 | | |
| M66EN[0] | 67 | 72 | 82 | | |
| $\overline{\text{PRSNT1}}$[3] | 113 | 120 | 134 | I | Present signals. These inputs are provided by hot-plug slots to indicate that an add-in card is physically present in the slot and to power requirements to the system. Only one of these signals must be tied to ground to indicate a card is present in an expansion slot. A set of $\overline{\text{PRSNT1}}$ and $\overline{\text{PRSNT2}}$ inputs are provided for each hot-plug slot. |
| $\overline{\text{PRSNT1}}$[2] | 94 | 101 | 115 | | |
| $\overline{\text{PRSNT1}}$[1] | 75 | 80 | 90 | | |
| $\overline{\text{PRSNT1}}$[0] | 55 | 58 | 64 | | |
| $\overline{\text{PRSNT2}}$[3] | 114 | 121 | 135 | | |
| $\overline{\text{PRSNT2}}$[2] | 96 | 103 | 117 | | |
| $\overline{\text{PRSNT2}}$[1] | 76 | 81 | 91 | | |
| $\overline{\text{PRSNT2}}$[0] | 57 | 60 | 66 | | |
| $\overline{\text{PWRFAULT}}$[3] | 110 | 117 | 131 | I | Power fault. This input is provided per slot power switch to indicate if there is a power fault. The HPC3130A can be programmed to generate an interrupt through $\overline{\text{INTR}}$ when this input is asserted. |
| $\overline{\text{PWRFAULT}}$[2] | 92 | 99 | 111 | | |
| $\overline{\text{PWRFAULT}}$[1] | 72 | 77 | 87 | | |
| $\overline{\text{PWRFAULT}}$[0] | 53 | 56 | 62 | | |
| $\overline{\text{PWRGOOD}}$[3] | 111 | 118 | 132 | I | Power good. This input is provided per slot power switch to indicate when power is successfully switched. The HPC3130A can be programmed to generate an interrupt through $\overline{\text{INTR}}$ when this input is asserted. |
| $\overline{\text{PWRGOOD}}$[2] | 93 | 100 | 113 | | |
| $\overline{\text{PWRGOOD}}$[1] | 73 | 78 | 88 | | |
| $\overline{\text{PWRGOOD}}$[0] | 54 | 57 | 63 | | |

## Terminal Functions (Continued)

### slot control and status functions (continued)

| TERMINAL | | | | I/O | FUNCTION |
|---|---|---|---|---|---|
| NAME | NO. 120 | NO. 128 | NO. 144 | | |
| PWRON/$\overline{\text{OFF}}$[3]<br>PWRON/$\overline{\text{OFF}}$[2]<br>PWRON/$\overline{\text{OFF}}$[1]<br>PWRON/$\overline{\text{OFF}}$[0] | 109<br>91<br>71<br>52 | 116<br>98<br>76<br>55 | 130<br>109<br>86<br>61 | O | Power ON/OFF. This output is provided per slot and is driven to the power switch to control the slot power state. |
| $\overline{\text{REQ64ON}}$[3]<br>$\overline{\text{REQ64ON}}$[2]<br>$\overline{\text{REQ64ON}}$[1]<br>$\overline{\text{REQ64ON}}$[0] | 116<br>98<br>78<br>59 | 123<br>105<br>83<br>62 | 137<br>119<br>93<br>70 | O | CBT switch control for $\overline{\text{SLOTREQ64}}$. A CBT switch can be implemented to reduce trace loading of the additional $\overline{\text{REQ64}}$ signal inherent to the HPC3130A controller. This output can be used to control the CBT switch. This output is only driven by the HPC3130A under programmed control. |
| REQ64ON[3]<br>REQ64ON[2]<br>REQ64ON[1]<br>REQ64ON[0] | 118<br>99<br>80<br>60 | 125<br>106<br>85<br>63 | 140<br>120<br>95<br>72 | O | CBT switch control for $\overline{\text{SLOTREQ64}}$. A CBT switch can be implemented to reduce trace loading of the additional $\overline{\text{REQ64}}$ signal inherent to the HPC3130A controller. This output can be used to control the CBT switch. This output is only driven by the HPC3130A under programmed control. |
| $\overline{\text{SLOTREQ64}}$[3]<br>$\overline{\text{SLOTREQ64}}$[2]<br>$\overline{\text{SLOTREQ64}}$[1]<br>$\overline{\text{SLOTREQ64}}$[0] | 7<br>8<br>9<br>10 | 8<br>9<br>10<br>11 | 10<br>11<br>12<br>13 | O | Slot request 64. This output is driven in conjunction with $\overline{\text{SLOTRST}}$ to the hot-plug slot to indicate to option cards whether or not they are plugged into a 64-bit slot. If a 64-bit device is plugged into a 32-bit slot, then it must ensure that its high-word path inputs do not oscillate and that there is not a significant power drain through the input buffer. This output is only driven by the HPC3130A under programmed control. |
| $\overline{\text{SLOTRST}}$[3]<br>$\overline{\text{SLOTRST}}$[2]<br>$\overline{\text{SLOTRST}}$[1]<br>$\overline{\text{SLOTRST}}$[0] | 115<br>97<br>77<br>58 | 122<br>104<br>82<br>61 | 136<br>118<br>92<br>68 | O | Slot PCI reset. This output is driven to the hot-plug slot to reset it after power up. When a card is inserted into a hot-plug slot it must be reset independent of the other PCI devices on the bus. This output is only driven by the HPC3130A under programmed control. |

## HPC3130A applications

This section discusses the various features of the HPC3130A in detail, and presents design considerations including a general connection sequencing guideline.

### system implementation

Figure 1 illustrates the HPC3130A implementation. The PCI bus signals are switched to the hot–plug PCI slot by the $\overline{\text{BUSON}}$ output, which controls a CBT switch. The PCI clock, PCI reset, M66EN, and $\overline{\text{REQ64}}$ must not be routed through the CBT switch. The HPC3130A drives the slot PCI reset and $\overline{\text{SLOTREQ64}}$, which can be controlled by internal HPC3130A registers. The $\overline{\text{SLOTREQ64}}$ requires special consideration during reset, as described. The PCI clock to the slot is driven by a clock driver, which is enabled by the HPC3130A $\overline{\text{CLKON}}$ output.

The HPC3130A also provides other features such as mechanical detection circuits, attention indicators, and interrupt signaling. The mechanical detection circuitry using the $\overline{\text{DETECT}}$[1,0] inputs is displayed as a dotted line and is an optional feature. Two attention indicator outputs, ATTN[1,0], are provided: one indicator to draw the attention of the user to a particular slot for insertion/removal, and one optional indicator that can be used to indicate fault conditions. Additional features, such as 66-MHz capability and automatic sequencing, are discussed in the following sections.

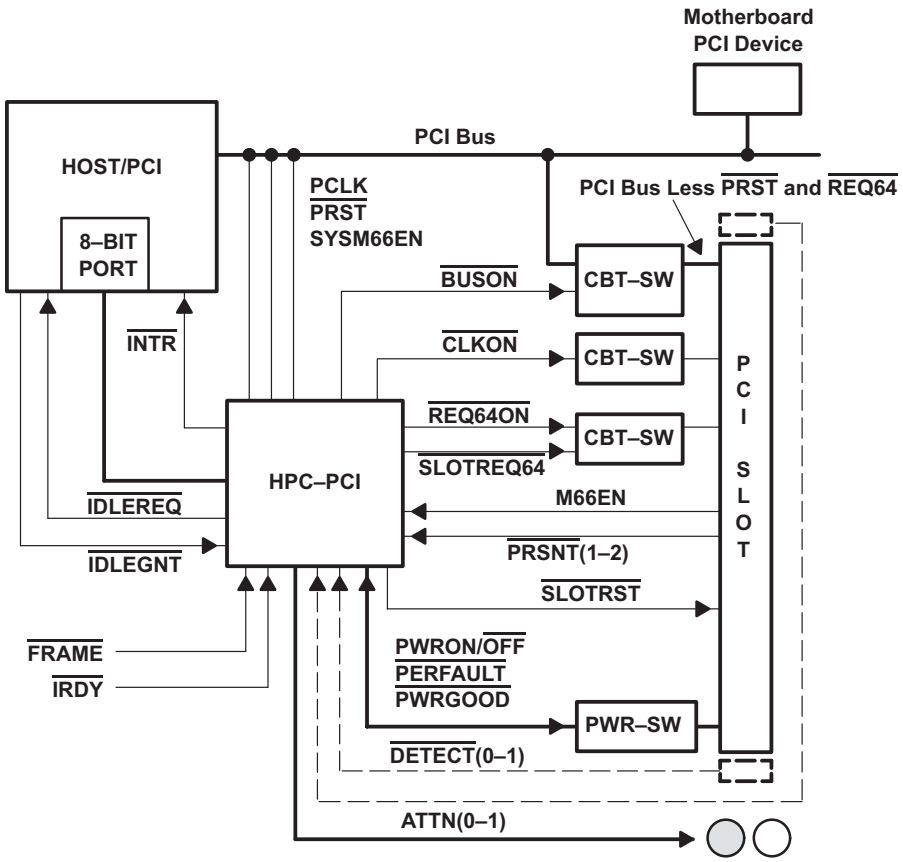



**Figure 1. HPC3130A Implementation**

The HPC3130A internal registers can be accessed through either a two-wire serial interface or an 8-bit generic parallel bus (ISA-like). The above figure illustrates the 8-bit port configuration. Not shown in the diagram is the SMODE chip input that must be wired low to indicate parallel bus interface mode. Also not shown in the diagram is the external chip-select logic required to select the HPC3130A in ISA bus cycles.

**serial interface**

The internal registers can be accessed either through a two-wire serial interface or through an 8-bit generic parallel interface. The SMODE input selects one of these modes.

The HPC3130A implements a two-pin serial slave interface with one clock signal (SCL) and one data signal (SDA). This serial interface can operate with a serial clock frequency up to 400 kHz. Both SCL and SDA require pullup resistors for the serial slave interface to function properly.

All data transfers are initiated by the serial bus master. The beginning of a data transfer is indicated by a START condition (S) when the SDA line transitions to a low state while SCL is in a high state as illustrated in Figure 2. The end of a requested data transfer is indicated by a STOP condition (P), which is the low-to-high transition of SDA while SCL is in the high state. Data on SDA must remain stable during the high state of the SCL signal. Changes on the SDA signal during the high state of SCL will be interpreted as control signals, that is, a START or STOP condition.

The SCL is an input into the HPC3130A and SDA is bidirectional.

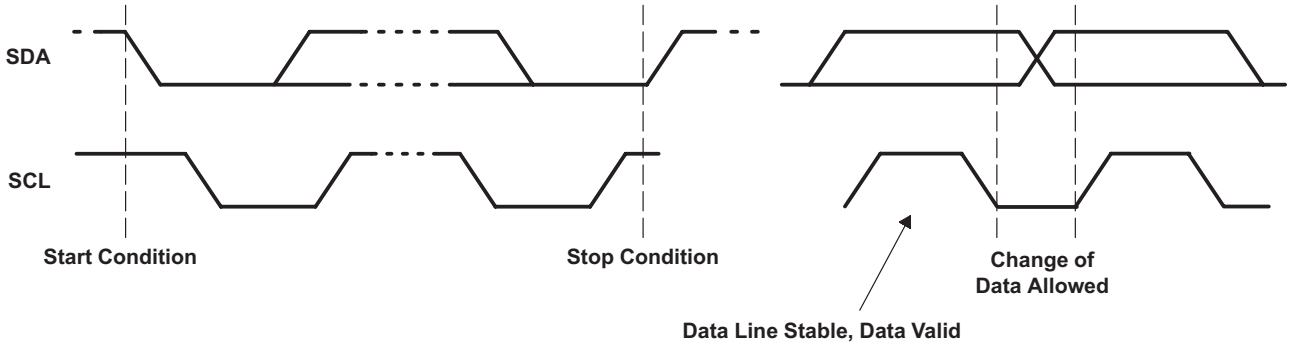



**Figure 2. Serial Bus Start/Stop Conditions and Bit Transfers**

Data is transferred on the bus in 8-bit bytes. The number of bytes that can be transmitted during a data transfer is unlimited; however, each byte must be completed with an acknowledge bit. An acknowledge (ACK) is indicated by the receiver pulling down the SDA signal so that it remains low during the high state of the SCL signal as shown in Figure 3.

