PDIUSBD12

USB peripheral controller with parallel bus

Rev. 12 — 8 April 2010 **Product data sheet**

1. General description

The PDIUSBD12 is a cost- and feature-optimized USB peripheral controller. 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 wide variety available. This flexibility cuts down development time, risks and costs, by allowing the use of the existing architecture, minimizing firmware investments. This results in the fastest way to develop the most cost-effective USB peripheral solution.

The PDIUSBD12 fully conforms to *Universal Serial Bus Specification Rev. 2.0*, supporting data transfer at full-speed (12 Mbit/s). 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. The PDIUSBD12 is ideally suited for many peripherals, such as printer, scanner, external mass storage (Zip drive) and digital still camera. 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.

It also incorporates features, such as SoftConnect[1], GoodLink[2], 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 peripherals.

2. Features

- Complies with *Universal Serial Bus specification Rev. 2.0*
- Supports data transfer at full-speed (12 Mbit/s)
- High performance USB peripheral controller with integrated SIE, FIFO memory, transceiver and voltage regulator
- Compliant with most device class specifications
- High-speed (2 MB/s) parallel interface to any external microcontroller or microprocessor

1. SoftConnect is a trademark of ST-Ericsson.
2. GoodLink is a trademark of ST-Ericsson.

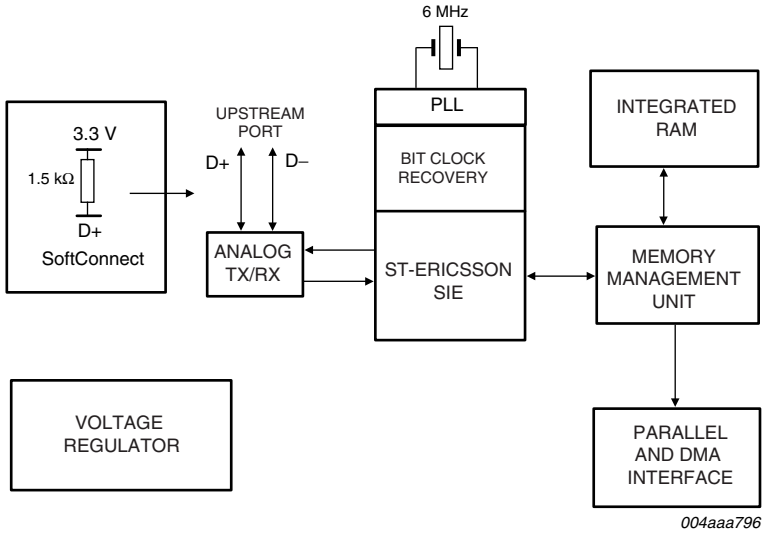
- Fully autonomous DMA operation
- Integrated 320 B of multi-configuration FIFO memory
- Double buffering scheme for main endpoint increases throughput and eases real-time data transfer
- Data transfer rates: 1 MB/s achievable in bulk mode, 1 Mbit/s achievable in isochronous mode
- Bus-powered capability 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 (POR) and low-voltage reset circuit
- Available in TSSOP28 pin package
- Full industrial grade operation from −40 °C to +85 °C
- Full-scan design with high fault coverage (> 99 %) ensures high quality
- Operation with dual voltages: 3.3 V ± 0.3 V or extended 5 V supply range of 4.0 V to 5.5 V
- Multiple interrupt modes to facilitate both bulk and isochronous transfers

3. Ordering information

Table 1. Ordering information

Commercial product code	Package description	Packing	Minimum sellable quantity
PDIUSBD12PWAA	TSSOP28; 28 leads; body width 4.4 mm	single tube dry pack	1275 pieces
PDIUSBD12PWTM	TSSOP28; 28 leads; body width 4.4 mm	13 inch tape and reel non-dry pack	2500 pieces

4. Block diagram



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

Fig 1. Block diagram

5. Pinning information

5.1 Pinning

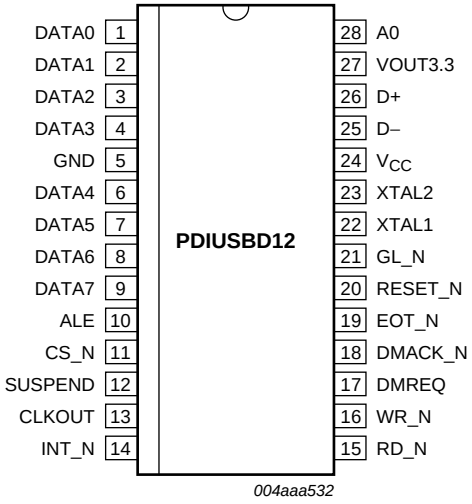


Fig 2. Pin configuration

5.2 Pin description

Table 2. Pin description

Symbol	Pin	Type[1]	Description
DATA0	1	IO2	bit 0 of bidirectional data; slew-rate controlled
DATA1	2	IO2	bit 1 of bidirectional data; slew-rate controlled
DATA2	3	IO2	bit 2 of bidirectional data; slew-rate controlled
DATA3	4	IO2	bit 3 of bidirectional data; slew-rate controlled
GND	5	P	ground
DATA4	6	IO2	bit 4 of bidirectional data; slew-rate controlled
DATA5	7	IO2	bit 5 of bidirectional data; slew-rate controlled
DATA6	8	IO2	bit 6 of bidirectional data; slew-rate controlled
DATA7	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 or data bus. Permanently tied to LOW for separate address or data bus configuration.
CS_N	11	I	chip select (active LOW) When the CS_N pin is LOW, ensure that the RESET_N pin is in inactive state; otherwise, the device will enter test mode.
SUSPEND	12	I, OD4	device is in the 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 is present on-chip, so the pin can be tied HIGH to V_{CC} When the RESET_N pin is LOW, ensure that the CS_N pin is in inactive state; otherwise, the device will enter test mode.
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 the external clock signal, instead of the crystal, is connected to XTAL1, then XTAL2 should be floated
V_{CC}	24	P	voltage supply (4.0 V to 5.5 V) To operate the IC at 3.3 V, supply 3.3 V to both the V_{CC} and VOUT3.3 pins.
D–	25	A	USB D– data line

