PDIUSBD12

USB interface device with parallel bus

Rev. 08 — 20 December 2001 **Product data**

1. Description

The PDIUSBD12 is a cost and feature optimized USB device. It is normally used in microcontroller based systems and communicates with the system microcontroller over the high-speed general purpose parallel interface. It also supports local DMA transfer.

This modular approach to implementing a USB interface 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 minimize firmware investments. This results in the fastest way to develop the most cost effective USB peripheral solution.

The PDIUSBD12 fully conforms to the *USB specification Rev. 2.0 (basic speed)*. It is also designed to be compliant with most device class specifications: Imaging Class, Mass Storage Devices, Communication Devices, Printing Devices, and Human Interface Devices. As such, the PDIUSBD12 is ideally suited for many peripherals like Printer, Scanner, External Mass Storage (Zip Drive), Digital Still Camera, etc. It offers an immediate cost reduction for applications that currently use SCSI implementations.

The PDIUSBD12 low suspend power consumption along with the 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 bus powered peripherals.

In addition, it also incorporates features like SoftConnect™, GoodLink™, programmable clock output, low frequency crystal oscillator, and integration of termination resistors. All of these features contribute to significant cost savings in the system implementation and at the same time ease the implementation of advanced USB functionality into the peripherals.

2. Features

- Complies with the *Universal Serial Bus specification Rev. 2.0 (basic speed)*
- High performance USB interface device with integrated SIE, FIFO memory, transceiver and voltage regulator
- Compliant with most Device Class specifications
- High-speed (2 Mbytes/s) parallel interface to any external microcontroller or microprocessor
- Fully autonomous DMA operation
- Integrated 320 bytes of multi-configuration FIFO memory

PHILIPS

- Double buffering scheme for main endpoint increases throughput and eases real-time data transfer
- Data transfer rates: 1 Mbytes/s achievable in Bulk mode, 1 Mbits/s achievable in Isochronous mode
- Bus-powered capability with very good EMI performance
- Controllable LazyClock output during suspend
- Software controllable connection to the USB bus (SoftConnect™)
- Good USB connection indicator that blinks with traffic (GoodLink™)
- Programmable clock frequency output
- Complies with the ACPI, OnNOW and USB power management requirements
- Internal Power-on reset and low-voltage reset circuit
- Available in SO28 and TSSOP28 pin packages
- Full industrial grade operation from –40 to +85 °C
- Higher than 8 kV in-circuit ESD protection lowers cost of extra components
- Full-scan design with high fault coverage (>99%) ensures high quality
- Operation with dual voltages:
  3.3 ±0.3 V or extended 5 V supply range of 4.0 to 5.5 V
- Multiple interrupt modes to facilitate both bulk and isochronous transfers.

3. Pinning information

3.1 Pinning

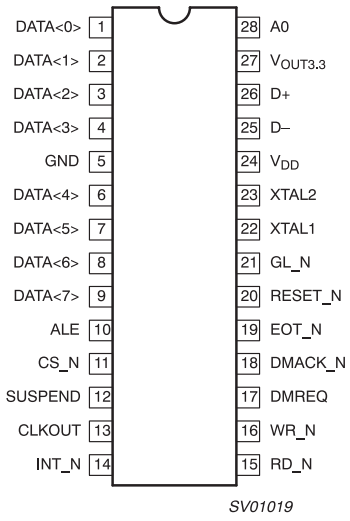


Fig 1. Pin configuration.

3.2 Pin description

Table 1: Pin description

Symbol	Pin	Type[1]	Description
DATA <0>	1	IO2	Bit 0 of bidirectional data. Slew-rate controlled.
DATA <1>	2	IO2	Bit 1 of bidirectional data. Slew-rate controlled.
DATA <2>	3	IO2	Bit 2 of bidirectional data. Slew-rate controlled.
DATA <3>	4	IO2	Bit 3 of bidirectional data. Slew-rate controlled.
GND	5	P	Ground.
DATA <4>	6	IO2	Bit 4 of bidirectional data. Slew-rate controlled.
DATA <5>	7	IO2	Bit 5 of bidirectional data. Slew-rate controlled.
DATA <6>	8	IO2	Bit 6 of bidirectional data. Slew-rate controlled.
DATA <7>	9	IO2	Bit 7 of bidirectional data. Slew-rate controlled.
ALE	10	I	Address Latch Enable. The falling edge is used to close the latch of the address information in a multiplexed address/ data bus. Permanently tied LOW for separate address/ data bus configuration.
CS_N	11	I	Chip Select (Active LOW).
SUSPEND	12	I,OD4	Device is in Suspend state.
CLKOUT	13	O2	Programmable Output Clock (slew-rate controlled).
INT_N	14	OD4	Interrupt (Active LOW).
RD_N	15	I	Read Strobe (Active LOW).
WR_N	16	I	Write Strobe (Active LOW).
DMREQ	17	O4	DMA Request.
DMACK_N	18	I	DMA Acknowledge (Active LOW).
EOT_N	19	I	End of DMA Transfer (Active LOW). Double up as V_{BUS} sensing. EOT_N is only valid when asserted together with DMACK_N and either RD_N or WR_N.
RESET_N	20	I	Reset (Active LOW and asynchronous). Built-in Power-on reset circuit present on chip, so pin can be tied HIGH to V_{CC}.
GL_N	21	OD8	GoodLink LED indicator (Active LOW)
XTAL1	22	I	Crystal Connection 1 (6 MHz).
XTAL2	23	O	Crystal Connection 2 (6 MHz). If external clock signal, instead of crystal, is connected to XTAL1, then XTAL2 should be floated.
V_{CC}	24	P	Voltage supply (4.0 – 5.5 V). To operate the IC at 3.3 V, supply 3.3 V to both V_{CC} and $V_{OUT3.3}$ pins.
D–	25	A	USB D– data line.
D+	26	A	USB D+ data line.

Table 1: Pin description...*continued*

Symbol	Pin	Type[1]	Description
$V_{OUT3.3}$	27	P	3.3 V regulated output. To operate the IC at 3.3 V, supply a 3.3 V to both V_{CC} and $V_{OUT3.3}$ pins.
A0	28	I	Address bit. A0 = 1 selects command instruction; A0 = 0 selects the data phase. This bit is a don't care in a multiplexed address and data bus configuration and should be tied HIGH.

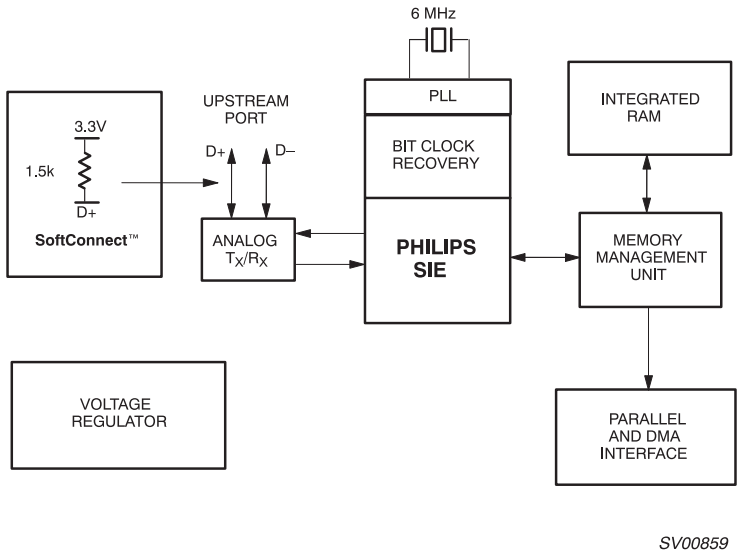
[1] O2 : Output with 2 mA drive
OD4: Output Open Drain with 4 mA drive
OD8: Output Open Drain with 8 mA drive
IO2: Input and Output with 2 mA drive
O4 : Output with 4 mA drive.

4. Ordering information

Table 2: Ordering information

Packages	Temperature range	Outside North America	North America	Pkg. Dwg. #
28-pin plastic SO	–40 °C to +85 °C	PDIUSBD12 D	PDIUSBD12 D	SOT136-1
28-pin plastic TSSOP	–40 °C to +85 °C	PDIUSBD12 PW	PDIUSBD12PW DH	SOT361-1

5. Block diagram



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

Fig 2. Block diagram.

6. Functional description

6.1 Analog transceiver

The integrated transceiver interfaces directly to the USB cables through termination resistors.

6.2 Voltage regulator

A 3.3 V regulator is integrated on-chip to supply the analog transceiver. This voltage is also provided as an output to connect to the external 1.5 kΩ pull-up resistor. Alternatively, the PDIUSBD12 provides SoftConnect technology with an integrated 1.5 kΩ pull-up resistor.