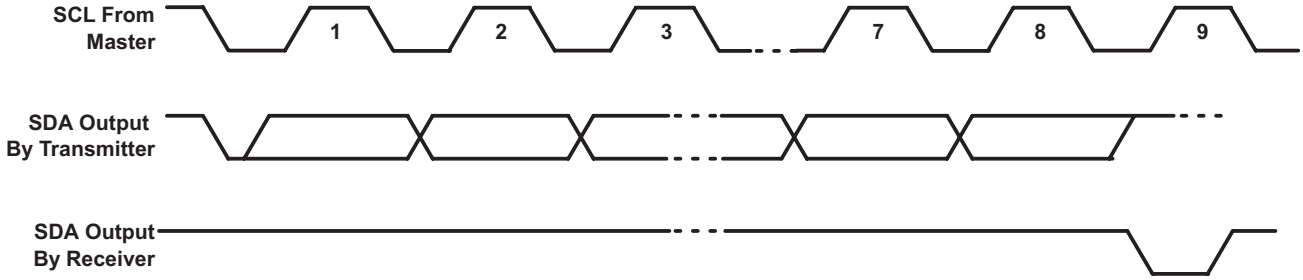



**Figure 3. Serial Bus Protocol – Acknowledge**

The HPC3130A serial bus slave interface protocol for write transactions is illustrated in Figure 4. The R/$\overline{W}$ command bit is set to zero to indicate a write transaction. For a write operation, the HPC3130A requires a word address field after the slave address. This address field is comprised of eight bits. Upon receipt of the word address, the HPC3130A responds with an acknowledge, and waits for the next eight bits of data, again responding with an acknowledge. After all the data bytes are transferred, the master then terminates the transfer by generating a STOP condition. The device automatically increments the address for subsequent data words. After the receipt of each word, the low order address bits are internally incremented by one.

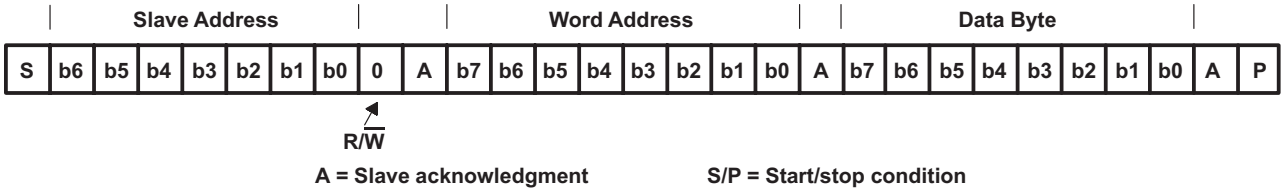



**Figure 4. Serial Bus Protocol – Byte Write**

A byte read operation is illustrated in Figure 5. The read protocol is very similar to the write protocol, except the R/$\overline{W}$ command bit must be set to one to indicate a read data transfer. First the master issues a write command that includes the START condition and the slave address field (with the R/$\overline{W}$ bit set to write), followed by the address of the word it is to read. This procedure sets the internal address counter of the HPC3130A to the desired address. After the word address acknowledgment is received by the master, the master immediately reissues a START condition followed by another slave address field with the R/$\overline{W}$ bit set to read. The HPC3130A responds with an acknowledgment and transmits the eight data bits stored in the addressed location. If the

master responds with an acknowledge signal, indicating that it requires additional data, the HPC3130A continues to output data for each received acknowledge signal. The master terminates the sequential read operation by not responding with an acknowledge signal, and issues a STOP condition.

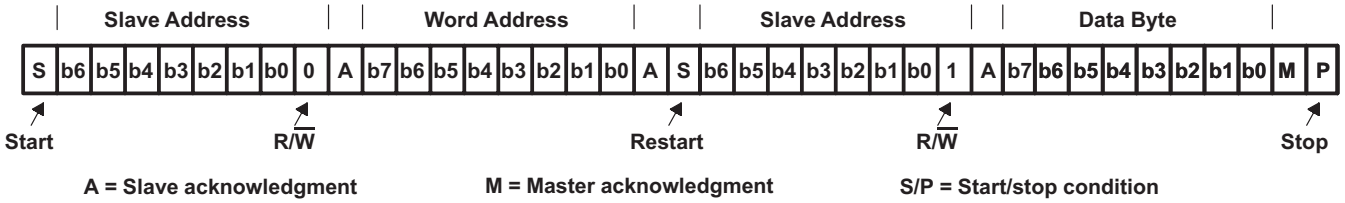



**Figure 5. Serial Bus Protocol – Byte Read**

### parallel interface

The HPC3130A also implements an 8-bit parallel interface mode. When this mode is selected, the HPC3130A internal register addressed by the A[4:0] inputs can be accessed for a read/write transaction using the $\overline{CS}$, $\overline{RD}$, $\overline{WR}$ strobes. The following signals have pullups on the mother board: $\overline{IO16}$, $\overline{M16}$, $\overline{NOWS}$, CHRDY, $\overline{MEMR}$, $\overline{MEMW}$, $\overline{IOR}$, $\overline{IOW}$, SD[15:0] to implement default states. Figure 6 shows write access using the default 8-bit standard ISA bus cycle with four wait states. A read cycle is similar.

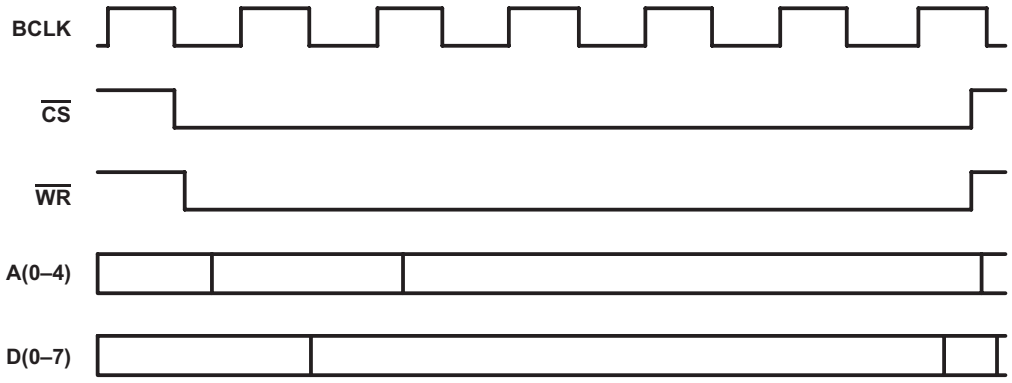



**Figure 6. Parallel Bus Write Cycle**

### connection sequencing

Before an add-in card is hot plugged and made available to the slot, the various pins in the HPC3130A have to be controlled in a specific sequence. The HPC3130A provides the software interface to sequence the power to the slot, clocks, and signals to the add-in cards that are being live inserted. The switch-timing block is used to control the exact timing when the CBT switches are enabled.

The initial software sequencing is done by setting individual bits in the hot plug control register in the following sequence. First, the SLTPWR_CTL bit is set high to drive the PWRON/$\overline{OFF}$ signal high. After the power to the slot is applied, the SLOTRST_O bit is set low to drive the $\overline{SLOTRST}$ output. Next the CLKON_O bit is set low to enable the PCI clock to the slot. Also, the REQ64_O bit is set to a value of 0 and the SLOTREQ64 bit is set to indicate to the add-in card whether it is inserted into a 64-bit or 32-bit slot. SLOTREQ64 is set low for a 64-bit slot and is set high for a 32-bit slot.

After initial software sequencing of the above signals is complete, the next step is to enable the CBT switches. This can be done either by using the software to manually set the BUS_CTL bit or using the HPC3130A via the automatic connection sequence mode located in the general configuration register.

**connection sequencing (continued)**

If automatic sequencing mode 1 is selected, then the BUS_CTL bit controls the sequencing by using idling protocol. When this bit is set to zero, the switch–timing block will arbitrate for the PCI bus by asserting $\overline{\text{IDLEREQ}}$. Subsequently, $\overline{\text{IDLEGNT}}$ is asserted and the HPC3130A waits for bus idle condition. When $\overline{\text{FRAME}}$ and $\overline{\text{IRDY}}$ are deasserted, the CBT switches are enabled. Following this, the $\overline{\text{SLOTRST}}$, $\overline{\text{SLOTREQ64}}$ and $\overline{\text{REQ64ON}}$ are deasserted. Figure 7 depicts the sequencing of events when automatic sequencing mode 1 is enabled.

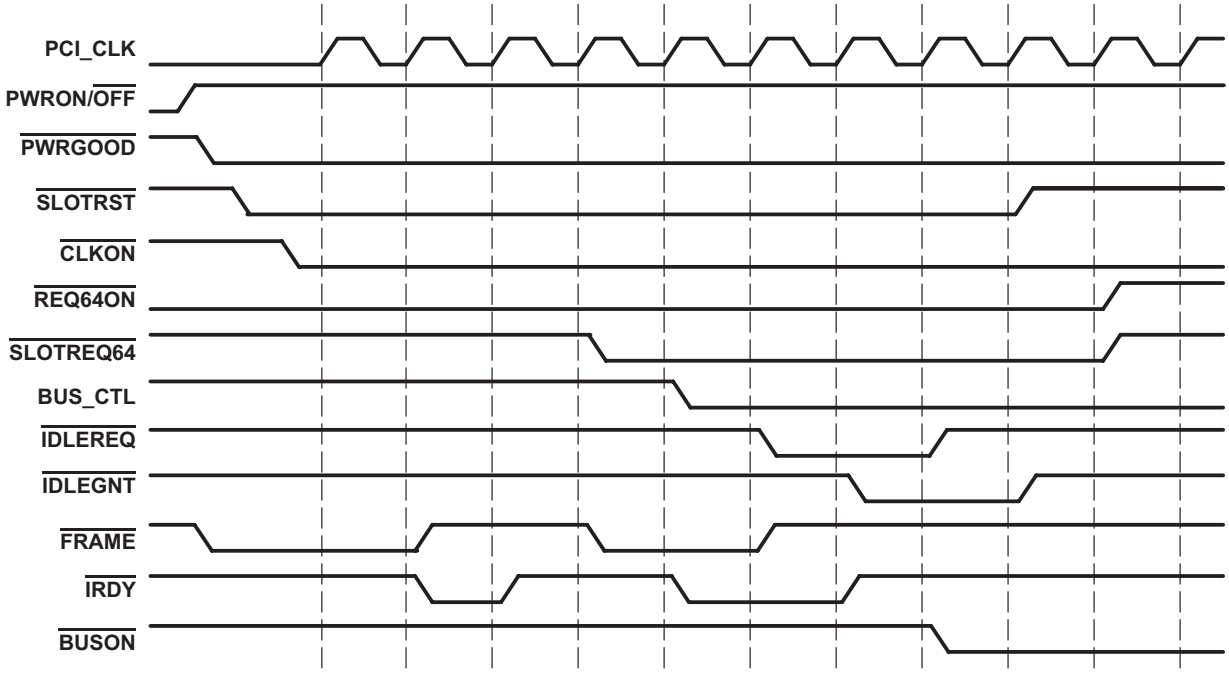



**Figure 7. Automatic Connection Sequencing Mode 1**

If automatic sequencing mode 2 is selected, then the BUS_CTL bit controls the sequencing by using idling protocol. When this bit is set to zero, the switch timing block arbitrates for the PCI bus by asserting $\overline{\text{IDLEREQ}}$. Subsequently, $\overline{\text{IDLEGNT}}$ is asserted and the HPC3130A waits for the bus idle condition. When $\overline{\text{FRAME}}$ and $\overline{\text{IRDY}}$ are deasserted, the $\overline{\text{SLOTRST}}$ is deasserted. Following this, the $\overline{\text{SLOTREQ64}}$ and $\overline{\text{REQ64ON}}$ are deasserted and the CBT switches are enabled. Figure 8 depicts the sequencing of events when automatic sequencing mode 2 is enabled.

