# ST-NXP Wireless

## IMPORTANT NOTICE

Dear customer,

As from August 2nd 2008, the wireless operations of NXP have moved to a new company, ST-NXP Wireless.

As a result, the following changes are applicable to the attached document.

- **Company name - NXP B.V.** is replaced with **ST-NXP Wireless**.

- **Copyright** - the copyright notice at the bottom of each page "© NXP B.V. 200x. All rights reserved", shall now read: "© ST-NXP Wireless 200x - All rights reserved".

- **Web site** - http://www.nxp.com is replaced with http://www.stnwireless.com

- **Contact information** - the list of sales offices previously obtained by sending an email to salesaddresses@nxp.com , is now found at http://www.stnwireless.com under Contacts.

If you have any questions related to the document, please contact our nearest sales office. Thank you for your cooperation and understanding.

ST-NXP Wireless

# ISP1583

## Hi-Speed USB peripheral controller

Rev. 07 — 22 September 2008 Product data sheet

## 1. General description

The ISP1583 is a cost-optimized and feature-optimized Hi-Speed Universal Serial Bus (USB) peripheral controller. It fully complies with Ref. 1 “Universal Serial Bus Specification Rev. 2.0”, supporting data transfer at high-speed (480 Mbit/s) and full-speed (12 Mbit/s).

The ISP1583 provides high-speed USB communication capacity to systems based on microcontrollers or microprocessors. It communicates with a microcontroller or microprocessor of a system through a high-speed general-purpose parallel interface.

The ISP1583 supports automatic detection of Hi-Speed USB system operation. Original USB fall-back mode allows the device to remain operational under full-speed conditions. It is designed as a generic USB peripheral controller so that it can fit into all existing device classes, such as imaging class, mass storage devices, communication devices, printing devices and human interface devices.

The ISP1583 is a low-voltage device, which supports I/O pad voltages from 1.65 V to 3.6 V.

The internal generic Direct Memory Access (DMA) block allows easy integration into data streaming applications. In addition, the various configurations of the DMA block are tailored for mass storage applications.

The modular approach to implementing a USB peripheral controller allows the designer to select the optimum system microcontroller from the wide variety available. The ability to reuse existing architecture and firmware shortens the development time, eliminates risk and reduces cost. The result is fast and efficient development of the most cost-effective USB peripheral solution.

The ISP1583 also incorporates features such as SoftConnect, a reduced frequency crystal oscillator and integrated termination resistors. These features allow significant cost savings in system design and easy implementation of advanced USB functionality into PC peripherals.

## 2. Features

- Complies fully with:
  - Ref. 1 “Universal Serial Bus Specification Rev. 2.0”
  - Most device class specifications
  - ACPI, OnNow and USB power management requirements
- Supports data transfer at high-speed (480 Mbit/s) and full-speed (12 Mbit/s)
- Direct interface to ATA/ATAPI peripherals; applicable only in split bus mode
- High performance USB peripheral controller with integrated Serial Interface Engine (SIE), Parallel Interface Engine (PIE), FIFO memory and data transceiver

- Automatic Hi-Speed USB mode detection and Original USB fall-back mode
- Supports sharing mode
- Supports I/O voltage range of 1.65 V to 3.6 V
- Supports $V_{BUS}$ sensing
- High-speed DMA interface
- Configurable direct access data path from the microprocessor to an ATA device
- Fully autonomous and multiconfiguration DMA operation
- Seven IN endpoints, seven OUT endpoints, and a fixed control IN and OUT endpoint
- Integrated physical 8 kB of multiconfiguration FIFO memory
- Endpoints with double buffering to increase throughput and ease real-time data transfer
- Bus-independent interface with most microcontrollers and microprocessors
- 12 MHz crystal oscillator with integrated PLL for low EMI
- Software-controlled connection to the USB bus (SoftConnect)
- Low-power consumption in operation and power-down modes; suitable for use in bus-powered USB devices
- Supports Session Request Protocol (SRP) that adheres to Ref. 2 “On-The-Go Supplement to the USB Specification Rev. 1.3”
- Internal power-on and low-voltage reset circuits; also supports software reset
- Operation over the extended USB bus voltage range (DP, DM and $V_{BUS}$)
- 5 V tolerant I/O pads
- Operating temperature range from –40 °C to +85 °C
- Available in HVQFN64 and TFBGA64 halogen-free and lead-free packages

## 3. Applications

- Personal digital assistant
- Mass storage device, for example: CD, DVD, Magneto-Optical (MO) and Zip drives
- Digital video camera
- Digital still camera
- 3G mobile phone
- MP3 player
- Communication device, for example: router and modem
- Printer
- Scanner

# 4. Ordering information

**Table 1. Ordering information**

| Type number | Package | | |
|---|---|---|---|
| | Name | Description | Version |
| ISP1583BS | HVQFN64 | plastic thermal enhanced very thin quad flat package; no leads; 64 terminals; body 9 × 9 × 0.85 mm | SOT804-1 |
| ISP1583ET | TFBGA64 | plastic thin fine-pitch ball grid array package; 64 balls; body 6 × 6 × 0.8 mm | SOT543-1 |
| ISP1583ET1[1] | TFBGA64 | plastic thin fine-pitch ball grid array package; 64 balls; body 4 × 4 × 0.8 mm | SOT969-1 |
| ISP1583ET2[1] | TFBGA64 | plastic thin fine-pitch ball grid array package; 64 balls; body 6 × 6 × 0.8 mm | SOT543-1 |

[1] Contains solder ball material SAC105.

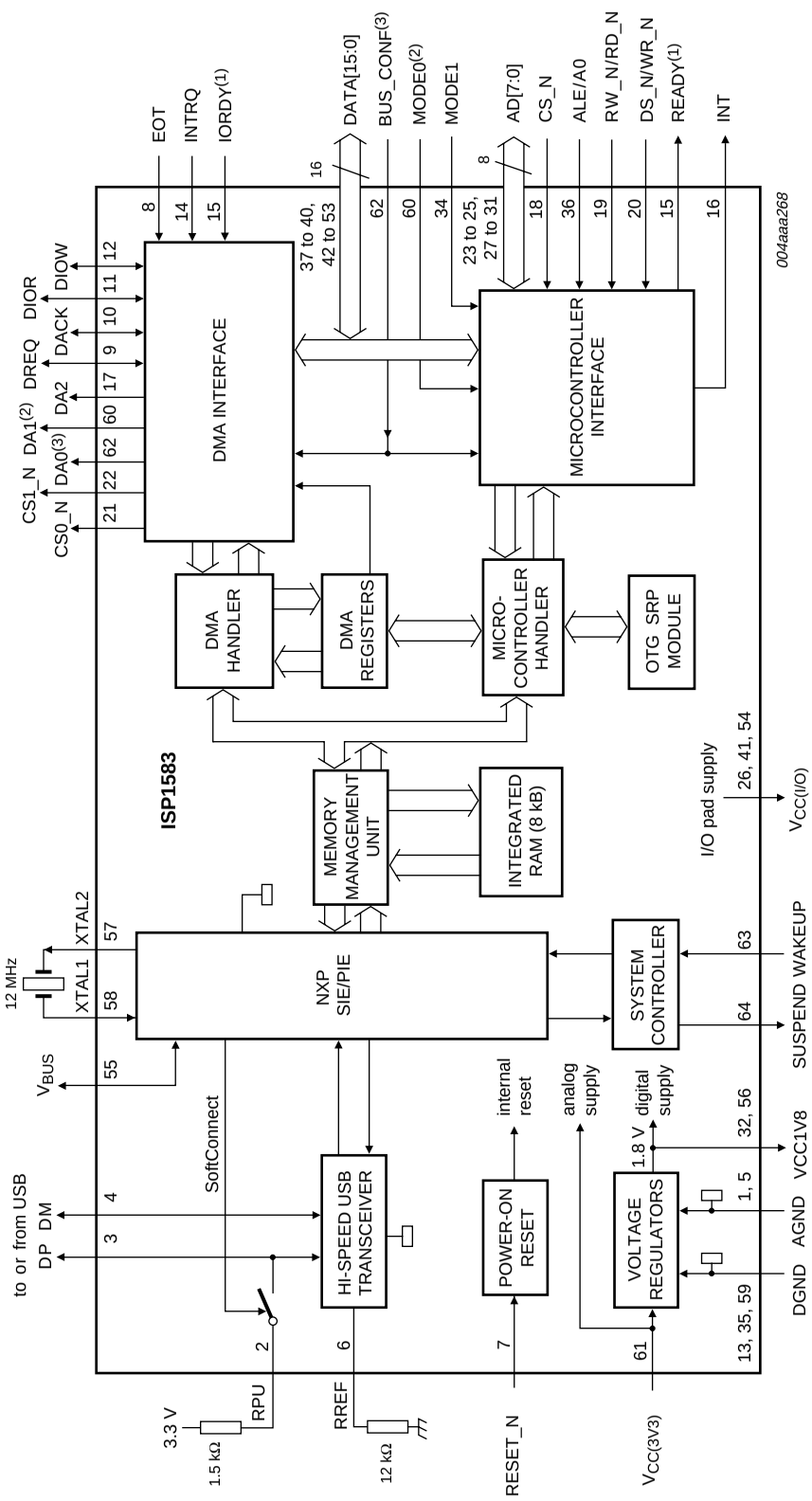
# 5. Marking

**Table 2. Marking codes**

| Type number | Marking code[1] |
|---|---|
| ISP1583BS | ISP1583BS |
| ISP1583ET | ISP1583 |
| ISP1583ET1 | 1583 |
| ISP1583ET2 | 1583ET2 |

[1] The package marking is the first line of text on the IC package and can be used for IC identification.

## 6. Block diagram




The figure shows the ISP1583BS pinout. For the ISP1583ET1, ISP1583ET2 and ISP1583ET2 ballouts, see Table 3.

The direction of pins DREQ, DACK, DIOR and DIOW is determined by bit MASTER (DMA Hardware register) and bit ATA_MODE (DMA Configuration register).

(1) Pin 15 is shared by READY and IORDY.

(2) Pin 60 is shared by MODE0 and DA1.

(3) Pin 62 is shared by BUS_CONF and DA0.

**Fig 1. Block diagram**

# 7. Pinning information

## 7.1 Pinning

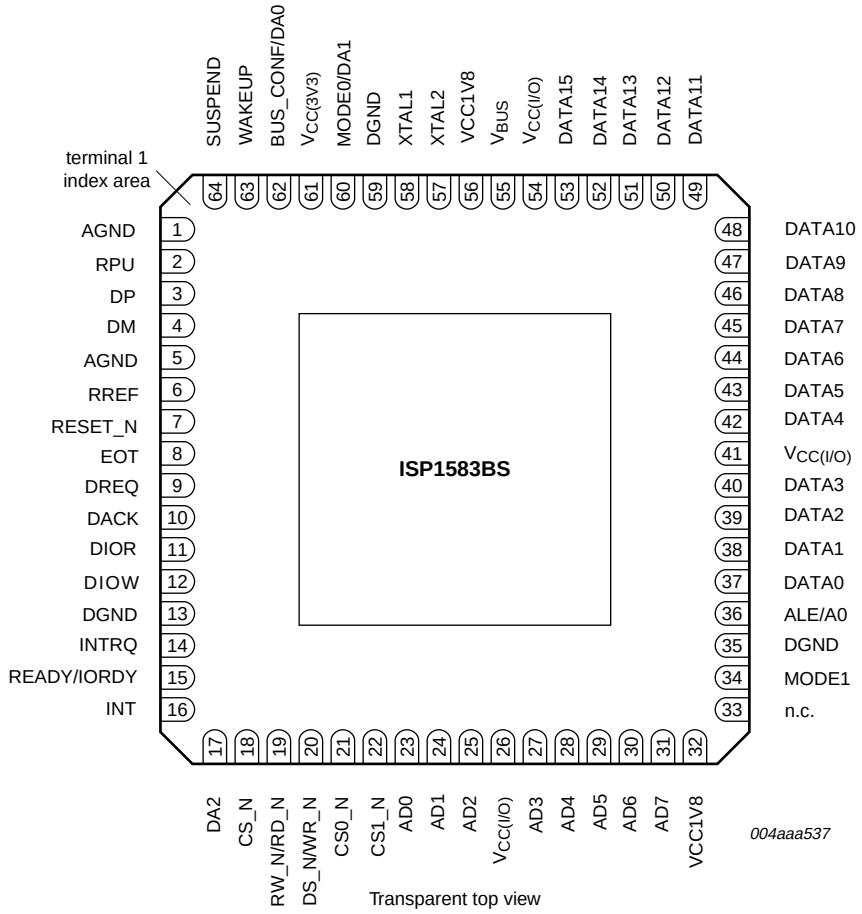



**Fig 2. Pin configuration ISP1583BS (top view)**

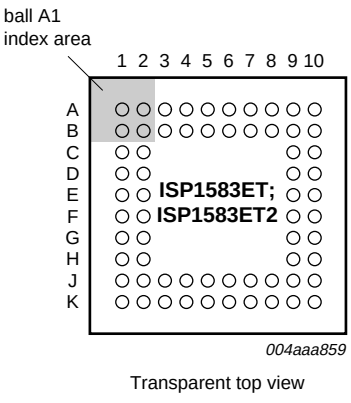



**Fig 3. Pin configuration ISP1583ET and ISP1583ET2 (top view)**

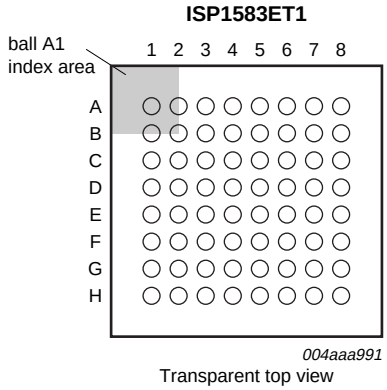



**Fig 4. Pin configuration ISP1583ET1 (top view)**

## 7.2 Pin description

**Table 3. Pin description**

| Symbol[1] | Pin | | | Type[2] | Description |
|---|---|---|---|---|---|
| | ISP1583BS | ISP1583ET; ISP1583ET2 | ISP1583ET1 | | |
| AGND | 1 | D2 | D2 | - | analog ground |
| RPU | 2 | A1 | A1 | A | pull-up resistor connection; this pin must be connected to 3.3 V through an external 1.5 kΩ resistor to pull up pin DP |
| DP | 3 | B1 | B1 | A | USB D+ line connection (analog) |
| DM | 4 | C1 | C1 | A | USB D– line connection (analog) |
| AGND | 5 | - | - | - | analog ground |
| RREF | 6 | D1 | D1 | A | external bias resistor connection; this pin must be connected to ground through a 12.0 kΩ ± 1 % resistor |
| RESET_N | 7 | E2 | C3 | I | reset input (500 µs); a LOW level produces an asynchronous reset; connect to $V_{CC(3V3)}$ for power-on reset (internal POR circuit)<br>When the RESET_N pin is LOW, ensure that the WAKEUP pin does not go from LOW to HIGH; otherwise the device will enter test mode.<br>TTL; 5 V tolerant |
| EOT | 8 | E1 | D3 | I | end-of-transfer input (programmable polarity); used in DMA slave mode only; when not in use, connect this pin to $V_{CC(I/O)}$ through a 10 kΩ resistor<br>input pad; TTL; 5 V tolerant |
| DREQ | 9 | F2 | E1 | I/O | DMA request input or output (programmable polarity); the signal direction depends on bit MASTER in register DMA Hardware (see Table 59):<br>• Input: DMA master mode if bit MASTER = 1<br>• Output: DMA slave mode if bit MASTER = 0<br>When not in use, in the default setting, this pin must be connected to ground through a 10 kΩ resistor.<br>bidirectional pad; 4 ns slew-rate control; TTL; 5 V tolerant |

