# DATA SHEET

## PDIUSBH12
USB 2-port hub

Product specification
Supersedes data of 1998 May 13

1999 Feb 25

Philips
Semiconductors

PHILIPS

# USB 2-port hub PDIUSBH12

## FEATURES

- Complies with the Universal Serial Bus specification Rev. 1.0
- Complies with the ACPI, OnNOW, and USB power management requirements
- Compliant with USB Human Interface Devices and Monitor Control Class
- Compliant with System Management Bus Specification Rev. 1.0
- Two downstream ports with per packet connectivity and auto speed detection
- Supports up to 3 embedded functions
- Integrated SIE (Serial Interface Engine), FIFO memory and transceivers
- Automatic USB protocol handling
- High speed slave $I^2C$ Interface (up to 1 Mbit/s)
- Compatible with the PDIUSBH11 hardware and software
- Software controllable connection to USB bus (SoftConnect™)
- Good USB downstream connection indicators that blink with traffic (GoodLink™)
- Low frequency 12 MHz crystal oscillator eases EMI design issues
- Programmable output clock frequency
- Bus powered capability with very low suspend current
- Controllable LazyClock output at 30 kHz (nominal) during suspend
- Single 3.3V supply with 5V tolerant I/O
- Available in 28-pin DIP and SO packages
- Full-scan design with high fault coverage (>99%) insures high quality
- Higher than 8 KV in-circuit ESD protection lowers cost of extra components

## DESCRIPTION

The Universal Serial Bus Hub PDIUSBH12 is a cost and feature optimized second generation USB Hub with 2 downstream ports and 3 embedded functions (compound hub). It is normally used in any microcontroller-based system and communicates with the system microcontroller over the high speed $I^2C$ serial bus. This modular approach to implementing a hub and embedded functions allows the designer to choose the optimum system microcontroller from the available wide variety. This flexibility cuts down the development time, risks and costs by allowing the use of the existing architecture and the firmware investments. This results in the fastest way to develop the most cost-effective USB peripheral solutions that need hub functionality. The PDIUSBH12 is ideally suited for computer monitors, docking stations, keyboards and many other applications that use the $I^2C$ or the SMBus based architecture.

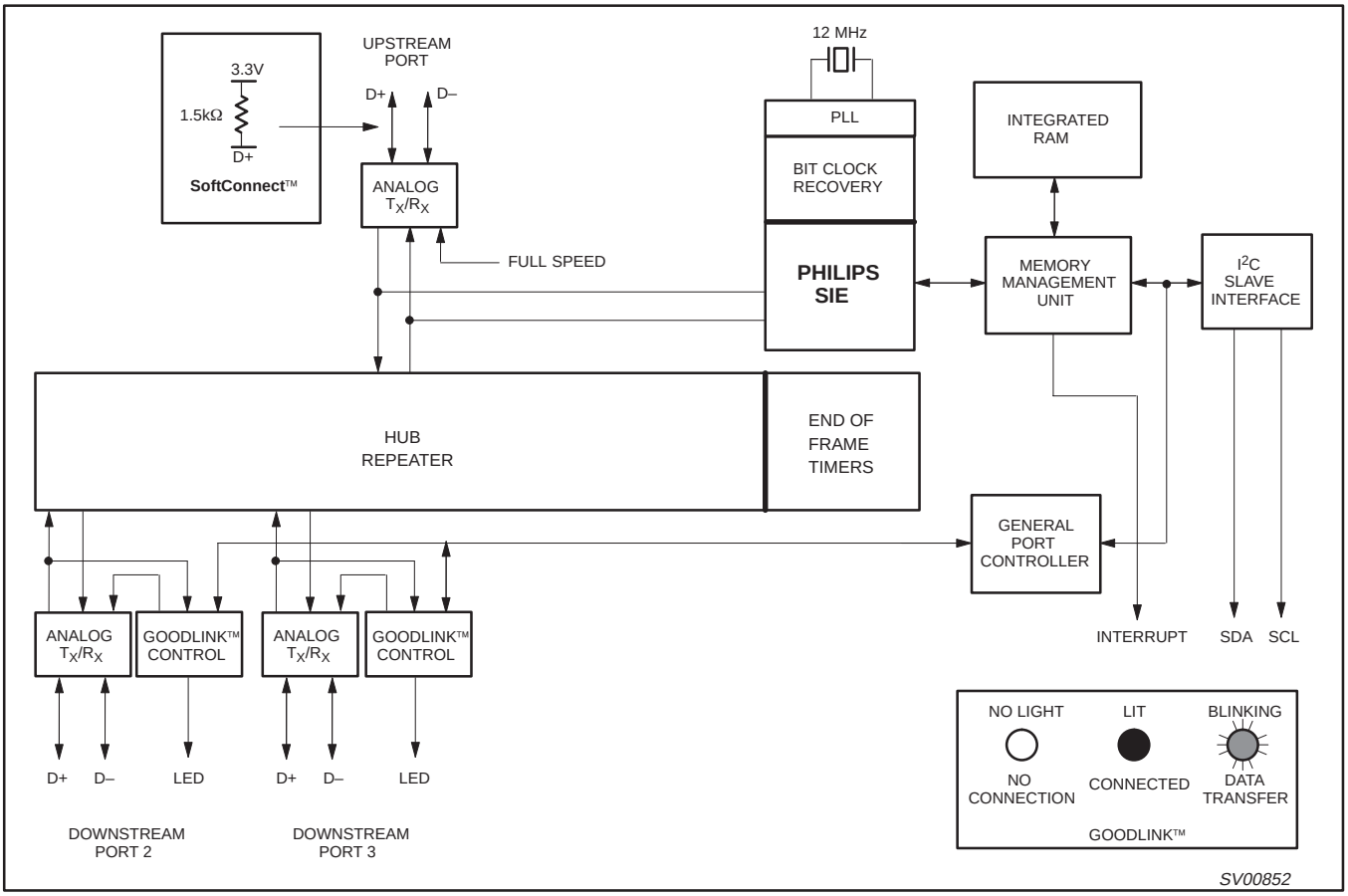
The PDIUSBH12 conforms to the USB specification Rev 1.0, $I^2C$ serial interface and the SMBus specifications. It is fully compliant with the Human Interface Device Class and Monitor Control Class specifications. Its low suspend power consumption along with the programmable LazyClock output allows for easy implementation of equipment that is compliant to the ACPI, OnNow and USB power management requirements. The low operating power allows the implementation of the bus powered or the compound powered hub function.

The PDIUSBH12 is fully backward compatible to the first generation PDIUSBH11 hardware and software. This allows an easy running change in the manufacturing line to realize the cost savings. In addition, it also incorporates the feature enhancements like SoftConnect™, GoodLink™, LazyClock, programmable clock output, lower frequency crystal oscillator, additional embedded functions and integration of termination resistors. All of these feature enhancements contribute to significant cost savings in the system implementation and at the same time ease the implementation of advanced USB functionality into the peripherals.

## ORDERING INFORMATION

| PACKAGES | TEMPERATURE RANGE | OUTSIDE NORTH AMERICA | NORTH AMERICA | PKG. DWG. # |
|---|---|---|---|---|
| 28-pin plastic SO | –40°C to +85°C | PDIUSBH12 D | PDIUSBH12 D | SOT136-1 |
| 28-pin plastic DIP | –40°C to +85°C | PDIUSBH12 N | PDIUSBH12 N | SOT117-1 |

## BLOCK DIAGRAM




**NOTE:**

1. This is a conceptual block diagram and does not include each individual signal.

### Analog Transceivers
These transceivers interface directly to the USB cables through some termination resistors. They are capable of transmitting and receiving serial data at both “full speed” (12 Mbit/s) and “low speed” (1.5 Mbit/s) data rates.

### Hub Repeater
The hub repeater is responsible for managing connectivity on a per packet basis. It implements packet signaling connectivity and resume connectivity.

Low speed devices can be connected to downstream ports since the repeater will not propagate upstream packets to downstream ports, to which low speed devices are connected, unless they are preceded by a PREAMBLE PID.

### End of Frame Timers
This block contains the specified EOF1 and EOF2 timers which are used to detect loss-of-activity and babble error conditions in the hub repeater. The timers also maintain the low-speed keep-alive strobe which is sent at the beginning of a frame.

### General and Individual Port Controller
The general and individual port controllers together provide status and control of individual downstream ports. Via the $I^2C$-interface a microcontroller can access the downstream ports and request or change the status of each individual port.

Any change in the status or settings of the individual port will result in an interrupt request. Via an interrupt register, the servicing microcontroller can look up the downstream port which generated the interrupt and request its new status. Any port status change can then be reported to the host via the hub status change (interrupt) endpoint.

### PLL
A 12 MHz to 48 MHz clock multiplier PLL (Phase-Locked Loop) is integrated on-chip. This allows for the use of low-cost 12 MHz crystal. EMI is also minimized due to lower frequency crystal. No external components are needed for the operation of the PLL.

### Bit Clock Recovery
The bit clock recovery circuit recovers the clock from the incoming USB data stream using 4X over-sampling principle. It is able to track jitter and frequency drift specified by the USB specification.

### Philips Serial Interface Engine (PSIE)
The Philips SIE implements the full USB protocol layer. It is completely hardwired for speed and needs no firmware intervention. The functions of this block include: synchronization pattern recognition, parallel/serial conversion, bit stuffing/de-stuffing, CRC checking/generation, PID verification/generation, address recognition, handshake evaluation/generation.

### Memory Management Unit (MMU) and Integrated RAM
The MMU and the integrated RAM is used to handle the large difference in data rate between USB, running in bursts of 12 Mbit/s and the $I^2C$ interface to the microcontroller, running at up to 1 Mbit/s. This allows the microcontroller to read and write USB packets at its own speed through $I^2C$.

### $I^2C$ Slave Interface
This block implements the necessary $I^2C$ interface protocol. A slave $I^2C$ allows for simple micro-coding. An interrupt is used to alert the microcontroller whenever the PDIUSBH12 needs attention. As a slave $I^2C$ device, the PDIUSBH12 $I^2C$ clock: SCL is an input and is controlled by the microcontroller. The $I^2C$ interface can run up to 1 Mbit/s.

### SoftConnect™
The connection to the USB is accomplished by bringing D+ (for high-speed USB device) high through a 1.5 kΩ pull-up resistor. In the PDIUSBH12, the 1.5 kΩ pull-up resistor is integrated on-chip and is not connected to $V_{CC}$ by default. Similarly, the 15 kΩ pull-down resistors are integrated on-chip and are not connected to GND by default. The connection of the internal resistors to Vcc is established through a command sent by the external/system microcontroller. This allows the system microcontroller to complete its initialization sequence before deciding to establish connection to the USB. Re-initialization of the USB bus connection can also be affected without requiring the pull out of the cable.

The PDIUSBH12 will check for USB VBUS availability before the connection can be established. VBUS sensing is provided through OCURRENT_N pin. See the pin description for details. Sharing of VBUS sensing and overcurrent sensing can be easily accomplished by using VBUS voltage as the pull-up voltage for the open drain output of the overcurrent indication device.

It should be noted that the tolerance of the internal resistors is higher (30%) than that specified by the USB specification (5%). However, the overall $V_{SE}$ voltage specification for the connection can still be met with good margin. The decision to make use of this feature lies with the users.

SoftConnect™ is a patent pending technology from Philips Semiconductors.

### GoodLink™
Good downstream USB connection indication is provided through GoodLink™ technology. When the port is enabled and there is at least one valid upstream traffic from the port, the LED indicator will be ON. The LED indicator will blink on every valid upstream traffic. A valid upstream traffic is defined as traffic with a good SOP and terminated by a good EOP. During global suspend, all LEDs will be OFF.