**connection sequencing (continued)**

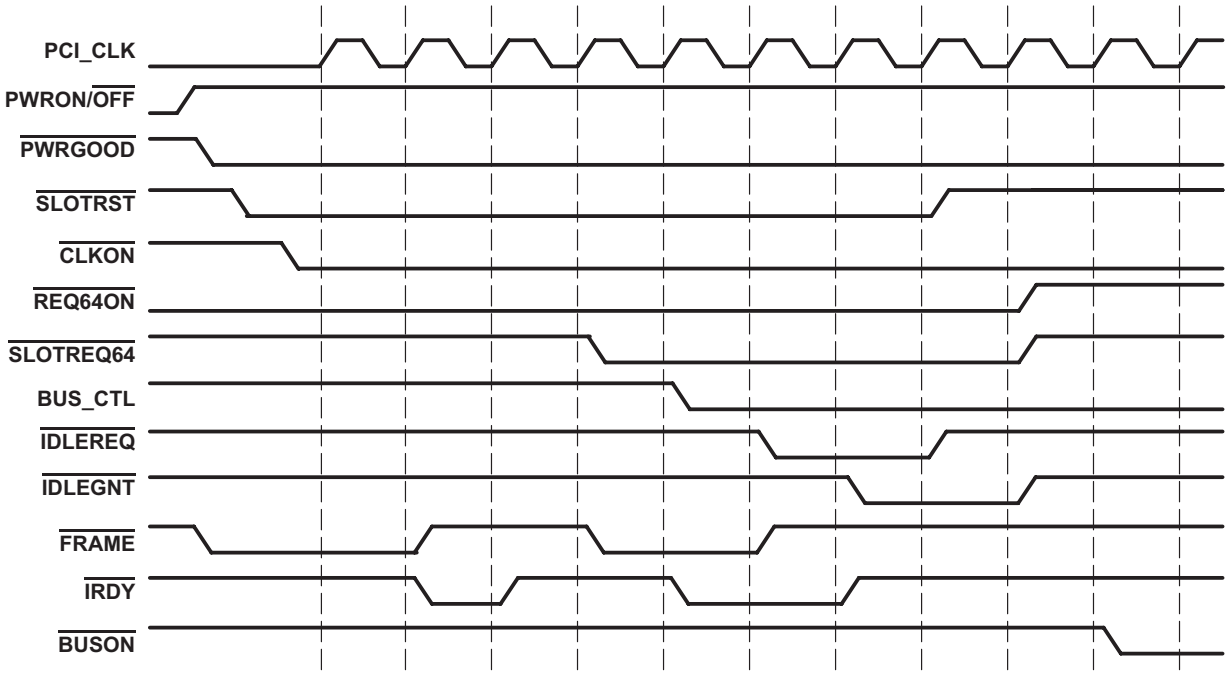



**Figure 8. Automatic Connection Sequencing Mode 2**

The protocol described above is identical to the PCI $\overline{\text{REQ}}$/$\overline{\text{GNT}}$ protocol used by PCI bus masters, which also must wait for bus idle through PCI $\overline{\text{FRAME}}$ and $\overline{\text{IRDY}}$ before initiating a PCI cycle. If the HPC3130A is connected to PCI $\overline{\text{FRAME}}$ and $\overline{\text{IRDY}}$, the HPC3130A arbitrates for the bus; although it does not drive the PCI bus or assert $\overline{\text{FRAME}}$ to start a cycle.

There are some issues with this implementation such as bus parking and additional loading on the PCI $\overline{\text{FRAME}}$ and $\overline{\text{IRDY}}$ signals, which need to be considered when designing a system. The system designer may have a level of confidence that PCI adapter cards can tolerate connection to a non-idle bus. In the scenario where the HPC3130A is not connected to the bus, then the $\overline{\text{FRAME}}$, $\overline{\text{IRDY}}$, and $\overline{\text{IDLEGNT}}$ must be wired to valid logic levels and the automatic sequencing will start without any relationship to the bus.

**connection sequencing (continued)**

$\overline{\text{IDLEREQ}}$, $\overline{\text{IDLEGNT}}$, $\overline{\text{FRAME}}$, and $\overline{\text{IRDY}}$ pins are not connected in manual mode. In contrast, during automatic connection, the HPC3130A requests the PCIBus to make sure it is idle before it sequences through the connection sequence.

In manual mode, software has to perform each part of the connection sequence. Figure 9 is an example of the manual connection sequence. There could be several other ways to implement this protocol.

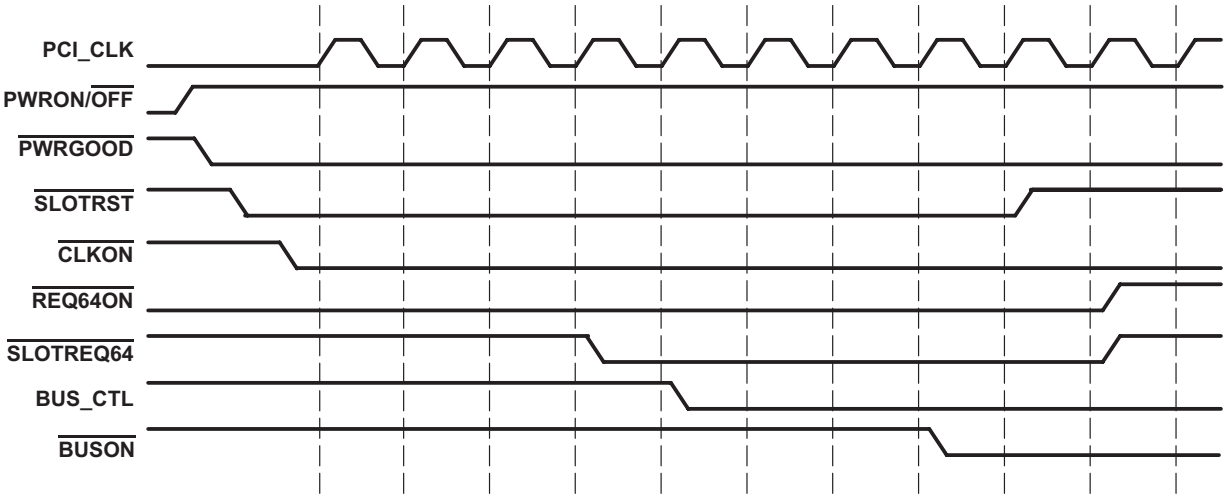



**Figure 9. Manual Connection Sequencing**

**disconnecting sequence**

The HPC3130A provides two mechanisms to isolate a PCI slot from the PCI bus so an add–in card can be removed. One of the mechanisms is called manual sequencing and is enabled by default. This mechanism allows the software to control the entire disconnect sequence. This means it is the software's responsibility to remove a powered slot from the PCI bus without any impact to the system.

The second mechanism is called autosequencing and can be enabled by programming either Auto-Sequence 1 or Auto-Sequence 2 in the general configuration register. Unlike the connection sequence, which has two different autoconnection sequencing modes, the autodisconnect sequence has only one mode of operation. The steps in the autodisconnect sequence are as follows:

1. Software may assert $\overline{\text{SLOTRST}}$ to the appropriate slot by writing a 0 to the SLOTRST_O bit in the hot-plug control register. This is optional.
2. Next the software must set the BUS_CTL bit in the hot-plug control register to a 1.
3. Once the BUS_CTL bit is set to a 1, the HPC3130A asserts $\overline{\text{IDLEREQ}}$.
4. Once $\overline{\text{IDLEGNT}}$ is asserted and $\overline{\text{FRAME}}$ and $\overline{\text{IRDY}}$ are deasserted, the HPC3130A deasserts $\overline{\text{BUSON}}$ and $\overline{\text{CLKON}}$ and asserts $\overline{\text{REQ64ON}}$ to isolate the slot from the PCI bus.
5. The HPC3130A then drives PWRON/$\overline{\text{OFF}}$ low to power off the slot.

**disconnecting sequence (continued)**

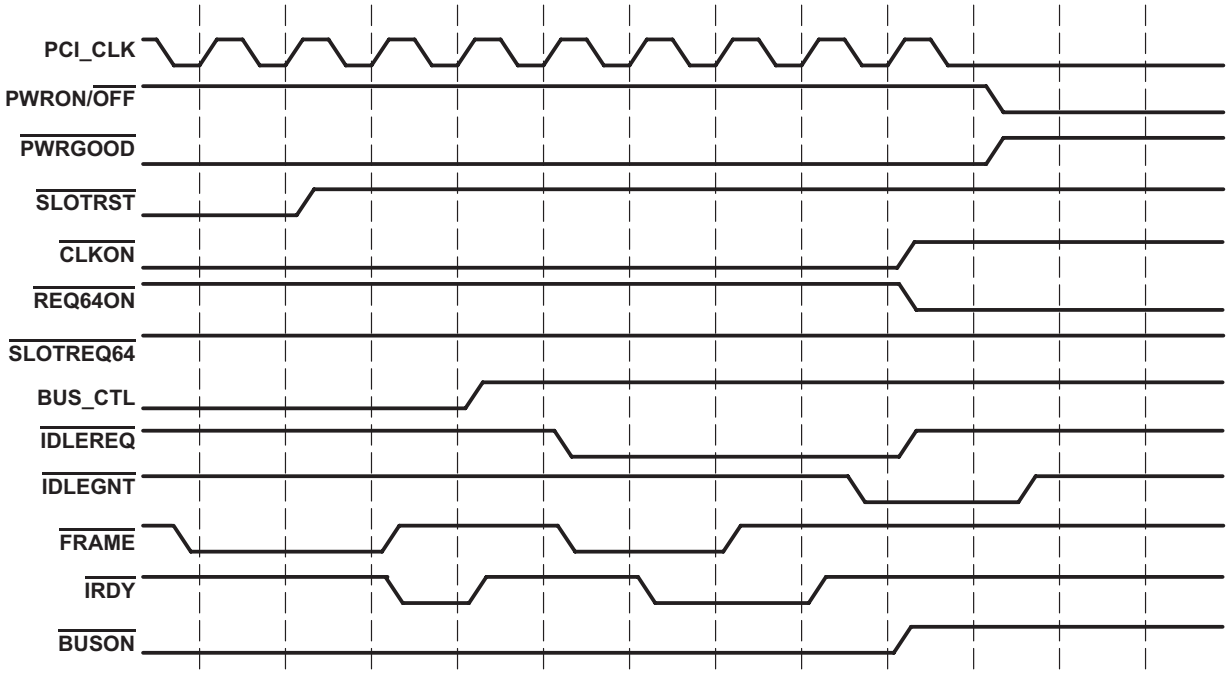



**Figure 10. Automatic Disconnect Sequencing Mode**

In manual disconnect mode, software has to perform each part of the disconnection sequence. Figure 11 is an example of the manual disconnection sequence. There could be several other ways to implement this protocol. During manual disconnection, the PCI bus may or may not idle; it will depend on the software implementation of the system.