**Table 3. Pin description** *...continued*

| Symbol[1] | Pin | | | Type[2] | Description |
|---|---|---|---|---|---|
| | ISP1583BS | ISP1583ET; ISP1583ET2 | ISP1583ET1 | | |
| DACK | 10 | F1 | E2 | I/O | DMA acknowledge input or output (programmable polarity); the signal direction depends on bit MASTER in register DMA Hardware (see Table 59):<br>• Input: DMA slave mode if bit MASTER = 0<br>• Output: DMA master mode if bit MASTER = 1<br>When not in use, in the default setting, this pin must be connected to $V_{CC(I/O)}$ through a 10 kΩ resistor.<br>bidirectional pad; 4 ns slew-rate control; TTL; 5 V tolerant |
| DIOR | 11 | G2 | E3 | I/O | DMA read strobe input or output (programmable polarity); the signal direction depends on bit MASTER in register DMA Hardware (see Table 59):<br>• Input: DMA slave mode if bit MASTER = 0<br>• Output: DMA master mode if bit MASTER = 1<br>When not in use, in the default setting, this pin must be connected to $V_{CC(I/O)}$ through a 10 kΩ resistor.<br>bidirectional pad; 4 ns slew-rate control; TTL; 5 V tolerant |
| DIOW | 12 | G1 | F1 | I/O | DMA write strobe input or output (programmable polarity); the signal direction depends on bit MASTER in register DMA Hardware (see Table 59):<br>• Input: DMA slave mode if bit MASTER = 0<br>• Output: DMA master mode if bit MASTER = 1<br>When not in use, in the default setting, this pin must be connected to $V_{CC(I/O)}$ through a 10 kΩ resistor.<br>bidirectional pad; 4 ns slew-rate control; TTL; 5 V tolerant |
| DGND | 13 | H2 | F2 | - | digital ground |
| INTRQ | 14 | H1 | G2 | I | interrupt request input; from the ATA/ATAPI peripheral; use a 10 kΩ resistor to pull down<br>input pad; TTL; 5 V tolerant |
| READY/ IORDY | 15 | J1 | G1 | I/O | **Signal ready output —** Used in generic processor mode:<br>• LOW: the ISP1583 is processing a previous command or data and is not ready for the next command or data transfer<br>• HIGH: the ISP1583 is ready for the next microprocessor read or write<br>**I/O ready input —** Used in split bus mode to access ATA/ATAPI peripherals (PIO mode only)<br>bidirectional pad; 10 ns slew-rate control; TTL; 5 V tolerant |
| INT | 16 | K1 | H1 | O | interrupt output; programmable polarity (active HIGH or LOW) and signaling (edge or level triggered)<br>CMOS output; 8 mA drive |
| DA2[3] | 17 | J2 | H2 | O | address output to select the Task File register of an ATA/ATAPI device; see Table 61<br>CMOS output; 8 mA drive |
| CS_N | 18 | K2 | E4 | I | chip selection input<br>input pad; TTL; 5 V tolerant |

**Table 3. Pin description** *...continued*

| Symbol[1] | Pin | | | Type[2] | Description |
|---|---|---|---|---|---|
| | ISP1583BS | ISP1583ET; ISP1583ET2 | ISP1583ET1 | | |
| RW_N/ RD_N | 19 | J3 | F3 | I | **Read or write input —** For the Freescale mode, this function is determined by pin MODE0 = LOW during power-up.<br>**Read input —** For the 8051 mode, this function is determined by pin MODE0 = HIGH during power-up.<br>input pad; TTL; 5 V tolerant |
| DS_N/ WR_N | 20 | K3 | H3 | I | **Data selection input —** For the Freescale mode, this function is determined by pin MODE0 = LOW at power-up.<br>**Write input —** For the 8051 mode, this function is determined by pin MODE0 = HIGH at power-up.<br>input pad; TTL; 5 V tolerant |
| CS0_N[3] | 21 | J4 | G3 | O | chip selection output 0 for the ATA/ATAPI device; see Table 61<br>CMOS output; 8 mA drive |
| CS1_N[3] | 22 | K4 | F4 | O | chip selection output 1 for the ATA/ATAPI device; see Table 61<br>CMOS output; 8 mA drive |
| AD0 | 23 | K5 | G4 | I/O | bit 0 of the multiplexed address and data<br>bidirectional pad; 4 ns slew-rate control; TTL; 5 V tolerant |
| AD1 | 24 | J5 | H4 | I/O | bit 1 of the multiplexed address and data bus<br>bidirectional pad; 4 ns slew-rate control; TTL; 5 V tolerant |
| AD2 | 25 | K6 | F5 | I/O | bit 2 of the multiplexed address and data bus<br>bidirectional pad; 4 ns slew-rate control; TTL; 5 V tolerant |
| $V_{CC(I/O)}$[4] | 26 | J6 | E5 | - | I/O pad supply voltage (1.65 V to 3.6 V); see Section 8.16 |
| AD3 | 27 | K7 | H5 | I/O | bit 3 of the multiplexed address and data bus<br>bidirectional pad; 4 ns slew-rate control; TTL; 5 V tolerant |
| AD4 | 28 | J7 | G5 | I/O | bit 4 of the multiplexed address and data bus<br>bidirectional pad; 4 ns slew-rate control; TTL; 5 V tolerant |
| AD5 | 29 | K8 | G6 | I/O | bit 5 of the multiplexed address and data bus<br>bidirectional pad; 4 ns slew-rate control; TTL; 5 V tolerant |
| AD6 | 30 | J8 | H6 | I/O | bit 6 of the multiplexed address and data bus<br>bidirectional pad; 4 ns slew-rate control; TTL; 5 V tolerant |
| AD7 | 31 | K9 | H7 | I/O | bit 7 of the multiplexed address and data bus<br>bidirectional pad; 4 ns slew-rate control; TTL; 5 V tolerant |
| VCC1V8[4] | 32 | K10 | H8 | - | voltage regulator output (1.8 V ± 0.15 V); tapped out voltage from the internal regulator; this regulated voltage cannot drive external devices; decouple this pin using a 0.1 µF capacitor; see Section 8.16 |
| n.c. | 33 | - | - | - | not connected |

**Table 3. Pin description** *...continued*

| Symbol[1] | Pin | | | Type[2] | Description |
|---|---|---|---|---|---|
| | ISP1583BS | ISP1583ET; ISP1583ET2 | ISP1583ET1 | | |
| MODE1 | 34 | J10 | G8 | I | mode selection input 1; used in split bus mode only:<br>• LOW: ALE function (address latch enable)<br>• HIGH (connect to $V_{CC(I/O)}$): A0 function (address/data indicator)<br>**Remark:** When operating in generic processor mode, set pin MODE1 as HIGH.<br>input pad; TTL; 5 V tolerant |
| DGND | 35 | H9 | F8 | - | digital ground |
| ALE/A0 | 36 | H10 | F7 | I | **Address latch enable input —** When pin MODE1 = LOW during power-up, a falling edge on this pin latches the address on the multiplexed address and data bus AD[7:0].<br>**Address and data selection input —** When pin MODE1 = HIGH during power-up, the function is determined by the level on this pin (detected on the rising edge of the WR_N pulse):<br>• HIGH: bus AD[7:0] is a register address<br>• LOW: bus AD[7:0] is register data; used in split bus mode only<br>**Remark:** When operating in generic processor mode with pin MODE1 = HIGH, this pin must be pulled down using a 10 kΩ resistor.<br>input pad; TTL; 5 V tolerant |
| DATA0 | 37 | G9 | F6 | I/O | bit 0 of the bidirectional data bus<br>bidirectional pad; 4 ns slew-rate control; TTL; 5 V tolerant |
| DATA1 | 38 | G10 | E6 | I/O | bit 1 of the bidirectional data bus<br>bidirectional pad; 4 ns slew-rate control; TTL; 5 V tolerant |
| DATA2 | 39 | F9 | E7 | I/O | bit 2 of the bidirectional data bus<br>bidirectional pad; 4 ns slew-rate control; TTL; 5 V tolerant |
| DATA3 | 40 | F10 | E8 | I/O | bit 3 of the bidirectional data bus<br>bidirectional pad; 4 ns slew-rate control; TTL; 5 V tolerant |
| $V_{CC(I/O)}$[4] | 41 | E9 | D7 | - | I/O pad supply voltage (1.65 V to 3.6 V); see Section 8.16 |
| DATA4 | 42 | E10 | D6 | I/O | bit 4 of the bidirectional data bus<br>bidirectional pad; 4 ns slew-rate control; TTL; 5 V tolerant |
| DATA5 | 43 | D10 | D8 | I/O | bit 5 of the bidirectional data bus<br>bidirectional pad; 4 ns slew-rate control; TTL; 5 V tolerant |
| DATA6 | 44 | D9 | D5 | I/O | bit 6 of the bidirectional data bus<br>bidirectional pad; 4 ns slew-rate control; TTL; 5 V tolerant |
| DATA7 | 45 | C10 | D4 | I/O | bit 7 of the bidirectional data bus<br>bidirectional pad; 4 ns slew-rate control; TTL; 5 V tolerant |
| DATA8 | 46 | C9 | C8 | I/O | bit 8 of the bidirectional data bus<br>bidirectional pad; 4 ns slew-rate control; TTL; 5 V tolerant |
| DATA9 | 47 | B10 | C7 | I/O | bit 9 of the bidirectional data bus<br>bidirectional pad; 4 ns slew-rate control; TTL; 5 V tolerant |

**Table 3. Pin description** *...continued*

| Symbol[1] | Pin | | | Type[2] | Description |
|---|---|---|---|---|---|
| | ISP1583BS | ISP1583ET; ISP1583ET2 | ISP1583ET1 | | |
| DATA10 | 48 | A10 | A8 | I/O | bit 10 of the bidirectional data bus<br>bidirectional pad; 4 ns slew-rate control; TTL; 5 V tolerant |
| DATA11 | 49 | A9 | C4 | I/O | bit 11 of the bidirectional data bus<br>bidirectional pad; 4 ns slew-rate control; TTL; 5 V tolerant |
| DATA12 | 50 | B8 | B7 | I/O | bit 12 of the bidirectional data bus<br>bidirectional pad; 4 ns slew-rate control; TTL; 5 V tolerant |
| DATA13 | 51 | A8 | A7 | I/O | bit 13 of the bidirectional data bus<br>bidirectional pad; 4 ns slew-rate control; TTL; 5 V tolerant |
| DATA14 | 52 | B7 | C6 | I/O | bit 14 of the bidirectional data bus<br>bidirectional pad; 4 ns slew-rate control; TTL; 5 V tolerant |
| DATA15 | 53 | A7 | C5 | I/O | bit 15 of the bidirectional data bus<br>bidirectional pad; 4 ns slew-rate control; TTL; 5 V tolerant |
| $V_{CC(I/O)}$[4] | 54 | B6 | B6 | - | I/O pad supply voltage (1.65 V to 3.6 V); see Section 8.16 |
| $V_{BUS}$ | 55 | B5 | A6 | A | **USB bus power sensing input —** Used to detect whether the host is connected or not; connect a 1 µF electrolytic or tantalum capacitor and a 1 MΩ pull-down resistor to ground; see Section 8.14<br>**$V_{BUS}$ pulsing output —** In OTG mode; connect a 1 µF electrolytic or tantalum capacitor and a 1 MΩ pull-down resistor to ground; see Section 8.14<br>5 V tolerant |
| VCC1V8[4] | 56 | A6 | B5 | - | voltage regulator output (1.8 V ± 0.15 V); tapped out voltage from the internal regulator; this regulated voltage cannot drive external devices; decouple this pin using 4.7 µF and 0.1 µF capacitors; see Section 8.16 |
| XTAL2 | 57 | A5 | A5 | O | crystal oscillator output (12 MHz); connect a fundamental parallel-resonant crystal; leave this pin open when using an external clock source on pin XTAL1; see Table 100 |
| XTAL1 | 58 | A4 | A4 | I | crystal oscillator input (12 MHz); connect a fundamental parallel-resonant crystal or an external clock source (leaving pin XTAL2 unconnected); see Table 100 |
| DGND | 59 | B4 | B4 | - | digital ground |
| MODE0/ DA1[3] | 60 | A3 | C2 | I/O | **Mode selection input 0 —** Selects the read/write strobe functionality in generic processor mode during power-up:<br>• LOW: for the Freescale mode; the function of pin 19 is RW_N and pin 20 is DS_N<br>• HIGH (connect to $V_{CC(I/O)}$): for the 8051 mode; the function of pin 19 is RD_N and pin 20 is WR_N<br>**Address selection output —** Selects the Task File register of an ATA/ATAPI device during normal operation; see Table 61<br>bidirectional pad; 10 ns slew-rate control; TTL; 5 V tolerant |
| $V_{CC(3V3)}$[4] | 61 | B3 | B3 | - | regulator supply voltage (3.3 V ± 0.3 V); this pin supplies the internal regulator; see Section 8.16 |

**Table 3. Pin description** *...continued*

| Symbol[1] | Pin | | | Type[2] | Description |
|---|---|---|---|---|---|
| | ISP1583BS | ISP1583ET; ISP1583ET2 | ISP1583ET1 | | |
| BUS_ CONF/ DA0[3] | 62 | B2 | A3 | I/O | **Bus configuration input —** Selects bus mode during power-up:<br>• LOW: split bus mode; multiplexed 8-bit address and data bus on AD[7:0], separate DMA data bus DATA[15:0][5]<br>• HIGH (connect to $V_{CC(I/O)}$): generic processor mode; separate 8-bit address on AD[7:0], 16-bit processor data bus on DATA[15:0]; DMA is multiplexed on processor bus DATA[15:0]<br>**Address selection output —** Selects the Task File register of an ATA/ATAPI device at normal operation; see Table 61<br>bidirectional pad; 10 ns slew-rate control; TTL; 5 V tolerant |
| WAKEUP | 63 | A2 | B2 | I | wake-up input; when this pin is at the HIGH level, the chip is prevented from going into the suspend state and wake-up the chip when already in suspend mode; when not in use, connect this pin to ground through a 10 kΩ resistor<br>When the RESET_N pin is LOW, ensure that the WAKEUP pin does not go from LOW to HIGH; otherwise the device will enter test mode.<br>input pad; TTL; 5 V tolerant |
| SUSPEND | 64 | C2 | A2 | O | suspend state indicator output; used as a power switch control output to power-off or power-on external devices when going into suspend mode or recovering from suspend mode<br>CMOS output; 8 mA drive |
| DGND | - | B9 | - | - | digital ground |
| DGND | exposed die pad | J9 | B8, G7 | - | ground supply; down bonded to the exposed die pad (heat sink); to be connected to DGND during the PCB layout |

[1] Symbol names ending with underscore N, for example, NAME_N, represent active LOW signals.