This feature provides a user-friendly indicator on the status of the hub, the connected downstream devices and the USB traffic. It is a useful field diagnostics tool to isolate the faulty equipment. This feature helps lower the field support and the hotline costs.

## ENDPOINT DESCRIPTIONS

There are two endpoint configuration modes supported by the PDIUSBH12, the Single Embedded Function mode and the Multiple (3) Embedded Function mode. The Single Embedded Function mode is the default at power up reset. The Multiple (3) Embedded Function mode can be configured by writing a zero to bit 7 of the first byte of the Set Mode command. Either mode is backward compatible to the PDIUSBH11.

### Table 1. SINGLE EMBEDDED FUNCTION MODE (DEFAULT AT POWER UP)

| FUNCTION | PORTS | ENDPOINT # | ENDPOINT INDEX | TRANSFER TYPE | DIRECTION | MAX PACKET SIZE (BYTES) |
|---|---|---|---|---|---|---|
| Hub | 0: Upstream<br>2–3: Downstream | 0 | 0<br>1 | Control | OUT<br>IN | 8<br>8 |
| | | 1 | – | Interrupt | IN | 1 |
| Embedded Function 1 | 1 | 0 | 2<br>3 | Control | OUT<br>IN | 8<br>8 |
| | | 1 | 5<br>4 | Generic | OUT<br>IN | 8<br>8 |
| | | 2 | 6<br>7 | Generic | OUT<br>IN | 8<br>8 |
| | | 3 | 8<br>9 | Generic | OUT<br>IN | 8<br>8 |

**NOTE:**
1. Hub interrupt endpoint is not indexed.
2. Generic endpoint can be used for Interrupt or Bulk endpoint.

### Table 2. MULTIPLE (3) EMBEDDED FUNCTION MODE

| FUNCTION | PORTS | ENDPOINT # | ENDPOINT INDEX | TRANSFER TYPE | DIRECTION | MAX PACKET SIZE (BYTES) |
|---|---|---|---|---|---|---|
| Hub | 0: Upstream<br>2–3: Downstream | 0 | 0<br>1 | Control | OUT<br>IN | 8<br>8 |
| | | 1 | – | Interrupt | IN | 1 |
| Embedded Function 1 | 1 | 0 | 2<br>3 | Control | OUT<br>IN | 8<br>8 |
| | | 1 | 5<br>4 | Generic | OUT<br>IN | 8<br>8 |
| Embedded Function 6 | 6 | 0 | 10<br>11 | Control | OUT<br>IN | 8<br>8 |
| | | 1 | 6<br>7 | Generic | OUT<br>IN | 8<br>8 |
| Embedded Function 7 | 7 | 0 | 12<br>13 | Control | OUT<br>IN | 8<br>8 |
| | | 1 | 8<br>9 | Generic | OUT<br>IN | 8<br>8 |

## PINNING

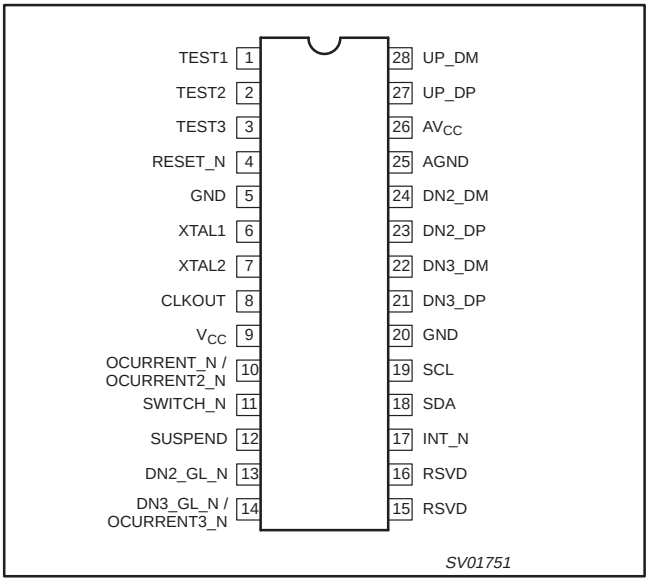
The PDIUSBH12 has two modes of operation. The first mode (Mode 0) configures the pins DNx_GL_N for GoodLink™ LED indication. The second mode (Mode 1) configures the LED pins as per port overcurrent condition pins. An overcurrent condition on any port can be uniquely identified in Mode 1. However, all downstream ports are disabled as a result of a single overcurrent condition. In addition to the two modes of operation, the PDIUSBH12 can also be configured to take either a 48 MHz crystal oscillator (for backward compatibility to PDIUSBH11) or a 12 MHz crystal.

The internal 4X clock multiplier PLL will be activated when 12 MHz input XTAL mode is selected. Also, the output clock frequency is now programmable rather than fixed to 12 MHz. The output clock frequency can be programmed through the Set Mode command. All these new features are added while maintaining backward compatibility to the PDIUSBH11 through TEST2 and TEST1 pins.

| TEST2 TEST1 | MODE | INPUT XTAL FREQUENCY (MHz) | OUTPUT CLOCK FREQUENCY (AT REST) |
|---|---|---|---|
| 00 | MODE 0 (GoodLink™) | 48 | 12MHz |
| 01 | MODE 0 (GoodLink™) | 12 | 4 MHz |
| 10 | MODE 1 (Individual Overcurrent) | 12 | 4 MHz |
| 11 | MODE 1 (Individual Overcurrent) | 48 | 12 MHz |

**NOTE:**
1. Pin TEST3 should always be connected to Ground at all times.

### Pin configuration




**NOTE:**
Pin 10 and Pin 14 show alternative pin functions, depending on mode of operation (Mode 0 or Mode 1) as described in *Pin Description*.

### Pin description (MODE 0 – Good Link™)

| PIN NO. | PIN SYMBOL | TYPE | DRIVE | DESCRIPTION |
|---|---|---|---|---|
| 1 | TEST1 | Input | | Connect to Ground for 48MHz crystal input.<br>Connect to $V_{CC}$ for 12MHz crystal input. |
| 2 | TEST2 | Input | | Connect to Ground |
| 3 | TEST3 | Input | | Connect to Ground |
| 4 | RESET_N | Input | ST | Power-on reset |
| 5 | GND | Power | | Ground reference |
| 6 | XTAL1 | Input | | Crystal connection 1 (48 or 12MHz depending on TEST1 pin) |
| 7 | XTAL2 | Output | | Crystal connection 2 (48 or 12MHz depending on TEST1 pin) |
| 8 | CLKOUT | Output | 3mA | Programmable output clock for external devices |
| 9 | $V_{CC}$ | Power | | Voltage supply 3.3V ± 0.3V |
| 10 | OCURRENT_N | Input | ST | Over-current notice to the device. This pin is also used to sense the USB VBUS. A LOW on this pin of less than 2 seconds is interpreted as an overcurrent notice; longer than 2 seconds is interpreted as loss of VBUS. |
| 11 | SWITCH_N | Output | OD6 | Enables power to downstream ports |
| 12 | SUSPEND | Output | OD6 | Device is in suspended state |
| 13 | DN2_GL_N | Output | OD6 | Downstream port 2 GoodLink™ LED indicator |
| 14 | DN3_GL_N | Output | OD6 | Downstream port 3 GoodLink™ LED indicator |
| 15 | RSVD | Input | | Reserved. Connect to GND for normal operation. |
| 16 | RSVD | Input | | Reserved. Connect to GND for normal operation. |
| 17 | INT_N | Output | OD6 | Connect to microcontroller interrupt |
| 18 | SDA | I/O | OD6 | $I^2C$ bi-directional data |
| 19 | SCL | I/O | OD6 | $I^2C$ bit-clock |
| 20 | GND | Power | | Ground reference |
| 21 | DN3_DP | AI/O | | Downstream port 3 $D^+$ connection |
| 22 | DN3_DM | AI/O | | Downstream port 3 $D^-$ connection |
| 23 | DN2_DP | AI/O | | Downstream port 2 $D^+$ connection |
| 24 | DN2_DM | AI/O | | Downstream port 2 $D^-$ connection |
| 25 | AGND | Power | | Analog Ground reference |
| 26 | $AV_{CC}$ | Power | | Analog voltage supply 3.3V ± 0.3V |
| 27 | UP_DP | AI/O | | Upstream $D^+$ connection |
| 28 | UP_DM | AI/O | | Upstream $D^-$ connection |

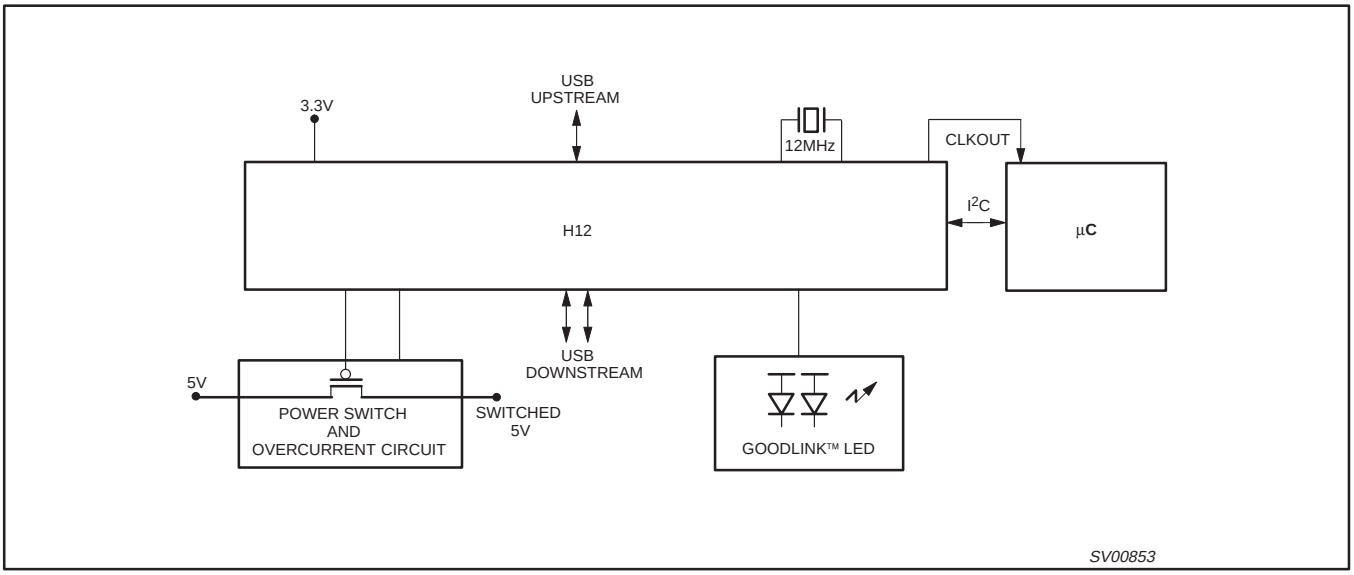
**NOTE:**
1. Signals ending in _N indicate active low signals.
   ST: Schmitt Trigger
   OD6: Open Drain with 6 mA drive
   AI/O: Analog I/O