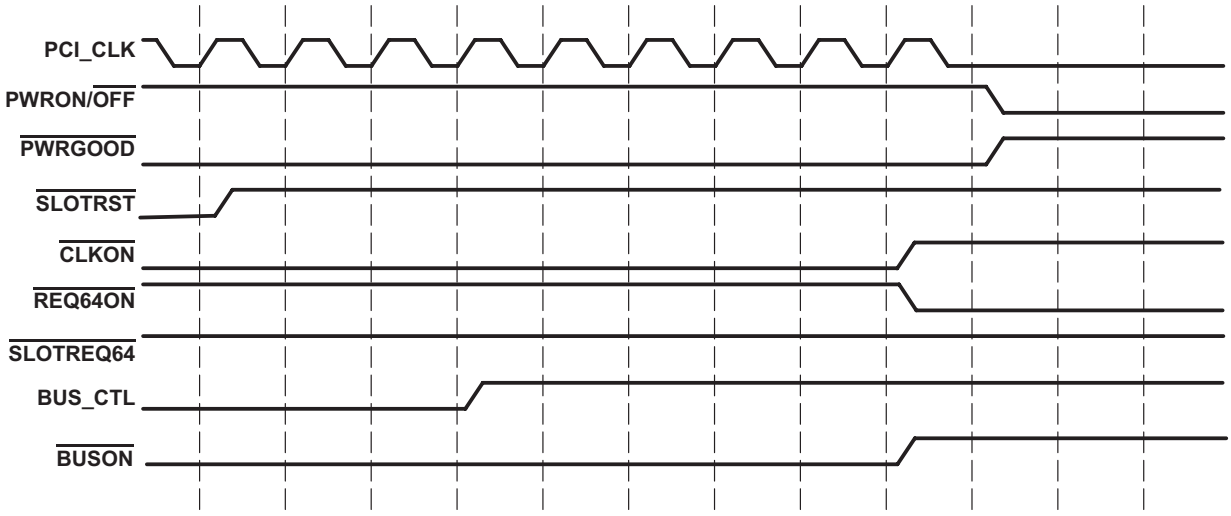



**Figure 11. Manual Disconnect Sequencing Mode**

### 64-bit implementation

When inserted into a slot, a 64-bit PCI card tests whether that slot is 64-bit or 32-bit. If $\overline{\text{REQ64}}$ is low when $\overline{\text{PCI\_RST}}$ is deasserted, that signals the card that the slot is 64-bit; otherwise, the slot is 32-bit. The HPC3130A provides a mechanism to notify the PCI card it is connected to a 64-bit PCI slot. The mechanism uses REQ64ON and $\overline{\text{REQ64ON}}$ as CBT switch enables and $\overline{\text{SLOTREQ64}}$ as the slot specific $\overline{\text{REQ64}}$ signal. As depicted in figures 12 and 13, $\overline{\text{REQ64ON}}$ enables the CBT switch that routes $\overline{\text{SLOTREQ64}}$ to the slot, and REQ64ON enables the CBT switch that routes $\overline{\text{SYSTEMREQ64}}$ to the slot. $\overline{\text{SYSTEMREQ64}}$ will always be routed to the slot except when the slot is disconnected from the bus or during a connection sequence. It is very important during a connection sequence that the software clears bit 3 in the *hot plug control register* to drive $\overline{\text{SLOTREQ64}}$ low. The HPC3130A will deassert $\overline{\text{SLOTREQ64}}$ during the automatic connection sequence.

In a 32-bit PCI bus implementation, the $\overline{\text{SLOTREQ64}}$, $\overline{\text{REQ64ON}}$, and REQ64ON outputs are not connected.

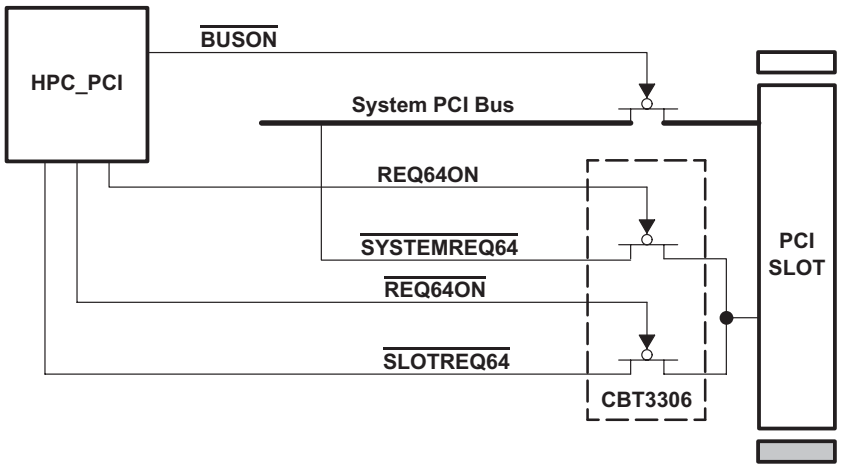



**Figure 12. $\overline{\text{SLOTREQ64}}$ Implementation Using CBT3306**

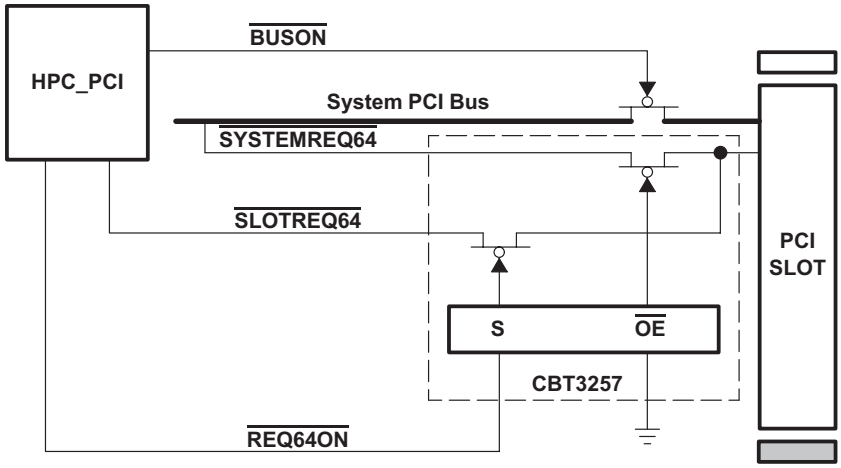



**Figure 13. $\overline{\text{SLOTREQ64}}$ Implementation Using CBT3257**

### 66-MHz PCI support

The HPC3130A supports up to two slots in a 66-MHz system: slot 0 and slot 1. These hot-plug slot interfaces include the M66EN signal and the SYSM66EN signal communicates hot-plug slot capability to the system. Figure 14 illustrates the 66-MHz support function in the HPC3130A.

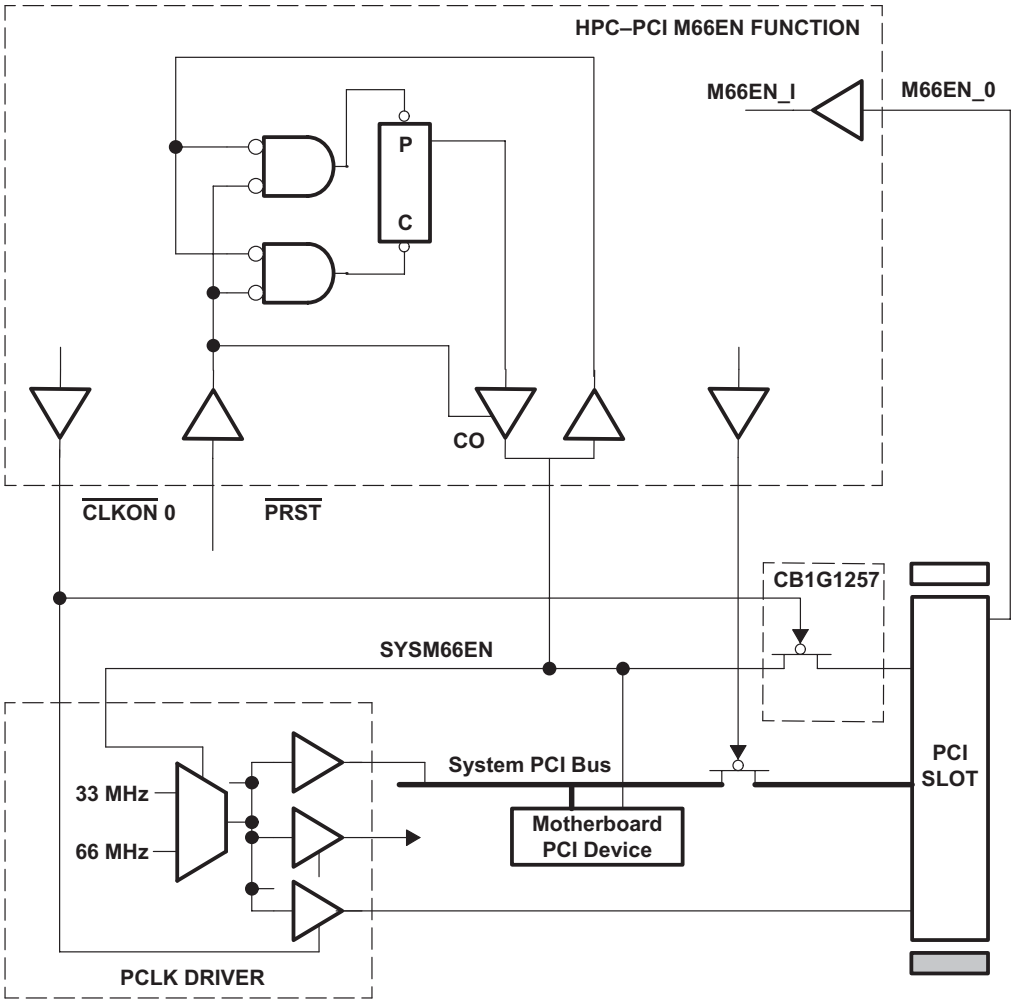



**Figure 14. HPC3130A 66-MHz System Support**

When the $\overline{\text{PCIRST}}$ signal is deasserted, the SYSM66EN signal is latched and selects the system frequency.

Before an adapter card can be inserted, the slot is prepared for insertion by deasserting $\overline{\text{BUSON}}$, deasserting $\overline{\text{CLKON}}$, and powering down the slot. When an adapter card is inserted into either hot-plug slot 0 or slot 1, $\overline{\text{PWRON/OFF}}$ is high, and power is applied, the value M66EN from the card can be read through internal registers. If the bus is operating at 66-MHz and a 33-MHz adapter card is inserted, then the software ensures that the card is never connected to the bus. If the bus is operating at 33 MHz and a 66-MHz adapter card is inserted, then the latched SYSM66EN state can be driven to the slot by enabling the CBT switch using $\overline{\text{CLKON}}$.

SYSM66EN and $\overline{\text{CLKON}}$ input to a clock driver circuit to control the PCI clock frequency. The 66-MHz support designed into the HPC3130A allows option cards to indicate PCI clock frequency capabilities upon PCI reset; however, does not allow an inserted hot-plug card to alter the clock frequency of an operating PCI bus.

## configuration and control registers

The HPC3130A register set is accessible through either a generic parallel bus interface or a two-wire serial interface. Eight bytes of register space are provided per slot. Since the HPC3130A supports four slots, a total of 32 bytes of registers is implemented. Register and bit descriptions are provided in the following sections and indicate the bits that are common to all slots. The bit default values are given, which represent the state of the HPC3130A after a PCI reset event. After a PCI reset, the HPC3130A drives outputs to a state such that the slots appear as if they were not a hot-plug platform.

The register map that follows provides the register overview. Byte addressing is required when accessing the internal registers. Read transactions from reserved registers return zeros.

**Table 7. Register Map**

| SLOT | REGISTER NAME | ADDRESS | SLOT | REGISTER NAME | ADDRESS |
|---|---|---|---|---|---|
| Slot 0 | General configuration register | 0x00 | Slot 2 | General configuration register | 0x10 |
| Slot 0 | Hot-plug slot status register | 0x01 | Slot 2 | Hot-plug slot status register | 0x11 |
| Slot 0 | Hot-plug slot control register | 0x02 | Slot 2 | Hot-plug slot control register | 0x12 |
| Slot 0 | Attention indicator control | 0x03 | Slot 2 | Attention indicator control | 0x13 |
| Slot 0 | Reserved | 0x04 | Slot 2 | Reserved | 0x14 |
| Slot 0 | Reserved | 0x05 | Slot 2 | Reserved | 0x15 |
| Slot 0 | Interrupt event status register | 0x06 | Slot 2 | Interrupt event status register | 0x16 |
| Slot 0 | Interrupt event enable register | 0x07 | Slot 2 | Interrupt event enable register | 0x17 |
| Slot 1 | General configuration register | 0x08 | Slot 3 | General configuration register | 0x18 |
| Slot 1 | Hot-plug slot status register | 0x09 | Slot 3 | Hot-plug slot status register | 0x19 |
| Slot 1 | Hot-plug slot control register | 0x0A | Slot 3 | Hot-plug slot control register | 0x1A |
| Slot 1 | Attention indicator control | 0x0B | Slot 3 | Attention indicator control | 0x1B |
| Slot 1 | Reserved | 0x0C | Slot 3 | Reserved | 0x1C |
| Slot 1 | Reserved | 0x0D | Slot 3 | Reserved | 0x1D |
| Slot 1 | Interrupt event status register | 0x0E | Slot 3 | Interrupt event status register | 0x1E |
| Slot 1 | Interrupt event enable register | 0x0F | Slot 3 | Interrupt event enable register | 0x1F |

## configuration and control registers (continued)

### general configuration register

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Name | General configuration register | | | | | | | |
| Type | R | R | R | R | R/W | R/W | R | R/W |
| Default | 0 | 0 | 1 | 1 | 0 | 0 | X | 0 |

Register: General configuration
Type: Read-only, Read/Write
Offset: 00h (slot 0), 08h (slot 1), 10h (slot 2), 18h (slot 3)
Default: 0Xh
Description: This register is for general configurations and indications. The automatic PCI bus connection sequencing is enabled through this register, and the register access mode is indicated. This register is shared among all four slots.