[2] All outputs and I/O pins can source 4 mA, unless otherwise specified.

[3] Control signals are not 3-stated.

[4] Add a decoupling capacitor (0.1 μF) to all the supply pins. For better EMI results, add a 0.01 μF capacitor in parallel to 0.1 μF.

[5] The DMA bus is in 3-state until a DMA command (see Section 9.4.1) is executed.

# 8. Functional description

The ISP1583 is a high-speed USB peripheral controller. It implements the Hi-Speed USB or the Original USB physical layer, and the packet protocol layer. It concurrently maintains up to 16 USB endpoints (control IN, control OUT, and seven IN and seven OUT configurable) along with endpoint EP0 set up, which accesses the set-up buffer. The Ref. 1 “Universal Serial Bus Specification Rev. 2.0”, Chapter 9 protocol handling is executed using the external firmware.

The ISP1583 has a fast general-purpose interface to communicate with most types of microcontrollers and microprocessors. This microcontroller interface is configured using pins BUS_CONF/DA0, MODE1 and MODE0/DA1 to accommodate most interface types. Two bus configurations are available, selected using input BUS_CONF/DA0 during power-up:

- **Generic processor mode (pin BUS_CONF/DA0 = HIGH)**:
  - AD[7:0]: 8-bit address bus (selects target register)
  - DATA[15:0]: 16-bit data bus (shared by processor and DMA)
  - Control signals: RW_N and DS_N or RD_N and WR_N (selected using pin MODE0/DA1), CS_N
  - DMA interface (generic slave mode only): Uses lines DATA[15:0] as data bus, DIOR and DIOW as dedicated read and write strobes
- **Split bus mode (pin BUS_CONF/DA0 = LOW)**:
  - AD[7:0]: 8-bit local microprocessor bus (multiplexed address and data)
  - DATA[15:0]: 16-bit DMA data bus
  - Control signals: CS_N, ALE or A0 (selected using pin MODE1), RW_N and DS_N or RD_N and WR_N (selected using pin MODE0/DA1)
  - DMA interface (master or slave mode): Uses DIOR and DIOW as dedicated read and write strobes

For high-bandwidth data transfer, the integrated DMA handler can be invoked to transfer data to or from external memory or devices. The DMA interface can be configured by writing to proper DMA registers (see Section 9.4).

The ISP1583 supports Hi-Speed USB and Original USB signaling. The USB signaling speed is automatically detected.

The ISP1583 has 8 kB of internal FIFO memory, which is shared among enabled USB endpoints, including control IN and control OUT endpoints, and set-up token buffer.

There are seven IN and seven OUT configurable endpoints, and two fixed control endpoints that are 64 bytes long. Any of the seven IN and seven OUT endpoints can be separately enabled or disabled. The endpoint type (interrupt, isochronous or bulk) and packet size of these endpoints can be individually configured, depending on the requirements of the application. Optional double buffering increases the data throughput of these data endpoints.

The ISP1583 requires 3.3 V power supply. It has 5 V tolerant I/O pads and an internal 1.8 V regulator to power the digital logic. The I/O voltage can range from 1.65 V to 3.6 V.

Table 4. Endpoint access and programmability

| Endpoint identifier | Maximum packet size | Double buffering | Endpoint type | Direction |
|---|---|---|---|---|
| EP0SETUP | 8 bytes (fixed) | no | set-up token | OUT |
| EP0RX | 64 bytes (fixed) | no | control OUT | OUT |
| EP0TX | 64 bytes (fixed) | no | control IN | IN |
| EP1RX | programmable | yes | programmable | OUT |
| EP1TX | programmable | yes | programmable | IN |
| EP2RX | programmable | yes | programmable | OUT |
| EP2TX | programmable | yes | programmable | IN |
| EP3RX | programmable | yes | programmable | OUT |
| EP3TX | programmable | yes | programmable | IN |
| EP4RX | programmable | yes | programmable | OUT |
| EP4TX | programmable | yes | programmable | IN |
| EP5RX | programmable | yes | programmable | OUT |
| EP5TX | programmable | yes | programmable | IN |
| EP6RX | programmable | yes | programmable | OUT |
| EP6TX | programmable | yes | programmable | IN |
| EP7RX | programmable | yes | programmable | OUT |
| EP7TX | programmable | yes | programmable | IN |

The ISP1583 operates on a 12 MHz crystal oscillator. An integrated 40 × PLL clock multiplier generates the internal sampling clock of 480 MHz.

## 8.1 DMA interface, DMA handler and DMA registers

The DMA block can be subdivided into two blocks: DMA handler and DMA interface.

The firmware writes to the DMA Command register to start a DMA transfer (see Table 51). The command opcode determines whether a generic DMA, Parallel I/O (PIO) or Multi-word DMA (MDMA) transfer will start. The handler interfaces to the same FIFO (internal RAM) as used by the USB core. On receiving the DMA command, the DMA handler directs the data from the endpoint FIFO to the external DMA device or from the external DMA device to the endpoint FIFO.

The DMA interface configures the timing and the DMA handshake. Data can be transferred using either the DIOR and DIOW strobes or the DACK and DREQ handshakes. DMA configurations are set up by writing to the DMA Configuration register (see Table 56 and Table 57).

For an IDE-based storage interface, applicable DMA modes are PIO and MDMA (Multi-word DMA; ATA).

For a generic DMA interface, DMA modes that can be used are Generic DMA (GDMA) slave.

**Remark:** The DMA endpoint buffer length must be a multiple of 4 bytes.

For details on DMA registers, see Section 9.4.

## 8.2 Hi-Speed USB transceiver

The analog transceiver directly interfaces to the USB cable through integrated termination resistors. The high-speed transceiver requires an external resistor (12.0 kΩ ± 1 %) between pin RREF and ground to ensure an accurate current mirror that generates the Hi-Speed USB current drive. A full-speed transceiver is integrated as well. This makes the ISP1583 compliant to Hi-Speed USB and Original USB, supporting both the high-speed and full-speed physical layers. After automatic speed detection, the NXP Serial Interface Engine (SIE) sets the transceiver to use either high-speed or full-speed signaling.

## 8.3 MMU and integrated RAM

The Memory Management Unit (MMU) manages the access to the integrated RAM that is shared by the USB, microcontroller handler and DMA handler. Data from the USB bus is stored in the integrated RAM, which is cleared only when the microcontroller has read the corresponding endpoint, or the DMA controller has written all data from the RAM of the corresponding endpoint to the DMA bus. The OUT endpoint buffer can also be forcibly cleared by setting bit CLBUF in the Control Function register. A total of 8 kB RAM is available for buffering.

## 8.4 Microcontroller interface and microcontroller handler

The microcontroller interface allows direct interfacing to most microcontrollers and microprocessors. The interface is configured at power-up through pins BUS_CONF/DA0, MODE1 and MODE0/DA1.

When pin BUS_CONF/DA0 = HIGH, the microcontroller interface switches to generic processor mode in which AD[7:0] is the 8-bit address bus and DATA[15:0] is the separate 16-bit data bus. If pin BUS_CONF/DA0 = LOW, the interface is in split bus mode, where AD[7:0] is the local microprocessor bus (multiplexed address and data) and DATA[15:0] is solely used as the DMA bus.

When pin MODE0/DA1 = HIGH, pins RW_N/RD_N and DS_N/WR_N are the read and write strobes (8051 mode). If pin MODE0/DA1 = LOW, pins RW_N/RD_N and DS_N/WR_N represent the direction and data strobes (Freescale mode).

When pin MODE1 = LOW, pin ALE/A0 is used to latch the multiplexed address on pins AD[7:0]. When pin MODE1 = HIGH, pin ALE/A0 is used to indicate address or data. Pin MODE1 is only used in split bus mode; in generic processor mode, it must be tied to $V_{CC(I/O)}$.

The microcontroller handler allows the external microcontroller to access the register set in the NXP SIE, as well as the DMA handler. The initialization of the DMA configuration is done through the microcontroller handler.

## 8.5 OTG SRP module

The OTG supplement defines a Session Request Protocol (SRP), which allows a B-device to request the A-device to turn on $V_{BUS}$ and start a session. This protocol allows the A-device, which may be battery-powered, to conserve power by turning off $V_{BUS}$ when there is no bus activity while still providing a means for the B-device to initiate bus activity.

Any A-device, including a PC or laptop, can respond to SRP. Any B-device, including a standard USB peripheral, can initiate SRP.

The ISP1583 is a device that can initiate SRP.

## 8.6 NXP high-speed transceiver

### 8.6.1 NXP Parallel Interface Engine (PIE)

In the High-Speed (HS) transceiver, the NXP PIE interface uses a 16-bit parallel bidirectional data interface. The functions of the HS module also include bit-stuffing or de-stuffing and Non-Return-to-Zero Inverted (NRZI) encoding or decoding logic.

### 8.6.2 Peripheral circuit

To maintain a constant current driver for HS transmit circuits and to bias other analog circuits, an internal band gap reference circuit and an RREF resistor form the reference current. This circuit requires an external precision resistor (12.0 kΩ ± 1 %) connected to the analog ground.

### 8.6.3 HS detection

The ISP1583 handles more than one electrical state, Full-Speed (FS) or High-Speed (HS), under the USB specification. When the USB cable is connected from the peripheral to the host controller, the ISP1583 defaults to the FS state, until it sees a bus reset from the host controller.

During the bus reset, the peripheral initiates an HS chirp to detect whether the host controller supports Hi-Speed USB or Original USB. If the HS handshake shows that there is an HS host connected, then the ISP1583 switches to the HS state.

In the HS state, the ISP1583 must observe the bus for periodic activity. If the bus remains inactive for 3 ms, the peripheral switches to the FS state to check for a Single-Ended Zero (SE0) condition on the USB bus. If an SE0 condition is detected for the designated time (100 µs to 875 µs; refer to Ref. 1 “Universal Serial Bus Specification Rev. 2.0”, Section 7.1.7.6), the ISP1583 switches to the HS chirp state to perform an HS detection handshake. Otherwise, the ISP1583 remains in the FS state, adhering to the bus-suspend specification.

### 8.6.4 Isolation

Ensure that the DP and DM lines are maintained in a clean state, without any residual voltage or glitches. Once the ISP1583 is reset and the clock is available, ensure that there are no erroneous pulses or glitches even of very small amplitude on the DP and DM lines.

**Remark:** If there are any erroneous unwanted pulses or glitches detected by the ISP1583 DP and DM lines, there is a possibility of the ISP1583 clocking this state into the internal core, causing unknown behaviors.

## 8.7 NXP Serial Interface Engine (SIE)

The NXP 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 de-stuffing, CRC checking or generation, Packet IDentifier (PID) verification or generation, address recognition, handshake evaluation or generation.

## 8.8 SoftConnect

The USB connection is established by pulling pin DP (for full-speed devices) to HIGH through a 1.5 kΩ pull-up resistor. In the ISP1583, an external 1.5 kΩ pull-up resistor must be connected between pin RPU and 3.3 V. The RPU pin connects the pull-up resistor to pin DP, when bit SOFTCT in the Mode register is set (see Table 24 and Table 25). After a hardware reset, the pull-up resistor is disconnected by default (bit SOFTCT = 0). The USB bus reset does not change the value of bit SOFTCT.

When $V_{BUS}$ is not present, the SOFTCT bit must be set to logic 0 to comply with the back-drive voltage.

## 8.9 Reconfiguring endpoints

The ISP1583 endpoints have a limitation when implementing a composite device with at least two functionalities that require the support of alternate settings, for example, the video class and audio class devices. The ISP1583 endpoints cannot be reconfigured on the fly because it is implemented as a FIFO base. The internal RAM partition will be corrupted if there is a need to reconfigure endpoints on the fly because of alternate settings request, causing data corruption.

For details and work-around, refer to Ref. 3 “Using ISP1582/3 in a composite device application with alternate settings (AN10071)”.

## 8.10 System controller

The system controller implements the USB power-down capabilities of the ISP1583. Registers are protected against data corruption during wake-up following a resume (from the suspend state) by locking the write access, until an unlock code is written to the Unlock Device register (see Table 90 and Table 91).

## 8.11 Modes of operation

The ISP1583 has two bus configuration modes, selected using pin BUS_CONF/DA0 at power-up:

- Split bus mode (BUS_CONF/DA0 = LOW): 8-bit multiplexed address and data bus, and separate 8-bit or 16-bit DMA bus
- Generic processor mode (BUS_CONF/DA0 = HIGH): separate 8-bit address and 16-bit data bus

Details of bus configurations for each mode are given in Table 5. Typical interface circuits for each mode are given in Section 14.

**Table 5. Bus configuration modes**

| Pin BUS_CONF/ DA0 | PIO width | DMA width | | Description |
|---|---|---|---|---|
| | | WIDTH = 0 | WIDTH = 1 | |
| LOW | AD[7:0] | D[7:0] | D[15:0] | split bus mode:<br>• Multiplexed address and data on pins AD[7:0]<br>• Separate 8-bit or 16-bit DMA bus on pins DATA[15:0] |
| HIGH | A[7:0] and D[15:0] | D[7:0] | D[15:0] | generic processor mode:<br>• Separate 8-bit address on pins AD[7:0]<br>• 16-bit data (PIO and DMA) on pins DATA[15:0] |

## 8.12 Pins status

Table 6 illustrates the behavior of ISP1583 pins with $V_{CC(I/O)}$ and $V_{CC(3V3)}$ in various operating conditions.