Table 2. Pin description *…continued*

Symbol	Pin	Type[1]	Description
D+	26	A	USB D+ data line
VOUT3.3	27	P	3.3 V regulated output; to operate the IC at 3.3 V, supply a 3.3 V to both the V_{CC} and VOUT3.3 pins
A0	28	I	address bit **A0 = 1 —** Selects the 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 to HIGH.

[1] P: power or ground; A: analog; I: input; O: Output; 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.

6. Functional description

6.1 Analog transceiver

The integrated transceiver directly interfaces to 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 the SoftConnect technology with an integrated 1.5 kΩ pull-up resistor.

6.3 PLL

A 6 MHz-to-48 MHz clock multiplier Phase-Locked Loop (PLL) is integrated on-chip. This allows the use of a low-cost 6 MHz crystal. ElectroMagnetic Interference (EMI) is also minimized because of 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 × over-sampling principle. It can track jitter and frequency drift specified by *Universal Serial Bus Specification Rev. 2.0*.

6.5 ST-Ericsson serial interface engine

The ST-Ericsson 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 or serial conversion, bit stuffing or discarding stuffed bits, CRC checking or generation, PID verification or generation, address recognition, and handshake evaluation or generation.

6.6 SoftConnect

The connection to the USB is accomplished by connecting D+ (for full-speed USB device) to 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 or 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. The V_{BUS} sensing is provided using pin EOT_N. For details, see Section 5.2. Sharing of the V_{BUS} sensing and EOT_N can be easily accomplished by using the V_{BUS} voltage as the pull-up voltage for the normally open-drain output of the DMA controller pin.

Remark: The tolerance of internal resistors is higher (25 %) than that specified in *Universal Serial Bus Specification Rev. 2.0* (5 %). The overall voltage specification for the connection, however, can still be met with good margin. The decision to make sure of this feature lies with users.

6.7 GoodLink

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

This feature provides a user-friendly indication 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 difference between MMU and the integrated RAM buffer lies in the speed between USB, running in bursts of 12 Mbit/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 and 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-bit address line (occupying two locations). The PDIUSBD12 supports both multiplexed and non-multiplexed address and data bus. The PDIUSBD12 also supports Direct Memory Access (DMA) transfer that 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 to 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 directly connected 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.

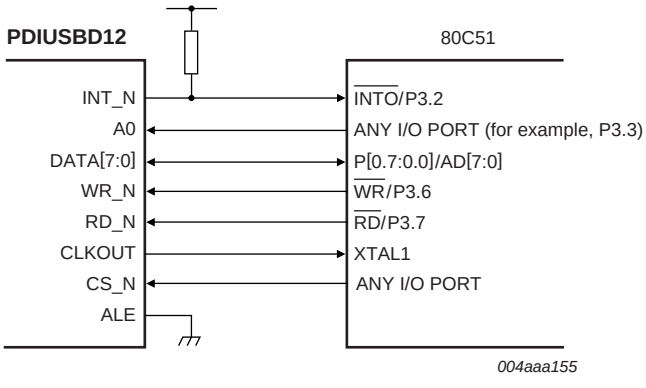


Fig 3. Example of a parallel interface to an 80C51 microcontroller

7. Direct Memory Access (DMA) transfer

DMA allows an efficient transfer of a block of data between the host and local shared memory. Using a DMA Controller (DMAC), the data transfer between the main endpoint (endpoint 2) of the PDIUSBD12 and the local shared memory can occur autonomously, without the local CPU intervention.

Preceding any DMA transfer, the local CPU receives from the host the necessary setup information and accordingly programs the DMA controller. Typically, the DMA controller is set up for demand transfer mode, and the Byte Count register and the address counter are programmed with the correct 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 is 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 acknowledgment by 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 the DMA operation is completed.

For the DMA read operation, the DMREQ pin will only be activated whenever the buffer is full, signaling 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 completely fill up the buffer (less than 64 B or 128 B 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 data is sent to the host. The only exception is at the end of DMA transfer, when the reception of the EOT_N pin will stop the 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 can 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.

The 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 single address mode, the DMA transfer is done using the DREQ, DMACK_N, EOT_N, WR_N and RD_N control lines. In 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 the 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, temporarily storing data 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 four operating modes, depending on the Set Mode command. The four 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 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 the USB access and the local CPU access, increasing throughput. Buffer switching is automatically handled. This results in transparent buffer operation.
- DMA operation. This can be interleaved with normal I/O operation to other endpoints.
- Automatic pointer handling during the 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 B
Set Endpoint Enable	device	D8	write 1 B
Set Mode	device	F3	write 2 B
Set DMA	device	FB	write or read 1 B
Data flow commands			
Read Interrupt register	device	F4	read 2 B
Select Endpoint	control OUT	00	read 1 B (optional)
	control IN	01	read 1 B (optional)
	endpoint 1 OUT	02	read 1 B (optional)
	endpoint 1 IN	03	read 1 B (optional)
	endpoint 2 OUT	04	read 1 B (optional)
	endpoint 2 IN	05	read 1 B (optional)
Read Last Transaction Status	control OUT	40	read 1 B
	control IN	41	read 1 B
	endpoint 1 OUT	42	read 1 B
	endpoint 1 IN	43	read 1 B
	endpoint 2 OUT	44	read 1 B
	endpoint 2 IN	45	read 1 B
Read Buffer	selected endpoint	F0	read n B
Write Buffer	selected endpoint	F0	write n B