**Table 8. General Configuration Register**

| BIT | TYPE | NAME | FUNCTION |
|---|---|---|---|
| 7–4 | R | RSVD | Reserved for revision ID. These bits return 0011b for this device. |
| 3–2 | R/W | SEQUENCING | Automatic PCI bus connection sequencing. These bits control the sequencing used to connect the hot-plug slot to the PCI bus.<br>00 = Manual sequencing through register accesses<br>01 = Auto–Sequence 1: Enable CBT switches before deasserting $\overline{\text{RST}}$<br>10 = Auto–Sequence 2: Enable CBT switches after deasserting $\overline{\text{RST}}$<br>11 = Reserved |
| 1 | R | SYSM66STAT | Status of SYSM66EN. This bit represents the latched value of SYSM66EN during a PCI reset. A value of 1 indicates the PCI bus is operating at a frequency greater than 33 MHz. A value of 0 indicates the PCI bus is operating at 33 MHz or less. |
| 0 | R/W | PROTECTEN | Protection enable. This bit enables a protection mechanism provided by the HPC3130A. When this bit is enabled and either of the $\overline{\text{DETECT}}$[1:0] inputs are high, the HPC3130A drives the $\overline{\text{BUS\_ON}}$ and $\overline{\text{CLKON}}$ outputs high. The HPC3130A also drives PWRON/$\overline{\text{OFF}}$ and $\overline{\text{REQ64ON}}$ outputs low. |

## configuration and control registers (continued)

### hot-plug slot status register

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Name | Hot-plug slot status register | | | | | | | |
| Type | R | R | R | R | R | R | R | R |
| Default | X | X | X | X | X | X | X | X |

Register: Hot-plug slot status
Type: Read-only
Offset: 01h (slot 0), 09h (slot 1), 11h (slot 2), 19h (slot 3)
Default: XXh
Description: This register reports card detection, power status, and other chip input from the hot-plug slot interface. All bits in this register are read only, and the data read from each bit represents the logical value of the data input from the corresponding terminal.

**Table 9. Hot-Plug Slot Status Register**

| BIT | TYPE | NAME | FUNCTION |
|---|---|---|---|
| 7 | R | BUSON | Bus on. This bit returns the logical value of the $\overline{\text{BUSON}}$ terminal output. |
| 6 | R | M66EN_I | M66EN input. This bit returns the logical value of the M66EN terminal input. |
| 5 | R | PWRGOOD_I | Power good input. This bit returns the logical value of the $\overline{\text{PWRGOOD}}$ terminal input. |
| 4 | R | PWRFAULT_I | Power fault input. This bit returns the logical value of the $\overline{\text{PWRFAULT}}$ terminal input. |
| 3 | R | DETECT1_I | Mech detect 1 input. This bit returns the logical value of the $\overline{\text{DETECT1}}$ terminal input. |
| 2 | R | DETECT0_I | Mech detect 0 input. This bit returns the logical value of the $\overline{\text{DETECT0}}$ terminal input. |
| 1 | R | PRSNT2_I | Card present 2 input. This bit returns the logical value of the $\overline{\text{PRSNT2}}$ terminal input. |
| 0 | R | PRSNT1_I | Card present 1 input. This bit returns the logical value of the $\overline{\text{PRSNT1}}$ terminal input. |

## configuration and control registers (continued)

### hot-plug slot control register

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Name | Hot-plug slot control register | | | | | | | |
| Type | R | R | R/W | R/W | R/W | R/W | R/W | R/W |
| Default | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 1 |

Register: Hot-plug slot control
Type: Read-only, Read/Write
Offset: 02 (slot 0), 0Ah (slot 1), 12h (slot 2), 1Ah (slot 3)
Default: 2Dh
Description: This register applies power, resets, and provides general control of a hot-plug slot connection to the system PCI bus.

**Table 10. Hot-Plug Slot Control Register**

| BIT | TYPE | NAME | FUNCTION |
|---|---|---|---|
| 7 | R | RSVD | Reserved. This bit returns 0 when read. |
| 6 | R | RSVD | Reserved. This bit returns 0 when read. |
| 5 | R/W | SLTPWR_CTL | Slot power On/$\overline{\text{Off}}$ control. The data written to this bit represents the logical value of the data to drive the PWRON/$\overline{\text{OFF}}$ output and is used to control the power state of a hot-plug slot. If the PROTECT$\overline{\text{EN}}$ bit in the general configuration register is set to 1, then a logic high can only be driven by the PWRON/$\overline{\text{OFF}}$ output if the $\overline{\text{DETECT}}$[1:0] inputs are low. |
| 4 | R/W | BUS_CTL | PCI bus CBT-switch control. When manual sequencing is enabled, then the value written to this bit represents the logical value of the data driven to the $\overline{\text{BUSON}}$ output, and it is used to connect/disconnect a hot-plug slot to/from the PCI bus.<br>If an auto sequencing mode is enabled in the general configuration register, then this bit functions as follows:<br>1 = By setting this bit, the hot-plug slot gets disconnected from the PCI bus. This is accomplished by asserting $\overline{\text{IDLEREQ}}$, then waiting for $\overline{\text{IDLEGNT}}$ assertion and deassertion of $\overline{\text{FRAME}}$ and $\overline{\text{IRDY}}$ before driving $\overline{\text{BUSON}}$ high, $\overline{\text{CLKON}}$ high, $\overline{\text{REQ64ON}}$ low, and PWRON/$\overline{\text{OFF}}$ low.<br>0 = By clearing this bit, the hot-plug slot gets connected to the PCI bus. This is accomplished by asserting $\overline{\text{IDLEREQ}}$, then waiting for $\overline{\text{IDLEGNT}}$ assertion and deassertion of $\overline{\text{FRAME}}$ and $\overline{\text{IRDY}}$ before driving $\overline{\text{BUSON}}$ low. Also verifies assertion of $\overline{\text{DETECT}}$[1:0] if the protection enable bit is enabled in the general configuration register. |
| 3 | R/W | SLOTREQ64 | Slot request 64-bit control. The data written to this bit represents the logical value of the data driven to the $\overline{\text{SLOTREQ64}}$ output and is used during reset of a slot after power is applied. This input indicates to an option card whether or not it is connected to a 64-bit slot. |
| 2 | R/W | REQ64_O | $\overline{\text{REQ64}}$ CBT switch control. The data written to this bit represents the logical value of the data driven to the $\overline{\text{REQ64ON}}$ output and is used to control the CBT switch that implements the $\overline{\text{REQ64}}$ PCI signal. |
| 1 | R/W | CLKON_O | $\overline{\text{CLKON}}$ CBT switch control. The data written to this bit represents the logical value of the data driven to the $\overline{\text{CLKON}}$ output and is used to control the clock driver to the hot-plug slot. |
| 0 | R/W | SLOTRST_O | Slot reset control. The data written to this bit represents the logical value of the data driven to the $\overline{\text{SLOTRST}}$ output and is used to reset a hot-plug slot after power is applied. |

## configuration and control registers (continued)

### attention indicator control register

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Name | Attention indicator control register | | | | | | | |
| Type | R | R | R | R | R/W | R/W | R/W | R/W |
| Default | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Register: Attention indicator control
Type: Read-only, Read/Write
Offset: 03 (slot 0), 0Bh (slot 1), 13h (slot 2), 1Bh (slot 3)
Default: 00h
Description: This register controls the attention indicators. The timing for the indicators is based upon the PCI clock and the M66EN input.

**Table 11. Attention Indicator Control Register**

| BIT | TYPE | NAME | FUNCTION |
|---|---|---|---|
| 7 | R | RSVD | Reserved. This bit returns 0 when read. |
| 6 | R | RSVD | Reserved. This bit returns 0 when read. |
| 5 | R | RSVD | Reserved. This bit returns 0 when read. |
| 4 | R | RSVD | Reserved. This bit returns 0 when read. |
| 3–2 | R/W | ATTN1_CTL | Attention indicator 1 control. These bits control the state of ATTN1 per slot and are programmed as follows:<br>00 = Drive low<br>01 = Slow blink – 1 cycle per second<br>10 = Fast blink – 2 cycles per second<br>11 = Drive high |
| 1–0 | R/W | ATTN0_CTL | Attention indicator 0 control. These bits control the state of ATTN0 per slot and are programmed as follows:<br>00 = Drive low<br>01 = Slow blink – 1 cycle per second<br>10 = Fast blink – 2 cycles per second<br>11 = Drive high |

## configuration and control registers (continued)

### interrupt event status register

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Name | Interrupt event status register | | | | | | | |
| Type | R | R/C | R/C | R/C | R/C | R/C | R/C | R/C |
| Default | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Register: Interrupt event status
Type: Read-only, Read/Clear
Offset: 06h (slot 0), 0Eh (slot 1), 16h (slot 2), 1Eh (slot 3)
Default: 00h
Description: This register reads interrupt status, and clears the interrupt. All functional bits in this register are readable and cleared by a write back of 1. The HPC3130A can be programmed to generate an interrupt, signaled through the open-drain $\overline{\text{INTR}}$, after detecting various events. Each event is individually enabled through the interrupt event enable register.

**Table 12. Interrupt Event Status Register**

| BIT | TYPE | NAME | FUNCTION |
|---|---|---|---|
| 7 | R | RSVD | Reserved. This bit returns 0 when read. |
| 6 | R/C | BUS_S | PCI Bus CBT switch status. This bit is set when the $\overline{\text{BUSON}}$ output changes state, and is cleared by a write back of 1. The BUS event is intended for use with the idling protocol. |
| 5 | R/C | PWRGOOD_S | Power good status. This bit is set when the $\overline{\text{PWRGOOD}}$ input changes state. |
| 4 | R/C | PWRFAULT_S | Power fault status. This bit is set when the $\overline{\text{PWRFAULT}}$ input is asserted. |
| 3 | R/C | DETECT1_S | Mechanical detect 1 status. This bit is set when the $\overline{\text{DETECT1}}$ input changes state. |
| 2 | R/C | DETECT0_S | Mechanical detect 0 status. This bit is set when the $\overline{\text{DETECT0}}$ input changes state. |
| 1 | R/C | PRSNT2_S | Card present 2 status. This bit is set when the $\overline{\text{PRSNT2}}$ input changes state. |
| 0 | R/C | PRSNT1_S | Card present 1 status. This bit is set when the $\overline{\text{PRSNT1}}$ input changes state. |

## configuration and control registers (continued)

### interrupt event enable register

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Name | Interrupt event enable register | | | | | | | |
| Type | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Default | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Register: Interrupt event enable
Type: Read-only, Read/Write
Offset: 07h (slot 0), 0Fh (slot 1), 17h (slot 2), 1Fh (slot 3)
Default: 00h
Description: This register is used to enable interrupts, signaled through the open-drain $\overline{\text{INTR}}$, after detecting various events. Event status is reported through the interrupt event status register.