**Table 6. ISP1583 pin status**

| $V_{CC(3V3)}$ | $V_{CC(I/O)}$ | State | Pin | | |
|---|---|---|---|---|---|
| | | | Input | Output | I/O |
| 0 V | 0 V | dead[1] | unknown | unknown | unknown |
| 0 V | 1.65 V to 3.6 V | plug-out[2] | high-Z | unknown | high-Z |
| 0 V → 3.3 V | 1.65 V to 3.6 V | plug-in[3] | high-Z | unknown | high-Z |
| 3.3 V | 1.65 V to 3.6 V | reset | state depends on how the pin is driven | output | high-Z |
| 3.3 V | 1.65 V to 3.6 V | after reset | state depends on how the pin is driven | output | state depends on how the pin is configured |

[1] Dead: the USB cable is plugged out, and $V_{CC(I/O)}$ is not available.

[2] Plug-out: the USB cable is not present, but $V_{CC(I/O)}$ is available.

[3] Plug-in: the USB cable is being plugged in, and $V_{CC(I/O)}$ is available.

Table 7 illustrates the behavior of output pins with $V_{CC(I/O)}$ and $V_{CC(3V3)}$ in various operating conditions.

**Table 7. ISP1583 output pin status**

| $V_{CC(3V3)}$ | $V_{CC(I/O)}$ | State | INT | SUSPEND |
|---|---|---|---|---|
| 0 V | 0 V | dead[1] | LOW | HIGH |
| 0 V | 1.65 V to 3.6 V | plug-out[2] | LOW | HIGH |
| 0 V → 3.3 V | 1.65 V to 3.6 V | plug-in[3] | LOW | HIGH |
| 3.3 V | 1.65 V to 3.6 V | reset | HIGH | LOW |
| 3.3 V | 1.65 V to 3.6 V | after reset | HIGH | LOW |

[1] Dead: the USB cable is plugged out, and $V_{CC(I/O)}$ is not available.

[2] Plug-out: the USB cable is not present, but $V_{CC(I/O)}$ is available.

[3] Plug-in: the USB cable is being plugged in, and $V_{CC(I/O)}$ is available.

## 8.13 Interrupt

### 8.13.1 Interrupt output pin

The Interrupt Configuration register of the ISP1583 controls the behavior of the INT output pin. The polarity and signaling mode of the INT pin can be programmed by setting bits INTPOL and INTLVL of the Interrupt Configuration register (R/W: 10h); see Table 28. Bit GLINTENA of the Mode register (R/W: 0Ch) is used to enable pin INT; see Table 25. Default settings after reset are active LOW and level mode. When pulse mode is selected, a pulse of 60 ns is generated when the OR-ed combination of all interrupt bits changes from logic 0 to logic 1.

Figure 5 shows the relationship between interrupt events and pin INT.

Each of the indicated USB and DMA events is logged in a status bit of the Interrupt register and the DMA Interrupt Reason register, respectively. Corresponding bits in the Interrupt Enable register and the DMA Interrupt Enable register determine whether an event will generate an interrupt.

Interrupts can be masked globally by means of bit GLINTENA of the Mode register.

Field CDBGMOD[1:0] of the Interrupt Configuration register controls the generation of INT signals for the control pipe. Field DDBGMODIN[1:0] of the Interrupt Configuration register controls the generation of INT signals for the IN pipe. Field DDBGMODOUT[1:0] of the Interrupt Configuration register controls the generation of INT signals for the OUT pipe; see Table 29.

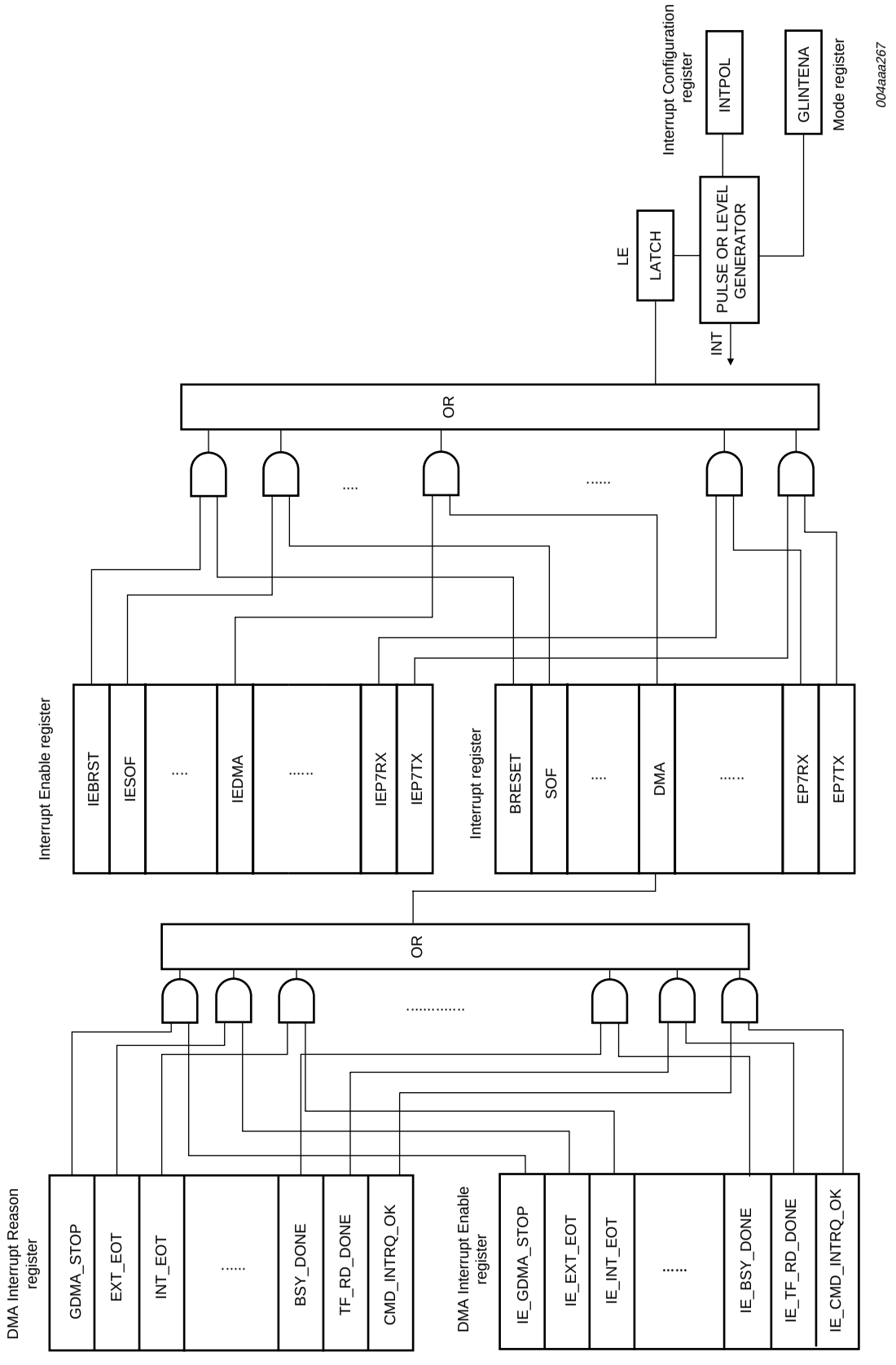



**Fig 5. Interrupt logic**

### 8.13.2 Interrupt control

Bit GLINTENA in the Mode register is a global interrupt enable or disable bit. The behavior of this bit is given in Figure 6.

The following illustrations are only applicable for level trigger.

Event A: When an interrupt event occurs (for example, SOF interrupt) with bit GLINTENA set to logic 0, an interrupt will not be generated at pin INT. It will, however, be registered in the corresponding Interrupt register bit.

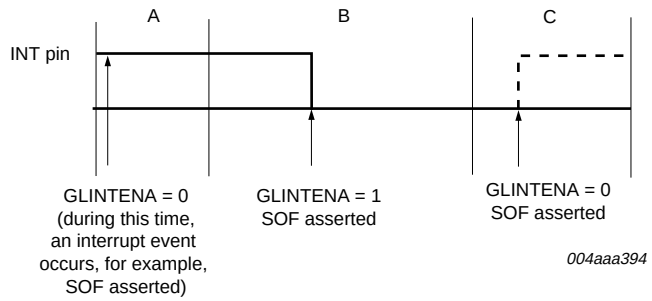
Event B: When bit GLINTENA is set to logic 1, pin INT is asserted because bit SOF in the Interrupt register is already set.

Event C: If the firmware sets bit GLINTENA to logic 0, pin INT will still be asserted. The bold line shows the desired behavior of pin INT.

Deassertion of pin INT can be achieved either by clearing all the bits in the Interrupt register or the DMA Interrupt Reason register, depending on the event.

**Remark:** When clearing an interrupt event, perform write to all the bytes of the register.

For more information on interrupt control, see Section 9.2.2, Section 9.2.5 and Section 9.5.1.



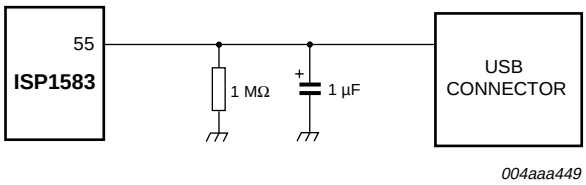

Pin INT: HIGH = deassert; LOW = assert (individual interrupts are enabled).

**Fig 6. Behavior of bit GLINTENA**

## 8.14 $V_{BUS}$ sensing

The $V_{BUS}$ pin is one of the ways to wake up the clock when the ISP1583 is suspended with bit CLKAON set to logic 0 (clock off option).

To detect whether the host is connected or not, that is $V_{BUS}$ sensing, a 1 MΩ resistor and a 1 µF electrolytic or tantalum capacitor must be added to damp the overshoot on plug in.




The figure shows the ISP1583BS pinout. For the ISP1583ET, ISP1583ET1 and ISP1583ET2 ballouts, see Table 3.

**Fig 7. Resistor and electrolytic or tantalum capacitor needed for $V_{BUS}$ sensing**

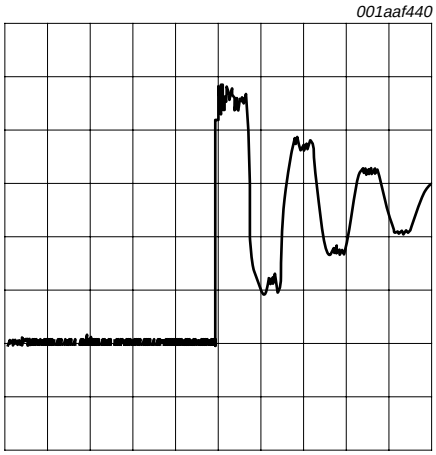



**Fig 8. Oscilloscope reading: no resistor and capacitor in the network**

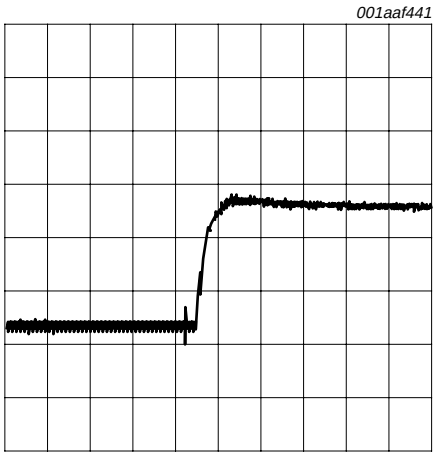



**Fig 9. Oscilloscope reading: with resistor and capacitor in the network**

## 8.15 Power-on reset

The ISP1583 requires a minimum pulse width of 500 µs.

The RESET_N pin can either be connected to $V_{CC(3V3)}$ (using the internal POR circuit) or externally controlled (by the microcontroller, ASIC, and so on). When $V_{CC(3V3)}$ is directly connected to the RESET_N pin, internal pulse width $t_{PORP}$ will typically be 200 ns.

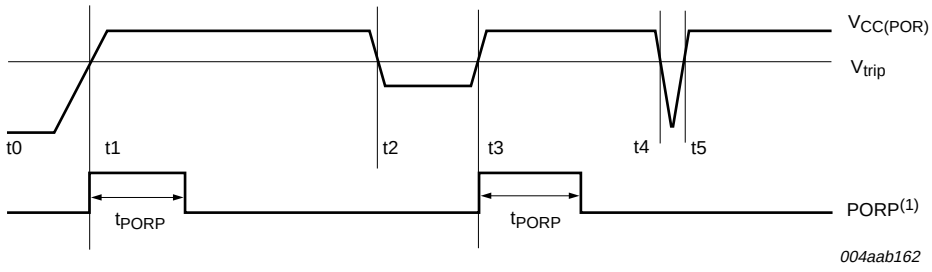
The power-on reset function can be explained by viewing the dips at t2 to t3 and t4 to t5 on the $V_{CC(POR)}$ curve (Figure 10).

**t0 —** The internal POR starts with a HIGH level.

**t1 —** The detector will see the passing of the trip level and a delay element will add another $t_{PORP}$ before it drops to LOW.

**t2 to t3 —** The internal POR pulse will be generated whenever $V_{CC(POR)}$ drops below $V_{trip}$ for more than 11 µs.

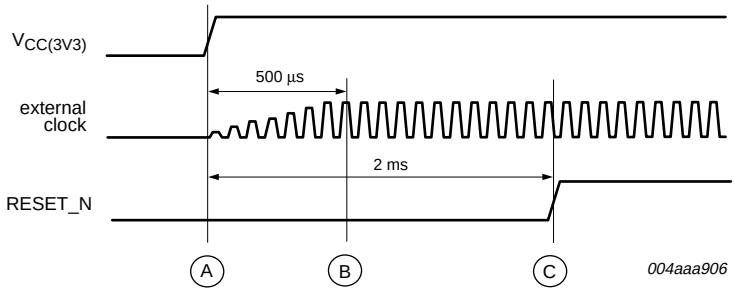
**t4 to t5 —** The dip is too short (< 11 µs) and the internal POR pulse will not react and will remain LOW.




(1) PORP = Power-On Reset Pulse.

**Fig 10. POR timing**

Figure 11 shows the availability of the clock with respect to the external POR.




Power on $V_{CC(3V3)}$ at A.

Stable external clock is to be available at B.