Table 4. Command summary *…continued*

Name	Destination	Code (Hex)	Transaction
Set Endpoint Status	control OUT	40	write 1 B
	control IN	41	write 1 B
	endpoint 1 OUT	42	write 1 B
	endpoint 1 IN	43	write 1 B
	endpoint 2 OUT	44	write 1 B
	endpoint 2 IN	45	write 1 B
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 B

11. Command description

11.1 Command procedure

There are three basic types of commands: initialization, data flow and general. 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

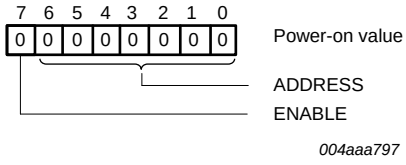
Initialization commands are used during the enumeration process of the USB network. These commands are used to enable function endpoints. They are also used to set the USB assigned address.

11.2.1 Set Address/Enable

Code (Hex) — D0

Transaction — write 1 B

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



ADDRESS: The value written becomes the address.

ENABLE: Logic 1 enables this function.

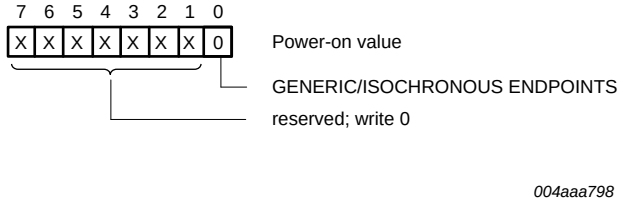
Fig 4. Set Address/Enable command: bit allocation

11.2.2 Set Endpoint Enable

Code (Hex) — D8

Transaction — write 1 B

The generic or isochronous endpoints can only be enabled when the function is enabled using the Set Address/Enable command.



GENERIC OR ISOCHRONOUS ENDPOINTS: Logic 1 indicates that generic or isochronous endpoints are enabled.

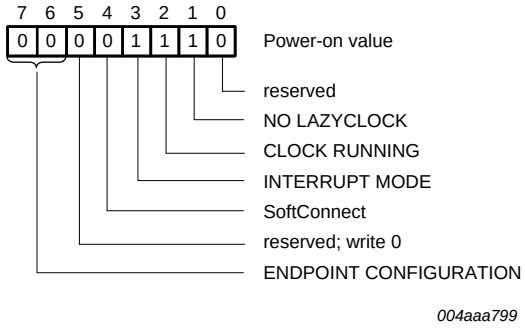
Fig 5. Set Endpoint Enable command: bit allocation

11.2.3 Set Mode

Code (Hex) — F3

Transaction — write 2 B

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

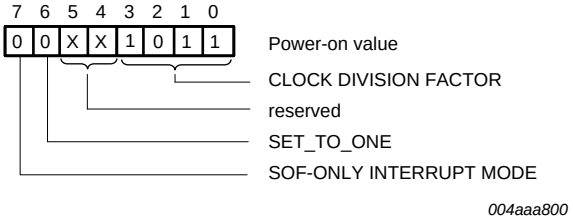


For bit allocation, see Table 5.

Fig 6. Set Mode command, configuration byte: bit allocation

Table 5. Set Mode command, configuration byte: bit allocation

Bit	Symbol	Description
7 to 6	ENDPOINT CONFIGURATION	These two bits set 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) For details, see Section 8.
4	SoftConnect	Logic 1 indicates that the upstream pull-up resistor will be connected if V_{BUS} is available. Logic 0 means that the upstream resistor will not be connected. The programmed value will not be changed by a bus reset.
3	INTERRUPT MODE	Logic 1 indicates that all errors and 'NAK' are reported and will generate an interrupt. Logic 0 indicates that only OK is reported. The programmed value will not be changed by a bus reset.
2	CLOCK RUNNING	Logic 1 indicates that internal clocks and PLL are always running even during the suspend state. Logic 0 indicates that the internal clock, crystal oscillator and PLL are stopped, whenever not needed. To meet the strict suspend current requirement, this bit must be set to logic 0. The programmed value will not be changed by a bus reset.
1	NO LAZYCLOCK	Logic 1 indicates that CLKOUT will not switch to LazyClock. Logic 0 indicates that CLKOUT switches to LazyClock 1 ms after the SUSPEND pin goes HIGH. LazyClock frequency is 30 kHz ± 40 %. The programmed value will not be changed by a bus reset.



For bit allocation, see Table 6.

Fig 7. Set Mode command, clock division factor byte: bit allocation

Table 6. Set Mode command, clock division factor byte: bit allocation

Bit	Symbol	Description
7	SOF-ONLY INTERRUPT MODE	Setting this bit to logic 1 will cause the interrupt line to be activated because of the Start-Of-Frame (SOF) clock only, regardless of the setting of PIN-INTERRUPT MODE, bit 5 of Set DMA.
6	SET_TO_ONE	This bit must be set to logic 1 before any DMA read or DMA write operation. This bit should always be set to logic 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 an output frequency of 4 MHz that can then be programmed up or down by the user. The minimum value is 1, giving a frequency range of 4 MHz 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

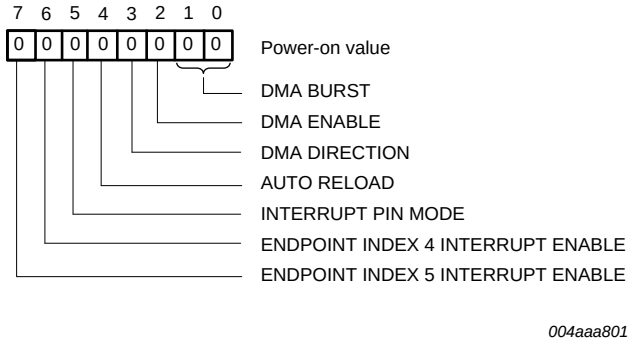
Code (Hex) — FB

Transaction — read or write 1 B

The Set DMA command is followed by one data write or read to or from the DMA Configuration register.