6.3 PLL

A 6 MHz to 48 MHz clock multiplier PLL (Phase-Locked Loop) is integrated on-chip. This allows for the use of a low-cost 6 MHz crystal. EMI is also minimized due to the lower frequency crystal. No external components are needed for the operation of the PLL.

6.4 Bit clock recovery

The bit clock recovery circuit recovers the clock from the incoming USB data stream using 4× oversampling principle. It is able to track jitter and frequency drift specified by the USB specification.

6.5 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, and handshake evaluation/generation.

6.6 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 PDIUSBD12, the 1.5 kΩ pull-up resistor is integrated on-chip and is not connected to V_{CC} by default. The connection 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 performed without requiring to pull out the cable.

The PDIUSBD12 will check for USB V_{BUS} availability before the connection can be established. V_{BUS} sensing is provided through pin EOT_N. See Section 3.2 “Pin description” for details. Sharing of V_{BUS} sensing and EOT_N can be easily accomplished by using V_{BUS} voltage as the pull-up voltage for the normally open-drain output of the DMA controller pin.

It should be noted that the tolerance of the internal resistors is higher (25%) 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 sure of this feature lies with the users.

6.7 GoodLink

Good USB connection indication is provided through GoodLink technology. During enumeration, the LED indicator will blink ON momentarily corresponding to the enumeration traffic. When the PDIUSBD12 is successfully enumerated and configured, the LED indicator will be permanently ON. Subsequent successful (with acknowledgement) transfer to and from the PDIUSBD12 will blink OFF the LED. During suspend, the LED will be OFF.

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

6.8 Memory Management Unit (MMU) and Integrated RAM

The MMU and the integrated RAM buffer the difference in speed between USB, running in bursts of 12 Mbits/s and the parallel interface to the microcontroller. This allows the microcontroller to read and write USB packets at its own speed.

6.9 Parallel and DMA Interface

A generic parallel interface is defined for ease-of-use, speed, and allows direct interfacing to major microcontrollers. To a microcontroller, the PDIUSBD12 appears as a memory device with 8-bit data bus and 1 address bit (occupying 2 locations). The PDIUSBD12 supports both multiplexed and non-multiplexed address and data bus. The PDIUSBD12 also supports DMA (Direct Memory Access) transfer which allows the main endpoint (endpoint 2) to directly transfer to and from the local shared memory. Both single-cycle and burst mode DMA transfers are supported.

6.10 Example of parallel interface to an 80C51 microcontroller

In the example shown in Figure 3, the ALE pin is permanently tied LOW to signify a separate address and data bus configuration. The A0 pin of the PDIUSBD12 connects to any of the 80C51 I/O ports. This port controls the command or data phase to the PDIUSBD12. The multiplexed address and data bus of the 80C51 can now be connected directly to the data bus of the PDIUSBD12. The address phase will be ignored by the PDIUSBD12. The clock input signal of the 80C51 (pin XTAL1) can be provided by output CLKOUT of the PDIUSBD12.

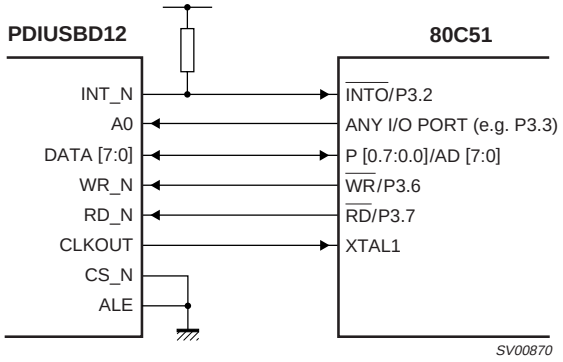


Fig 3. Example of a parallel interface to an 80C51 microcontroller.

7. DMA transfer

Direct Memory Address (DMA) allows an efficient transfer of a block of data between the host and local shared memory. Using a DMA controller, data transfer between the PDIUSBD12's main endpoint (endpoint 2) and local shared memory can happen autonomously without local CPU intervention.

Preceding any DMA transfer, the local CPU receives from the host the necessary setup information and programs the DMA controller accordingly. Typically, the DMA controller is set up for demand transfer mode and the byte count register and the address counter are programmed with the right values. In this mode, transfers occur only when the PDIUSBD12 requests them and are terminated when the byte count register reaches zero. After the DMA controller has been programmed, the DMA enable bit of the PDIUSBD12 is set by the local CPU to initiate the transfer.

The PDIUSBD12 can be programmed for single-cycle DMA or burst mode DMA. In single-cycle DMA, the DMREQ pin is deactivated for every single acknowledgement by the DMACK_N before being re-asserted. In burst mode DMA, the DMREQ pin is kept active for the number of bursts programmed in the device before going inactive. This process continues until the PDIUSBD12 receives a DMA termination notice through pin EOT_N. This will generate an interrupt to notify the local CPU that DMA operation is completed.

For DMA read operation, the DMREQ pin will only be activated whenever the buffer is full, signalling that the host has successfully transferred a packet to the PDIUSBD12. With the double buffering scheme, the host can start filling up the second buffer while the first buffer is being read out. This parallel processing increases the effective throughput. When the host does not fill up the buffer completely (less than 64 bytes or 128 bytes for single direction ISO configuration), the DMREQ pin will be deactivated at the last byte of the buffer regardless of the current DMA burst count. It will be re-asserted on the next packet with a refreshed DMA burst count.

Similarly, for DMA write operations, the DMREQ pin remains active whenever the buffer is not full. When the buffer is filled up, the packet is sent over to the host on the next IN token and DMREQ will be reactivated if the transfer was successful. Also, the double buffering scheme here will improve throughput. For non-isochronous transfer (bulk and interrupt), the buffer needs to be completely filled up by the DMA write

operation before the data is sent to the host. The only exception is at the end of DMA transfer, when the reception of pin EOT_N will stop DMA write operation and the buffer content will be sent to the host on the next IN token.

For isochronous transfers, the local CPU and DMA controller have to guarantee that they are able to sink or source the maximum packet size in one USB frame (1 ms).

The assertion of pin DMACK_N automatically selects the main endpoint (endpoint 2), regardless of the current selected endpoint. The DMA operation of the PDIUSBD12 can be interleaved with normal I/O access to other endpoints.

DMA operation can be terminated by resetting the DMA enable register bit or the assertion of EOT_N together with DMACK_N and either RD_N or WR_N.

The PDIUSBD12 supports DMA transfer in single address mode and it can also work in dual address mode of the DMA controller. In the single address mode, DMA transfer is done via the DREQ, DMACK_N, EOT_N, WR_N and RD_N control lines. In the dual address mode, pins DMREQ, DMACK_N and EOT_N are **not** used; instead CS_N, WR_N and RD_N control signals are used. The I/O mode Transfer Protocol of PDIUSBD12 needs to be followed. The source of the DMAC is accessed during the read cycle and the destination during the write cycle. Transfer needs to be done in two separate bus cycles, storing the data temporarily in the DMAC.

8. Endpoint description

The PDIUSBD12 endpoints are sufficiently generic to be used by various device classes ranging from Imaging, Printer, Mass Storage and Communication device classes. The PDIUSBD12 endpoints can be configured for 4 operating modes depending on the Set mode command. The 4 modes are:

Mode 0 Non-isochronous transfer (Non-ISO mode)

Mode 1 Isochronous output only transfer (ISO-OUT mode)

Mode 2 Isochronous input only transfer (ISO-IN mode)

Mode 3 Isochronous input and output transfer (ISO-I/O mode).

Table 3: Endpoint Configuration

Endpoint number	Endpoint index	Transfer type	Direction[1]	Max. Packet size (bytes)
Mode 0 (Non-ISO mode)				
0	0	Control	OUT	16
	1		IN	16
1	2	Generic[2]	OUT	16
	3		IN	16
2	4	Generic[2][3]	OUT	64 [4]
	5		IN	64 [4]
Mode 1 (ISO-OUT mode)				
0	0	Control	OUT	16
	1		IN	16
1	2	Generic[2]	OUT	16
	3		IN	16
2	4	Isochronous[3]	OUT	128[4]
Mode 2 (ISO-IN mode)				
0	0	Control	OUT	16
	1		IN	16
1	2	Generic[2]	OUT	16
	3		IN	16
2	5	Isochronous[3]	IN	128[4]
Mode 3 (ISO-I/O mode)				
0	0	Control	OUT	16
	1		IN	16
1	2	Generic[2]	OUT	16
	3		IN	16
2	4	Isochronous[3]	OUT	64[4]
	5		IN	64[4]

[1] IN: input for the USB host; OUT: output from the USB host.