### Pin description (MODE 1 – Individual Overcurrent)

| PIN NO | PIN SYMBOL | TYPE | DRIVE | DESCRIPTION |
|---|---|---|---|---|
| 1 | TEST1 | Input | | Connect to $V_{CC}$ for 48MHz crystal input.<br>Connect to Ground for 12MHz crystal input. |
| 2 | TEST2 | Input | | Connect to $V_{CC}$ |
| 3 | TEST3 | Input | | Connect to Ground |
| 4 | RESET_N | Input | ST | Power-on reset |
| 5 | GND | Power | | Ground reference |
| 6 | XTAL1 | Input | | Crystal connection 1 (48 or 12MHz depending on TEST1 pin) |
| 7 | XTAL2 | Output | | Crystal connection 2 (48 or 12MHz depending on TEST1 pin) |
| 8 | CLKOUT | Output | 3mA | Programmable output clock for external devices |
| 9 | $V_{CC}$ | Power | | Voltage supply 3.3V ± 0.3V |
| 10 | OCURRENT2_N | Input | ST | Downstream port 2 over-current notice. This pin is also use to sense the USB VBUS. A LOW on this pin of less than 2 seconds is interpreted as an overcurrent notice; longer than 2 seconds is interpreted as loss of VBUS. |
| 11 | SWITCH_N | Output | OD6 | Enables power to downstream ports |
| 12 | SUSPEND | Output | OD6 | Device is in suspended state |
| 13 | DN2_GL_N | Output | OD6 | Downstream port 2 GoodLink™ LED indicator |
| 14 | OCURRENT3_N | Input | ST | Downstream port 3 over-current notice |
| 15 | RSVD | Input | | Reserved. Connect to GND for normal operation. |
| 16 | RSVD | Input | | Reserved. Connect to GND for normal operation. |
| 17 | INT_N | Output | OD6 | Connect to microcontroller interrupt |
| 18 | SDA | I/O | OD6 | $I^2C$ bi-directional data |
| 19 | SCL | I/O | OD6 | $I^2C$ bit-clock |
| 20 | GND | Power | | Ground reference |
| 21 | DN3_DP | AI/O | | Downstream port 3 $D^+$ connection |
| 22 | DN3_DM | AI/O | | Downstream port 3 $D^-$ connection |
| 23 | DN2_DP | AI/O | | Downstream port 2 $D^+$ connection |
| 24 | DN2_DM | AI/O | | Downstream port 2 $D^-$ connection |
| 25 | AGND | Power | | Analog Ground reference |
| 26 | $AV_{CC}$ | Power | | Analog voltage supply 3.3V ± 0.3V |
| 27 | UP_DP | AI/O | | Upstream $D^+$ connection |
| 28 | UP_DM | AI/O | | Upstream $D^-$ connection |

**NOTE:**
1. Signals ending in _N indicate active low signals.
   ST: Schmitt Trigger
   OD6: Open Drain with 6 mA drive
   AI/O: Analog I/O

## APPLICATION DIAGRAM




## I²C Interface

The I²C bus is used to interface to an external microcontroller needed to control the operation of the hub. For cost consideration, the target system microcontroller can be shared and utilized for this purpose. The PDIUSBH12 implements a slave I²C interface. When the PDIUSBH12 needs to communicate with the microcontroller it asserts an interrupt signal. The microcontroller services this interrupt by reading the appropriate status register on the PDIUSBH12 through the I²C bus. (For more information about the I²C serial bus, refer to the *I²C Handbook*, Philips order number 9397 750 00013).

The I²C interface on the PDIUSBH12 defines two types of transactions:

1. **command transaction**
   A command transaction is used to define which data (e.g., status byte, buffer data, ...) will be read from / written to the USB interface in the next data transaction. A data transaction usually follows a command transaction.
2. **data transaction**
   A data transaction reads data from / writes data to the USB interface. The meaning of the data is dependent on the command transaction which was sent before the data transaction.

Two addresses are used to differentiate between command and data transactions. Writing to the command address is interpreted as a command, while reading from / writing to the data address is used to transfer data between the PDIUSBH12 and the controller.

## ADDRESS TABLE

| TYPE OF ADDRESS | PHYSICAL ADDRESS (MSB to LSB) |
|---|---|
| Command | 0011 011 (binary) |
| Data | 0011 010 (binary) |

## Protocol

An I²C transaction starts with a ‘Start Condition’, followed by an address. When the address matches either the command or data address the transaction starts and runs until a ‘Stop Condition’ or another ‘Start Condition’ (repeated start) occurs.

The command address is write-only and is unable to do a read. The next bytes in the message are interpreted as commands. Several command bytes can be sent after one command address. Each of the command bytes is acknowledged and passed on to the Memory Management Unit inside the PDIUSBH12.

When the start condition address matches the data address, the next bytes are interpreted as data. When the RW bit in the address indicates a ‘master writes data to slave’ (=‘0’) the bytes are received, acknowledged and passed on to the Memory Management Unit. If the RW bit in the address indicates a ‘master reads data from slave’ (=‘1’) the PDIUSBH12 will send data to the master. The I²C-master must acknowledge all data bytes except the last one. In this way the I²C interface knows when the last byte has been transmitted and it then releases the SDA line so that the master controller can generate the STOP condition.

Repeated start support allows another packet to be sent without generating a Stop Condition.

### Timing

The I²C interface in the PDIUSBH12 can support clock speeds up to 1MHz.

## COMMAND SUMMARY

Some commands have the same command code (e.g., Read Buffer and Write Buffer). In these cases, the direction of the Data Phase (read or write) indicates which command is executed.

| COMMAND NAME | RECIPIENT | CODING | DATA PHASE |
|---|---|---|---|
| **Initialization Commands** | | | |
| Set Address / Enable | Hub | D0h | Write 1 byte |
| | Embedded Function 1 | D1h | Write 1 byte |
| | Embedded Function 6 | D2h | Write 1 byte |
| | Embedded Function 7 | D3h | Write 1 byte |
| Set Endpoint Enable | Hub + Embedded Functions | D8h | Write 1 byte |
| Set Mode | Hub + Embedded Functions | F3h | Write 2 bytes |
| **Data Flow Commands** | | | |
| Read Interrupt Register | | F4h | Read 2 bytes |
| Select Endpoint | Hub Control OUT | 00h | Read 1 byte (optional) |
| | Hub Control IN | 01h | Read 1 byte (optional) |
| | Other Endpoints | 00h + Endpoint Index | Read 1 byte (optional) |
| Read Last Transaction Status | Hub Control OUT | 40h | Read 1 byte |
| | Hub Control IN | 41h | Read 1 byte |
| | Other Endpoints | 40h + Endpoint Index | Read 1 byte |
| Read Endpoint Status | Hub Control OUT | 80h | Read 1 byte |
| | Hub Control IN | 81h | Read 1 byte |
| | Other Endpoints | 80h + Endpoint Index | Read 1 byte |
| Read Buffer | Selected Endpoint | F0h | Read n bytes |
| Write Buffer | Selected Endpoint | F0h | Write n bytes |
| Set Endpoint Status | Hub Control OUT | 40h | Write 1 byte |
| | Hub Control IN | 41h | Write 1 byte |
| | Other Endpoints | 40h + Endpoint Index | Write 1 byte |
| Acknowledge Setup | Selected Endpoint | F1h | None |
| Clear Buffer | Selected Endpoint | F2h | None |
| Validate Buffer | Selected Endpoint | FAh | None |
| **Hub Commands** | | | |
| Clear Port Feature | Port 2 | E0h | Write 1 byte |
| | Port 3 | E1h | Write 1 byte |
| Set Port Feature | Port 2 | E8h | Write 1 byte |
| | Port 3 | E9h | Write 1 byte |
| Get Port Status | Port 2 | E0h | Read 1 or 2 bytes |
| | Port 3 | E1h | Read 1 or 2 bytes |
| Set Status Change Bits | | F7h | Write 1 byte |
| **General Commands** | | | |
| Send Resume | | F6h | None |
| Read Current Frame Number | | F5h | Read 1 or 2 bytes |

## COMMAND DESCRIPTIONS

### Command Procedure

There are four basic types of commands: Initialization, Data, Hub Specific, and General commands. Respectively, these are used to initialize the hub and embedded function; for data flow between the hub, embedded function, and the host; some hub specific commands for controlling individual downstream ports; and some general commands.

### Initialization Commands

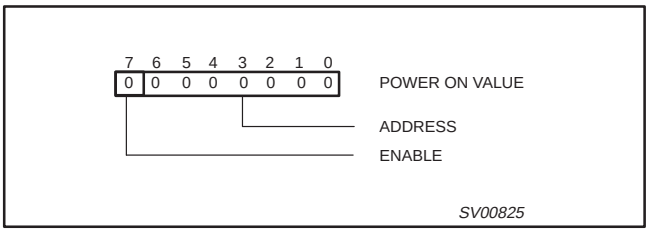
Initialization commands are used during the enumeration process of the USB network. These commands are used to enable the hub and embedded function endpoints. They are also used to set the USB assigned address.

#### Set Address / Enable

**Command : D0h (Hub), D1h, D2h, D3h, (Embedded Functions)**

**Data : Write 1 byte**

This command is used to set the USB assigned address and enable the hub or embedded functions respectively. The hub powers up enabled and needs not be enabled by the firmware at power up initialization.



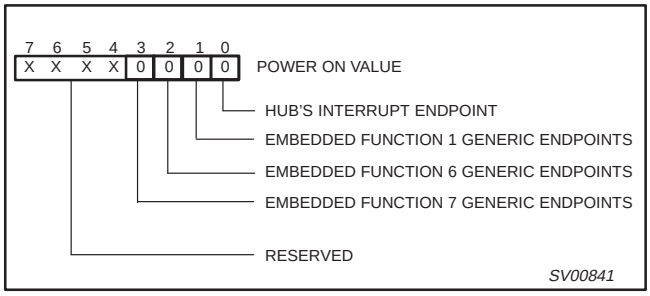

| | |
|---|---|
| Address | The value written becomes the address. |
| Enable | A ‘1’ enables this function. |

#### Set Endpoint Enable

**Command : D8h**

**Data : Write 1 byte**

The hub’s interrupt endpoint and the embedded functions generic endpoints can only be enabled when the corresponding hub/function is enabled via the Set Address/Enable command.




| | |
|---|---|
| Hub’s Interrupt Endpoint | A value of ‘1’ indicates the hub’s interrupt endpoint is enabled. |
| Embedded Function 1 Generic Endpoint | A value of ‘1’ indicates the embedded function 1 generic endpoints are enabled. |
| Embedded Function 6 Generic Endpoint | A value of ‘1’ indicates the embedded function 6 generic endpoints are enabled. |
| Embedded Function 7 Generic Endpoint | A value of ‘1’ indicates the embedded function 7 generic endpoints are enabled. |

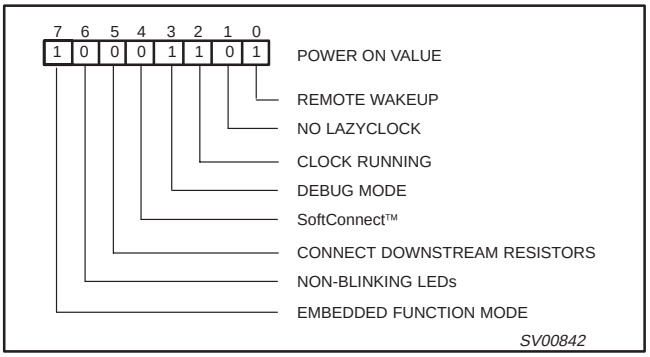
#### Set Mode

**Command : F3h**

**Data : Write 2 bytes**

The Set Mode command is followed by two data writes. The first byte contains the configuration byte values. The second byte is the clock division factor byte.

## Configuration Byte




Remote Wakeup — A ‘1’ indicates that a remote wakeup feature is ON. Bus reset will set this bit to ‘1’.

No LazyClock — A ‘1’ indicates that CLKOUT will not switch to LazyClock. A ‘0’ indicates that the CLKOUT switches to LazyClock 1ms after the Suspend pin goes high. LazyClock frequency is 30KHz (± 40%). The programmed value will not be changed by a bus reset.

Clock Running — A ‘1’ indicates that the internal clocks and PLL are always running even during Suspend state. A ‘0’ indicates that the internal clock, crystal oscillator and PLL are stopped whenever not needed. To meet the strict Suspend current requirement, this bit needs to be set to ‘0’. The programmed value will not be changed by a bus reset.