11.2.4.1 DMA Configuration register

During the DMA operation, the 2 B buffer header (status and byte length information) is not transferred to or from the local CPU. This allows the 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 command. For DMA write operations, the header will be automatically added by the PDIUSBD12. This provides a clean and simple DMA data transfer.



For bit allocation, see Table 7.

Fig 8. Set DMA command: bit allocation

Table 7. Set DMA command: bit allocation

Bit	Symbol	Description
7	ENDPOINT INDEX 5 INTERRUPT ENABLE	Logic 1 allows an interrupt to be generated whenever the endpoint buffer is validated (see Section 11.3.8 command). Normally turned off for the DMA operation to reduce unnecessary CPU servicing.
6	ENDPOINT INDEX 4 INTERRUPT ENABLE	Logic 1 allows an interrupt to be generated whenever the endpoint buffer contains a valid packet. Normally turned off for the DMA operation to reduce unnecessary CPU servicing.
5	INTERRUPT PIN MODE	Logic 0 signifies a normal interrupt pin mode in which an interrupt is generated as a logical OR of all the bits in interrupt registers. Logic 1 signifies that the interrupt will occur when SOF clock is seen on the upstream USB bus. The other normal interrupts are still active.
4	AUTO RELOAD	When this bit is set to logic 1, the DMA operation will automatically restart.
3	DMA DIRECTION	This bit determines the direction of data flow during a DMA transfer. Logic 1 means from the external shared memory to the PDIUSBD12 (DMA write); logic 0 means from the PDIUSBD12 to the external shared memory (DMA read).
2	DMA ENABLE	Writing logic 1 to this bit will start the DMA operation through the assertion of pin DMREQ. The main endpoint buffer must be full (for DMA read) or empty (for DMA write), before DMREQ is asserted. In single cycle DMA mode, the DMREQ is deactivated on 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 logic 0 and terminate the DMA operation. The DMA operation can also be terminated by writing logic 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

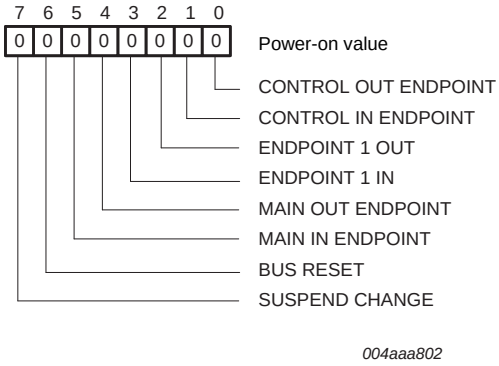
Data flow commands are used to manage the data transmission between USB endpoints and the external microcontroller. Much of the data flow is initiated using an interrupt to the microcontroller. The microcontroller utilizes these commands to access and determine whether endpoint FIFOs have valid data.

11.3.1 Read Interrupt register

Code (Hex) — F4

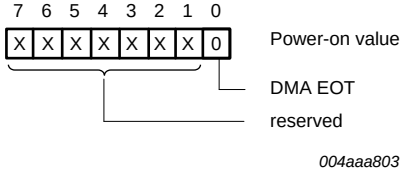
Transaction — read 2 B

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 Interrupt registers.



For bit allocation, see Table 8.

Fig 9. Interrupt register, byte 1: bit allocation



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

Fig 10. Interrupt register, byte 2: bit allocation

Table 8. Read Interrupt register, byte 1: bit allocation

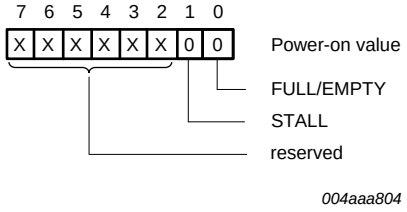
Bit	Symbol	Description
7	SUSPEND CHANGE	When the PDIUSBD12 does not receive three SOFs, it will go into the suspend state and the SUSPEND CHANGE bit will be HIGH. Any change to the suspend or awake state will set this bit to HIGH and generate an interrupt.
6	BUS RESET	After a bus reset, an interrupt will be generated and this bit will be logic 1. A bus reset is identical to a hardware reset through the RESET_N pin, except 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 B (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: Logic 1 indicates that the buffer is full, logic 0 indicates an empty buffer.

STALL: Logic 1 indicates that 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 B

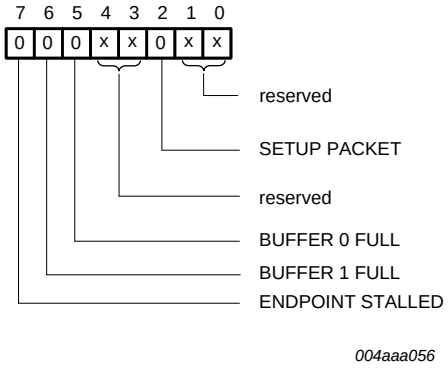


Fig 12. Read Endpoint Status: bit allocation

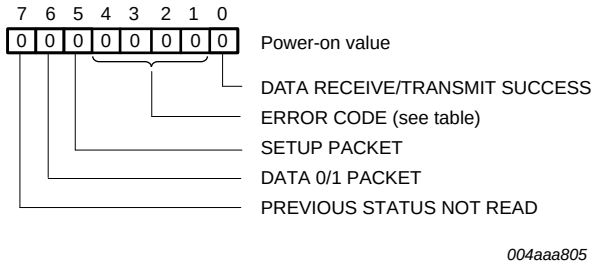
11.3.4 Read Last Transaction Status register

Code (Hex) — 40 to 45

Transaction — read 1 B

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. The status information is overwritten for each new transaction because it keeps track of every transaction.