[2] Generic endpoints can be used either as Bulk or Interrupt endpoint.

[3] The main endpoint (endpoint number 2) is double-buffered to ease synchronization with the real-time applications and to increase throughput. This endpoint supports DMA access.

[4] Denotes double buffering. The size shown is for a single buffer.

9. Main endpoint

The main endpoint (endpoint number 2) is the primary endpoint for sinking or sourcing relatively large amounts of data. It implements the following features to ease this task:

- Double buffering. This allows parallel operation between USB access and local CPU access thus increasing throughput. Buffer switching is handled automatically. This results in transparent buffer operation.
- DMA (Direct Memory Access) operation. This can be interleaved with normal I/O operation to other endpoints.
- Automatic pointer handling during DMA operation. No local CPU intervention is necessary when ‘crossing’ the buffer boundary.
- Configurable endpoint for either isochronous transfer or non-isochronous (bulk and interrupt) transfer.

10. Command summary

Table 4: Command summary

Name	Destination	Code (Hex)	Transaction
Initialization commands			
Set Address/Enable	Device	D0	Write 1 byte
Set Endpoint Enable	Device	D8	Write 1 byte
Set mode	Device	F3	Write 2 bytes
Set DMA	Device	FB	Write/Read 1 byte
Data flow commands			
Read Interrupt Register	Device	F4	Read 2 bytes
Select Endpoint	Control OUT	00	Read 1 byte (optional)
	Control IN	01	Read 1 byte (optional)
	Endpoint 1 OUT	02	Read 1 byte (optional)
	Endpoint 1 IN	03	Read 1 byte (optional)
	Endpoint 2 OUT	04	Read 1 byte (optional)
	Endpoint 2 IN	05	Read 1 byte (optional)
Read Last Transaction Status	Control OUT	40	Read 1 byte
	Control IN	41	Read 1 byte
	Endpoint 1 OUT	42	Read 1 byte
	Endpoint 1 IN	43	Read 1 byte
	Endpoint 2 OUT	44	Read 1 byte
	Endpoint 2 IN	45	Read 1 byte
Read Buffer	Selected Endpoint	F0	Read n bytes
Write Buffer	Selected Endpoint	F0	Write n bytes

Table 4: Command summary…*continued*

Name	Destination	Code (Hex)	Transaction
Set Endpoint Status	Control OUT	40	Write 1 byte
	Control IN	41	Write 1 byte
	Endpoint 1 OUT	42	Write 1 byte
	Endpoint 1 IN	43	Write 1 byte
	Endpoint 2 OUT	44	Write 1 byte
	Endpoint 2 IN	45	Write 1 byte
Acknowledge Setup	Selected Endpoint	F1	None
Clear Buffer	Selected Endpoint	F2	None
Validate Buffer	Selected Endpoint	FA	None
General commands			
Send Resume		F6	None
Read Current Frame Number		F5	Read 1 or 2 bytes

11. Command description

11.1 Command procedure

There are three basic types of commands: Initialization, Data Flow and General commands. Respectively, these are used to initialize the function; for data flow between the function and the host; and some general commands.

11.2 Initialization commands

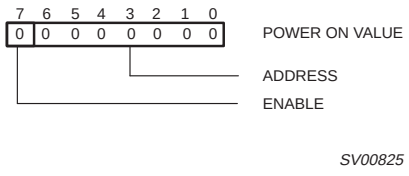
Initialization commands are used during the enumeration process of the USB network. These commands are used to enable the function endpoints. They are also used to set the USB assigned address.

11.2.1 Set Address/Enable

Code (Hex) — D0

Transaction — write 1 byte

This command is used to set the USB assigned address and enable the function.



ADDRESS: The value written becomes the address.

ENABLE: A ‘1’ enables this function.

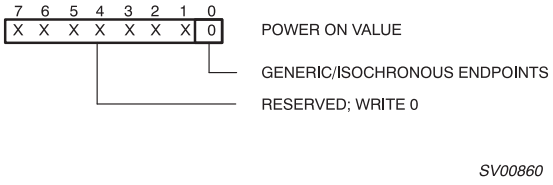
Fig 4. Set Address/Enable command: bit allocation.

11.2.2 Set endpoint enable

Code (Hex) — D8

Transaction — write 1 byte

The generic/Isochronous endpoints can only be enabled when the function is enabled via the Set Address/Enable command.



GENERIC/ISOCHRONOUS ENDPOINT: A value of '1' indicates the generic/isochronous endpoints are enabled.

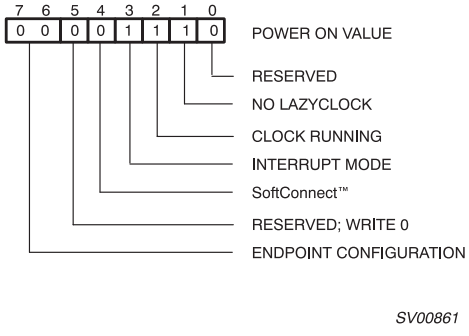
Fig 5. Set endpoint enable command: bit allocation.

11.2.3 Set mode

Code (Hex) — F3

Transaction — write 2 bytes

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

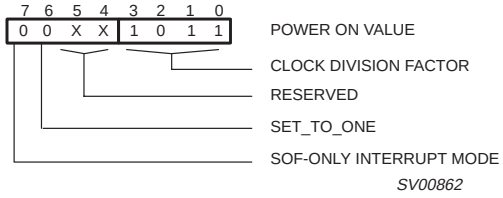


See Table 5 for bit allocation.

Fig 6. Set mode command, Configuration byte.

Table 5: Set mode command, Configuration byte: bit allocation

Bit	Symbol	Description
7 to 6	ENDPOINT CONFIGURATION	These two bits set the endpoint configurations as follows: mode 0 (Non-ISO mode) mode 1 (ISO-OUT mode) mode 2 (ISO-IN mode) mode 3 (ISO-I/O mode) See Section 8 “Endpoint description” for more details.
4	SoftConnect	A ‘1’ indicates that the upstream pull-up resistor will be connected if V_{BUS} is available. A ‘0’ means that the upstream resistor will not be connected. The programmed value will not be changed by a bus reset.
3	INTERRUPT MODE	A ‘1’ indicates that all errors and “NAKing” are reported and will generate an interrupt. A ‘0’ indicates that only OK is reported. The programmed value will not be changed by a bus reset.
2	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.
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 30 kHz ± 40%. The programmed value will not be changed by a bus reset.



See Table 6 for bit allocation.

Fig 7. Set mode command, Clock division factor byte.

Table 6: Clock division factor byte: bit allocation

Bit	Symbol	Description
7	SOF-ONLY INTERRUPT MODE	Setting this bit to 1 will cause the interrupt line to be activated due to the Start Of Frame clock (SOF) only, regardless of the setting of Pin-Interrupt mode, bit 5 of set DMA.
6	SET_TO_ONE	This bit needs to be set to 1 prior to any DMA read or DMA write operation. This bit should always be set to 1 after power. It is zero after Power-on reset.
3 to 0	CLOCK DIVISION FACTOR	The value indicates the clock division factor for CLKOUT. The output frequency is 48 MHz/(N+1) where N is the Clock Division Factor. The reset value is 11. This will produce the output frequency of 4 MHz which can then be programmed up or down by the user. The minimum value is 1 giving the range of frequency from 4 to 24 MHz. The minimum value of N is 0, giving a maximum frequency of 48 MHz. The maximum value of N is 11 giving a minimum frequency of 4 MHz. The PDIUSBD12 design ensures no glitching during frequency change. The programmed value will not be changed by a bus reset.

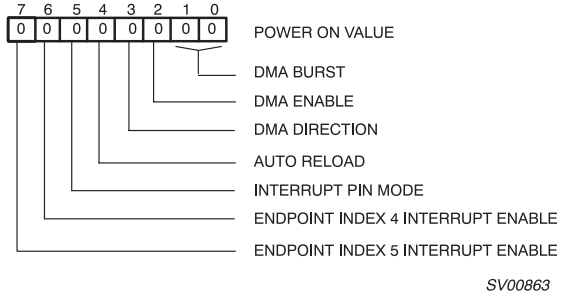
11.2.4 Set DMA

Code (Hex) — FB

Transaction — read/write 1 byte

The set DMA command is followed by one data write/read to/from the DMA configuration register.

DMA Configuration register: During DMA operation, the two-byte buffer header (status and byte length information) is not transferred to/from the local CPU. This allows DMA data to be continuous and not interleaved by chunks of these headers. For DMA read operations, the header will be skipped by the PDIUSBD12. See Section 11.3.5 “Read buffer” command. For DMA write operations, the header will be automatically added by the PDIUSBD12. This provides for a clean and simple DMA data transfer.