Debug Mode — A ‘1’ indicates that all errors and “NAKing” are reported and a ‘0’ indicates that only OK and babbling are reported. The programmed value will not be changed by a bus reset.

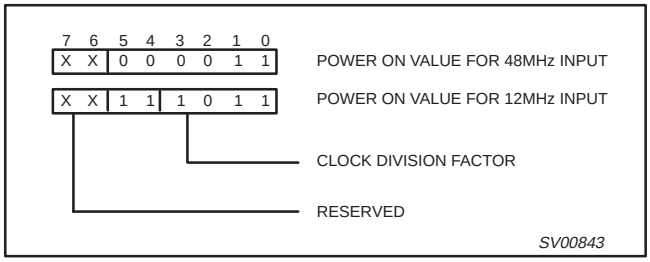
SoftConnect™ — A ‘1’ indicates that the upstream pull-up resistor will be connected if VBUS is available. A ‘0’ means that the upstream resistor will not be connected. The programmed value will not be changed by a bus reset.

Connect Downstream Resistors — A ‘1’ indicates that downstream resistors are connected. A ‘0’ means that downstream resistors are not connected. The programmed value will not be changed by a bus reset.

Non-blinking LEDs — A ‘1’ indicates that GoodLink™ LEDs will NOT blink when there is traffic. Leave this bit at ‘0’ to achieve blinking LEDs. The programmed value will not be changed by a bus reset.

Embedded Function Mode — A ‘1’ indicates single embedded function mode. A ‘0’ indicates multiple (3) embedded function mode. See *endpoint descriptions* for details. The programmed value will not be changed by a bus reset.

## Clock Division Factor Byte




Clock Division Factor — The value indicates clock division factor for CLKOUT. The output frequency is 48 MHz/(N+1) where N is the Clock Division Factor. The reset value is 3. This will give a default output frequency at CLKOUT pin of 12 MHz, thus maintaining backward compatibility to the PDIUSBH11. When the 12 MHz input crystal frequency is selected, the reset value is 11. This will produce the lowest output frequency of 4 MHz which can then be programmed up by the user. The PDIUSBH12 design ensures no glitching during frequency change. The programmed value will not be changed by a bus reset.

## Data Flow Commands

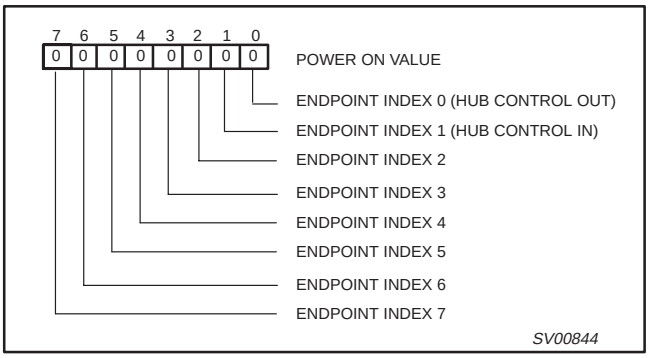
Data flow commands are used to manage the data transmission between the USB endpoints and the monitor. Much of the data flow is initiated via an interrupt to the microcontroller. The microcontroller utilizes these commands to access and determine whether the endpoint FIFOs have valid data.

### Read Interrupt Register

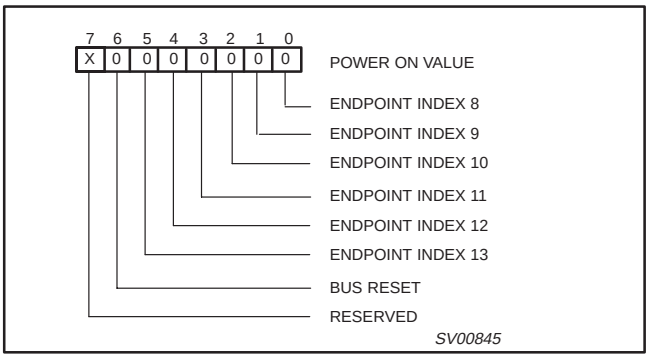
**Command : F4h**

**Data : Read 2 bytes**

**Interrupt Register Byte 1**




**Interrupt Register Byte 2**




This command indicates the origin of an interrupt. A ‘1’ indicates an interrupt occurred at this endpoint. The bits are cleared by reading the endpoint status register through Read Endpoint Status command.

After a bus reset an interrupt will be generated and bit 6 of the Interrupt Register Byte 2 will be ‘1’. [In the PDIUSBH11, the bus reset event is indicated by the absence of a ‘1’ in any bit of the Interrupt Register. Note that the backward compatibility is still maintained because in the PDIUSBH11, the Interrupt Register Byte 2 does not exist.]

The bus reset interrupt is internally cleared by reading the interrupt register. A bus reset is completely identical to the hardware reset through the RESET_N pin with the sole difference of interrupt notification.

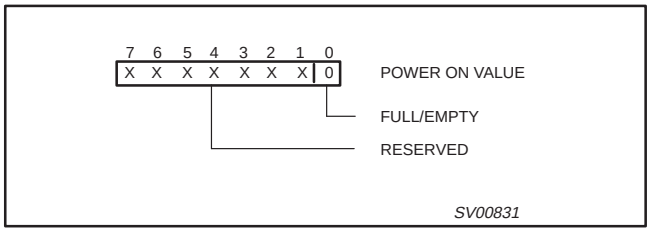
The hub interrupt endpoint is handled internally by the PDIUSBH12 hardware without the need of microcontroller intervention.

### Select Endpoint

**Command : 00-0Dh**

**Data : Optional Read 1 byte**

The Select Endpoint command initializes an internal pointer to the start of the Selected buffer. Optionally, this command can be followed by a data read, which returns ‘0’ if the buffer is empty and ‘1’ if the buffer is full.




| | |
|---|---|
| Full/Empty | A ‘1’ indicates the buffer is full, ‘0’ indicates an empty buffer. |

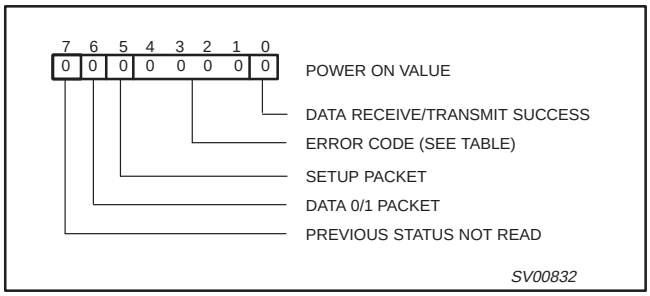
### Read Last Transaction Status

**Command : 40–4Dh**

**Data : Read 1 byte**

The Read Last Transaction Status command is followed by one data read that returns the status of the last transaction of the endpoint. This command also resets the corresponding interrupt flag in the interrupt register, and clears the status, indicating that it was read.

This command is useful for debugging purposes. Since it keeps track of every transaction, the status information is overwritten for each new transaction.




| | |
|---|---|
| Data Receive/Transmit Success | A ‘1’ indicates data has been received or transmitted successfully. |
| Error Code | See Table 3, Error Codes. |
| Setup Packet | A ‘1’ indicates the last successful received packet had a SETUP token (this will always read ‘0’ for IN buffers. |
| Data 0/1 Packet | A ‘1’ indicates the last successful received or sent packet had a DATA1 PID. |
| Previous Status not Read | A ‘1’ indicates a second event occurred before the previous status was read. |

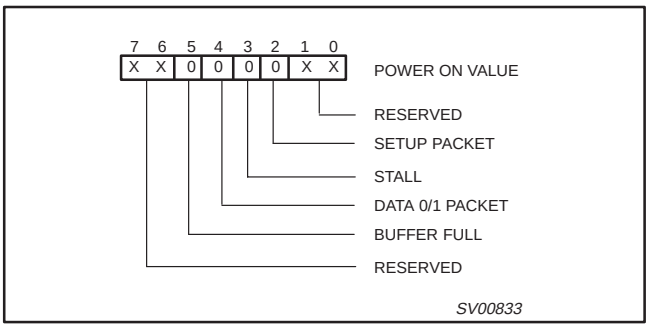
**Table 3. Error codes**

| ERROR CODE | RESULT |
|---|---|
| 0000 | No Error |
| 0001 | PID encoding Error; bits 7–4 are not the inversion of bits 3–0 |
| 0010 | PID unknown; encoding is valid, but PID does not exist |
| 0011 | Unexpected packet; packet is not of the type expected (= token, data or acknowledge), or SETUP token to a non-control endpoint |
| 0100 | Token CRC Error |
| 0101 | Data CRC Error |
| 0110 | Time Out Error |
| 0111 | Babble Error |
| 1000 | Unexpected End-of-packet |
| 1001 | Sent or received NAK |
| 1010 | Sent Stall, a token was received, but the endpoint was stalled |
| 1011 | Overflow Error, the received packet was longer than the available buffer space |
| 1101 | Bitstuff Error |
| 1111 | Wrong DATA PID; the received DATA PID was not the expected one |

## Read Endpoint Status

**Command : 80–8Dh**

**Data : Read 1 byte**




| | |
|---|---|
| Setup Packet | A ‘1’ indicates the last received packet had a SETUP token. |
| STALL | A ‘1’ indicates the endpoint is stalled. |
| Data 0/1 Packet | A ‘1’ indicates if the last received or sent packet had a DATA1 PID. |
| Buffer Full | A ‘1’ indicates that the buffer is full. |

## Read Buffer

**Command : F0h**

**Data : Read multiple bytes (max 10)**

The Read Buffer command is followed by a number of data reads, which return the contents of the selected endpoint data buffer. After each read, the internal buffer pointer is incremented by 1.

The buffer pointer is not reset to the buffer start by the Read Buffer command. This means that reading or writing a buffer can be interrupted by any other command (except for Select Endpoint), or can be done by more than one $I^2C$ transaction (read the first 2 bytes to get the number of data bytes, then read the rest in other transactions).

The data in the buffer are organized as follows:

- byte 0: Reserved: can have any value
- byte 1: Number/length of data bytes
- byte 2: Data byte 1
- byte 3: Data byte 2

## Write Buffer

**Command : F0h**

**Data : Write multiple bytes (max 10)**

The Write Buffer command is followed by a number of data writes, which load the endpoints buffer. The data must be organized in the same way as described in the Read Buffer command. The first byte (reserved) should always be 0. As in the Read Buffer command, the data can be split up into different $I^2C$ data transactions.

**WARNING:**
There is no protection against writing or reading over a buffer’s boundary or against writing into an OUT buffer or reading from an IN buffer. Any of these actions could cause an incorrect operation. Data in an OUT buffer are only meaningful after a successful transaction.

## Clear Buffer

**Command : F2h**

**Data : None**

When a packet is received completely, an internal endpoint buffer full flag is set. All subsequent packets will be refused by returning a NAK. When the microcontroller has read the data, it should free the buffer by the Clear Buffer command. When the buffer is cleared new packets will be accepted.

## Validate Buffer

**Command : FAh**

**Data : None**

When the microprocessor has written data into an IN buffer, it should set the buffer full flag by the Validate Buffer command. This indicates that the data in the buffer are valid and can be sent to the host when the next IN token is received.

### Set Endpoint Status

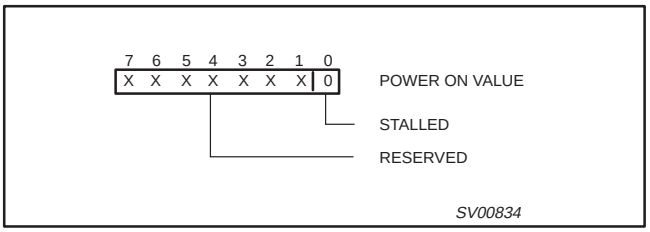
**Command : 40–4Dh**

**Data : Write 1 byte**

A stalled control endpoint is automatically unstalled when it receives a SETUP token, regardless of the content of the packet. If the endpoint should stay in its stalled state, the microcontroller can re-stall it.