For bit allocation, see Table 9.

Fig 13. Read Last Transaction Status register: bit allocation

Table 9. Read Last Transaction Status register: bit allocation

Bit	Symbol	Description
7	PREVIOUS STATUS NOT READ	Logic 1 indicates a second event occurred before the previous status was read.
6	DATA0/1 PACKET	Logic 1 indicates the last successful received or sent packet had a DATA1 PID.
5	SETUP PACKET	Logic 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
0	DATA RECEIVE/TRANSMIT SUCCESS	Logic 1 indicates that data has been successfully received or transmitted.

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 (EOP)
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	bit stuff error
1111	wrong DATA PID; the received DATA PID was not what was expected

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 that return contents of the selected endpoint data buffer. After each read, the internal buffer pointer is incremented by 1.

The buffer pointer is 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 is organized as follows:

- Byte 0: reserved; can have any value
- Byte 1: number or length of data bytes
- Byte 2: data byte 1
- Byte 3: data byte 2
- and so on

The first two bytes will be skipped in the DMA read operation. Therefore, the first read will get data byte 1, the second read will get data byte 2, and so on. 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 that load the endpoints buffer. Data must be organized in the same way as described in the Read Buffer command. The first byte (reserved) should always be 0.

During the DMA write operation, the first two bytes will be bypassed. Therefore, the first write will write into data byte 1, the second write will write into data byte 2, and so on. For non-isochronous transfer (bulk or interrupt), the buffer must be completely filled before data is sent to the host and a switch to the next buffer occurs. The exception is at the end of the 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 is only meaningful after a successful transaction. The exception is during the 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 completely received, an internal endpoint buffer full flag is set. All subsequent packets will be refused by returning a NAK. When the microcontroller has read data, it should free the buffer using 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 using the Validate Buffer command. This indicates that data in the buffer is valid and can be sent to the host when the next IN token is received.

11.3.9 Set Endpoint Status

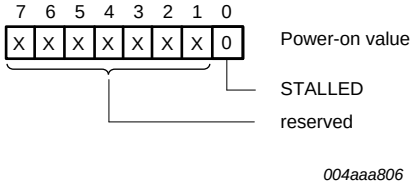
Code (Hex) — 40 to 45

Transaction — write 1 B

A stalled control endpoint is automatically un-stalled 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 un-stalled (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 DATA0 PID, if it is an IN buffer it writes a DATA0 PID.

Even when un-stalled, writing logic 0 to Set Endpoint Status initializes the endpoint.



STALLED: Logic 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 using 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 B

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.

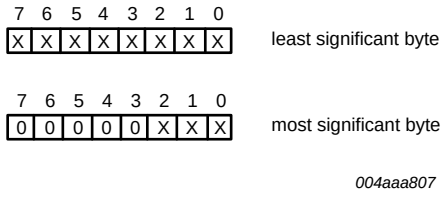


Fig 15. Read Current Frame Number

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 the clock division factor byte of the Set Mode command (see Table 6).

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

[3] Normal interrupts from the 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_{lu}	latch-up current	$V_I < 0$ V or $V_I > V_{CC}$		-	100	mA
V_{esd}	electrostatic discharge voltage	$I_{LI} < 1$ µA	[1]	–2000	+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 through a 1.5 kΩ resistor.

14. Recommended operating conditions

Table 13. Recommended operating conditions

Symbol	Parameter	Conditions		Min	Typ	Max	Unit
V_{CC1}	supply voltage 1	apply V_{CC1} to V_{CC} pin only	[1]	4.0	-	5.5	V
V_{CC2}	supply voltage 2	apply V_{CC2} to both the V_{CC} and VOUT3.3 pins	[2]	3.0	-	3.6	V
V_I	input voltage			0	-	5.5	V
$V_{I/O}$	voltage on an input/output pin			0	-	5.5	V
$V_{IA(I/O)}$	input voltage on analog I/O pins			0	-	3.6	V
V_O	output voltage			0	-	V_{CC}	V
T_{amb}	ambient temperature	see Section 15 and Section 16 per device	[3]	–40	-	+85	°C

[1] Supply voltage (main mode).

[2] Supply voltage (alternate mode).

[3] Operating ambient temperature in free air.

15. Static characteristics

Table 14. Static 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	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

Table 14. Static characteristics (digital pins) *...continued*

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
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 output current	open-drain pins	–5	-	+5	μA
I_L	leakage current		–5	-	+5	μA
I_S	suspend current	oscillator stopped and inputs to GND/V_{CC}	-	-	15	μA
I_{CC}	supply current		-	15		mA

Table 15. Static characteristics (AI/O pins)

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
Leakage current						
$I_{L(dl)}$	data line leakage current	hi-Z; 0 V < V_{IN} < 3.3 V	–10	-	+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}	LOW-level output voltage	R_L of 1.5 kΩ to 3.6 V	-	-	0.3	V
V_{OH}	HIGH-level output voltage	R_L of 15 kΩ to GND	2.8	-	3.6	V
Capacitance						
C_{in}	input capacitance	pin to GND	-	-	20	pF
Output resistance						
Z_{DRV}	driver output impedance	steady state drive [1]	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–.