See Table 7 for bit allocation.

Fig 8. Set DMA command.

Table 7: Set DMA command: bit allocation

Bit	Symbol	Description
7	ENDPOINT INDEX 5 INTERRUPT ENABLE	A ‘1’ allows for an interrupt to be generated whenever the endpoint buffer is validated (see Section 11.3.8 “Validate buffer” command). Normally turned off for DMA operation to reduce unnecessary CPU servicing.
6	ENDPOINT INDEX 4 INTERRUPT ENABLE	A ‘1’ allows for an interrupt to be generated whenever the endpoint buffer contains a valid packet. Normally turned off for DMA operation to reduce unnecessary CPU servicing.
5	INTERRUPT PIN MODE	A ‘0’ signifies a normal interrupt pin mode where an interrupt is generated as a logical OR of all the bits in the interrupt registers. A ‘1’ signifies that the interrupt will occur when Start of Frame clock (SOF) is seen on the upstream USB bus. The other normal interrupts are still active.
4	AUTO RELOAD	When this bit is set to ‘1’, the DMA operation will automatically restart.
3	DMA DIRECTION	This bit determines the direction of data flow during a DMA transfer. A ‘1’ means external shared memory to PDIUSBD12 (DMA Write); a ‘0’ means PDIUSBD12 to the external shared memory (DMA Read).
2	DMA ENABLE	Writing a ‘1’ to this bit will start DMA operation through the assertion of pin DMREQ. The main endpoint buffer needs to be full (for DMA Read) or empty (for DMA Write) before DMREQ will be asserted. In a single cycle DMA mode, the DMREQ is deactivated upon receiving DMACK_N. In burst mode DMA, the DMREQ is deactivated after the number of burst is exhausted. It is then asserted again for the next burst. This process continues until EOT_N is asserted together with DMACK_N and either RD_N or WR_N, which will reset this bit to ‘0’ and terminate the DMA operation. The DMA operation can also be terminated by writing a ‘0’ to this bit.
1 to 0	DMA BURST	Selects the burst length for DMA operation: 00 Single-cycle DMA 01 Burst (4-cycle) DMA 10 Burst (8-cycle) DMA 11 Burst (16-cycle) DMA

11.3 Data flow commands

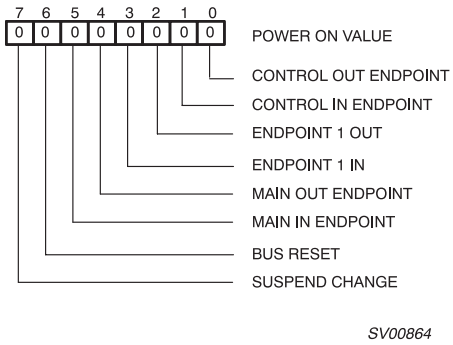
Data flow commands are used to manage the data transmission between the USB endpoints and the external microcontroller. Much of the data flow is initiated via an interrupt to the microcontroller. The microcontroller utilizes these commands to access and determine whether the endpoint FIFOs have valid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

11.3.1 Read interrupt register

Code (Hex) — F4

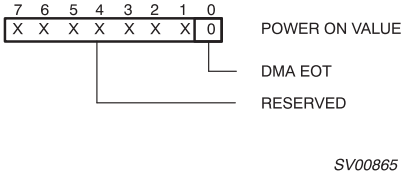
Transaction — read 2 bytes

This command indicates the origin of an interrupt. The endpoint interrupt bits (bits 0 to 5) are cleared by reading the endpoint last transaction status register through Read Last Transaction Status command. The other bits are cleared after reading the interrupt registers.



See Table 8 for bit allocation.

Fig 9. Interrupt Register, byte 1.



DMA EOT: This bit signifies that DMA operation is completed.

Fig 10. Interrupt Register, byte 2: bit allocation.

Table 8: Read interrupt register, byte 1: bit allocation

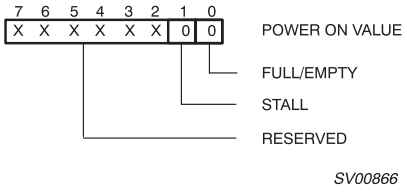
Bit	Symbol	Description
7	SUSPEND CHANGE	When the PDIUSBD12 did not receive 3 SOFs, it will go into suspend state and the Suspend Change bit will be HIGH. Any change to the suspend or awake state will set this bit HIGH and generate an interrupt.
6	BUS RESET	After a bus reset an interrupt will be generated this bit will be '1'. A bus reset is identical to a hardware reset through the RESET_N pin with the exception that a bus reset generates an interrupt notification and the device is enabled at default address 0.

11.3.2 Select Endpoint

Code (Hex) — 00 to 05

Transaction — read 1 byte (optional)

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 this byte.



FULL/EMPTY: A '1' indicates the buffer is full, '0' indicates an empty buffer.

STALL: A '1' indicates the selected endpoint is in the stall state.

Fig 11. Select Endpoint command: bit allocation.

11.3.3 Read Endpoint status

Code (Hex) — 80 to 85

Transaction — read 1 byte

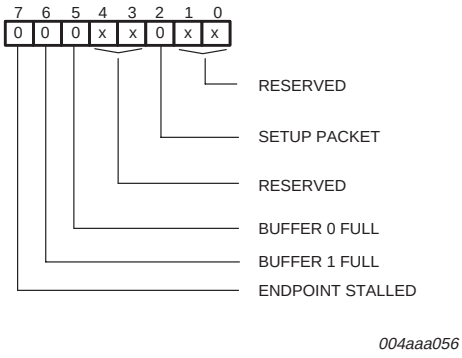


Fig 12. Read Endpoint status: bit allocation.

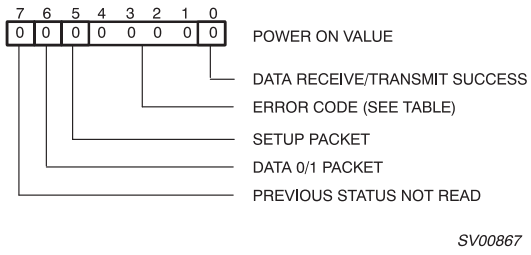
11.3.4 Read last transaction status register

Code (Hex) — 40 to 45

Transaction — 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.



See Table 9 for bit allocation.

Fig 13. Read last transaction status register.

Table 9: Read last transaction status register: bit allocation

Bit	Symbol	Description
7	PREVIOUS STATUS NOT READ	A '1' indicates a second event occurred before the previous status was read.
6	DATA 0/1 PACKET	A '1' indicates the last successful received or sent packet had a DATA1 PID.
5	SETUP PACKET	A '1' indicates the last successful received packet had a SETUP token (this will always read '0' for IN buffers).
4 to 1	ERROR CODE	See Table 10 "Error codes".
0	DATA RECEIVE/TRANSMIT SUCCESS	A '1' indicates data has been received or transmitted successfully.

Table 10: Error codes

Error code (Binary)	Description
0000	No Error
0001	PID encoding Error; bits 7 to 4 are not the inversion of bits 3 to 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	Never happens
1000	Unexpected End-Of-Packet
1001	Sent or received NAK
1010	Sent Stall, a token was received, but the endpoint was stalled

Table 10: Error codes…*continued*

Error code (Binary)	Description
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

11.3.5 Read buffer

Code (Hex) — F0

Transaction — read multiple bytes (max. 130)

The Read Buffer command is followed by a number of data reads, which returns 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 top of the buffer by the Read Buffer command. This means that reading or writing a buffer can be interrupted by any other command (except for Select Endpoint).

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
- etc.

The first two bytes will be skipped in the DMA read operation. Thus, the first read will get Data byte 1, the second read will get Data byte 2, etc. The PDIUSBD12 can determine the last byte of this packet through the EOP termination of the USB packet.

11.3.6 Write buffer

Code (Hex) — F0

Transaction — write multiple bytes (max. 130)

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’.

During DMA write operation, the first two bytes will be bypassed. Thus, the first write will write into Data byte 1, the second write will write into Data byte 2, etc. For non-isochronous transfer (bulk or interrupt), the buffer should be completely filled before the data is sent to the host and a switch to the next buffer occurs. The exception is at the end of DMA transfer indicated by activation of EOT_N, when the current buffer content (completely full or not) will be sent to the host.

Remark: 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. The exception is during DMA operation on the main endpoint (endpoint 2), in which case the pointer is automatically pointed to the second buffer after reaching the boundary (double buffering scheme).

11.3.7 Clear buffer

Code (Hex) — F2

Transaction — 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.

11.3.8 Validate buffer

Code (Hex) — FA

Transaction — 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.