**Table 13. Interrupt Event Enable Register**

| BIT | TYPE | NAME | FUNCTION |
|---|---|---|---|
| 7 | R | RSVD | Reserved. This bit returns 0 when read. |
| 6 | R/W | BUS_E | PCI bus CBT switch event enable. When this bit is set, an $\overline{\text{INTR}}$ is signaled when the $\overline{\text{BUSON}}$ output changes state. The BUS event is intended for use with the idling protocol. |
| 5 | R/W | PWRGOOD_E | Power good event enable. When this bit is set, an $\overline{\text{INTR}}$ is signaled when the $\overline{\text{PWRGOOD}}$ input changes state. |
| 4 | R/W | PWRFAULT_E | Power fault event enable. When this bit is set, an $\overline{\text{INTR}}$ is signaled when $\overline{\text{PWRFAULT}}$ input is asserted. |
| 3 | R/W | DETECT1_E | Mechanical detect 1 event enable. Enables $\overline{\text{INTR}}$ events on $\overline{\text{DETECT1}}$ input state changes. |
| 2 | R/W | DETECT0_E | Mechanical detect 0 event enable. Enables $\overline{\text{INTR}}$ events on $\overline{\text{DETECT0}}$ input state changes. |
| 1 | R/W | PRSNT2_E | Card present 2 event enable. Enables $\overline{\text{INTR}}$ events on $\overline{\text{PRSNT2}}$ input state changes. |
| 0 | R/W | PRSNT1_E | Card present 1 event enable. Enables $\overline{\text{INTR}}$ events on $\overline{\text{PRSNT1}}$ input state changes. |

**absolute maximum ratings over operating free-air temperature range†**

| | | |
|---|---|---|
| Supply voltage range, $V_{CC}$ | | –0.5 V to 3.6 V |
| $V_{CCP}$ and $V_{CC5V}$ | | –0.5 V to 5.5 V |
| Input voltage range, $V_I$: | PCI | –0.5 V to $V_{CCP}$ + 0.5 V |
| | SLOT | –0.5 V to $V_{CC5V}$ + 0.5 V |
| | BUSON | –0.5 V to $V_{CC5V}$ + 0.5 V |
| | Parallel | –0.5 V to $V_{CC5V}$ + 0.5 V |
| | Parallel/serial | –0.5 V to $V_{CC5V}$ + 0.5 V |
| | Miscellaneous | –0.5 V to $V_{CC5V}$ + 0.5 V |
| Output voltage range, $V_O$: | PCI | –0.5 V to $V_{CCP}$ + 0.5 V |
| | SLOT | –0.5 V to $V_{CC5V}$ + 0.5 V |
| | BUSON | –0.5 V to $V_{CC5V}$ + 0.5 V |
| | Parallel | –0.5 V to $V_{CC5V}$ + 0.5 V |
| | Parallel/serial | –0.5 V to $V_{CC5V}$ + 0.5 V |
| | Miscellaneous | –0.5 V to $V_{CC5V}$ + 0.5 V |
| Input clamp current, $I_{IK}$ ($V_I$ < 0 or $V_I$ > $V_{CC}$) (see Note 1) | | ±20 mA |
| Output clamp current, $I_{OK}$ ($V_O$ < 0 or $V_O$ > $V_{CC}$) (see Note 2) | | ±20 mA |
| Storage temperature range | | –65°C to 150°C |
| Junction temperature, $T_J$ | | 150°C |

† Stresses beyond those listed under “absolute maximum ratings” may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under “recommended operating conditions” is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

NOTES: 1. Applies to external input and bidirectional buffers. For 5-V tolerant use $V_I$ > $V_{CC5V}$. For universal PCI, use $V_I$ > $V_{CCP}$.
2. Applies to external output and bidirectional buffers. For 5-V tolerant use $V_O$ > $V_{CC5V}$. For universal PCI, use $V_O$ > $V_{CCP}$.

## recommended operating conditions

| | | | MIN | NOM | MAX | UNIT |
|---|---|---|---|---|---|---|
| $V_{CC}$ | Core voltage | Commercial | 3 | 3.3 | 3.6 | V |
| $V_{CCP}$ | PCI I/O voltage | Commercial | 3 | 5 | 5.5 | V |
| $V_{CC5V}$ | Slot I/O voltage | Commercial | 3 | 5 | 5.5 | V |
| $V_{CC5V}$ | Parallel I/O voltage | Commercial | 3 | 5 | 5.5 | V |
| $V_{CC5V}$ | Serial I/O voltage | Commercial | 3 | 5 | 5.5 | V |
| $V_{CC5V}$ | Miscellaneous I/O voltage | Commercial | 3 | 5 | 5.5 | V |
| $V_I$ | Input voltage | PCI | 0 | | $V_{CCP}$ | V |
| | | Slot | 0 | | $V_{CC5V}$ | |
| | | BUSON | 0 | | $V_{CC5V}$ | |
| | | Parallel | 0 | | $V_{CC5V}$ | |
| | | Parallel/Serial | 0 | | $V_{CC5V}$ | |
| | | Miscellaneous | 0 | | $V_{CC5V}$ | |
| $V_O$ | Output voltage† | PCI | 0 | | $V_{CC}$ | V |
| | | Slot | 0 | | $V_{CC}$ | |
| | | BUSON | 0 | | $V_{CC}$ | |
| | | Parallel | 0 | | $V_{CC}$ | |
| | | Parallel/Serial | 0 | | $V_{CC}$ | |
| | | Miscellaneous | 0 | | $V_{CC}$ | |
| $V_{IH}$ | High-level input voltage | PCI | 2 | | $V_{CCP}$ | V |
| | | Slot | 2 | | $V_{CC5V}$ | |
| | | BUSON | 2 | | $V_{CC5V}$ | |
| | | Parallel | 2 | | $V_{CC5V}$ | |
| | | Parallel/Serial | 2 | | $V_{CC5V}$ | |
| | | Miscellaneous | 2 | | $V_{CC5V}$ | |
| $V_{IL}$ | Low-level input voltage | PCI | 0 | | 0.8 | V |
| | | Slot | 0 | | 0.8 | |
| | | BUSON | 0 | | 0.8 | |
| | | Parallel | 0 | | 0.8 | |
| | | Parallel/Serial | 0 | | 0.8 | |
| | | Miscellaneous | 0 | | 0.8 | |
| $t_t$ | Input transition time ($t_r$ and $t_f$) (10% to 90%) | PCI | 0 | | 6 | ns |
| | | Slot | 0 | | 6 | |
| | | BUSON | 0 | | 6 | |
| | | Parallel/Serial | 0 | | 6 | |
| | | Miscellaneous | 0 | | 6 | |
| $T_A$ | Operating ambient temperature range | | 0 | 25 | 70 | °C |
| $T_J$ | Virtual junction temperature‡ | | 0 | 25 | 115 | °C |

† Applies to external output buffers.

‡ These junction temperatures reflect simulation conditions. Customer is responsible for verifying junction temperature.

## serial bus interface

| | | | STANDARD MODE | | FAST MODE | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| $V_{IL}$ | Low-level input voltage | Fixed:<br>$V_{CC5V}$ related: | –0.5<br>–0.5 | 1.5<br>0.3 $V_{CC5V}$ | –0.5<br>–0.5 | 1.5<br>0.3 $V_{CC5V}$ | V |
| $V_{IH}$ | High-level input voltage | Fixed:<br>$V_{CC5V}$ related: | 0<br>0.7 $V_{CC5V}$ | $V_{CC5V}$ max + 0.5<br>$V_{CC5V}$ max + 0.5 | 3<br>0.7 $V_{CC5V}$ | $V_{CC5V}$ max + 0.5<br>$V_{CC5V}$ max + 0.5 | V |

## electrical characteristics over recommended operating conditions

| PARAMETER | | PINS | OPERATION | TEST CONDITIONS | MIN | MAX | UNIT |
|---|---|---|---|---|---|---|---|
| $V_{OH}$ | High-level output voltage | PCI□ | 3.3 V | $I_{OH}$ = –0.5 mA | 0.9 $V_{CC}$ | | V |
| | | | 5 V | $I_{OH}$ = –2 mA | $V_{CC}$ – 0.6 | | |
| | | Slot§ | | $I_{OH}$ = –8mA | $V_{CC}$ – 0.6 | | |
| | | BUSON¶ | | $I_{OH}$ = –8mA | $V_{CC}$ – 0.6 | | |
| | | Parallel# | | $I_{OH}$ = –4mA | $V_{CC}$ – 0.6 | | |
| | | Parallel/Serial‖ | | $I_{OH}$ = –4mA | $V_{CC}$ – 0.6 | | |
| | | Miscellaneous☆ | | $I_{OH}$ = –8mA | $V_{CC}$ – 0.6 | | |
| $V_{OL}$ | Low-level output voltage | PCI□ | 3.3 V | $I_{OL}$ = 1.5 mA | | 0.1 $V_{CC}$ | V |
| | | | 5 V | $I_{OL}$ = 6 mA | | 0.55 | |
| | | Slot§ | | $I_{OL}$ = 8mA | | 0.5 | |
| | | BUSON¶ | | $I_{OL}$ = 8mA | | 0.5 | |
| | | Parallel# | | $I_{OL}$ = 4mA | | 0.5 | |
| | | Parallel/Serial‖ | | $I_{OL}$ = 4mA | | 0.5 | |
| | | Miscellaneous☆ | | $I_{OL}$ = 2mA | | 0.5 | |
| $I_{IL}$† | Low-level input current‡ | PCI□ | | $V_I$ = GND | | –20 | μA |
| | | Slot§ | | $V_I$ = GND | | –20 | |
| | | BUSON¶ | | $V_I$ = GND | | –20 | |
| | | Parallel# | | $V_I$ = GND | | –20 | |
| | | Parallel/Serial‖ | | $V_I$ = GND | | –20 | |
| | | Miscellaneous☆ | | $V_I$ = GND | | –20 | |
| $I_{IH}$† | High-level input current‡ | PCI□ | | $V_I$ = $V_{CC}$ | | 20 | μA |
| | | Slot§ | | $V_I$ = $V_{CC}$ | | 20 | |
| | | BUSON¶ | | $V_I$ = $V_{CC}$ | | 20 | |
| | | Parallel# | | $V_I$ = $V_{CC}$ | | 20 | |
| | | Parallel/Serial‖ | | $V_I$ = $V_{CC}$ | | 20 | |
| | | Miscellaneous☆ | | $V_I$ = $V_{CC}$ | | 20 | |
| $I_{OZ}$ | High-impedance-state output current | | | $V_I$ = $V_{CC}$ or GND | | ±20 | μA |

† Specifications apply only when pullup terminator is turned off.
‡ For I/O pins, the imput leakage current includes the off state output current $I_{OZ}$.
□ PCI terminals: 120 PBM — 12, 13, 14, 16, 18, 19, 20, 21, 23, 24, 25, 27. 128 PBK — 13, 14, 15, 17, 19, 20, 21, 22, 24, 25, 26, 28. 144 PGE — 15, 16, 17, 19, 21, 22, 23, 24, 26, 27, 28, 30.
§ Slot terminals: 120 PBM — 7, 8, 9, 10, 58, 77, 97, 115. 128 PBK — 8, 9, 10, 11, 61, 82, 104, 122. 144 PGE — 10, 11, 12, 13, 68, 92, 118, 136.
¶ BUSON terminals: 120 PBM — 63, 81, 100, 119. 128 PBK — 68, 86, 107, 126. 144 PGE — 77, 96, 121, 142
# Parallel terminals: 120 PBM — 35, 37, 38, 39, 40, 42. 128 PBK — 38, 40, 41, 42, 43, 45. 144 PGE — 44, 46, 47, 48, 49, 51
‖ Parallel/serial terminals: 120 PBM — 43, 44, 45, 47, 48, 49, 50. 128 PBK — 46, 47, 48, 50, 51, 52, 53. 144 PGE — 52, 53, 54, 56, 57, 58, 59.
☆ Miscellaneous terminals: All terminals other than PCI, slot, BUSON, parallel, and parallel/serial.