16. Dynamic characteristics

Table 16. Dynamic characteristics (AI/O pins; full-speed)
C_L = 50 pF; R_{PU} = 1.5 kΩ on D+ to V_{CC}; unless otherwise specified.[1]

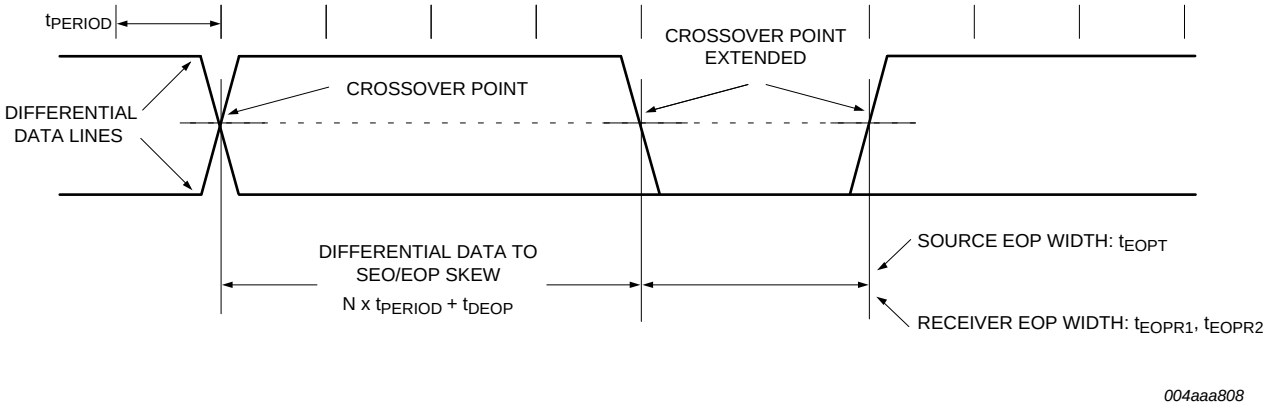
Symbol	Parameter	Conditions	Min	Typ	Max	Unit
Driver characteristics						
t_R	rise time	10 % to 90 %	4	-	20	ns
t_F	fall time	10 % to 90 %	4	-	20	ns
t_{RFM}	rise time/fall time matching (t_R/t_F)		90	-	110	%
V_{CRS}	output signal crossover voltage		1.3	-	2.0	V
Driver timing						
t_{EOPT}	source EOP width	see Figure 16	160	-	175	ns
t_{DEOP}	differential data to EOP transition skew	see Figure 16	–2	-	+5	ns

Table 16. Dynamic characteristics (AI/O pins; full-speed) *…continued*
C_L = 50 pF; R_{PU} = 1.5 kΩ on D+ to V_{CC}; unless otherwise specified.[1]

Symbol	Parameter	Conditions		Min	Typ	Max	Unit
Receiver timing:							
t_{JR1}	receiver data jitter tolerance to next transition			–18.5	-	+18.5	ns
t_{JR2}	receiver data jitter tolerance for paired transition			–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.



t_{PERIOD} is the bit duration corresponding with the USB data rate.

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

Table 17. Dynamic characteristics (parallel interface)

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
ALE timing						
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 timing						
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

Table 17. Dynamic characteristics (parallel interface) *…continued*

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
$t_{(WC-WD)}$	write command to write data		600	-	-	ns
Read timing						
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 the DMA access only on the module 64th byte and the second last (EOT – 1) byte.