11.3.9 Set endpoint status

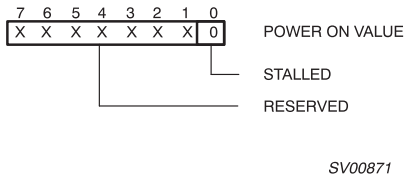
Code (Hex) — 40 to 45

Transaction — 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.

Fig 14. Set endpoint status: bit allocation.

11.3.10 Acknowledge setup

Code (Hex) — F1

Transaction — 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.

11.4 General commands

11.4.1 Send resume

Code (Hex) — F6

Transaction — 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.

11.4.2 Read current frame number

Code (Hex) — F5

Transaction — read 1 or 2 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.

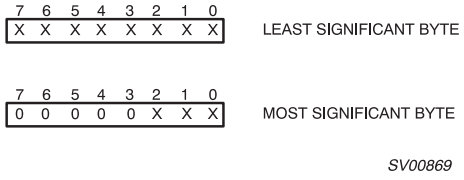


Fig 15. Read current frame number.

12. Interrupt modes

Table 11: Interrupt modes

SOF-ONLY INTERRUPT MODE[1]	INTERRUPT PIN MODE[2]	Interrupt types
0	0	Normal[3]
0	1	Normal + SOF [3]
1	X	SOF only

[1] Bit 7 of Clock division factor byte of Set mode command (see Table 6).

[2] Bit 5 of Set DMA command (see Table 7).

[3] Normal interrupts from Interrupt Register.

13. Limiting values

Table 12: Limiting values

In accordance with the Absolute Maximum Rating System (IEC 60134).

Symbol	Parameter	Conditions		Min	Max	Unit
V_{CC}	supply voltage			–0.5	+6.0	V
V_I	input voltage			–0.5	V_{CC} + 0.5	V
$I_{latchup}$	latchup current	$V_I < 0$ or $V_I > V_{CC}$		–	100	mA
V_{esd}	electrostatic discharge voltage	$I_{LI} < 1$ µA	[1][2]	–	±2000	V
T_{stg}	storage temperature			–60	+150	°C
P_{tot}	total power dissipation	V_{CC} = 5.5 V		–	95	mW

[1] Equivalent to discharging a 100 pF capacitor via a 1.5 kΩ resistor.

[2] Values are given for device only; in-circuit $V_{esd(max)}$ = ±8000 V.

Table 13: Recommended operating conditions

Symbol	Parameter	Conditions	Min	Max	Unit
V_{CC1}	DC supply voltage (Main mode)	apply V_{CC1} to V_{CC} pin only	4.0	5.5	V
V_{CC2}	DC supply voltage (Alternate mode)	apply V_{CC2} to both V_{CC} and $V_{out3.3}$ pins	3.0	3.6	V
V_I	DC input voltage		0	5.5	V
$V_{I/O}$	DC input voltage for I/O		0	5.5	V
$V_{AI/O}$	DC input voltage for analog I/O		0	3.6	V
V_O	DC output voltage		0	V_{CC}	V
T_{amb}	operating ambient temperature in free air	See Section 14 and Section 15 per device.	–40	+85	°C

14. Static characteristics

Table 14: DC characteristics (digital pins)

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
Input levels						
V_{IL}	LOW level input voltage		–	–	0.8	V
V_{IH}	HIGH level input voltage		2.0	–	–	V
V_{HYS}	hysteresis voltage	ST (Schmitt Trigger) pins	0.4	–	0.7	V
Output levels						
V_{OL}	LOW level output voltage	I_{OL} = rated drive	–	–	0.4	V
		I_{OL} = 20 µA			0.1	V
V_{OH}	HIGH level output voltage	I_{OH} = rated drive	2.4	–	–	V
		I_{OH} = 20 µA	V_{CC} – 0.1			V
Leakage current						
I_{OZ}	OFF-state current	OD (Open Drain) pins	–	–	±5	µA
I_L	input leakage current		–	–	±5	µA
I_S	suspend current	oscillator stopped and inputs to GND/V_{CC}	-	-	15	µA
I_O	operating current		-	15		mA

Table 15: DC characteristics (AI/O pins)

Symbol	Parameter	Conditions	Min	Max	Unit
Leakage current					
I_{LO}	Hi-Z state data line leakage	0 V < V_{IN} < 3.3 V	–	±10	µA
Input levels					
V_{DI}	differential input sensitivity	|(D+) – (D–)|	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 1.5 kΩ to 3.6 V	–	0.3	V
V_{OH}	static output HIGH	R_L of 15 kΩ to GND	2.8	3.6	V
Capacitance					
C_{IN}	transceiver capacitance	pin to GND	–	20	pF
Output resistance					
Z_{DRV} [1]	driver output resistance	steady state drive	29	44	Ω
Pull-up resistance					
Z_{PU}	pull-up resistance	SoftConnect = ON	1.1	1.9	kΩ

[1] Includes external resistors of 18 Ω ± 1% on D+ and D–.

15. Dynamic characteristics

Table 16: AC characteristics (AI/O pins; FULL speed)[1]

C_L = 50 pF; R_{PU} = 1.5 kΩ on D+ to V_{CC}; unless otherwise specified.

Symbol	Parameter	Conditions		Min	Max	Unit
Driver characteristics						
t_R	rise time	10% to 90%		4	20	ns
t_F	fall time	10% to 90%		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	see Figure 16		160	175	ns
t_{DEOP}	differential data to EOP transition skew	see Figure 16		–2	+5	ns
Receiver timings:						
t_{JR1}	receiver data jitter tolerance to next transition			–18.5	+18.5	ns
t_{JR2}	receiver data jitter tolerance for paired transitions			–9	+9	ns
t_{EOPR1}	EOP width at receiver	must reject as EOP; see Figure 16	[2]	40	-	ns
t_{EOPR2}	EOP width at receiver	must accept as EOP; see Figure 16	[2]	82	–	ns

[1] Test circuit, see Figure 22.

[2] Characterized but not implemented as production test. Guaranteed by design.

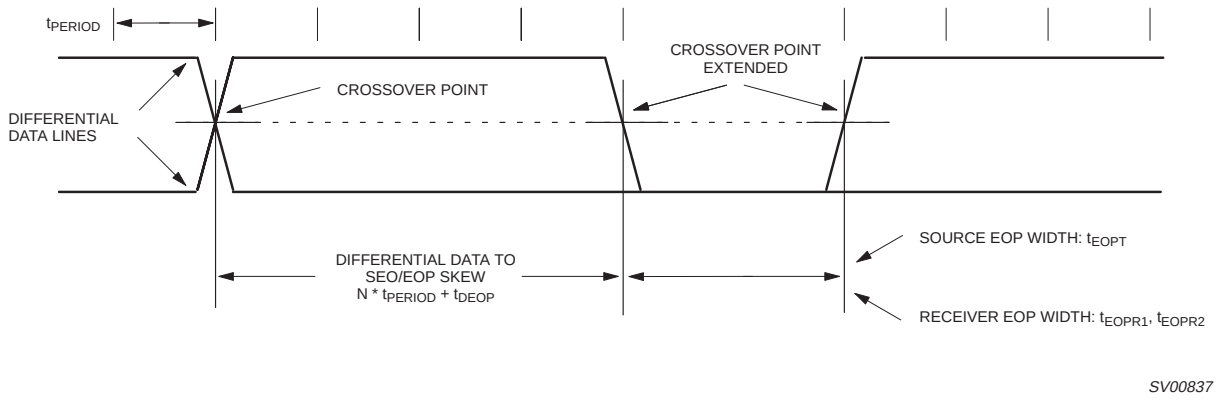


Fig 16. Differential data-to-EOP transition skew and EOP width.

Table 17: AC characteristics (parallel interface)

Symbol	Parameter	Conditions	Min	Max	Unit
ALE timings					
t_{LH}	ALE HIGH pulse width		20	–	ns
t_{AVLL}	address valid to ALE LOW time		10	–	ns
t_{LLAX}	ALE LOW to Address transition time		–	10	ns
Write timings					
t_{CLWL}	CS_N (DMACK_N) LOW to WR_N LOW time		0 [1]	–	ns
t_{WHCH}	WR_N HIGH to CS_N (DMACK_N) HIGH time		5	–	ns
t_{AVWL}	A0 Valid to WR_N LOW time		0 [1]	–	ns
			130 [2]	-	ns
t_{WHAX}	WR_N HIGH to A0 transition time		5	–	ns
t_{WL}	WR_N LOW pulse width		20	–	ns
t_{WDSU}	write data setup time		30	–	ns
t_{WDH}	write data hold time		10	–	ns
t_{WC}	write cycle time		500 [3]	–	ns
$t_{(WC - WD)}$	write command to write data		600	-	ns
Read timings					
t_{CLRL}	CS_N (DMACK_N) LOW to RD_N LOW time		0 [1]	–	ns
			130 [2]	-	ns
t_{RHCH}	RD_N HIGH to CS_N (DMACK_N) HIGH time		5	–	ns
t_{AVRL}	A0 Valid to RD_N LOW time		0 [1]	–	ns
t_{RL}	RD_N LOW pulse width		20	–	ns
t_{RLDD}	RD_N LOW to Data Driven time		–	20	ns
t_{RHDZ}	RD_N HIGH to Data Hi-Z time		–	20	ns
t_{RC}	read cycle time		500 [3]	–	ns
$t_{(WC - RD)}$	write command to read data		600	-	ns

[1] Can be negative.