When a stalled endpoint is unstalled (either by the Set Endpoint Status command or by receiving a SETUP token), it is also re-initialized. This flushes the buffer and if it is an OUT buffer it waits for a DATA 0 PID, if it is an IN buffer it writes a DATA 0 PID.

Even when unstalled, writing Set Endpoint Status to '0' initializes the endpoint.




Stalled A '1' indicates the endpoint is stalled.

### Acknowledge Setup

**Command : F1h**

**Data : None**

The arrival of a SETUP packet flushes the IN buffer and disables the Validate Buffer and Clear Buffer commands for both IN and OUT endpoints.

The microcontroller needs to re-enable these commands by the Acknowledge Setup command. This ensures that the last SETUP packet stays in the buffer and no packet can be sent back to the host until the microcontroller has acknowledged explicitly that it has seen the SETUP packet.

The microcontroller must send the Acknowledge Setup command to both the IN and OUT endpoints.

## Hub Commands

Hub commands are used to report connectivity and power status between the hub and the host. These commands allow the host to enable each port individually and get any change of status such as new connectivity information.

### Clear/Set Port Feature

**Command : E0–E1h (Clear) and E8h–E9h (Set)**

**Data : Write 1 byte**

The data written in the data phase is the feature code described in Table 4.

When the controller receives a Set Feature or a Clear Feature request, there are two possibilities:

The request applies to port 1, the embedded port. In this case the request should be handled internally by the controller.

If the request applies to ports 2 and 3, the controller should translate the request into a Set Feature or Clear Feature command towards the PDIUSBH12.

When the PDIUSBH12 is configured in mode 0, there is only one power switch output and one overcurrent input. This means that the F_PORT_POWER and C_PORT_OVERCURRENT features are not port specific. For these features, any of the Set / Clear Feature commands can be used. The specific port assignment is ignored.

When the PDIUSBH12 is configured in mode 1, there is still only one power switch output but there are two individual overcurrent input pins corresponding to each port. This means that the F_PORT_POWER feature is port specific and the C_PORT_OVERCURRENT feature is not port specific.

Setting the F_PORT_POWER feature turns the power on when it is off and turns the overcurrent detection on only when the power is already on. This allows it to have a short period of overcurrent condition at the moment that power is switched on. For this reason, the F_PORT_POWER feature needs to be set twice. Clearing this feature turns both the power and the overcurrent detection off.

**Table 4.**

| FEATURE | FEATURE CODE | SET | CLEAR |
|---|---|---|---|
| F_PORT_ENABLE | 0 | Enables a port | Disables a port |
| F_PORT_SUSPEND | 1 | Suspends a port | Resumes a port |
| FC_PORT_RESET | 2 | Resets a port | Clears a port Reset Change bit |
| F_PORT_POWER | 3 | Powers all ports | Unpowers all ports |
| C_PORT_CONNECTION | 4 | – | Clears a port Connection Change bit |
| C_PORT_ENABLE | 5 | – | Clears a port Enable Change bit |
| C_PORT_SUSPEND | 6 | – | Clears a port Suspend Change bit |
| C_PORT_OVERCURRENT | 7 | – | Clears a port (Mode 1) or hub (Mode 0) Overcurrent Change bit |

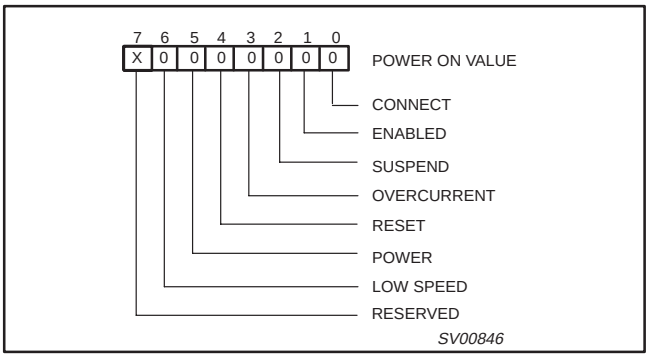
## Get Port Status

**Command : E0h–E1h**

**Data : Read 1 or 2 bytes**

The Get Port Status Command can be followed by one or two data reads. The first byte returned contains the port status. The second byte returned is the port status change byte.

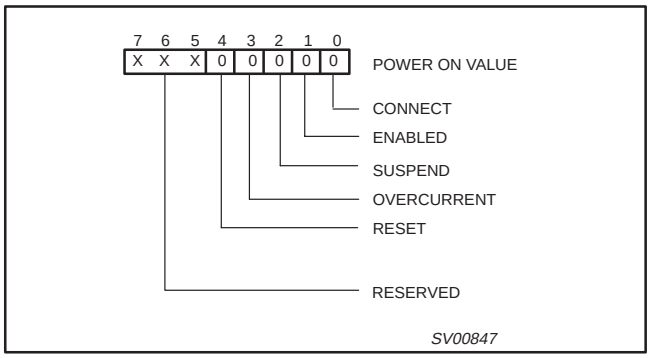
**Port Status Byte**




| | |
|---|---|
| Connect | A ‘1’ indicates that a device is connected on this port of the hub. |
| Enabled | A ‘1’ indicates that this port is enabled. |
| Suspend | A ‘1’ indicates that this port is suspended. |
| OverCurrent | A ‘1’ indicates that overcurrent condition exists on this port. In mode 0 of operation, this bit is the same for all ports. In mode 1, individual port overcurrent indication is possible. |
| Reset | A ‘1’ indicates that bus reset on this port is in progress. When reset is completed (nominal duration of 10 ms), this bit indicates a ‘0’. |
| Power | A ‘1’ indicates that power is supplied to downstream ports. Since the PDIUSBH12 supports gang mode power switching, this bit is the same for all ports. |
| Low Speed | A ‘1’ indicates that low speed device is connected to this port. This bit is only valid when Connect bit is a ‘1’. |

**Port Status Change Byte**

The description for the Port Status Change Byte is similar to the Port Status Byte except that the value of the bits are ‘1’ only when a change has occurred.



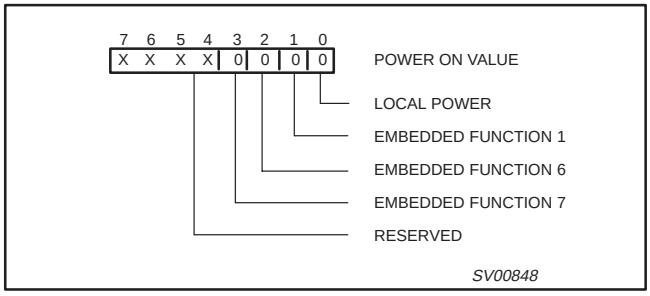

## Set Status Change Bits

**Command : F7h**

**Data : Write 1 byte**

For assembling the hub’s status change register, the device needs some additional information from the controller, i.e. the Local Power Status Change bit and the embedded function Status Change bit.

These are provided by the Set Status Change Bits command. This command is always followed by one data write which contains the Local Power Status Change bit at the LSB and the embedded function Status Change bit at position 1. All other bits should be 0.



## General Commands

### Send Resume

**Command : F6h**

**Data : None**

Sends an upstream resume signal for 10 ms. This command is normally issued when the device is in suspend. The RESUME command is not followed by a data read or write.

The PDIUSBH12 automatically sends a RESUME command when an event occurs downstream.

### Read Current Frame Number

**Command : F5h**

**Data : Read One or Two Bytes**

This command is followed by one or two data reads and returns the frame number of the last successfully received SOF. The frame number is returned Least Significant Byte first.


## Embedded Function

The USB host sees no difference between the embedded function and a function connected to one of the downstream ports. Some of the port commands sent by the host must be handled appropriately by the embedded function to appear as any other downstream port.

The microcontroller maintains a series of status and status change bits for the embedded function as described in the Get Port Status command section. From these bits, the Status Change bit for the embedded function is derived (i.e. the port specific Status Change bits). This Status Change bit is then provided to the PDIUSBH12 by the Set Status Change Bits command.

## Host Requests

### SetFeature PORT_RESET

Reinitialize the embedded function and set the Reset Change bit to indicate that the reset has completed. Reset the Enable Status bit, enable the embedded function and set its address to ‘0’ by the Set embedded function Address / Enable command. Disable the embedded function interrupt endpoint by the Set Endpoint Enable command.

### SetFeature PORT_ENABLE

Enable the function by the Set embedded function Address/Enable command. Set the Enable Status bit.

### SetFeature PORT_SUSPEND

Disable the function by the Set embedded function Address/Enable command. Reset the Enable Status bit and set the Suspend Status bit.

### ClearFeature PORT_ENABLE

Disable the function by the Set embedded function Address / Enable command. Reset the Enable Status bit.

### ClearFeature PORT_SUSPEND

Enable the function by the Set embedded function Address / Enable command. Set the Enable Status bit, reset the Suspend Status bit; set the Resume Status Change bit to indicate that the resume has completed.

### ClearFeature any Change Indicator

Clear the corresponding status change bit.

## Babbling condition

When the embedded function causes a babbling condition, the function is automatically disabled by the PDIUSBH12. As soon as the microcontroller detects the babbling error, it must set the Enable Status Change bit and reset the Enable Status bit.

## Remote WakeUp

There are three scenarios when a remote wakeup can occur. The following describes the course of actions for each of the cases:

1. The device is not suspended and the embedded port is suspended:
   Enable back the function by setting the enable bit in the Set Address/Enable register and update the following status bits in the microcontroller program: reset the Suspend Status bit, set the Enable Status bit and set the Suspend Status Change bit.
2. The device is suspended and the embedded port is suspended:
   Send an upstream Resume using the Send Resume command, enable back the function by setting the enable bit in the Set Address/Enable register and update the following status bits in the microcontroller program: reset the Suspend Status bit, set the Enable Status bit and set the Suspend Status Change bit.
3. The device is suspended and the embedded port is enabled:
   Send an upstream resume using the Send Resume command.

## RECOMMENDED OPERATING CONDITIONS

| SYMBOL | PARAMETER | TEST CONDITIONS | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| $V_{CC}$ | DC supply voltage | | 3.0 | 3.6 | V |
| $V_I$ | DC input voltage range | | 0 | 5.5 | V |
| $V_{I/O}$ | DC input voltage range for I/O | | 0 | 5.5 | V |
| $V_{AI/O}$ | DC input voltage range for analog I/O | | 0 | $V_{CC}$ | V |
| $V_O$ | DC output voltage range | | 0 | $V_{CC}$ | V |
| $T_{amb}$ | Operating ambient temperature range in free air | See DC and AC characteristics per device | –40 | 85 | °C |

## ABSOLUTE MAXIMUM RATINGS[1]

| SYMBOL | PARAMETER | TEST CONDITIONS | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| $V_{CC}$ | DC supply voltage | | –0.5 | +4.6 | V |
| $I_{IK}$ | DC input diode current | $V_I < 0$ | | –50 | mA |
| $V_I$ | DC input voltage | Note 2 | –0.5 | +5.5 | V |
| $V_{I/O}$ | DC input voltage range for I/O's | | –0.5 | $V_{CC}$ + 0.5 | V |
| $I_{OK}$ | DC output diode current | $V_O > V_{CC}$ or $V_O < 0$ | | ±50 | mA |
| $V_O$ | DC output voltage | Note 2 | –0.5 | $V_{CC}$ + 0.5 | V |
| $I_O$ | DC output sink or source current for other pins | $V_O$ = 0 to $V_{CC}$ | | ±15 | mA |
| $I_O$ | DC output sink or source current for D+/D– pins | $V_O$ = 0 to $V_{CC}$ | | ±50 | mA |
| $I_{GND}$, $I_{CC}$ | DC $V_{CC}$ or GND current | | | ±100 | mA |
| $V_{ESD}$ | Electrostatic discharge voltage | $I_{IL} < 1$ µA [3] | — | ±4000 [4] | V |
| $T_{STG}$ | Storage temperature range | | –60 | +150 | °C |
| $P_{TOT}$ | Power dissipation per package | | | | |

**NOTES:**
1. Stresses beyond those listed may cause damage to the device. These are stress ratings only and functional operation of the device at these or any other conditions beyond those listed in the RECOMMENDED OPERATING CONDITIONS table is not implied. Exposure to absolute maximum rated conditions for extended periods may affect device reliability.
2. The input and output voltage ratings may be exceeded if the input and output current ratings are observed.
3. Values are given for device only: in-circuit $V_{ESD(MAX)}$ = ±8000 V.
4. For open-drain pins $V_{ESD(MAX)}$ = ±2000 V.