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

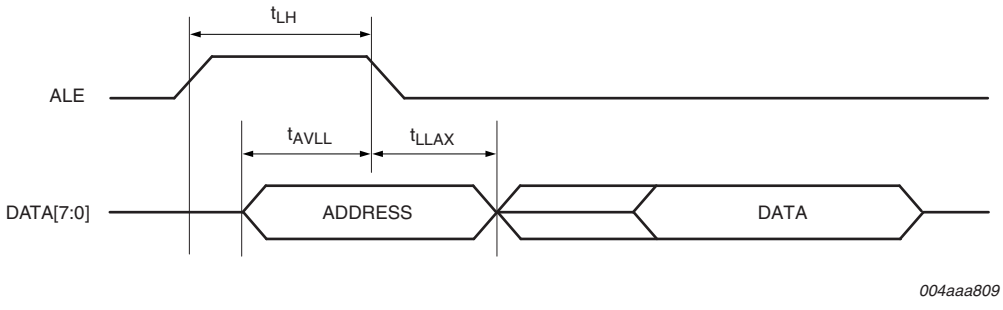


Fig 17. ALE timing

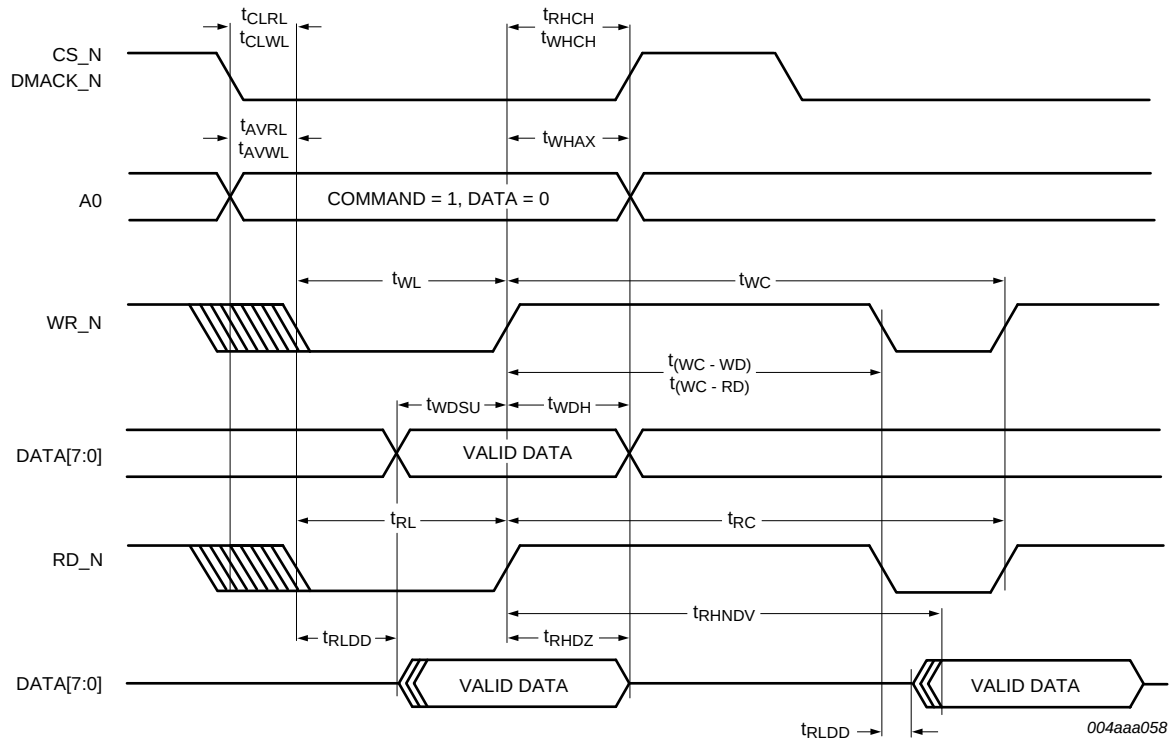
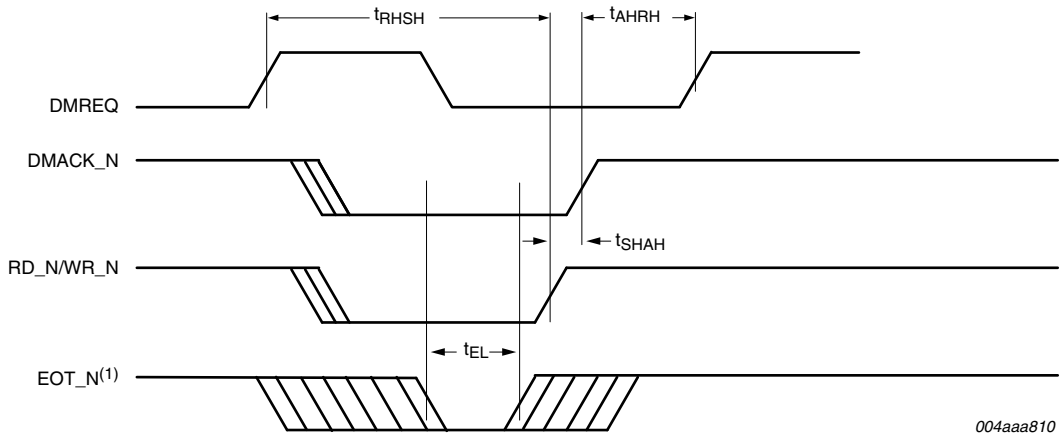


Fig 18. Parallel interface timing (I/O and DMA)

Table 18. Dynamic characteristics (DMA)

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
Single-cycle DMA timing						
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_N 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 timing						
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 timing						
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

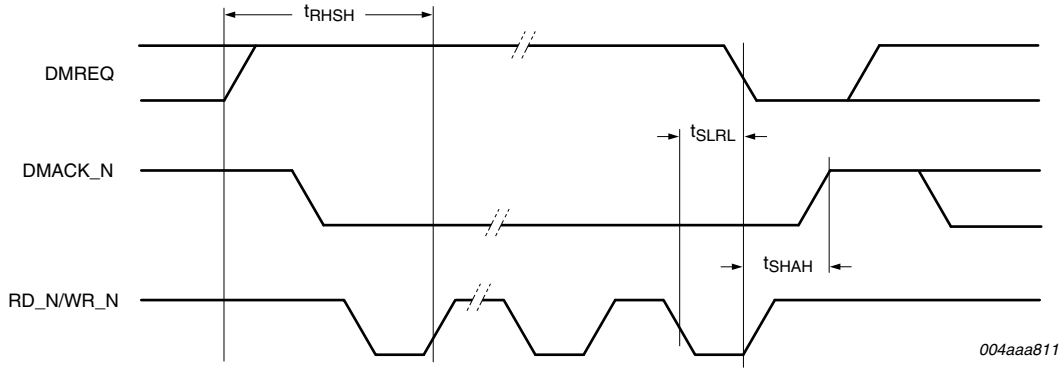


Fig 20. Burst DMA timing

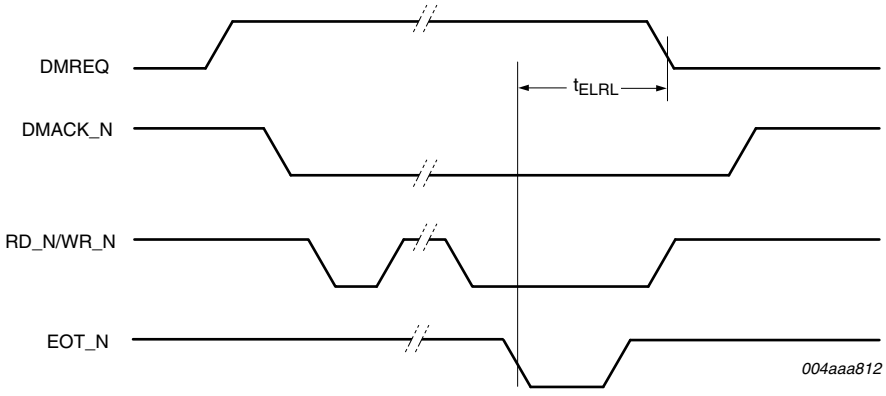


Fig 21. DMA terminated by EOT

17. 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.