## PCI clock/reset timing requirements over recommended ranges of supply voltage and operating free-air temperature

| PARAMETER | | ALTERNATE SYMBOL | TEST CONDITIONS | MIN | MAX | UNITS |
|---|---|---|---|---|---|---|
| $t_c$ | PCLK cycle time | $t_{cyc}$ | | 15 | | ns |
| $t_{wH}$ | PCLK high time | $t_{high}$ | | 6 | | ns |
| $t_{wL}$ | PCLK low time | $t_{low}$ | | 6 | | ns |
| dv/dt | PCLK slew rate | $t_{r,\ tf}$ | | 1.5 | 4 | V/ns |
| $t_w$ | RSTIN pulse width | $t_{rst}$ | | 1 | | ms |
| $t_{su}$ | PCLK active time at end of RSTIN | $t_{rst–clk}$ | | 100 | | ms |

## Pin definitions for 120-pin PBM package

| PARAMETER | PIN NUMBER |
|---|---|
| PCI | 14, 16, 18, 19, 20, 21, 25, 27 |
| Slot | 1, 2, 3, 4, 5, 7, 8, 9, 10, 52, 53, 54, 55, 57, 58, 59, 60, 64, 65, 66, 67, 69, 70, 71, 72, 73, 75, 76, 77, 78, 80, 82, 83, 84, 86, 87, 88, 91, 92, 93, 94, 96, 97, 98, 99, 102, 103, 104, 105, 106, 108, 109, 110, 111, 113, 114, 115, 116, 118, 120 |
| BUSON | 63, 81, 100, 119 |
| Parallel | 34, 35, 37, 38, 39, 40, 42 |
| Parallel/serial | 23, 24, 29, 30, 43, 44, 45, 47, 48, 49, 50 |
| Miscellaneous | 12, 13 |

## Pin definitions for 128-pin PBK package

| PARAMETER | PIN NUMBER |
|---|---|
| PCI | 15, 17, 19, 20, 21, 22, 26, 28, |
| Slot | 2, 3, 4, 5, 6, 8, 9, 10, 11, 55, 56, 57, 58, 60, 61, 62, 63, 69, 70, 71, 72, 74, 75, 76, 77, 78, 80, 81, 82, 83, 85, 87, 88, 89, 91, 92, 93, 98, 99, 100, 101, 103, 104, 105, 106, 109, 110, 111, 112, 113, 115, 116, 117, 118, 120, 121, 122, 123, 125, 127 |
| BUSON | 68, 86, 107, 126 |
| Parallel | 37, 38, 40, 41, 42, 43, 45, |
| Parallel/serial | 24, 25, 30, 31, 46, 47, 48, 50, 51, 52, 53 |
| Miscellaneous | 13, 14 |

## Pin definitions for 144-pin PGE package

| PARAMETER | PIN NUMBER |
|---|---|
| PCI | 17, 19, 21, 22, 23, 24, 28, 30 |
| Slot | 1, 3, 5, 7, 8, 10, 11, 12, 13, 61, 62, 63, 64, 66, 68, 70, 72, 79, 80, 81, 82, 84, 85, 86, 87, 88, 90, 91, 92, 93, 95, 97, 98, 99, 101, 102, 104, 109, 111, 113, 115, 117, 118, 119, 120, 123, 124, 125, 126, 127, 129, 130, 131, 132, 134, 135, 136, 137, 140, 144 |
| BUSON | 77, 96, 121, 142 |
| Parallel | 43, 44, 46, 47, 48, 49, 51 |
| Parallel/serial | 26, 27, 34, 36, 52, 53, 54, 56, 57, 58, 59 |
| Miscellaneous | 15, 16 |

## PCI timing requirements over recommended ranges of supply voltage and operating free–air temperature† (see Note 3)

| PARAMETER | | | ALTERNATE SYMBOL | TEST CONDITIONS | MIN | MAX | UNITS |
|---|---|---|---|---|---|---|---|
| $t_{pd}$ | Propagation delay time | PCLK to shared signal valid delay time | $t_{val}$ | $C_L$ = 50 pF, See Note 4, 1,2,3 | 2 | 11 | ns |
| $t_{en}$ | Enable time, high-impedance-to-active delay time from PCLK | | $t_{on}$ | | 2 | | ns |
| $t_{dis}$ | Disable time, active-to-high-impedance delay time from PCLK | | $t_{off}$ | | | 28 | ns |
| $t_{su}$ | Valid setup time, before PCLK | | $t_{su}$ | 3,4 | 3 | | ns |
| $t_h$ | Hold time, after PCLK high | | $t_h$ | 4 | 0 | | ns |

† Applies to external output buffers.

NOTES: 3. This data sheet uses the following conventions to describe time ( t ) intervals. The format is: $t_A$, where *subscript A* indicates the type of dynamic parameter being represented. One of the following is used: $t_{pd}$ = propagation delay time, $t_d$ = delay time, $t_{su}$ = setup time, and $t_h$ = hold time.

4. PCI shared signals are AD31–AD0, C/BE3–C/BE0, PCIFRAME, PCITRDY, PCIIRDY, PCISTOP, IDSEL, PCIDEVSEL, and PCIPAR.

## serial bus interface†

| | | | STANDARD MODE | | FAST MODE | | UNIT |
|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | |
| $f_{SCL}$ | SCL clock frequency (see Note 5) | | 0 | 100 | 0 | 400 | kHz |
| $t_{BUF}$ | Bus free time between a STOP and START condition | | 4.7 | | 1.3 | | µs |
| $t_{HD;STA}$ | Hold time (repeated) START condition. After this period, the first clock pulse is generated. | | 4 | | 0.6 | | µs |
| $t_{LOW}$ | LOW period of the SCL clock | | 4.7 | | 1.3 | | µs |
| $t_{HIGH}$ | HIGH period of the SCL clock | | 4 | | 0.6 | | µs |
| $t_{SU;STA}$ | Setup time for a repeated START condition | | 4.7 | | 0.6 | | µs |
| $t_{HD;DAT}$ | Data hold time (see Note 6) | For CBUS compatible masters: | 5 | | | | µs |
| | | For serial bus devices: | $0_1$ | | $0_1$ | $0.9_2$ | |
| $t_{SU;DAT}$ | Data setup time (see Note 7) | | 250 | | $100_3$ | | µs |
| $t_R$ | Rise time of both SDA and SCL signals | | | 1000 | 20 | 300 | µs |
| $t_F$ | Fall time of both SDA and SCL signals | | | 300 | 20 | 300 | µs |
| $t_{FSU;STO}$ | Setup time for STOP condition | | 4 | | 0.6 | | µs |

† All values refer to serial bus interface $V_{IH\ MIN}$ and $V_{IL\ MAX}$ levels

NOTES: 5. A device must internally provide a hold time of at least 300 ns for the SDA signal (referred to the $V_{IH\ MIN}$ of the SCL signal) in order to bridge the undefined region of the falling edge of SCL.

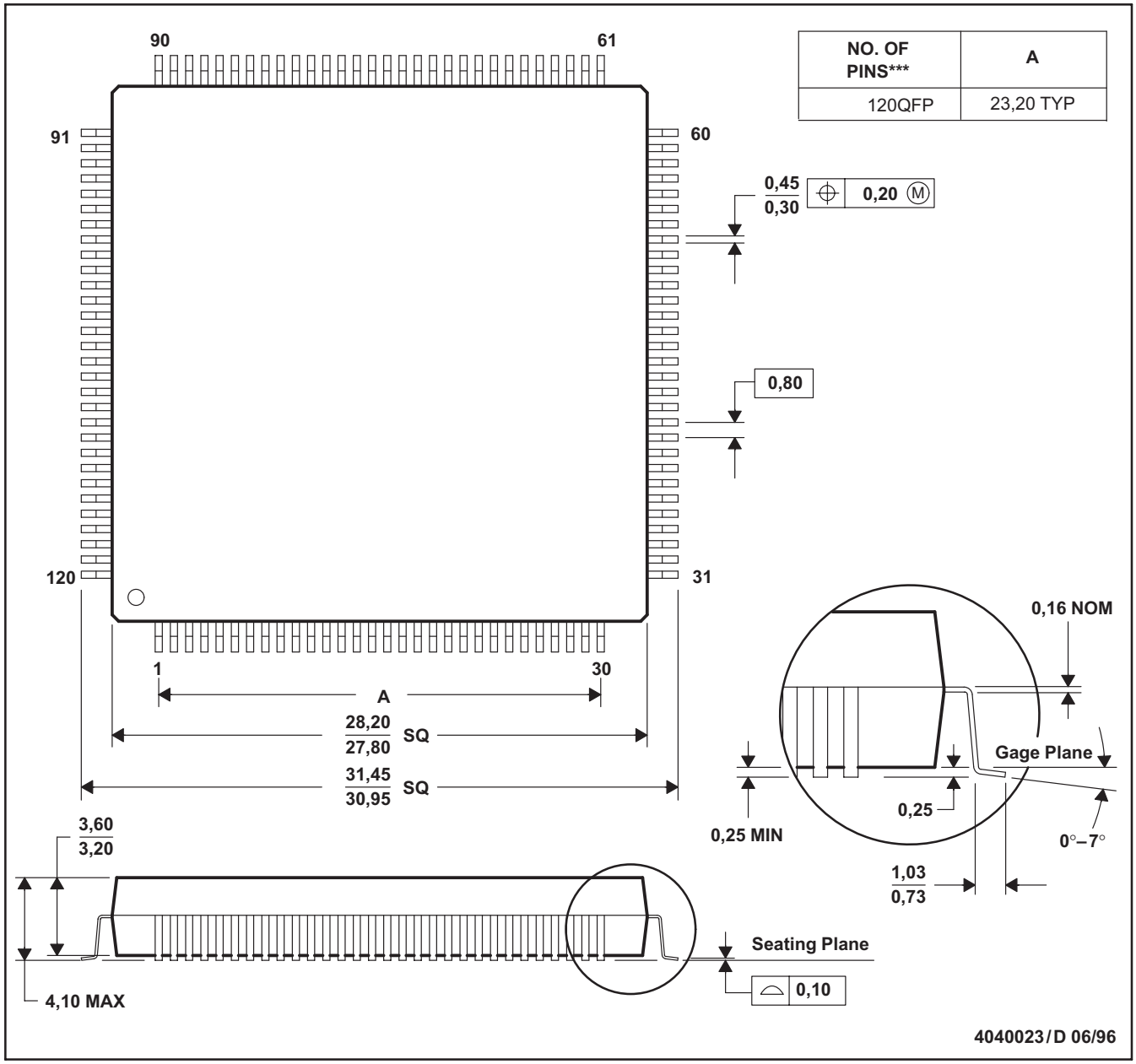
6. The maximum $t_{HD;DAT}$ has only to be met if the device does not stretch the LOW period ($t_{LOW}$ ) of the SDL signal.

7. A fast mode serial bus device can be used in a standard mode serial bus system, but the requirement $t_{SU;DAT} \geq 250$ ns must then be met. This will automatically be the case if the device does not stretch the LOW period of the SCL signal. If such a device does stretch the LOW period of the SCL signal, it must output the next data bit to the SDA line $t_{R\ MAX} + t_{SU;DAT} = 1000 + 250 = 1250$ ns (according to the *Standard Mode Serial Bus Specification*) before the SCL line is released.