## DC CHARACTERISTICS (Digital pins)

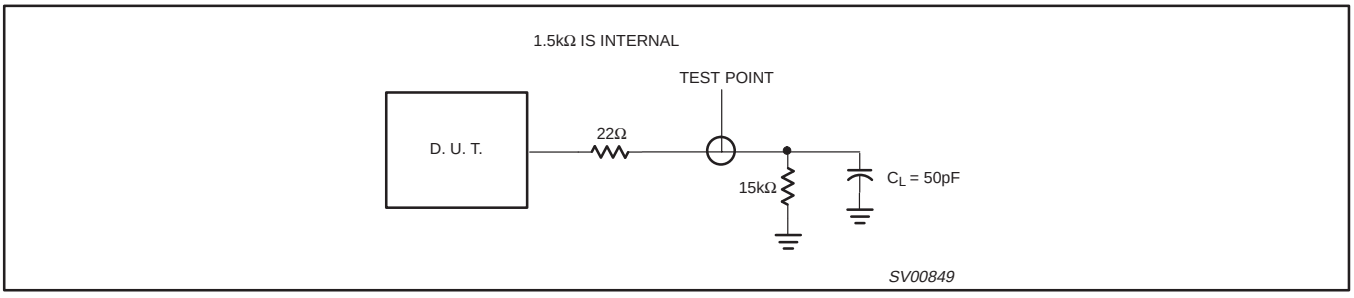
| SYMBOL | PARAMETER | TEST CONDITIONS | MIN | TYP | MAX | UNIT |
|---|---|---|---|---|---|---|
| **Input Levels** | | | | | | |
| $V_{IL}$ | LOW level input voltage | | | | 0.6 | V |
| VIH | HIGH level input voltage | | 2.7 | | | V |
| $V_{TLH}$ | LOW to HIGH threshold voltage | ST (Schmitt Trigger) pins | 1.4 | | 1.9 | V |
| $V_{THL}$ | HIGH to LOW threshold voltage | ST pins | 0.9 | | 1.5 | V |
| $V_{HYS}$ | Hysteresis voltage | ST pins | 0.4 | | 0.7 | V |
| **Output Levels** | | | | | | |
| $V_{OL}$ | LOW level output voltage | $I_{OL}$ = rated drive<br>$I_{OL}$ = 20 µA | | | 0.4<br>0.1 | V<br>V |
| $V_{OH}$ | HIGH level output voltage | $I_{OH}$ = rated drive<br>$I_{OH}$ = 20 µA | 2.4<br>$V_{CC}$ – 0.1 | | | V<br>V |
| **Leakage Current** | | | | | | |
| $I_{OZ}$ | OFF state current | OD (Open Drain) pins | | | ±1 | µA |
| $I_L$ | Input leakage current | | | | ±1 | µA |

## DC CHARACTERISTICS (AI/O pins)

| SYMBOL | PARAMETER | TEST CONDITIONS | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| **Leakage Current** | | | | | |
| $I_{LO}$ | Hi-Z state data line leakage | 0V < $V_{IN}$ < 3.3V | | ±10 | µA |
| **Input Levels** | | | | | |
| $V_{DI}$ | Differential input sensitivity | \|(D+) – (D–)\|[1] | 0.2 | | V |
| $V_{CM}$ | Differential common mode range | Includes $V_{DI}$ range | 0.8 | 2.5 | V |
| $V_{SE}$ | Single-ended receiver threshold | | 0.8 | 2.0 | V |
| **Output Levels** | | | | | |
| $V_{OL}$ | Static output LOW | $R_L$ of 1.5kΩ to 3.6V | | 0.3 | V |
| $V_{OH}$ | Static output HIGH | $R_L$ of 15kΩ to GND | 2.8 | 3.6 | V |
| **Capacitance** | | | | | |
| $C_{IN}$ | Transceiver capacitance | Pin to GND | | 20 | pF |
| **Output Resistance** | | | | | |
| $Z_{DRV}$[2] | Driver output resistance | Steady state drive | 28 | 43 | Ω |
| **Integrated Resistance** | | | | | |
| $Z_{PU}$ | Pull-up resistance | SoftConnect™ = ON | 1.1 | 1.9 | kΩ |
| $Z_{PD}$ | Pull-down resistance | Pull-down = ON | 11 | 19 | kΩ |

**NOTES:**
1. D+ is the symbol for the USB positive data pin: UP_DP, DN2_DP, DN3_DP.
   D– is the symbol for the USB negative data pin: UP_DM, DN2_DM, DN3_DM.
2. Includes external resistors of 22 Ω ± 1% each on D+ and D–.

## LOAD FOR D+/D–



## AC CHARACTERISTICS (AI/O pins, FULL speed)

| SYMBOL | PARAMETER | TEST CONDITIONS | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| **Driver characteristics** | | $C_L$ = 50pF;<br>$R_{pu}$ = 1.5kΩ on D+ to $V_{CC}$ | | | |
| | Transition Time: | | | | |
| $t_r$ | Rise time | Between 10% and 90% | 4 | 20 | ns |
| $t_f$ | Fall time | | 4 | 20 | ns |
| $t_{RFM}$ | Rise/fall time matching | ($t_r/t_f$) | 90 | 110 | % |
| $V_{CRS}$ | Output signal crossover voltage | | 1.3 | 2.0 | V |
| **Driver Timings** | | | | | |
| $t_{EOPT}$ | Source EOP width | Figure 1 | 160 | 175 | ns |
| $t_{DEOP}$ | Differential data to EOP transition skew | Figure 1 | –2 | 5 | ns |
| **Receiver Timings** | | | | | |
| | Receiver Data Jitter Tolerance | | | | |
| $t_{JR1}$ | To next transition | Characterized and not tested. | –18.5 | 18.5 | ns |
| $t_{JR2}$ | For paired transitions | Guaranteed by design. | –9 | 9 | ns |
| | EOP Width at Receiver | | | | |
| $t_{EOPR1}$ | Must reject as EOP | Figure 1 | 40 | | ns |
| $t_{EOPR2}$ | Must accept | | 82 | | ns |
| **Hub Timings** | | Full Speed downstream port. | | | |
| $t_{HDD}$ | Hub Differential Data Delay | Figure 2 | | 40 | ns |
| $t_{SOP}$ | Data bit width distortion after SOP | Figure 2 | –5 | 3 | ns |
| $t_{EOPDR}$ | Hub EOP Delay Relative to $t_{HDD}$ | Figure 3 | 0 | 15 | ns |
| $t_{HESK}$ | Hub EOP Output Width Skew | Figure 3 | –15 | +15 | ns |

## AC CHARACTERISTICS (AI/O pins, LOW speed)

| SYMBOL | PARAMETER | TEST CONDITIONS | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| **Driver characteristics** | | $C_L$ = 50pF and 350pF; $R_{pu}$ = 1.5kΩ on D– to $V_{CC}$ | | | |
| | Transition Time | Between 10% and 90% | | | |
| $t_{lr}$ | Rise time | $C_L$ = 50pF | 75 | | ns |
| | | $C_L$ = 350pF | | 300 | ns |
| $t_{lf}$ | Fall time | $C_L$ = 50pF | 75 | | ns |
| | | $C_L$ = 350pF | | 300 | ns |
| $t_{RFM}$ | Rise/fall time matching | ($t_r/t_f$) | 80 | 120 | % |
| $V_{LCRS}$ | Output signal crossover voltage | | 1.3 | 2.0 | V |
| **Driver Timings** | | | | | |
| $t_{LEOPT}$ | Source EOP width | Figure 1 | 1.25 | 1.50 | µs |
| $t_{LDEOP}$ | Differential data to EOP transition skew | Figure 1 | –40 | 100 | ns |
| **Receiver Timings** | | | | | |
| | EOP Width at Receiver | | | | |
| $t_{LEOPR1}$ | Must reject as EOP | Figure 1 | 330 | | ns |
| $t_{LEOPR2}$ | Must accept | | 675 | | ns |
| **Hub Timings** | | Low Speed downstream port. | | | |
| $t_{LHDD}$ | Hub Differential Data Delay | Figure 2 | | 300 | ns |
| $t_{LSOP}$ | Data bit width distortion after SOP | Figure 2 | –65 | 45 | ns |
| $t_{LEOPDR}$ | Hub EOP Delay Relative to $T_{HDD}$ | Figure 3 | 0 | 200 | ns |
| $t_{LHESK}$ | Hub EOP Output Width Skew | Figure 3 | –300 | +300 | ns |