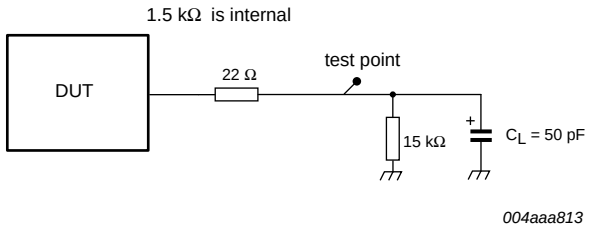
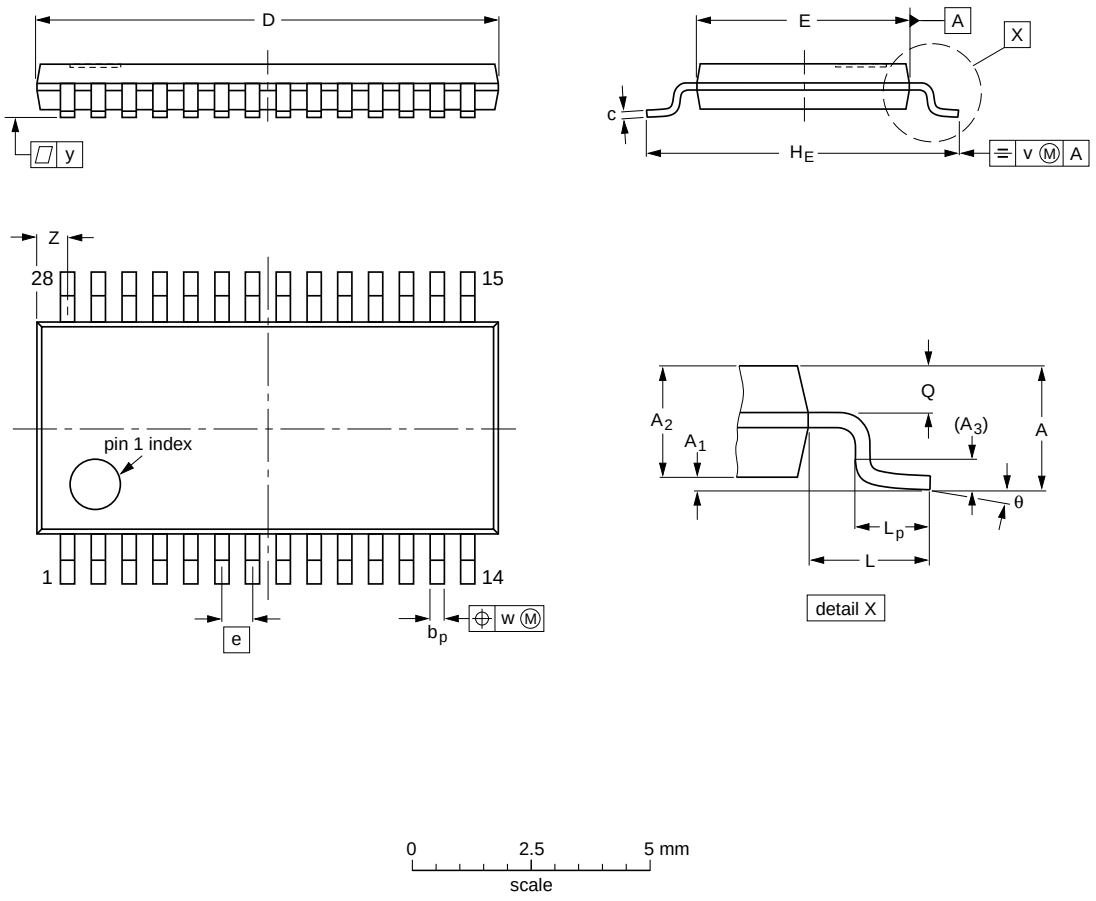


Fig 22. Load for D+ and D–

18. 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	1.1	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	0.75 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	JEITA			
SOT361-1		MO-153				~~99-12-27~~ 03-02-19

Fig 23. Package outline SOT361-1 (TSSOP28)

19. Abbreviations

Table 19. Abbreviations

Acronym	Description
ACPI	Advanced Configuration and Power Interface
CPU	Central Processing Unit
CRC	Cyclic Redundancy Code
DMA	Direct Memory Access
DMAC	Direct Memory Access Controller
EMI	ElectroMagnetic Interference
FIFO	First In, First Out
ISO	Isochronous
MMU	Memory Management Unit
NAK	Not Acknowledged
OD	Open-Drain
PID	Packet Identifier
PLL	Phase-Locked Loop
POR	Power-On Reset
RAM	Random Access Memory
SCSI	Small Computer System Interface
SIE	Serial Interface Engine
SOF	Start-Of-Frame
USB	Universal Serial Bus

20. Revision history

Table 20. Revision history

Revision	Release date	Data sheet status	Change notice
12	20100408	Product data sheet	-
Modifications:	• Updated the filename according to the latest standards. • Section 2 “Features”: removed the SO28 package.		
11	20090929	Product data sheet	-
10	20090123	Product data sheet	-
9	20060511	Product data sheet	-
8 (9397 750 08969)	20011220	Product data	-
7 (9397 750 08117)	20011127	Product data	-
6 (9397 750 04979)	20010423	Product data	-

21. Tables

22. Figures

23. Contents

Contact information at www.stericsson.com under Contacts

www.stericsson.com