[2] For DMA access only on the module 64th byte and the second last (EOT-1)byte.

[3] The t_{WC} and t_{RC} timings are valid for back-to-back data access only.

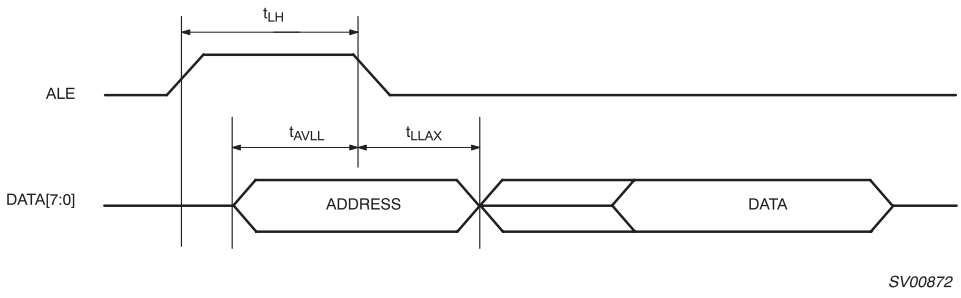


Fig 17. ALE timing.

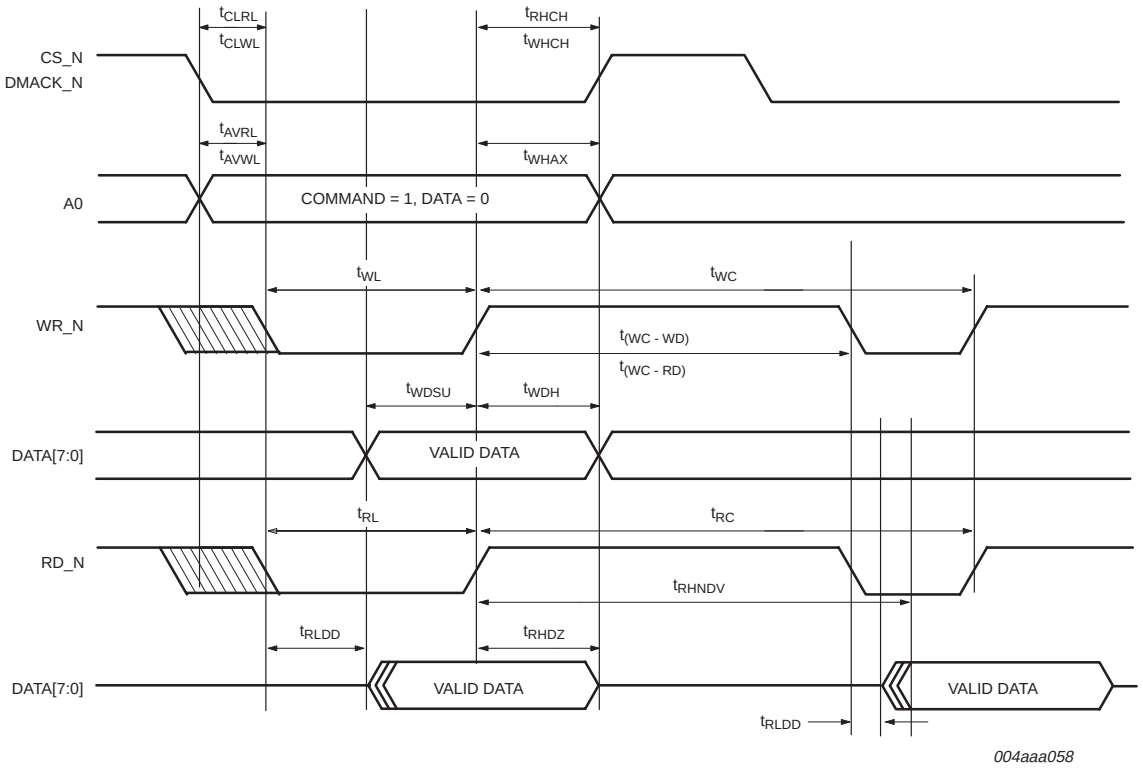
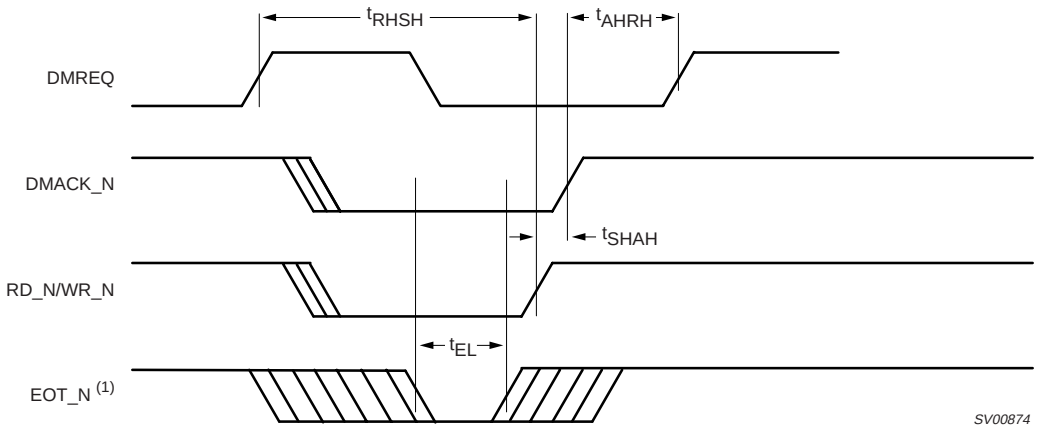


Fig 18. Parallel interface timing (I/O and DMA).

Table 18: AC characteristics (DMA)

Symbol	Parameter	Conditions	Min	Max	Unit
Single-cycle DMA timings					
t_{AHRH}	DMACK_N HIGH to DMREQ HIGH time		–	330	ns
t_{SHAH}	RD_N/WR_N HIGH to DMACK_N HIGH time		130	–	ns
t_{RHSH}	DMREQ HIGH to RD_N/WR HIGH time		120	–	ns
t_{EL}	EOT_N LOW pulse width	simultaneous DMACK_N, RD_N/WR_N and EOT_N LOW time	10	-	ns
Burst DMA timings					
t_{SLRL}	RD_N/WR_N LOW to DMREQ LOW time		-	40	ns
t_{RHNDV}	RD_N (only) HIGH to next data valid		–	420	ns
EOT timings					
t_{ELRL}	EOT_N LOW to DMREQ LOW time		–	40	ns



EOT_N is considered valid when DMACK_N, RD_N/WR_N and EOT_N are all LOW.

Fig 19. Single-cycle DMA timing.

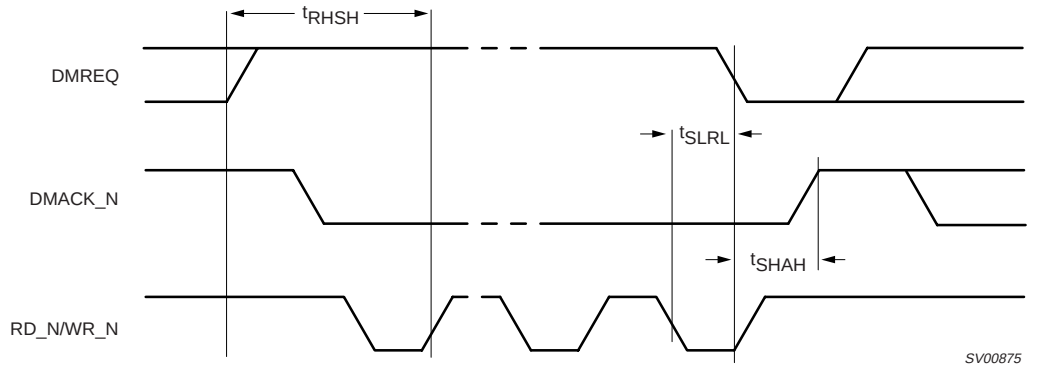


Fig 20. Burst DMA timing.