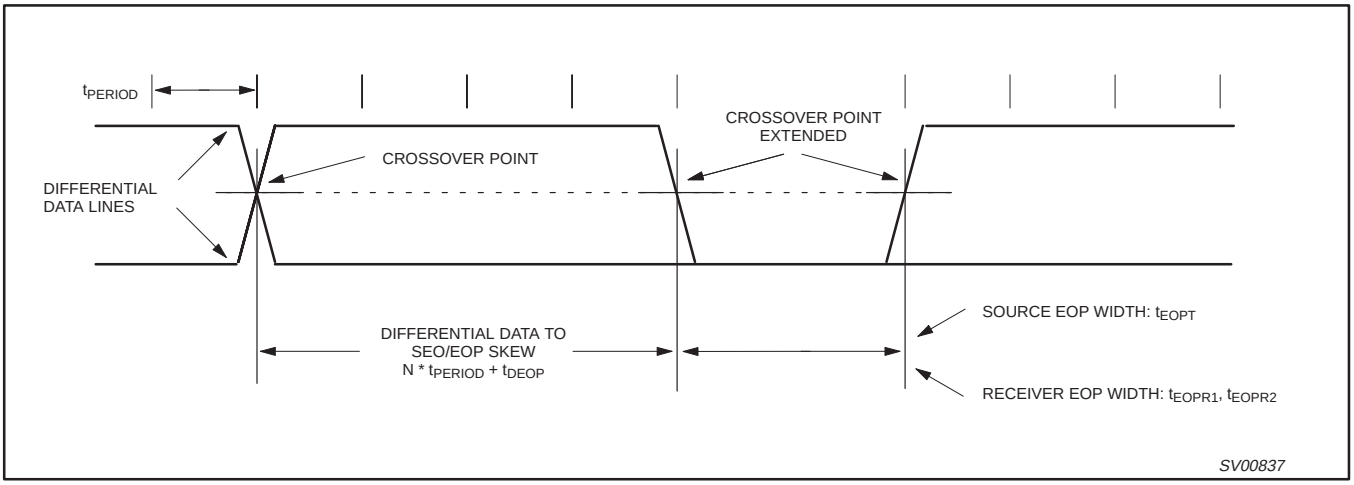



**Figure 1. Differential data to EOP transition skew and EOP width**

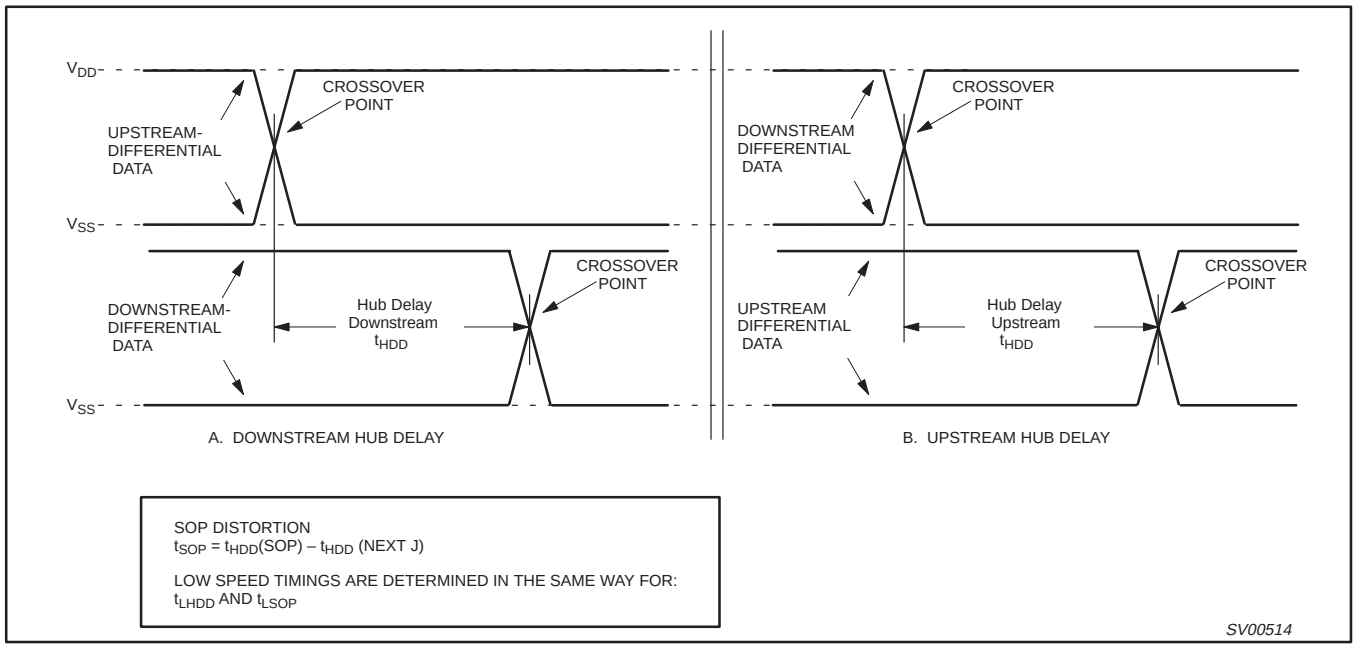



**Figure 2. Hub Differential Data Delay and SOP distortion**

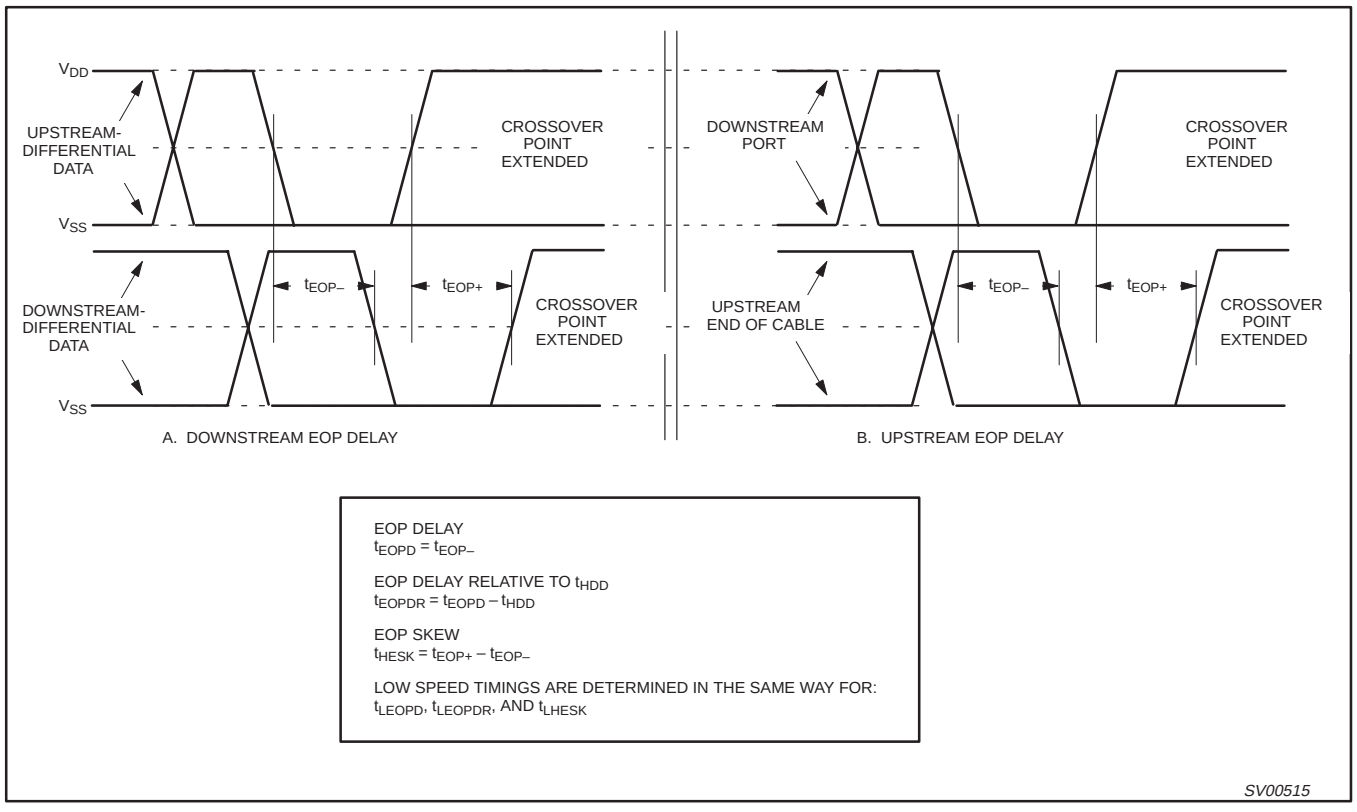



**Figure 3. Hub EOP Delay and EOP Skew**

## AC CHARACTERISTICS (I²C pins)

All timing values are valid within the operating supply voltage and ambient temperature range and reference to $V_{IL}$ and $V_{IH}$ with an input voltage swing of $V_{SS}$ and $V_{DD}$.

| SYMBOL | PARAMETER | TEST CONDITIONS | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| $f_{SCL}$ | SCL clock frequency | | | 1000 | kHz |
| $t_{BUF}$ | Bus free time | | 0.5 | | µs |
| $t_{SU;STA}$ | Start condition set-up time | | 0.25 | | µs |
| $t_{HD;STA}$ | Start condition hold time | | 0.25 | | µs |
| $t_{LOW}$ | SCL LOW time | | 0.45 | | µs |
| $t_{HIGH}$ | SCL HIGH time | | 0.45 | | µs |
| $t_r$ | SCL and SDA rise time | | | 0.3 | µs |
| $t_f$ | SCL and SDA fall time | | | 0.1 | µs |
| $t_{SU;DAT}$ | Data set-up time | | 100 | | ns |
| $t_{HD;DAT}$ | Data hold time | | 0 | | ns |
| $t_{VD;DAT}$ | SCL LOW to data out valid | | | 0.4 | µs |
| $t_{SU;STO}$ | Stop condition set-up time | | 0.25 | | µs |

A detailed description of the I²C-bus specification, with applications, is given in the brochure "*The I²C-bus and how to use it*". This brochure may be ordered using the Philips order number 9398 393 40011.

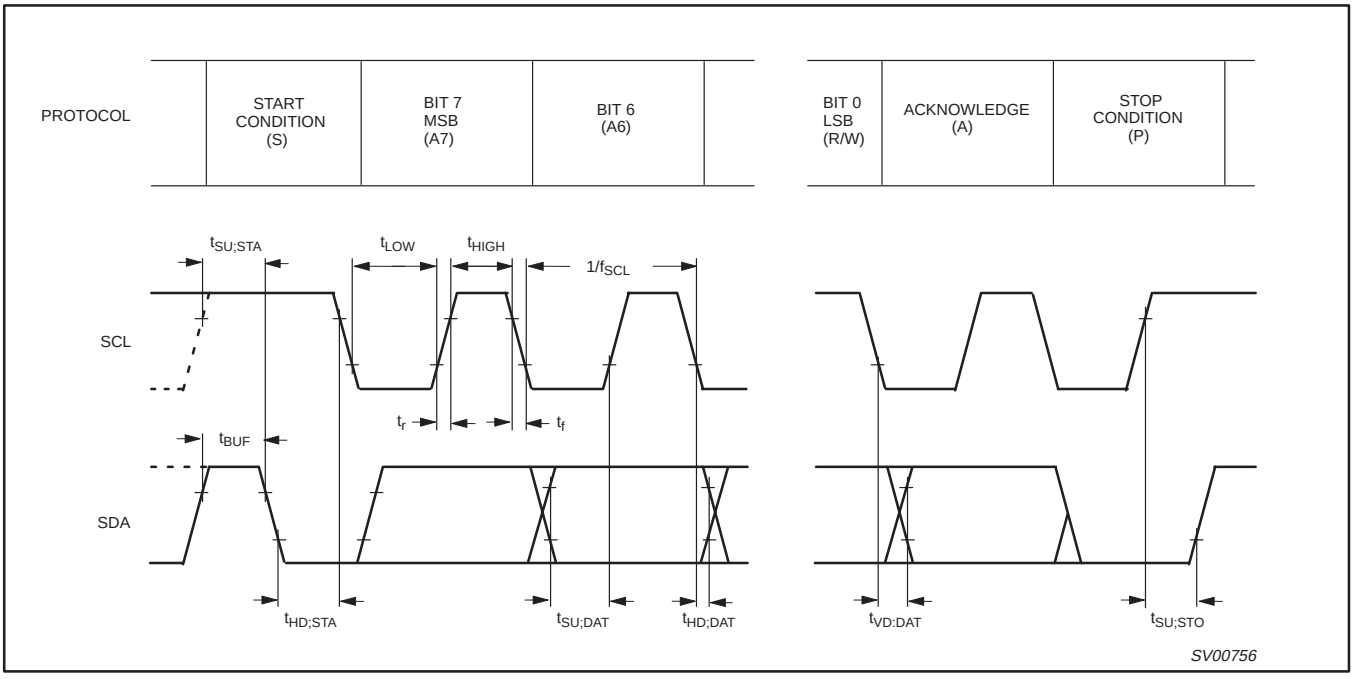
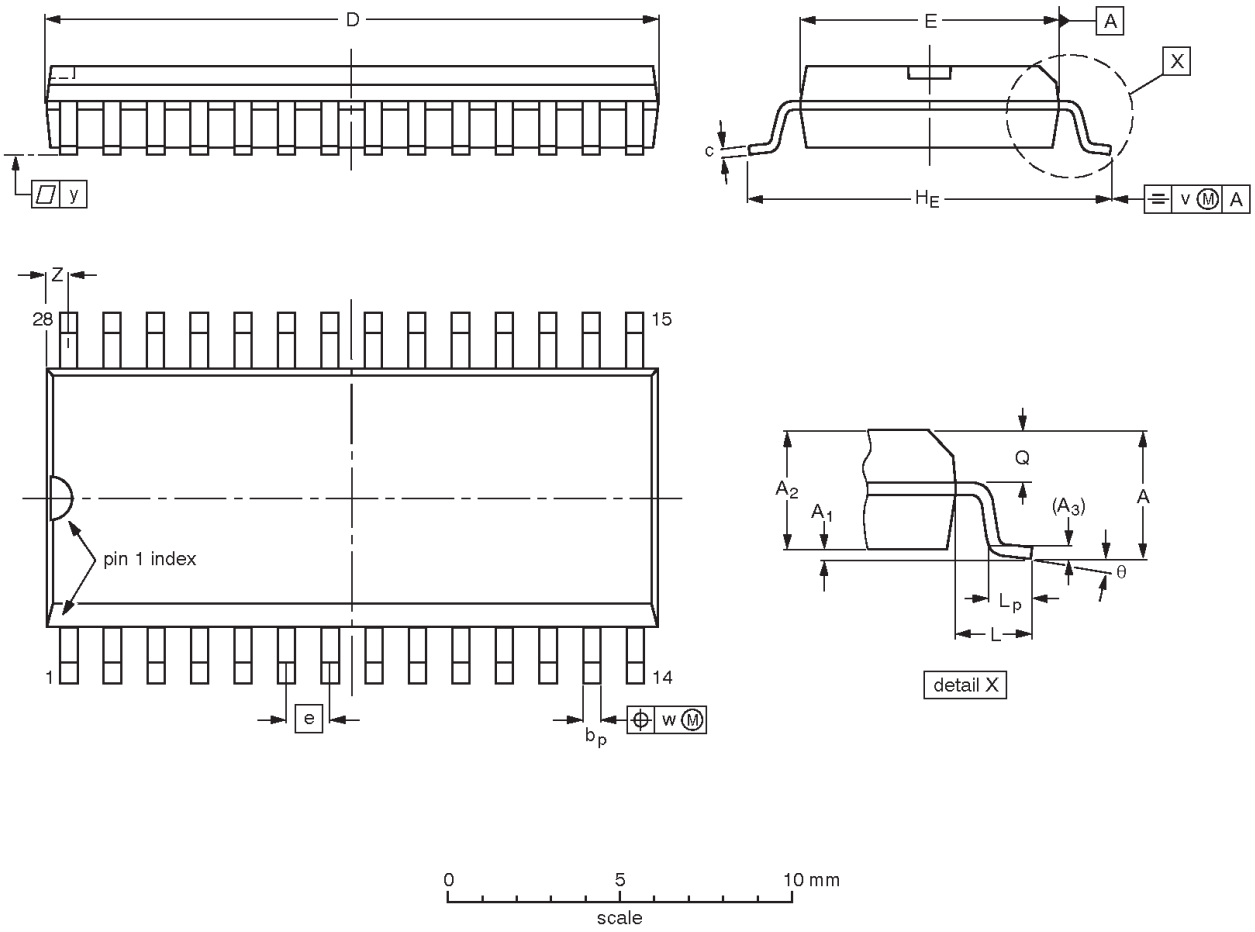