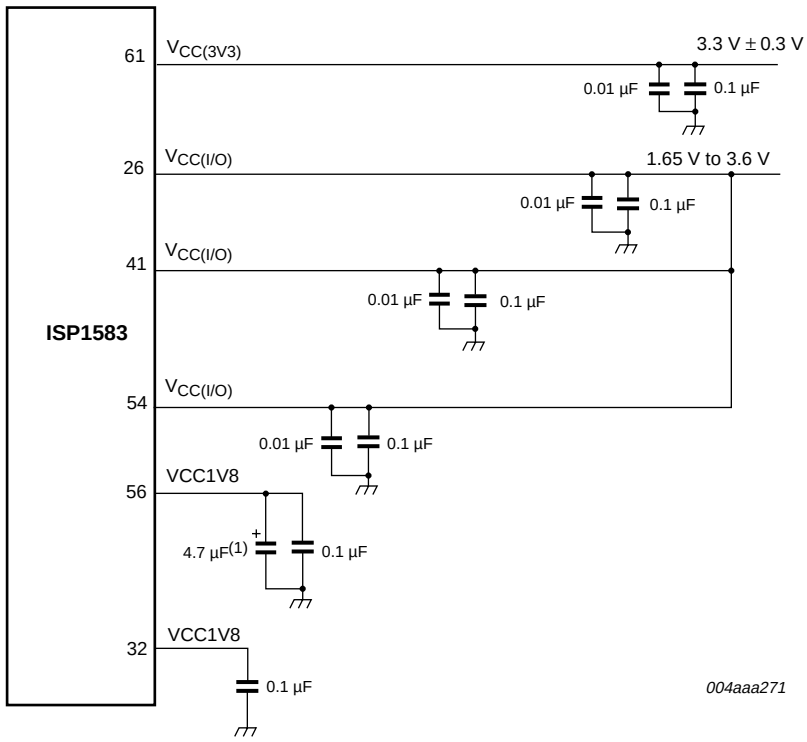
The ISP1583 is operational at C.

**Fig 11. Clock with respect to the external POR**

## 8.16 Power supply

The ISP1583 can be powered by 3.3 V ± 0.3 V, and from 1.65 V to 3.6 V at the interface. For details, see Figure 12. If the ISP1583 is powered by $V_{CC(3V3)}$ = 3.3 V, an integrated 3.3 V-to-1.8 V voltage regulator provides a 1.8 V supply voltage for the internal logic.

In sharing mode (that is, when $V_{CC(3V3)}$ is not present and $V_{CC(I/O)}$ is present), all I/O pins are input type, the interrupt pin is connected to ground, and the suspend pin is connected to $V_{CC(I/O)}$. See Table 7.




(1) At the $V_{CC}$ input (3.3 V) to the USB controller, if the ripple voltage is less than 20 mV, then 4.7 µF standard electrolytic or tantalum capacitors (tested ESR up to 10 Ω) should be OK at the VCC1V8 output. If the ripple voltage at the input is higher than 20 mV, then use 4.7 µF LOW ESR capacitors (ESR from 0.2 Ω to 2 Ω) at the VCC1V8 output. This is to improve the high-speed signal quality at the USB side.

The figure shows the ISP1583BS pinout. For the ISP1583ET, ISP1583ET1 and ISP1583ET2 ballouts, see Table 3.

**Fig 12. ISP1583 with a 3.3 V supply**

Table 8 shows power modes in which the ISP1583 can be operated.

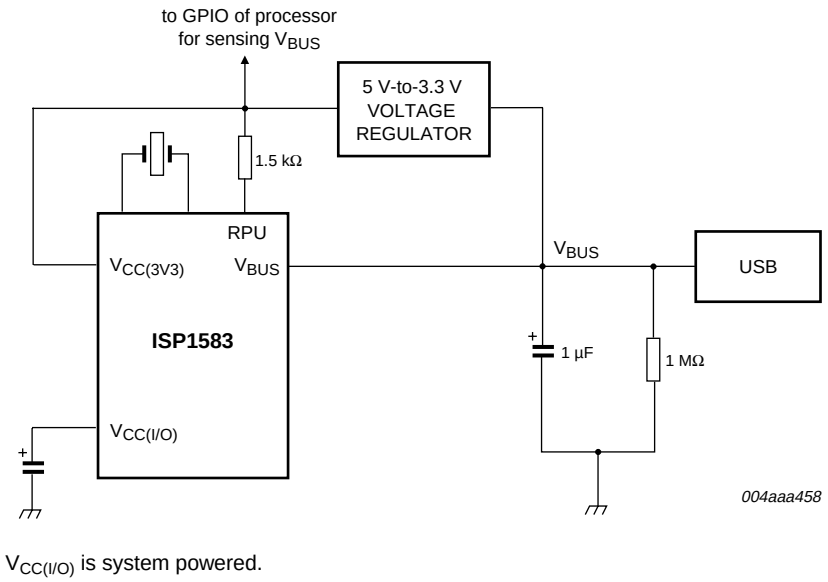
**Table 8. Power modes**

| $V_{CC(3V3)}$ | $V_{CC(I/O)}$ | Power mode |
|---|---|---|
| $V_{BUS}$[1] | $V_{BUS}$[2] | bus-powered |
| System-powered | system-powered | self-powered |
| $V_{BUS}$[1] | system-powered | power-sharing (hybrid) |

[1] The power supply to the IC ($V_{CC(3V3)}$) is 3.3 V. Therefore, if the application is bus-powered, a 3.3 V regulator needs to be used.

[2] $V_{CC(I/O)}$ can range from 1.65 V to 3.6 V. If the application is bus-powered, a voltage regulator must be used.

### 8.16.1 Power-sharing mode




$V_{CC(I/O)}$ is system powered.

**Fig 13. Power-sharing mode**

As can be seen in Figure 13, in power-sharing mode, $V_{CC(3V3)}$ is supplied by the output of the 5 V-to-3.3 V voltage regulator. The input to the regulator is from $V_{BUS}$. $V_{CC(I/O)}$ is supplied through the power source of the system. When the USB cable is plugged in, the ISP1583 goes through the power-on reset cycle. In this mode, OTG is disabled.

The processor will experience continuous interrupt because the default status of the interrupt pin when operating in sharing mode with $V_{BUS}$ not present is LOW. To overcome this, implement external $V_{BUS}$ sensing circuitry. The output from the voltage regulator can be connected to pin GPIO of the processor to qualify the interrupt from the ISP1583.

**Remark:** When the core power is applied, the ISP1583 must be reset using the RESET_N pin. The minimum width of the reset pulse width must be 2 ms.

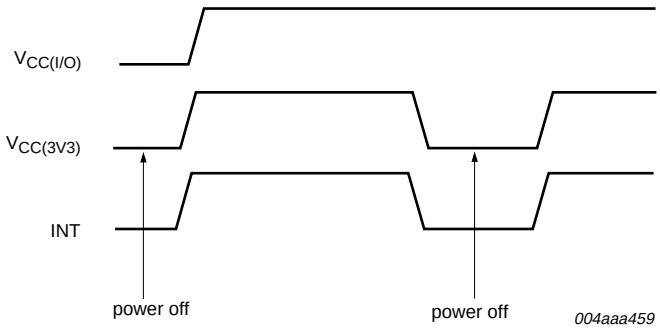



**Fig 14. Interrupt pin status during power off in power-sharing mode**

**Table 9. Operation truth table for SoftConnect**

| ISP1583 operation | Power supply | | | | Bit SOFTCT in Mode register |
|---|---|---|---|---|---|
| | $V_{CC(3V3)}$ | $V_{CC(I/O)}$ | RPU (3.3 V) | $V_{BUS}$ | |
| Normal bus operation | 3.3 V | 3.3 V | 3.3 V | 5 V | enabled |
| Core power is lost | 0 V | 3.3 V | 0 V | 0 V | not applicable |

**Table 10. Operation truth table for clock off during suspend**

| ISP1583 operation | Power supply | | | | Clock off during suspend |
|---|---|---|---|---|---|
| | $V_{CC(3V3)}$ | $V_{CC(I/O)}$ | RPU (3.3 V) | $V_{BUS}$ | |
| Clock will wake up:<br>After a resume and<br>After a bus reset | 3.3 V | 3.3 V | 3.3 V | 5 V | enabled |
| Core power is lost | 0 V | 3.3 V | 0 V | 0 V | not applicable |

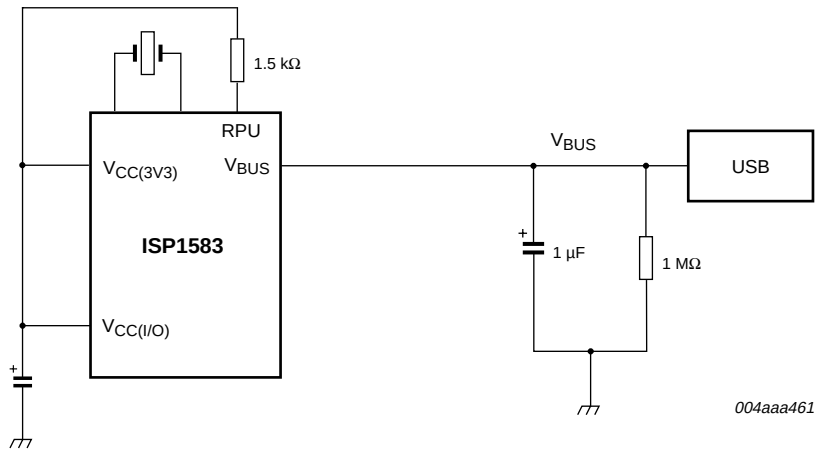
**Table 11. Operation truth table for back voltage compliance**

| ISP1583 operation | Power supply | | | | Bit SOFTCT in Mode register |
|---|---|---|---|---|---|
| | $V_{CC(3V3)}$ | $V_{CC(I/O)}$ | RPU (3.3 V) | $V_{BUS}$ | |
| Back voltage is not measured in this mode | 3.3 V | 3.3 V | 3.3 V | 5 V | enabled |
| Back voltage is not an issue because core power is lost | 0 V | 3.3 V | 0 V | 0 V | not applicable |

**Table 12. Operation truth table for OTG**

| ISP1583 operation | Power supply | | | | OTG register |
|---|---|---|---|---|---|
| | $V_{CC(3V3)}$ | $V_{CC(I/O)}$ | RPU (3.3 V) | $V_{BUS}$ | |
| SRP is not applicable | 3.3 V | 3.3 V | 3.3 V | 5 V | not applicable |
| OTG is not possible because $V_{BUS}$ is not present and so core power is lost | 0 V | 3.3 V | 0 V | 0 V | not applicable |

### 8.16.2 Self-powered mode




$V_{CC(I/O)}$ and $V_{CC(3V3)}$ are system powered.

**Fig 15. Self-powered mode**

In self-powered mode, $V_{CC(3V3)}$ and $V_{CC(I/O)}$ are supplied by the system. See Figure 15.

**Table 13. Operation truth table for SoftConnect**

| ISP1583 operation | Power supply | | | | Bit SOFTCT in Mode register |
|---|---|---|---|---|---|
| | $V_{CC(3V3)}$ | $V_{CC(I/O)}$ | RPU (3.3 V) | $V_{BUS}$ | |
| Normal bus operation | 3.3 V | 3.3 V | 3.3 V | 5 V | enabled |
| No pull up on DP | 3.3 V | 3.3 V | 3.3 V | 0 V[1] | disabled |

[1] When the USB cable is removed, SoftConnect is disabled.

**Table 14. Operation truth table for clock off during suspend**

| ISP1583 operation | Power supply | | | | Clock off during suspend |
|---|---|---|---|---|---|
| | $V_{CC(3V3)}$ | $V_{CC(I/O)}$ | RPU (3.3 V) | $V_{BUS}$ | |
| Clock will wake up:<br>After a resume and<br>After a bus reset | 3.3 V | 3.3 V | 3.3 V | 5 V | enabled |
| Clock will wake up:<br>After detecting the presence of $V_{BUS}$ | 3.3 V | 3.3 V | 3.3 V | 0 V → 5 V | enabled |

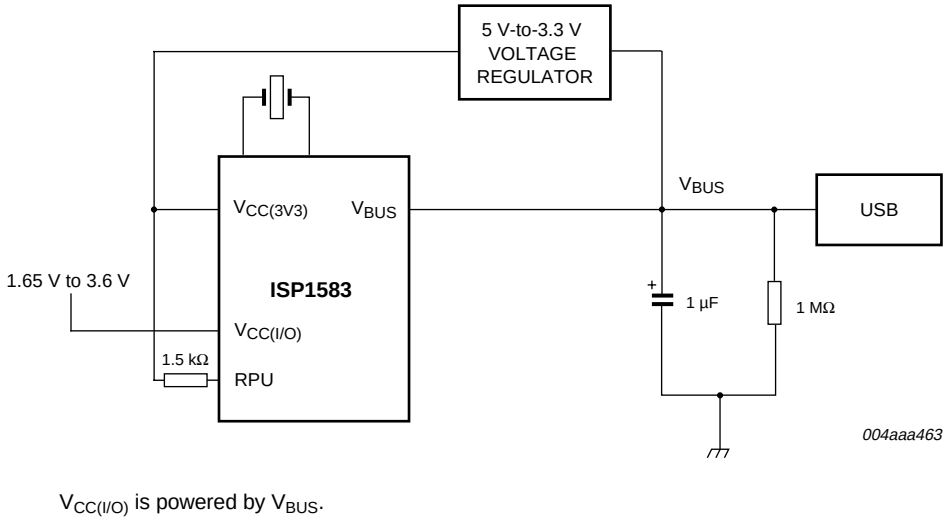
**Table 15. Operation truth table for back voltage compliance**

| ISP1583 operation | Power supply | | | | Bit SOFTCT in Mode register |
|---|---|---|---|---|---|
| | $V_{CC(3V3)}$ | $V_{CC(I/O)}$ | RPU (3.3 V) | $V_{BUS}$ | |
| Back voltage is not measured in this mode | 3.3 V | 3.3 V | 3.3 V | 5 V | enabled |
| Back voltage is not an issue because pull up on DP will not be present when $V_{BUS}$ is not present | 3.3 V | 3.3 V | 3.3 V | 0 V | disabled |

**Table 16. Operation truth table for OTG**

| ISP1583 operation | Power supply | | | | OTG register |
|---|---|---|---|---|---|
| | $V_{CC(3V3)}$ | $V_{CC(I/O)}$ | RPU (3.3 V) | $V_{BUS}$ | |
| SRP is not applicable | 3.3 V | 3.3 V | 3.3 V | 5 V | not applicable |
| SRP is possible | 3.3 V | 3.3 V | 3.3 V | 0 V | operational |

### 8.16.3 Bus-powered mode




$V_{CC(I/O)}$ is powered by $V_{BUS}$.

**Fig 16. Bus-powered mode**

In bus-powered mode (see Figure 16), $V_{CC(3V3)}$ and $V_{CC(I/O)}$ are supplied by the output of the 5 V-to-3.3 V voltage regulator. The input to the regulator is from $V_{BUS}$. On plugging the USB cable, the ISP1583 goes through the power-on reset cycle. In this mode, OTG is disabled.