## MECHANICAL DATA

**PBM (S-PQFP-G***)** **PLASTIC QUAD FLATPACK**

**120-PIN SHOWN**



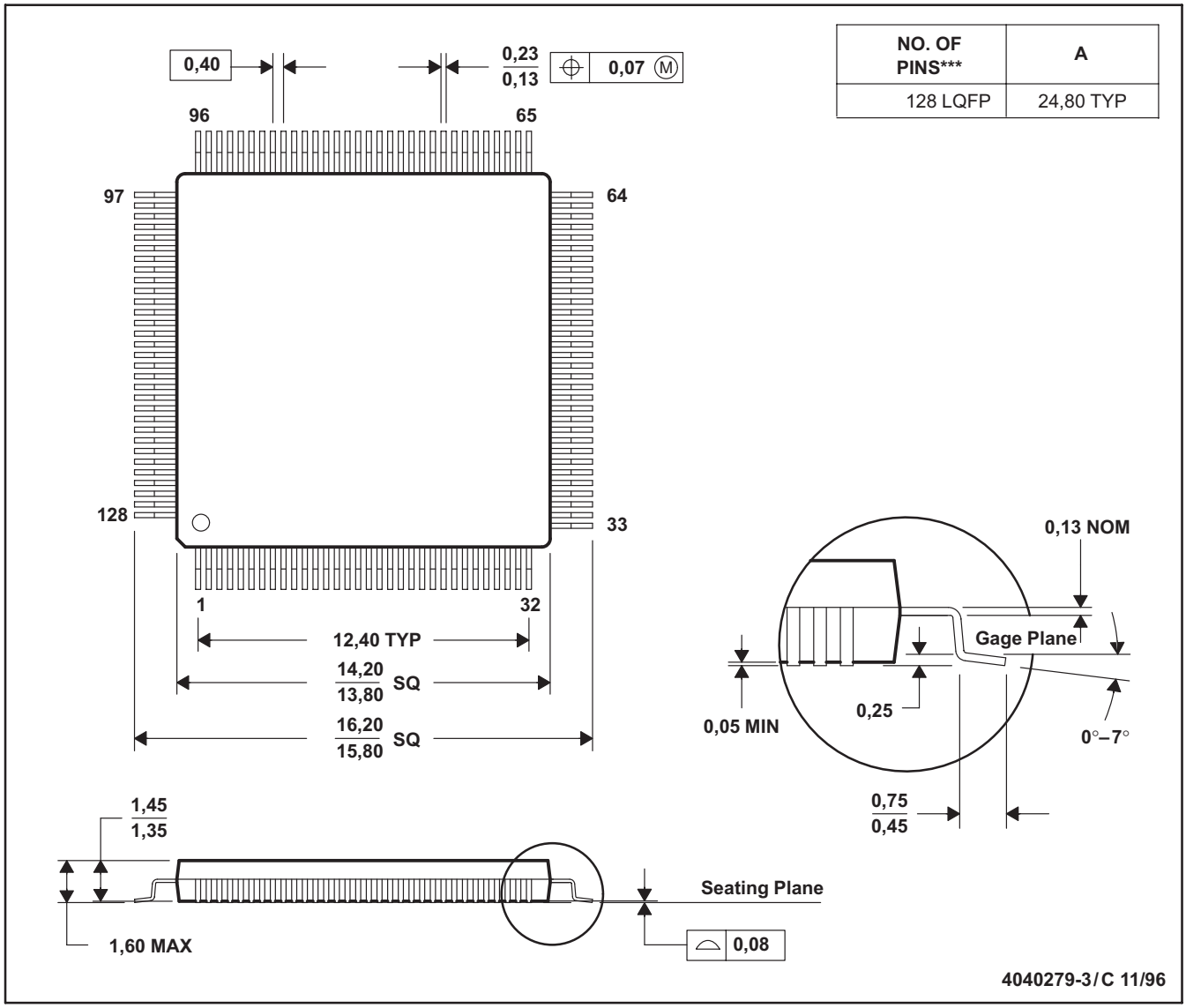

| NO. OF PINS*** | A |
|---|---|
| 120QFP | 23,20 TYP |

NOTES: A. All linear dimensions are in millimeters.
B. This drawing is subject to change without notice.
C. Falls within JEDEC MS-022

## MECHANICAL DATA

**PBK (S-PQFP-G128)** **PLASTIC QUAD FLATPACK**

**128-PIN SHOWN**



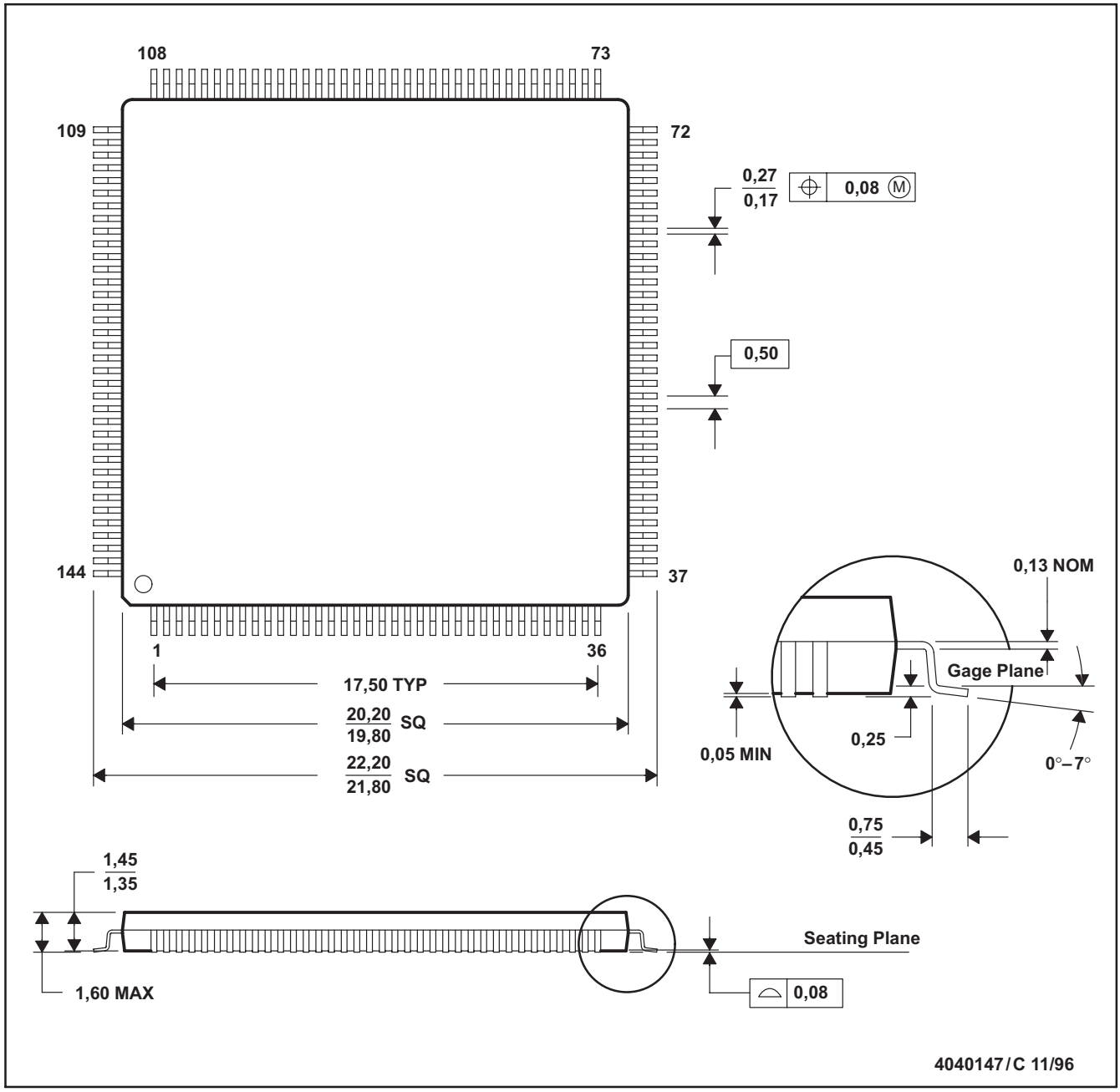

| NO. OF PINS*** | A |
|---|---|
| 128 LQFP | 24,80 TYP |

NOTES: A. All linear dimensions are in millimeters.
B. This drawing is subject to change without notice.
C. Falls within JEDEC MS-026

## MECHANICAL DATA

**PGE (S-PQFP-G144)** **PLASTIC QUAD FLATPACK**

**144-PIN SHOWN**




NOTES: A. All linear dimensions are in millimeters.
B. This drawing is subject to change without notice.
C. Falls within JEDEC MS-026

**IMPORTANT NOTICE**

Texas Instruments and its subsidiaries (TI) reserve the right to make changes to their products or to discontinue any product or service without notice, and advise customers to obtain the latest version of relevant information to verify, before placing orders, that information being relied on is current and complete. All products are sold subject to the terms and conditions of sale supplied at the time of order acknowledgement, including those pertaining to warranty, patent infringement, and limitation of liability.

TI warrants performance of its semiconductor products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are utilized to the extent TI deems necessary to support this warranty. Specific testing of all parameters of each device is not necessarily performed, except those mandated by government requirements.

CERTAIN APPLICATIONS USING SEMICONDUCTOR PRODUCTS MAY INVOLVE POTENTIAL RISKS OF DEATH, PERSONAL INJURY, OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE ("CRITICAL APPLICATIONS"). TI SEMICONDUCTOR PRODUCTS ARE NOT DESIGNED, AUTHORIZED, OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS. INCLUSION OF TI PRODUCTS IN SUCH APPLICATIONS IS UNDERSTOOD TO BE FULLY AT THE CUSTOMER'S RISK.

In order to minimize risks associated with the customer's applications, adequate design and operating safeguards must be provided by the customer to minimize inherent or procedural hazards.

TI assumes no liability for applications assistance or customer product design. TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of TI covering or relating to any combination, machine, or process in which such semiconductor products or services might be or are used. TI's publication of information regarding any third party's products or services does not constitute TI's approval, warranty or endorsement thereof.

## PACKAGING INFORMATION

| Orderable Device | Status (1) | Package Type | Package Drawing | Pins | Package Qty | Eco Plan (2) | Lead/Ball Finish | MSL Peak Temp (3) |
|---|---|---|---|---|---|---|---|---|
| HPC3130APBK | ACTIVE | LQFP | PBK | 128 | 90 | TBD | CU NIPDAU | Level-4-220C-72 HR |
| HPC3130APBM | ACTIVE | QFP | PBM | 120 | 24 | TBD | Call TI | Level-3-220C-168 HR |
| HPC3130APGE | ACTIVE | LQFP | PGE | 144 | 60 | Green (RoHS & no Sb/Br) | CU NIPDAU | Level-2-260C-1 YEAR |

(1) The marketing status values are defined as follows:
**ACTIVE:** Product device recommended for new designs.
**LIFEBUY:** TI has announced that the device will be discontinued, and a lifetime-buy period is in effect.
**NRND:** Not recommended for new designs. Device is in production to support existing customers, but TI does not recommend using this part in a new design.
**PREVIEW:** Device has been announced but is not in production. Samples may or may not be available.
**OBSOLETE:** TI has discontinued the production of the device.

(2) Eco Plan - The planned eco-friendly classification: Pb-Free (RoHS) or Green (RoHS & no Sb/Br) - please check http://www.ti.com/productcontent for the latest availability information and additional product content details.
**TBD:** The Pb-Free/Green conversion plan has not been defined.
**Pb-Free (RoHS):** TI's terms "Lead-Free" or "Pb-Free" mean semiconductor products that are compatible with the current RoHS requirements for all 6 substances, including the requirement that lead not exceed 0.1% by weight in homogeneous materials. Where designed to be soldered at high temperatures, TI Pb-Free products are suitable for use in specified lead-free processes.
**Green (RoHS & no Sb/Br):** TI defines "Green" to mean Pb-Free (RoHS compatible), and free of Bromine (Br) and Antimony (Sb) based flame retardants (Br or Sb do not exceed 0.1% by weight in homogeneous material)

(3) MSL, Peak Temp. -- The Moisture Sensitivity Level rating according to the JEDEC industry standard classifications, and peak solder temperature.

## IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

| **Products** | | **Applications** | |
|---|---|---|---|
| Amplifiers | amplifier.ti.com | Audio | www.ti.com/audio |
| Data Converters | dataconverter.ti.com | Automotive | www.ti.com/automotive |
| DSP | dsp.ti.com | Broadband | www.ti.com/broadband |
| Interface | interface.ti.com | Digital Control | www.ti.com/digitalcontrol |
| Logic | logic.ti.com | Military | www.ti.com/military |
| Power Mgmt | power.ti.com | Optical Networking | www.ti.com/opticalnetwork |
| Microcontrollers | microcontroller.ti.com | Security | www.ti.com/security |
| | | Telephony | www.ti.com/telephony |
| | | Video & Imaging | www.ti.com/video |
| | | Wireless | www.ti.com/wireless |

Mailing Address: Texas Instruments
Post Office Box 655303 Dallas, Texas 75265