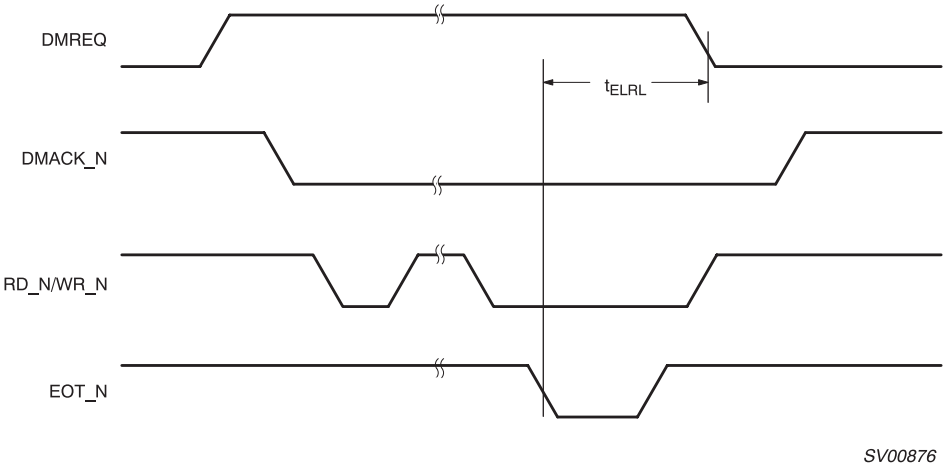


Fig 21. DMA terminated by EOT.

16. Test information

The dynamic characteristics of the analog I/O ports (D+ and D–) as listed in Table 16, were determined using the circuit shown in Figure 22.

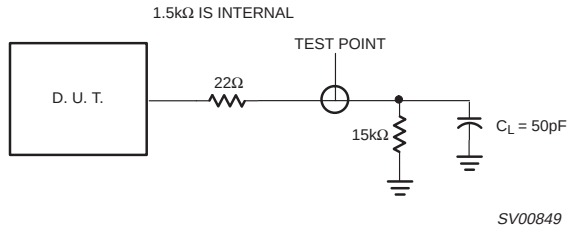
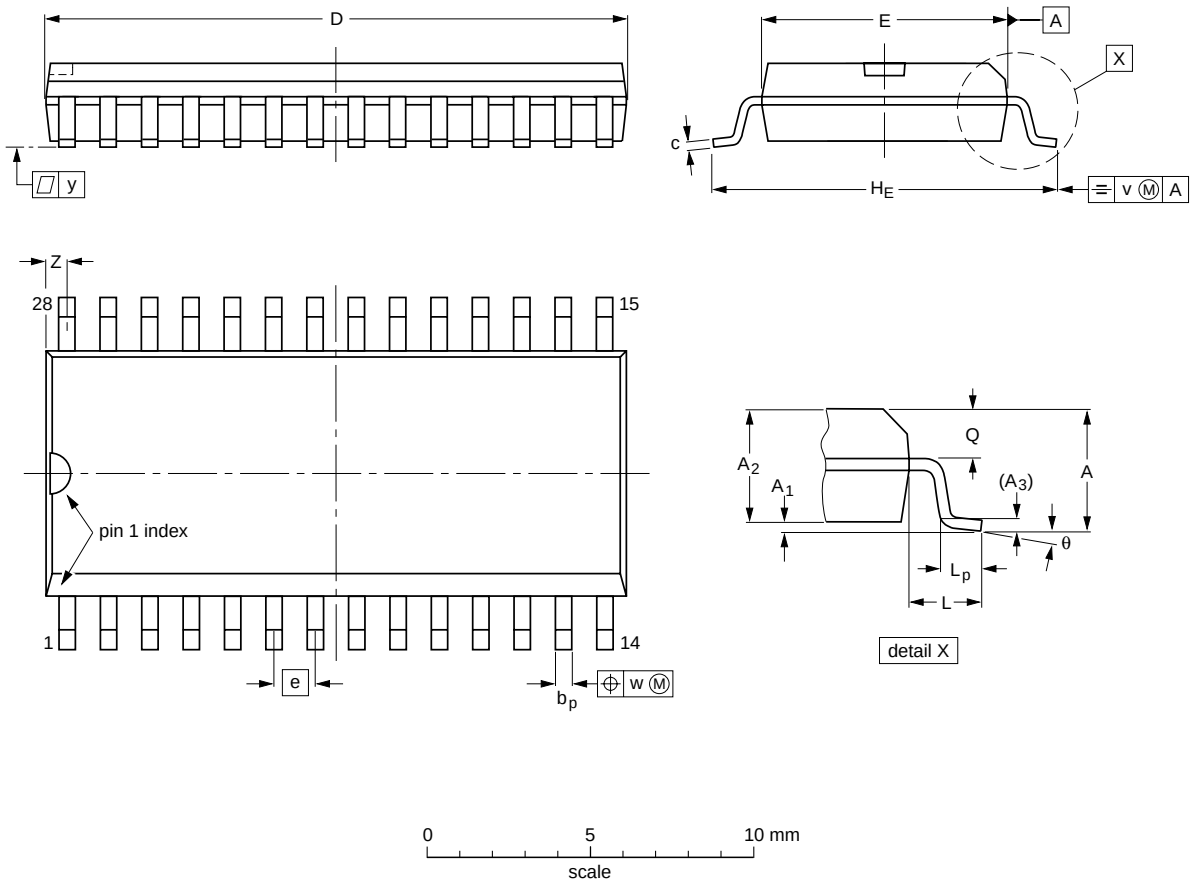


Fig 22. Load for D+/D–.

17. Package outline

SO28: plastic small outline package; 28 leads; body width 7.5 mm **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 0.10	2.45 2.25	0.25	0.49 0.36	0.32 0.23	18.1 17.7	7.6 7.4	1.27	10.65 10.00	1.4	1.1 0.4	1.1 1.0	0.25	0.25	0.1	0.9 0.4	8^o 0^o
inches	0.10	0.012 0.004	0.096 0.089	0.01	0.019 0.014	0.013 0.009	0.71 0.69	0.30 0.29	0.050	0.419 0.394	0.055	0.043 0.016	0.043 0.039	0.01	0.01	0.004	0.035 0.016	

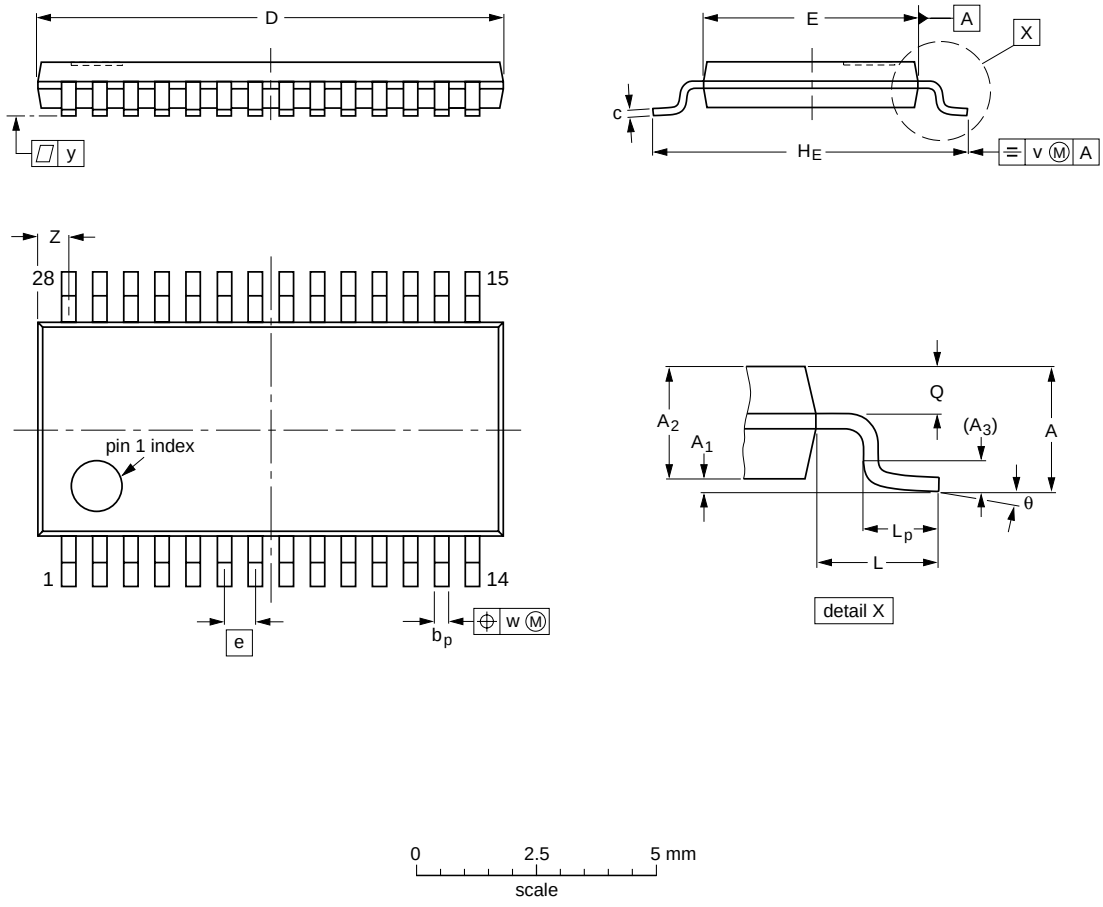
Note

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-013				~~97-05-22~~ 99-12-27

Fig 23. SO28 package outline.