**Table 17. Operation truth table for SoftConnect**

| ISP1583 operation | Power supply | | | | Bit SOFTCT in Mode register |
|---|---|---|---|---|---|
| | $V_{CC(3V3)}$ | $V_{CC(I/O)}$ | RPU (3.3 V) | $V_{BUS}$ | |
| Normal bus operation | 3.3 V | 3.3 V | 3.3 V | 5 V | enabled |
| Power is lost | 0 V | 0 V | 0 V | 0 V | not applicable |

**Table 18. Operation truth table for clock off during suspend**

| ISP1583 operation | Power supply | | | | Clock off during suspend |
|---|---|---|---|---|---|
| | $V_{CC(3V3)}$ | $V_{CC(I/O)}$ | RPU (3.3 V) | $V_{BUS}$ | |
| Clock will wake up:<br>After a resume and<br>After a bus reset | 3.3 V | 3.3 V | 3.3 V | 5 V | enabled |
| Power is lost | 0 V | 0 V | 0 V | 0 V | not applicable |

**Table 19. Operation truth table for back voltage compliance**

| ISP1583 operation | Power supply | | | | Bit SOFTCT in Mode register |
|---|---|---|---|---|---|
| | $V_{CC(3V3)}$ | $V_{CC(I/O)}$ | RPU (3.3 V) | $V_{BUS}$ | |
| Back voltage is not measured in this mode | 3.3 V | 3.3 V | 3.3 V | 5 V | enabled |
| Power is lost | 0 V | 0 V | 0 V | 0 V | not applicable |

**Table 20. Operation truth table for OTG**

| ISP1583 operation | Power supply | | | | OTG register |
|---|---|---|---|---|---|
| | $V_{CC(3V3)}$ | $V_{CC(I/O)}$ | RPU (3.3 V) | $V_{BUS}$ | |
| SRP is not applicable | 3.3 V | 3.3 V | 3.3 V | 5 V | not applicable |
| Power is lost | 0 V | 0 V | 0 V | 0 V | not applicable |

# 9. Register description

**Table 21. Register overview**

| Name | Destination | Address | Description | Size (bytes) | Reference |
|---|---|---|---|---|---|
| **Initialization registers** | | | | | |
| Address | device | 00h | USB device address and enable | 1 | Section 9.2.1 on page 31 |
| Mode | device | 0Ch | power-down options, global interrupt enable, SoftConnect | 2 | Section 9.2.2 on page 31 |
| Interrupt Configuration | device | 10h | interrupt sources, trigger mode, output polarity | 1 | Section 9.2.3 on page 33 |
| OTG | device | 12h | OTG implementation | 1 | Section 9.2.4 on page 34 |
| Interrupt Enable | device | 14h | interrupt source enabling | 4 | Section 9.2.5 on page 36 |
| **Data flow registers** | | | | | |
| Endpoint Index | endpoints | 2Ch | endpoint selection, data flow direction | 1 | Section 9.3.1 on page 38 |
| Control Function | endpoint | 28h | endpoint buffer management | 1 | Section 9.3.2 on page 39 |
| Data Port | endpoint | 20h | data access to endpoint FIFO | 2 | Section 9.3.3 on page 40 |
| Buffer Length | endpoint | 1Ch | packet size counter | 2 | Section 9.3.4 on page 41 |
| Buffer Status | endpoint | 1Eh | buffer status for each endpoint | 1 | Section 9.3.5 on page 42 |
| Endpoint MaxPacketSize | endpoint | 04h | maximum packet size | 2 | Section 9.3.6 on page 43 |
| Endpoint Type | endpoint | 08h | selects endpoint type: isochronous, bulk or interrupt | 2 | Section 9.3.7 on page 44 |
| **DMA registers** | | | | | |
| DMA Command | DMA controller | 30h | controls all DMA transfers | 1 | Section 9.4.1 on page 47 |
| DMA Transfer Counter | DMA controller | 34h | sets byte count for DMA transfer | 4 | Section 9.4.2 on page 49 |
| DMA Configuration | DMA controller | 38h | byte 0: sets GDMA configuration (counter enable, data strobing, bus width) | 1 | Section 9.4.3 on page 50 |
| | | 39h | byte 1: sets ATA configuration (IORDY enable, mode selection: ATA, MDMA, PIO) | 1 | |
| DMA Hardware | DMA controller | 3Ch | endian type, master or slave selection, signal polarity for DACK, DREQ, DIOW, DIOR, EOT | 1 | Section 9.4.4 on page 52 |

**Table 21. Register overview** *…continued*

| Name | Destination | Address | Description | Size (bytes) | Reference |
|---|---|---|---|---|---|
| Task File 1F0 | ATAPI peripheral | 40h | single address word register: byte 0 (lower byte) is accessed first | 2 | Section 9.4.5 on page 53 |
| Task File 1F1 | ATAPI peripheral | 48h | IDE device access | 1 | |
| Task File 1F2 | ATAPI peripheral | 49h | IDE device access | 1 | |
| Task File 1F3 | ATAPI peripheral | 4Ah | IDE device access | 1 | |
| Task File 1F4 | ATAPI peripheral | 4Bh | IDE device access | 1 | |
| Task File 1F5 | ATAPI peripheral | 4Ch | IDE device access | 1 | |
| Task File 1F6 | ATAPI peripheral | 4Dh | IDE device access | 1 | |
| Task File 1F7 | ATAPI peripheral | 44h | IDE device access (write only; reading returns FFh) | 1 | |
| Task File 3F6 | ATAPI peripheral | 4Eh | IDE device access | 1 | |
| Task File 3F7 | ATAPI peripheral | 4Fh | IDE device access | 1 | |
| DMA Interrupt Reason | DMA controller | 50h | shows reason (source) for DMA interrupt | 2 | Section 9.4.6 on page 56 |
| DMA Interrupt Enable | DMA controller | 54h | enables DMA interrupt sources | 2 | Section 9.4.7 on page 57 |
| DMA Endpoint | DMA controller | 58h | selects endpoint FIFO, data flow direction | 1 | Section 9.4.8 on page 57 |
| DMA Strobe Timing | DMA controller | 60h | strobe duration in MDMA mode | 1 | Section 9.4.9 on page 58 |
| DMA Burst Counter | DMA controller | 64h | DMA burst length | 2 | Section 9.4.10 on page 59 |
| **General registers** | | | | | |
| Interrupt | device | 18h | shows interrupt sources | 4 | Section 9.5.1 on page 59 |
| Chip ID | device | 70h | product ID code and hardware version | 3 | Section 9.5.2 on page 61 |
| Frame Number | device | 74h | last successfully received Start-Of-Frame: lower byte (byte 0) is accessed first | 2 | Section 9.5.3 on page 62 |
| Scratch | device | 78h | allows save or restore of firmware status during suspend | 2 | Section 9.5.4 on page 62 |
| Unlock Device | device | 7Ch | re-enables register write access after suspend | 2 | Section 9.5.5 on page 63 |
| Test Mode | PHY | 84h | direct setting of the DP and DM states, internal transceiver test (PHY) | 1 | Section 9.5.6 on page 63 |

## 9.1 Register access

Register access depends on the bus width used:

- 8-bit bus: multi-byte registers are accessed lower byte (LSByte) first
- 16-bit bus: for single-byte registers, the upper byte (MSByte) must be ignored

Endpoint specific registers are indexed using the Endpoint Index register. The target endpoint must be selected before accessing the following registers:

- Buffer Length
- Buffer Status
- Control Function
- Data Port
- Endpoint MaxPacketSize
- Endpoint Type

**Remark:** Write zero to all reserved bits, unless otherwise specified.

## 9.2 Initialization registers

### 9.2.1 Address register (address: 00h)

This register sets the USB assigned address and enables the USB device. Table 22 shows the Address register bit allocation.

Bits DEVADDR[6:0] will be cleared whenever a bus reset, a power-on reset or a soft reset occurs. Bit DEVEN will be cleared whenever a power-on reset or a soft reset occurs.

In response to standard USB request SET_ADDRESS, the firmware must write the (enabled) device address to the Address register, followed by sending an empty packet to the host. The new device address is activated when the device receives an acknowledgment from the host for the empty packet token.

**Table 22. Address register: bit allocation**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Symbol | DEVEN | DEVADDR[6:0] | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bus reset | unchanged | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**Table 23. Address register: bit description**

| Bit | Symbol | Description |
|---|---|---|
| 7 | DEVEN | **Device Enable**: Logic 1 enables the device. The device will not respond to the host, unless this bit is set. |
| 6 to 0 | DEVADDR[6:0] | **Device Address**: This field specifies the USB device address. |

### 9.2.2 Mode register (address: 0Ch)

This register consists of 2 bytes (bit allocation: see Table 24).

The Mode register controls resume, suspend and wake-up behavior, interrupt activity, soft reset, clock signals and SoftConnect operation.

**Table 24. Mode register: bit allocation**

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Symbol | TEST2 | TEST1 | TEST0 | reserved | | | DMA CLKON | VBUSSTAT |
| Reset | - | - | - | - | - | - | 0 | -[1] |
| Bus reset | - | - | - | - | - | - | 0 | -[1] |
| Access | R | R | R | - | - | - | R/W | R |
| **Bit** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| Symbol | CLKAON | SNDRSU | GOSUSP | SFRESET | GLINTENA | WKUPCS | PWRON | SOFTCT |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bus reset | unchanged | 0 | 0 | 0 | unchanged | 0 | unchanged | unchanged |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

[1] Value depends on the status of the $V_{BUS}$ pin.

**Table 25. Mode register: bit description**

| Bit | Symbol | Description |
|---|---|---|
| 15 | TEST2 | This bit reflects the MODE1 pin setting. Only for test purposes. |
| 14 | TEST1 | This bit reflects the MODE0/DA1 pin setting. Only for test purposes. |
| 13 | TEST0 | This bit reflects the BUS_CONF/DA0 pin setting. Only for test purposes. |
| 12 to 10 | - | reserved |
| 9 | DMACLKON | **DMA Clock On**:<br>**0 —** Power save mode; the DMA circuit will stop completely to save power.<br>**1 —** Supply clock to the DMA circuit. |
| 8 | VBUSSTAT | **$V_{BUS}$ Pin Status**: This bit reflects the $V_{BUS}$ pin status. |
| 7 | CLKAON | **Clock Always On**: Logic 1 indicates that internal clocks are always running when in the suspend state. Logic 0 switches off the internal oscillator and PLL when the device goes into suspend mode. The device will consume less power if this bit is set to logic 0. The clock is stopped about 2 ms after bit GOSUSP is set and then cleared. |
| 6 | SNDRSU | **Send Resume**: Writing logic 1, followed by logic 0 will generate a 10 ms upstream resume signal.<br>**Remark:** The upstream resume signal is generated 5 ms after this bit is set to logic 0. |
| 5 | GOSUSP | **Go Suspend**: Writing logic 1, followed by logic 0 will activate suspend mode. |
| 4 | SFRESET | **Soft Reset**: Writing logic 1, followed by logic 0 will enable a software-initiated reset to the ISP1583. A soft reset is similar to a hardware-initiated reset (using the RESET_N pin). |

**Table 25. Mode register: bit description** *...continued*

| Bit | Symbol | Description |
|---|---|---|
| 3 | GLINTENA | **Global Interrupt Enable**: Logic 1 enables all interrupts. Individual interrupts can be masked by clearing the corresponding bits in the Interrupt Enable register.<br>When this bit is not set, an unmasked interrupt will not generate an interrupt trigger on the interrupt pin. If global interrupt, however, is enabled while there is any pending unmasked interrupt, an interrupt signal will be immediately generated on the interrupt pin. (If the interrupt is set to pulse mode, the interrupt events that were generated before the global interrupt is enabled will not appear on the interrupt pin.) |
| 2 | WKUPCS | **Wake-up on Chip selection**: Logic 1 enables wake-up from suspend mode through a valid register read on the ISP1583. (A read will invoke the chip clock to restart. If you write to the register before the clock gets stable, it may cause malfunctioning.) |
| 1 | PWRON | **Power On**: The SUSPEND pin output control.<br>**0 —** The SUSPEND pin is HIGH when the ISP1583 is in the suspend state. Otherwise, the SUSPEND pin is LOW.<br>**1 —** When the device is woken up from the suspend state, there will be a 1 ms active HIGH pulse on the SUSPEND pin. The SUSPEND pin will remain LOW in all other states. |
| 0 | SOFTCT | **SoftConnect**: Logic 1 enables the connection of the 1.5 kΩ pull-up resistor on pin RPU to the DP pin. |

The status of the chip is shown in Table 26.

**Table 26. Status of the chip**

| Bus state | SoftConnect = on | SoftConnect = off |
|---|---|---|
| $V_{BUS}$ on | pull-up resistor on pin DP | pull-up resistor on pin DP is removed; suspend interrupt is generated after 3 ms of no bus activity |
| $V_{BUS}$ off | pull-up resistor on pin DP is present; suspend interrupt is generated after 3 ms of no bus activity | pull-up resistor on pin DP is removed; suspend interrupt is generated after 3 ms of no bus activity |

### 9.2.3 Interrupt Configuration register (address: 10h)

This 1-byte register determines the behavior and polarity of the INT output. The bit allocation is shown in Table 27. When the USB SIE receives or generates an ACK, NAK or NYET, it will generate interrupts, depending on three Debug mode fields.

**CDBGMOD[1:0] —** interrupts for control endpoint 0

**DDBGMODIN[1:0] —** interrupts for DATA IN endpoints 1 to 7

**DDBGMODOUT[1:0] —** interrupts for DATA OUT endpoints 1 to 7

The Debug mode settings for CDBGMOD, DDBGMODIN and DDBGMODOUT allow you to individually configure when the ISP1583 sends an interrupt to the external microprocessor. Table 29 lists available combinations.

Bit INTPOL controls the signal polarity of the INT output: active HIGH or LOW, rising or falling edge. For level-triggering, bit INTLVL must be made logic 0. By setting INTLVL to logic 1, an interrupt will generate a pulse of 60 ns (edge-triggering).

**Table 27. Interrupt Configuration register: bit allocation**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Symbol | CDBGMOD[1:0] | | DDBGMODIN[1:0] | | DDBGMODOUT[1:0] | | INTLVL | INTPOL |
| Reset | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 |
| Bus reset | 1 | 1 | 1 | 1 | 1 | 1 | unchanged | unchanged |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**Table 28. Interrupt Configuration register: bit description**

| Bit | Symbol | Description |
|---|---|---|
| 7 to 6 | CDBGMOD[1:0] | **Control Endpoint 0 Debug Mode**: For values, see Table 29 |
| 5 to 4 | DDBGMODIN[1:0] | **Data Debug Mode IN**: For values, see Table 29 |
| 3 to 2 | DDBGMODOUT[1:0] | **Data Debug Mode OUT**: For values, see Table 29 |
| 1 | INTLVL | **Interrupt Level**: Selects signaling mode on output INT (**0** = level; **1** = pulsed). In pulsed mode, an interrupt produces a 60 ns pulse. |
| 0 | INTPOL | **Interrupt Polarity**: Selects signal polarity on output INT (**0** = active LOW, **1** = active HIGH). |

**Table 29. Debug mode settings**

| Value | CDBGMOD | DDBGMODIN | DDBGMODOUT |
|---|---|---|---|
| 00h | interrupt on all ACK and NAK | interrupt on all ACK and NAK | interrupt on all ACK, NYET and NAK |
| 01h | interrupt on all ACK | interrupt on ACK | interrupt on ACK and NYET |
| 1Xh | interrupt on all ACK and first NAK[1] | interrupt on all ACK and first NAK[1] | interrupt on all ACK, NYET and first NAK[1] |

[1] First NAK: the first NAK on an IN or OUT token is generated after a set-up token and an ACK sequence.