**Figure 4. I²C-bus timing diagram**

**SO28: plastic small outline package; 28 leads; body width 7.5mm** **SOT136-1**




**DIMENSIONS (inch dimensions are derived from the original mm dimensions)**

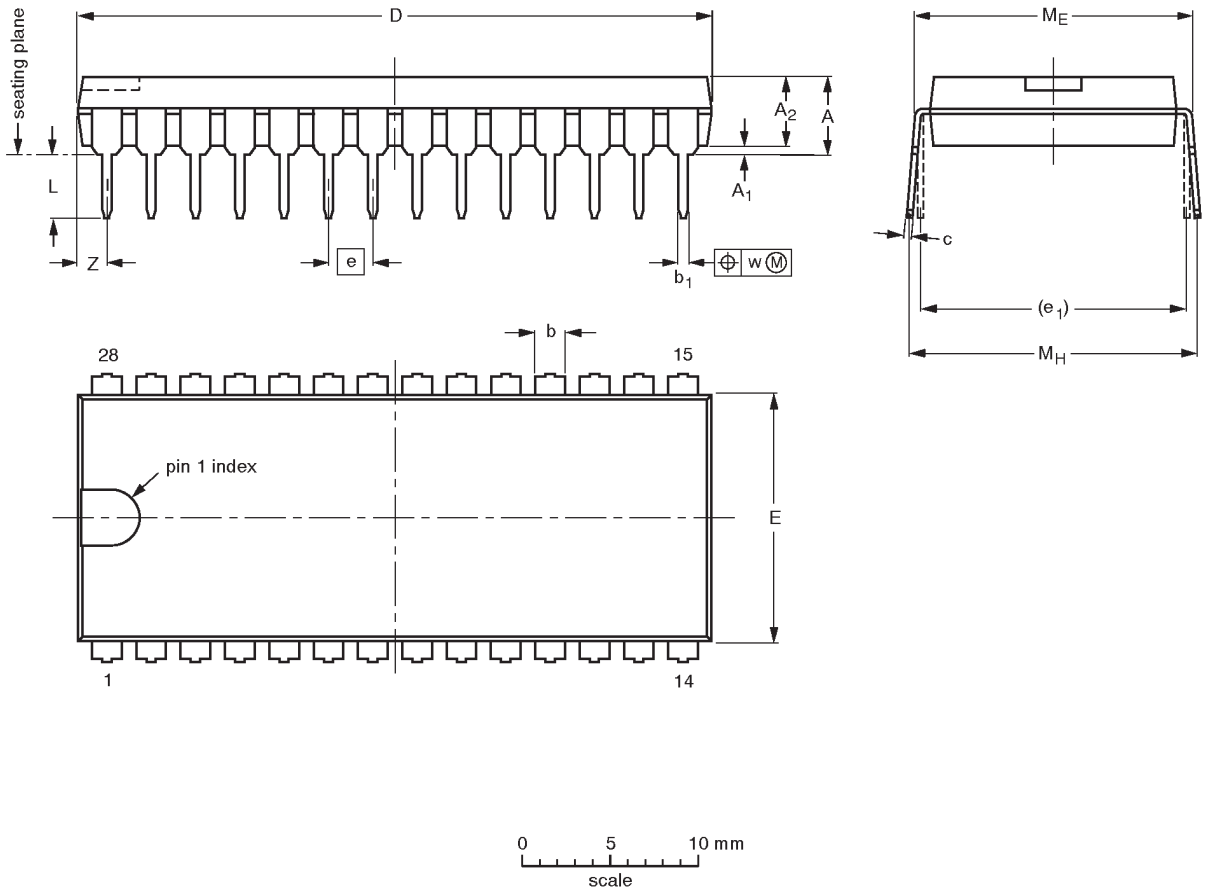
| UNIT | A max. | $A_1$ | $A_2$ | $A_3$ | $b_p$ | c | $D^{(1)}$ | $E^{(1)}$ | e | $H_E$ | L | $L_p$ | Q | v | w | y | $Z^{(1)}$ | θ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| mm | 2.65 | 0.30<br>0.10 | 2.45<br>2.25 | 0.25 | 0.49<br>0.36 | 0.32<br>0.23 | 18.1<br>17.7 | 7.6<br>7.4 | 1.27 | 10.65<br>10.00 | 1.4 | 1.1<br>0.4 | 1.1<br>1.0 | 0.25 | 0.25 | 0.1 | 0.9<br>0.4 | $8^o$<br>$0^o$ |
| inches | 0.10 | 0.012<br>0.004 | 0.096<br>0.089 | 0.01 | 0.019<br>0.014 | 0.013<br>0.009 | 0.71<br>0.69 | 0.30<br>0.29 | 0.050 | 0.419<br>0.394 | 0.055 | 0.043<br>0.016 | 0.043<br>0.039 | 0.01 | 0.01 | 0.004 | 0.035<br>0.016 | |

**Note**

1. Plastic or metal protrusions of 0.15 mm maximum per side are not included.

| OUTLINE VERSION | REFERENCES | | | | EUROPEAN PROJECTION | ISSUE DATE |
|---|---|---|---|---|---|---|
| | IEC | JEDEC | EIAJ | | | |
| SOT136-1 | 075E06 | MS-013AE | | | | ~~95-01-24~~<br>97-05-22 |

**DIP28: plastic dual in-line package; 28 leads (600 mil)** **SOT117-1**




**DIMENSIONS (inch dimensions are derived from the original mm dimensions)**

| UNIT | A max. | $A_1$ min. | $A_2$ max. | b | $b_1$ | c | D (1) | E (1) | e | $e_1$ | L | $M_E$ | $M_H$ | w | Z (1) max. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| mm | 5.1 | 0.51 | 4.0 | 1.7<br>1.3 | 0.53<br>0.38 | 0.32<br>0.23 | 36.0<br>35.0 | 14.1<br>13.7 | 2.54 | 15.24 | 3.9<br>3.4 | 15.80<br>15.24 | 17.15<br>15.90 | 0.25 | 1.7 |
| inches | 0.20 | 0.020 | 0.16 | 0.066<br>0.051 | 0.020<br>0.014 | 0.013<br>0.009 | 1.41<br>1.34 | 0.56<br>0.54 | 0.10 | 0.60 | 0.15<br>0.13 | 0.62<br>0.60 | 0.68<br>0.63 | 0.01 | 0.067 |

**Note**

1. Plastic or metal protrusions of 0.25 mm maximum per side are not included.

| OUTLINE VERSION | REFERENCES | | | | EUROPEAN PROJECTION | ISSUE DATE |
|---|---|---|---|---|---|---|
| | IEC | JEDEC | EIAJ | | | |
| SOT117-1 | 051G05 | MO-015AH | | | | ~~92-11-17~~<br>95-01-14 |

## Data sheet status

| Data sheet status | Product status | Definition [1] |
|---|---|---|
| Objective specification | Development | This data sheet contains the design target or goal specifications for product development. Specification may change in any manner without notice. |
| Preliminary specification | Qualification | This data sheet contains preliminary data, and supplementary data will be published at a later date. Philips Semiconductors reserves the right to make chages at any time without notice in order to improve design and supply the best possible product. |
| Product specification | Production | This data sheet contains final specifications. Philips Semiconductors reserves the right to make changes at any time without notice in order to improve design and supply the best possible product. |

[1] Please consult the most recently issued datasheet before initiating or completing a design.

## Definitions

**Short-form specification —** The data in a short-form specification is extracted from a full data sheet with the same type number and title. For detailed information see the relevant data sheet or data handbook.

**Limiting values definition —** Limiting values given are in accordance with the Absolute Maximum Rating System (IEC 134). Stress above one or more of the limiting values may cause permanent damage to the device. These are stress ratings only and operation of the device at these or at any other conditions above those given in the Characteristics sections of the specification is not implied. Exposure to limiting values for extended periods may affect device reliability.

**Application information —** Applications that are described herein for any of these products are for illustrative purposes only. Philips Semiconductors make no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

## Disclaimers

**Life support —** These products are not designed for use in life support appliances, devices or systems where malfunction of these products can reasonably be expected to result in personal injury. Philips Semiconductors customers using or selling these products for use in such applications do so at their own risk and agree to fully indemnify Philips Semiconductors for any damages resulting from such application.

**Right to make changes —** Philips Semiconductors reserves the right to make changes, without notice, in the products, including circuits, standard cells, and/or software, described or contained herein in order to improve design and/or performance. Philips Semiconductors assumes no responsibility or liability for the use of any of these products, conveys no license or title under any patent, copyright, or mask work right to these products, and makes no representations or warranties that these products are free from patent, copyright, or mask work right infringement, unless otherwise specified.

**Philips Semiconductors**
**811 East Arques Avenue**
**P.O. Box 3409**
**Sunnyvale, California 94088–3409**
**Telephone 800-234-7381**



print code Date of release: 02-99

Document order number: 9397 750 05352

Let's make things better.

**Philips Semiconductors**

PHILIPS