TSSOP28: plastic thin shrink small outline package; 28 leads; body width 4.4 mm **SOT361-1**



DIMENSIONS (mm are the original dimensions)

UNIT	A max.	A_1	A_2	A_3	b_p	c	D (1)	E (2)	e	H_E	L	L_p	Q	v	w	y	Z (1)	θ
mm	1.10	0.15 0.05	0.95 0.80	0.25	0.30 0.19	0.2 0.1	9.8 9.6	4.5 4.3	0.65	6.6 6.2	1.0	0.75 0.50	0.4 0.3	0.2	0.13	0.1	0.8 0.5	8^o 0^o

Notes

1. Plastic or metal protrusions of 0.15 mm maximum per side are not included.
2. Plastic interlead protrusions of 0.25 mm maximum per side are not included.

OUTLINE VERSION	REFERENCES				EUROPEAN PROJECTION	ISSUE DATE
	IEC	JEDEC	EIAJ			
SOT361-1		MO-153				~~95-02-04~~ 99-12-27

Fig 24. TSSOP28 package outline.

18. Soldering

18.1 Introduction to soldering surface mount packages

This text gives a very brief insight to a complex technology. A more in-depth account of soldering ICs can be found in our *Data Handbook IC26; Integrated Circuit Packages* (document order number 9398 652 90011).

There is no soldering method that is ideal for all surface mount IC packages. Wave soldering can still be used for certain surface mount ICs, but it is not suitable for fine pitch SMDs. In these situations reflow soldering is recommended.

18.2 Reflow soldering

Reflow soldering requires solder paste (a suspension of fine solder particles, flux and binding agent) to be applied to the printed-circuit board by screen printing, stencilling or pressure-syringe dispensing before package placement.

Several methods exist for reflowing; for example, convection or convection/infrared heating in a conveyor type oven. Throughput times (preheating, soldering and cooling) vary between 100 and 200 seconds depending on heating method.

Typical reflow peak temperatures range from 215 to 250 °C. The top-surface temperature of the packages should preferable be kept below 220 °C for thick/large packages, and below 235 °C small/thin packages.

18.3 Wave soldering

Conventional single wave soldering is not recommended for surface mount devices (SMDs) or printed-circuit boards with a high component density, as solder bridging and non-wetting can present major problems.

To overcome these problems the double-wave soldering method was specifically developed.

If wave soldering is used the following conditions must be observed for optimal results:

- Use a double-wave soldering method comprising a turbulent wave with high upward pressure followed by a smooth laminar wave.
- For packages with leads on two sides and a pitch (e):
  - larger than or equal to 1.27 mm, the footprint longitudinal axis is **preferred** to be parallel to the transport direction of the printed-circuit board;
  - smaller than 1.27 mm, the footprint longitudinal axis **must** be parallel to the transport direction of the printed-circuit board.

  The footprint must incorporate solder thieves at the downstream end.
- For packages with leads on four sides, the footprint must be placed at a 45° angle to the transport direction of the printed-circuit board. The footprint must incorporate solder thieves downstream and at the side corners.

During placement and before soldering, the package must be fixed with a droplet of adhesive. The adhesive can be applied by screen printing, pin transfer or syringe dispensing. The package can be soldered after the adhesive is cured.

Typical dwell time is 4 seconds at 250 °C. A mildly-activated flux will eliminate the need for removal of corrosive residues in most applications.

18.4 Manual soldering

Fix the component by first soldering two diagonally-opposite end leads. Use a low voltage (24 V or less) soldering iron applied to the flat part of the lead. Contact time must be limited to 10 seconds at up to 300 °C.

When using a dedicated tool, all other leads can be soldered in one operation within 2 to 5 seconds between 270 and 320 °C.

18.5 Package related soldering information

Table 19: Suitability of surface mount IC packages for wave and reflow soldering methods

Package	Soldering method	
	Wave	Reflow[1]
BGA, HBGA, LFBGA, SQFP, TFBGA	not suitable	suitable
HBCC, HLQFP, HSQFP, HSOP, HTQFP, HTSSOP, HVQFN, SMS	not suitable[2]	suitable
PLCC[3], SO, SOJ	suitable	suitable
LQFP, QFP, TQFP	not recommended[3][4]	suitable
SSOP, TSSOP, VSO	not recommended[5]	suitable

[1] All surface mount (SMD) packages are moisture sensitive. Depending upon the moisture content, the maximum temperature (with respect to time) and body size of the package, there is a risk that internal or external package cracks may occur due to vaporization of the moisture in them (the so called popcorn effect). For details, refer to the Drypack information in the *Data Handbook IC26; Integrated Circuit Packages; Section: Packing Methods*.

[2] These packages are not suitable for wave soldering. On versions with the heatsink on the bottom side, the solder cannot penetrate between the printed-circuit board and the heatsink. On versions with the heatsink on the top side, the solder might be deposited on the heatsink surface.

[3] If wave soldering is considered, then the package must be placed at a 45° angle to the solder wave direction. The package footprint must incorporate solder thieves downstream and at the side corners.

[4] Wave soldering is only suitable for LQFP, QFP and TQFP packages with a pitch (e) equal to or larger than 0.8 mm; it is definitely not suitable for packages with a pitch (e) equal to or smaller than 0.65 mm.

[5] Wave soldering is only suitable for SSOP and TSSOP packages with a pitch (e) equal to or larger than 0.65 mm; it is definitely not suitable for packages with a pitch (e) equal to or smaller than 0.5 mm.

19. Revision history

Table 20: Revision history

Rev	Date	CPCN	Description
08	20011220		**Product data; version 8. Supersedes PDIUSBD12_7 of 20011124 (9397 750 08969). Modifications:** • Added new USB logo to indicate PDIUSBD12 as a USB-IF certified product. • In Section 1 “Description” changed *USB specification Rev. 1.1* to *USB specification Rev. 2.0 (basic speed)*. • In Section 2 “Features” changed *Universal Serial Bus specification Rev. 1.1* to *Universal Serial Bus specification Rev. 2.0 (basic speed)*.
07	20011127		**Product data; version 7. Supersedes PDIUSBD12_6 of 20010423 (9397 750 08117).**
06	20010423		**Product data; version 6. Supersedes PDUIUSBD12_5 of 19990108 (9397 750 04979).**

20. Data sheet status

Data sheet status[1]	Product status[2]	Definition
Objective data	Development	This data sheet contains data from the objective specification for product development. Philips Semiconductors reserves the right to change the specification in any manner without notice.
Preliminary data	Qualification	This data sheet contains data from the preliminary specification. Supplementary data will be published at a later date. Philips Semiconductors reserves the right to change the specification without notice, in order to improve the design and supply the best possible product.
Product data	Production	This data sheet contains data from the product specification. Philips Semiconductors reserves the right to make changes at any time in order to improve the design, manufacturing and supply. Changes will be communicated according to the Customer Product/Process Change Notification (CPCN) procedure SNW-SQ-650A.

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

[2] The product status of the device(s) described in this data sheet may have changed since this data sheet was published. The latest information is available on the Internet at URL http://www.semiconductors.philips.com.

21. 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 60134). 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.

22. 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 licence 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.

23. Trademarks

ACPI — is an open industry specification for PC power management, co-developed by Intel Corp., Microsoft Corp. and Toshiba
GoodLink — is a trademark of Koninklijke Philips Electronics N.V.
OnNow — is a trademark of Microsoft Corp.
SoftConnect — is a trademark of Koninklijke Philips Electronics N.V.

Contact information

For additional information, please visit **http://www.semiconductors.philips.com**.
For sales office addresses, send e-mail to: **sales.addresses@www.semiconductors.philips.com**. **Fax: +31 40 27 24825**

9397 750 09238

Contents

Date of release: 20 December 2001 **Document order number: 9397 750 09238**

PHILIPS

Let's make things better.