### 9.2.4 OTG register (address: 12h)

The bit allocation of the OTG register is given in Table 30.

**Table 30. OTG register: bit allocation**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Symbol | reserved | | DP | BSESSVALID | INITCOND | DISCV | VP | OTG |
| Reset | - | - | 0 | - | - | 0 | 0 | 0 |
| Bus reset | - | - | 0 | - | - | 0 | 0 | 0 |
| Access | - | - | R/W | R/W | R/W | R/W | R/W | R/W |

**Table 31. OTG register: bit description**

| Bit | Symbol | Description[1] |
|---|---|---|
| 7 to 6 | - | reserved |
| 5 | DP | **Data Pulsing**: Used for data-line pulsing to toggle DP to generate the required data-line pulsing signal. The default value of this bit is logic 0. This bit must be cleared when data-line pulsing is completed. |
| 4 | BSESS VALID | **B-Session Valid**: The device can initiate another $V_{BUS}$ discharge sequence after data-line pulsing and $V_{BUS}$ pulsing, and before it clears this bit and detects a session valid.<br>This bit is latched to logic 1 once $V_{BUS}$ exceeds the B-device session valid threshold. Once set, it remains at logic 1. To clear this bit, write logic 1. (The ISP1583 continuously updates this bit to logic 1 when the B-session is valid. If the B-session is valid after it is cleared, it is set back to logic 1 by the ISP1583).<br>**0 —** It implies that SRP has failed. To proceed to a normal operation, the device can restart SRP, clear bit OTG or proceed to an error handling process.<br>**1 —** It implies that the B-session is valid. The device clears bit OTG, goes into normal operation mode, and sets bit SOFTCT (DP pull-up) in the Mode register. The OTG host has a maximum of 5 s before it responds to a session request. During this period, the ISP1583 may request to suspend. Therefore, the device firmware must wait for some time if it wishes to know the SRP result (success: if there is minimum response from the host within 5 s; failure: if there is no response from the host within 5 s). |
| 3 | INIT COND | **Initial Condition**: Write logic 1 to clear this bit. Wait for more than 2 ms and check the bit status. If it reads logic 0, it means that $V_{BUS}$ remains lower than 0.8 V, and DP or DM are at SE0 during the elapsed time. The device can then start a B-device SRP. If it reads logic 1, it means that the initial condition of SRP is violated. So, the device must abort SRP.<br>The bit is set to logic 1 by the ISP1583 when initial conditions are not met, and only writing logic 1 clears the bit. (If initial conditions are not met after this bit has been cleared, it will be set again).<br>**Remark:** This implementation does not cover the case if an initial SRP condition is violated when this bit is read and data-line pulsing is started. |
| 2 | DISCV | **Discharge $V_{BUS}$**: Set to logic 1 to discharge $V_{BUS}$. The device discharges $V_{BUS}$ before starting a new SRP. The discharge can take as long as 30 ms for $V_{BUS}$ to be charged less than 0.8 V. This bit must be cleared (write logic 0) before a session end. |
| 1 | VP | **$V_{BUS}$ Pulsing**: Used for $V_{BUS}$ pulsing to toggle VP to generate the required $V_{BUS}$ pulsing signal. This bit must be set for more than 16 ms and must be cleared before 26 ms. |
| 0 | OTG | **On-The-Go**:<br>**1 —** Enables the OTG function. The $V_{BUS}$ sensing functionality will be disabled.<br>**0 —** Normal operation. All OTG control bits will be masked. Status bits are undefined. |

[1] No interrupt is designed for OTG. The $V_{BUS}$ interrupt, however, may assert as a side effect during the $V_{BUS}$ pulsing.

When OTG is in progress, the $V_{BUS}$ interrupt may be set because $V_{BUS}$ is charged over the $V_{BUS}$ sensing threshold or the OTG host has turned on the $V_{BUS}$ supply to the device. Even if the $V_{BUS}$ interrupt is found during SRP, the device must complete data-line pulsing and $V_{BUS}$ pulsing before starting the B_SESSION_VALID detection.

OTG implementation applies to the device with self-power capability. If the device works in sharing mode, it must provide a switch circuit to supply power to the ISP1583 core during SRP.

#### 9.2.4.1 Session Request Protocol (SRP)

The ISP1583 can initiate an SRP. The B-device initiates SRP by data-line pulsing, followed by $V_{BUS}$ pulsing. The A-device can detect either data-line pulsing or $V_{BUS}$ pulsing.

The ISP1583 can initiate the B-device SRP by performing the following steps:

1. Set the OTG bit to start SRP.
2. Detect initial conditions by following the instructions given in bit INITCOND of the OTG register.
3. Start data-line pulsing: set bit DP of the OTG register to logic 1.
4. Wait for 5 ms to 10 ms.
5. Stop data-line pulsing: set bit DP of the OTG register to logic 0.
6. Start $V_{BUS}$ pulsing: set bit VP of the OTG register to logic 1.
7. Wait for 10 ms to 20 ms.
8. Stop $V_{BUS}$ pulsing: set bit VP of the OTG register to logic 0.
9. Discharge $V_{BUS}$ for about 30 ms: optional by using bit DISCV of the OTG register.
10. Detect bit BSESSVALID of the OTG register for a successful SRP with bit OTG cleared.
11. Once bit BSESSVALID is detected, turn on the SOFTCT bit to start normal bus enumeration.

The B-device must complete both data-line pulsing and $V_{BUS}$ pulsing within 100 ms.

**Remark:** When disabling OTG, data-line pulsing bit DP and $V_{BUS}$ pulsing bit VP must be cleared by writing logic 0.

### 9.2.5 Interrupt Enable register (address: 14h)

This register enables or disables individual interrupt sources. The interrupt for each endpoint can individually be controlled using the associated bits IEPnRX or IEPnTX, here n represents the endpoint number. All interrupts can be globally disabled using bit GLINTENA in the Mode register (see Table 24).

An interrupt is generated when the USB SIE receives or generates an ACK or NAK on the USB bus. The interrupt generation depends on Debug mode settings of bit fields CDBGMOD[1:0], DDBGMODIN[1:0] and DDBGMODOUT[1:0] in the Interrupt Configuration register.

All data IN transactions use the Transmit buffers (TX), which are handled by bits DDBGMODIN[1:0]. All data OUT transactions go through the Receive buffers (RX), which are handled by bits DDBGMODOUT[1:0]. Transactions on control endpoint 0 (IN, OUT and SETUP) are handled by bits CDBGMOD[1:0].

Interrupts caused by events on the USB bus (SOF, suspend, resume, bus reset, set up and high-speed status) can also be individually controlled. A bus reset disables all enabled interrupts, except bit IEBRST (bus reset), which remains logic 1.

The Interrupt Enable register consists of 4 bytes. The bit allocation is given in Table 32.

**Table 32. Interrupt Enable register: bit allocation**

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Symbol | reserved | | | | | | IEP7TX | IEP7RX |
| Reset | - | - | - | - | - | - | 0 | 0 |
| Bus reset | - | - | - | - | - | - | 0 | 0 |
| Access | - | - | - | - | - | - | R/W | R/W |
| **Bit** | **23** | **22** | **21** | **20** | **19** | **18** | **17** | **16** |
| Symbol | IEP6TX | IEP6RX | IEP5TX | IEP5RX | IEP4TX | IEP4RX | IEP3TX | IEP3RX |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| **Bit** | **15** | **14** | **13** | **12** | **11** | **10** | **9** | **8** |
| Symbol | IEP2TX | IEP2RX | IEP1TX | IEP1RX | IEP0TX | IEP0RX | reserved | IEP0SETUP |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | - | 0 |
| Bus reset | 0 | 0 | 0 | 0 | 0 | 0 | - | 0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | - | R/W |
| **Bit** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| Symbol | IEVBUS | IEDMA | IEHS_STA | IERESM | IESUSP | IEPSOF | IESOF | IEBRST |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**Table 33. Interrupt Enable register: bit description**

| Bit | Symbol | Description |
|---|---|---|
| 31 to 26 | - | reserved |
| 25 | IEP7TX | Logic 1 enables interrupt from the indicated endpoint. |
| 24 | IEP7RX | Logic 1 enables interrupt from the indicated endpoint. |
| 23 | IEP6TX | Logic 1 enables interrupt from the indicated endpoint. |
| 22 | IEP6RX | Logic 1 enables interrupt from the indicated endpoint. |
| 21 | IEP5TX | Logic 1 enables interrupt from the indicated endpoint. |
| 20 | IEP5RX | Logic 1 enables interrupt from the indicated endpoint. |
| 19 | IEP4TX | Logic 1 enables interrupt from the indicated endpoint. |
| 18 | IEP4RX | Logic 1 enables interrupt from the indicated endpoint. |
| 17 | IEP3TX | Logic 1 enables interrupt from the indicated endpoint. |
| 16 | IEP3RX | Logic 1 enables interrupt from the indicated endpoint. |
| 15 | IEP2TX | Logic 1 enables interrupt from the indicated endpoint. |
| 14 | IEP2RX | Logic 1 enables interrupt from the indicated endpoint. |
| 13 | IEP1TX | Logic 1 enables interrupt from the indicated endpoint. |
| 12 | IEP1RX | Logic 1 enables interrupt from the indicated endpoint. |
| 11 | IEP0TX | Logic 1 enables interrupt from control IN endpoint 0. |
| 10 | IEP0RX | Logic 1 enables interrupt from control OUT endpoint 0. |
| 9 | - | reserved |
| 8 | IEP0SETUP | Logic 1 enables interrupt for the set-up data received on endpoint 0. |

Table 33. Interrupt Enable register: bit description *...continued*

| Bit | Symbol | Description |
|---|---|---|
| 7 | IEVBUS | Logic 1 enables interrupt for $V_{BUS}$ sensing. |
| 6 | IEDMA | Logic 1 enables interrupt on the DMA Interrupt Reason register change detection. |
| 5 | IEHS_STA | Logic 1 enables interrupt on detecting a high-speed status change. |
| 4 | IERESM | Logic 1 enables interrupt on detecting a resume state. |
| 3 | IESUSP | Logic 1 enables interrupt on detecting a suspend state. |
| 2 | IEPSOF | Logic 1 enables interrupt on detecting a pseudo SOF. |
| 1 | IESOF | Logic 1 enables interrupt on detecting an SOF. |
| 0 | IEBRST | Logic 1 enables interrupt on detecting a bus reset. |

## 9.3 Data flow registers

### 9.3.1 Endpoint Index register (address: 2Ch)

The Endpoint Index register selects a target endpoint for register access by the microcontroller. The register consists of 1 byte, and the bit allocation is shown in Table 34.

The following registers are indexed:

- Buffer length
- Buffer status
- Control function
- Data port
- Endpoint MaxPacketSize
- Endpoint type

For example, to access the OUT data buffer of endpoint 1 using the Data Port register, the Endpoint Index register must first be written with 02h.

**Remark:** The Endpoint Index register and the DMA Endpoint register must not point to the same endpoint, irrespective of IN and OUT.

**Remark:** The delay time from the Write Endpoint Index register to the Read Data Port register must be at least 190 ns.

**Remark:** The delay time from the Write Endpoint Index register to the Write Data Port register must be at least 100 ns.

Table 34. Endpoint Index register: bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Symbol | reserved | | EP0SETUP | ENDPIDX[3:0] | | | | DIR |
| Reset | - | - | 1 | 0 | 0 | 0 | 0 | 0 |
| Bus reset | - | - | unchanged | 0 | 0 | 0 | 0 | 0 |
| Access | - | - | R/W | R/W | R/W | R/W | R/W | R/W |

Table 35. Endpoint Index register: bit description

| Bit | Symbol | Description |
|---|---|---|
| 7 to 6 | - | reserved |
| 5 | EP0SETUP | **Endpoint 0 Setup**: Selects the SETUP buffer for endpoint 0.<br>**0 —** Data buffer<br>**1 —** SETUP buffer<br>Must be logic 0 for access to endpoints other than set-up token buffer. |
| 4 to 1 | ENDPIDX[3:0] | **Endpoint Index**: Selects the target endpoint for register access of buffer length, buffer status, control function, data port, endpoint type and MaxPacketSize. |
| 0 | DIR | **Direction**: Sets the target endpoint as IN or OUT.<br>**0 —** Target endpoint refers to OUT (RX) FIFO<br>**1 —** Target endpoint refers to IN (TX) FIFO |

Table 36. Addressing of endpoint buffers

| Buffer name | EP0SETUP | ENDPIDX | DIR |
|---|---|---|---|
| SETUP | 1 | 00h | 0 |
| Control OUT | 0 | 00h | 0 |
| Control IN | 0 | 00h | 1 |
| Data OUT | 0 | 0Xh | 0 |
| Data IN | 0 | 0Xh | 1 |

### 9.3.2 Control Function register (address: 28h)

The Control Function register performs the buffer management on endpoints. It consists of 1 byte, and the bit configuration is given in Table 37. Register bits can stall, clear or validate any enabled endpoint. Before accessing this register, the Endpoint Index register must first be written to specify the target endpoint.

Table 37. Control Function register: bit allocation

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Symbol | reserved | | | CLBUF | VENDP | DSEN | STATUS | STALL |
| Reset | - | - | - | 0 | 0 | 0 | 0 | 0 |
| Bus reset | - | - | - | 0 | 0 | 0 | 0 | 0 |
| Access | - | - | - | R/W | R/W | W | R/W | R/W |

**Table 38. Control Function register: bit description**

| Bit | Symbol | Description |
|---|---|---|
| 7 to 5 | - | reserved. |
| 4 | CLBUF | **Clear Buffer**: Logic 1 clears the TX or RX buffer of the indexed endpoint. The RX buffer is automatically cleared once the endpoint is completely read. This bit is set only when it is necessary to forcefully clear the buffer.<br>**Remark:** If using double buffer, to clear both the buffers issue the CLBUF command two times. For details on clearing buffers, refer to Ref. 5 "ISP1582/83 and ISP1761 clearing an IN buffer (AN10045)". |
| 3 | VENDP | **Validate Endpoint**: Logic 1 validates data in the TX FIFO of an IN endpoint to send on the next IN token. In general, the endpoint is automatically validated when its FIFO byte count has reached endpoint MaxPacketSize. This bit is set only when it is necessary to validate the endpoint with the FIFO byte count, which is below endpoint MaxPacketSize.<br>**Remark:** Use either bit VENDP or register Buffer Length to validate endpoint FIFO with FIFO bytes. |
| 2 | DSEN | **Data Stage Enable**: This bit controls the response of the ISP1583 to a control transfer. After the completion of the set-up stage, firmware must determine whether a data stage is required. For control OUT, firmware will set this bit and the ISP1583 goes into the data stage. Otherwise, the ISP1583 will NAK the data stage transfer. For control IN, firmware will set this bit before writing data to the TX FIFO and validate the endpoint. If no data stage is required, firmware can immediately set the STATUS bit after the set-up stage.<br>**Remark:** The DSEN bit is cleared once the OUT token is acknowledged by the device and the IN token is acknowledged by the PC host. This bit cannot be read back and reading this bit will return logic 0. |
| 1 | STATUS | **Status Acknowledge**: Only applicable for control IN or OUT.<br>This bit controls the generation of ACK or NAK during the status stage of a SETUP transfer. It is automatically cleared when the status stage is completed, or when a SETUP token is received. No interrupt signal will be generated.<br>**0 —** Sends NAK<br>**1 —** Sends an empty packet following the IN token (peripheral-to-host) or ACK following the OUT token (host-to-peripheral)<br>**Remark:** The STATUS bit is cleared to zero once the zero-length packet is acknowledged by the device or the PC host.<br>**Remark:** Data transfers preceding the status stage must first be fully completed before the STATUS bit can be set. |
| 0 | STALL | **Stall Endpoint**: Logic 1 stalls the indexed endpoint. This bit is not applicable for isochronous transfers.<br>**Remark:** Stalling a data endpoint will confuse the Data Toggle bit about the stalled endpoint because the internal logic picks up from where it is stalled. Therefore, the Data Toggle bit must be reset by disabling and re-enabling the corresponding endpoint (by setting bit ENABLE to logic 0, followed by logic 1 in the Endpoint Type register) to reset the PID. |

### 9.3.3 Data Port register (address: 20h)

This 2-byte register provides direct access for a microcontroller to the FIFO of the indexed endpoint. The bit allocation is shown in Table 39.

**Peripheral-to-host (IN endpoint)**: After each write action, an internal counter is auto incremented (by two for a 16-bit access, by one for an 8-bit access) to the next location in the TX FIFO. When all bytes are written (FIFO byte count = endpoint MaxPacketSize), the

buffer is automatically validated. The data packet will then be sent on the next IN token. When it is necessary to validate the endpoint whose byte count is less than MaxPacketSize, it can be done using the Control Function register (bit VENDP) or the Buffer Length register.

**Remark:** The buffer can automatically be validated by using the Buffer Length register (see Table 41).

**Host-to-peripheral (OUT endpoint)**: After each read action, an internal counter is auto decremented (by two for a 16-bit access, by one for an 8-bit access) to the next location in the RX FIFO. When all bytes are read, buffer contents are automatically cleared. A new data packet can then be received on the next OUT token. Buffer contents can also be cleared using the Control Function register (bit CLBUF), when it is necessary to forcefully clear contents.

**Remark:** The delay time from the Write Endpoint Index register to the Read Data Port register must be at least 190 ns.

**Remark:** The delay time from the Write Endpoint Index register to the Write Data Port register must be at least 100 ns.

**Table 39. Data Port register: bit allocation**

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Symbol | DATAPORT[15:8] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| **Bit** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| Symbol | DATAPORT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**Table 40. Data Port register: bit description**

| Bit | Symbol | Description |
|---|---|---|
| 15 to 8 | DATAPORT[15:8] | data (upper byte) |
| 7 to 0 | DATAPORT[7:0] | data (lower byte) |

### 9.3.4 Buffer Length register (address: 1Ch)

This register determines the current packet size (DATACOUNT) of the indexed endpoint FIFO. The bit allocation is given in Table 41.

The Buffer Length register is automatically loaded with the FIFO size, when the Endpoint MaxPacketSize register is written (see Table 45). A smaller value can be written when required. After a bus reset, the Buffer Length register is made zero.

**IN endpoint**: When data transfer is performed in multiples of MaxPacketSize, the Buffer Length register is not significant. This register is useful only when transferring data that is not a multiple of MaxPacketSize. The following two examples demonstrate the significance of the Buffer Length register.

Example 1: Consider that the transfer size is 512 bytes and the MaxPacketSize is programmed as 64 bytes, the Buffer Length register need not be filled. This is because the transfer size is a multiple of MaxPacketSize, and MaxPacketSize packets will be automatically validated because the last packet is also of MaxPacketSize.

Example 2: Consider that the transfer size is 510 bytes and the MaxPacketSize is programmed as 64 bytes, the Buffer Length register must be filled with 62 bytes just before the microprocessor writes the last packet of 62 bytes. This ensures that the last packet, which is a short packet of 62 bytes, is automatically validated.

Use bit VENDP in the Control register if you are not using the Buffer Length register.

This is applicable only to PIO mode access.

**OUT endpoint**: The DATACOUNT value is automatically initialized to the number of data bytes sent by the host on each ACK.

**Remark:** When using a 16-bit microprocessor bus, the last byte of an odd-sized packet is output as the lower byte (LSByte).

**Remark:** Buffer Length is valid only after an interrupt is generated for the OUT endpoint.

**Table 41. Buffer Length register: bit allocation**

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Symbol | DATACOUNT[15:8] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| **Bit** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| Symbol | DATACOUNT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**Table 42. Buffer Length register: bit description**

| Bit | Symbol | Description |
|---|---|---|
| 15 to 0 | DATACOUNT[15:0] | **Data Count**: Determines the current packet size of the indexed endpoint FIFO. |

### 9.3.5 Buffer Status register (address: 1Eh)

This register is accessed using index. The endpoint index must first be set before accessing this register for the corresponding endpoint. It reflects the status of the double buffered endpoint FIFO.

**Remark:** This register is not applicable to the control endpoint.

**Remark:** For endpoint IN data transfer, firmware must ensure a 200 ns delay between writing of the data packet and reading the Buffer Status register. For endpoint OUT data transfer, firmware must also ensure a 200 ns delay between receiving the endpoint interrupt and reading the Buffer Status register. For more information, refer to Ref. 3 "Using ISP1582/3 in a composite device application with alternate settings (AN10071)".

Table 43 shows the bit allocation of the Buffer Status register.

**Table 43. Buffer Status register: bit allocation**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Symbol | reserved | | | | | | BUF1 | BUF0 |
| Reset | - | - | - | - | - | - | 0 | 0 |
| Bus reset | - | - | - | - | - | - | 0 | 0 |
| Access | - | - | - | - | - | - | R | R |

**Table 44. Buffer Status register: bit description**

| Bit | Symbol | Description |
|---|---|---|
| 7 to 2 | - | reserved |
| 1 to 0 | BUF[1:0] | **Buffer**:<br>**00 —** Buffers are not filled.<br>**01 —** One of the buffers is filled.<br>**10 —** One of the buffers is filled.<br>**11 —** Both the buffers are filled. |

### 9.3.6 Endpoint MaxPacketSize register (address: 04h)

This register determines the maximum packet size for all endpoints, except set-up token buffer, control IN and control OUT. The register contains 2 bytes, and the bit allocation is given in Table 45.

Each time the register is written, the Buffer Length register of the corresponding endpoint is re-initialized to the FFOSZ field value. Bits NTRANS control the number of transactions allowed in a single microframe (for high-speed isochronous and interrupt endpoints only).

**Table 45. Endpoint MaxPacketSize register: bit allocation**

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Symbol | reserved | | | NTRANS[1:0] | | FFOSZ[10:8] | | |
| Reset | - | - | - | 0 | 0 | 0 | 0 | 0 |
| Bus reset | - | - | - | 0 | 0 | 0 | 0 | 0 |
| Access | - | - | - | R/W | R/W | R/W | R/W | R/W |
| **Bit** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| Symbol | FFOSZ[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

Table 46. Endpoint MaxPacketSize register: bit description

| Bit | Symbol | Description |
|---|---|---|
| 15 to 13 | - | reserved |
| 12 to 11 | NTRANS[1:0] | **Number of Transactions** (HS mode only).<br>**00 —** 1 packet per microframe<br>**01 —** 2 packets per microframe<br>**10 —** 3 packets per microframe<br>**11 —** reserved<br>These bits are applicable only for isochronous or interrupt transactions. |
| 10 to 0 | FFOSZ[10:0] | **FIFO Size**: Sets the FIFO size, in bytes, for the indexed endpoint. Applies to both high-speed and full-speed operations. |

The ISP1583 supports all the transfers given in Ref. 1 “Universal Serial Bus Specification Rev. 2.0”.

Each programmable FIFO can be independently configured using its Endpoint MaxPacketSize register (R/W: 04h), but the total physical size of all enabled endpoints (IN plus OUT), including set-up token buffer, control IN and control OUT, must not exceed 8192 bytes.

### 9.3.7 Endpoint Type register (address: 08h)

This register sets the endpoint type of the indexed endpoint: isochronous, bulk or interrupt. It also serves to enable the endpoint and configure it for double buffering. Automatic generation of an empty packet for a zero-length TX buffer can be disabled using bit NOEMPKT. The register contains 2 bytes, and the bit allocation is shown in Table 47.

Table 47. Endpoint Type register: bit allocation

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Symbol | reserved | | | | | | | |
| Reset | - | - | - | - | - | - | - | - |
| Bus reset | - | - | - | - | - | - | - | - |
| Access | - | - | - | - | - | - | - | - |
| **Bit** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| Symbol | reserved | | | NOEMPKT | ENABLE | DBLBUF | ENDPTYP[1:0] | |
| Reset | - | - | - | 0 | 0 | 0 | 0 | 0 |
| Bus reset | - | - | - | 0 | 0 | 0 | 0 | 0 |
| Access | - | - | - | R/W | R/W | R/W | R/W | R/W |

Table 48. Endpoint Type register: bit description

| Bit | Symbol | Description |
|---|---|---|
| 15 to 5 | - | reserved |
| 4 | NOEMPKT | **No Empty Packet**: Logic 0 causes the ISP1583 to return a null length packet for the IN token after the DMA IN transfer is complete. For ATA mode or the DMA IN transfer, which does not require a null length packet after DMA completion, set to logic 1 to disable the generation of the null length packet. |
| 3 | ENABLE | **Endpoint Enable**: Logic 1 enables the FIFO of the indexed endpoint. The memory size is allocated as specified in the Endpoint MaxPacketSize register. Logic 0 disables the FIFO.<br>**Remark:** Stalling a data endpoint will confuse the Data Toggle bit on the stalled endpoint because the internal logic picks up from where it has stalled. Therefore, the Data Toggle bit must be reset by disabling and re-enabling the corresponding endpoint (by setting bit ENABLE to logic 0, followed by logic 1 in the Endpoint Type register) to reset the PID. |
| 2 | DBLBUF | **Double Buffering**: Logic 1 enables double buffering for the indexed endpoint. Logic 0 disables double buffering.<br>**Remark:** When performing a write to two empty buffers, ensure that a minimum of 200 ns delay is provided from the last write of the first buffer to the first write of the second buffer. Otherwise, the first few data bytes may not be written to the second buffer, causing data corruption. |
| 1 to 0 | ENDPTYP[1:0] | **Endpoint Type**: These bits select the endpoint type.<br>**00 —** not used<br>**01 —** Isochronous<br>**10 —** Bulk<br>**11 —** Interrupt |

## 9.4 DMA registers

Two types of Generic DMA transfers and three types of IDE-specified transfers can be done by writing the proper opcode in the DMA Command register.

Control bits are given in Table 49 (Generic DMA transfers) and Table 50 (IDE-specified transfers).

**GDMA read/write (opcode = 00h/01h) —** Generic DMA slave mode. Depending on the MODE[1:0] bits set in the DMA Configuration register, the DACK, DIOR or DIOW signal strobes data. These signals are driven by the external DMA controller.

GDMA slave mode can operate in either counter mode or EOT-only mode.

In counter mode, bit DIS_XFER_CNT in the DMA Configuration register must be set to logic 0. The DMA Transfer Counter register must be programmed before any DMA command is issued. The DMA transfer counter is set by writing from the LSByte to the MSByte (address: 34h to 37h). The DMA transfer count is internally updated only after the MSByte is written. Once the DMA transfer is started, the transfer counter starts decrementing and on reaching 0, bit DMA_XFER_OK is set and an interrupt is generated by the ISP1583. If the DMA master wishes to terminate the DMA transfer, it can issue an EOT signal to the ISP1583. This EOT signal overrides the transfer counter and can terminate the DMA transfer at any time.

In EOT-only mode, DIS_XFER_CNT must be set to logic 1. Although the DMA transfer counter can still be programmed, it will not have any effect on the DMA transfer. The DMA transfer will start once the DMA command is issued. Any of the following three ways will terminate this DMA transfer:

- Detecting an external EOT
- Detecting an internal EOT (short packet on an OUT token)
- Issuing a GDMA stop command

There are three interrupts programmable to differentiate the method of DMA termination: bits INT_EOT, EXT_EOT and DMA_XFER_OK in the DMA Interrupt Reason register (see Table 74).

**MDMA (master) read/write (opcode = 06h/07h) —** Generic DMA master mode. Depending on the MODE[1:0] bits set in the DMA Configuration register, the DACK, DIOR or DIOW signal strobes data. These signals are driven by the ISP1583.

In master mode, BURSTCOUNTER[12:0] in the DMA Burst Counter register, DIS_XFER_CNT in the DMA Configuration register and the external EOT signal are not applicable. The DMA transfer counter is always enabled and bit DMA_XFER_OK is set to 1 once the counter reaches 0.

**MDMA read/write (opcode = 06h/07h) —** Multi-word DMA mode for IDE transfers. The specification of this mode can be obtained from Ref. 4 “AT Attachment with Packet Interface Extension (ATA/ATAPI-4), ANSI INCITS 317-1998 (R2003)”. DIOR and DIOW are used as data strobes, while DREQ and DACK serve as handshake signals.

**Table 49. Control bits for Generic DMA transfers**

| Control bits | Description | | Reference |
|---|---|---|---|
| | GDMA read/write (opcode = 00h/01h) | MDMA (master) read/write (opcode = 06h/07h) | |
| **DMA Configuration register** | | | |
| ATA_MODE | set to logic 0 (non-ATA transfer) | set to logic 1 (ATA transfer) | Table 56 |
| DMA_MODE[1:0] | - | determines MDMA timing for DIOR and DIOW strobes | |
| DIS_XFER_CNT | disables use of DMA transfer counter | disables use of DMA transfer counter | |
| MODE[1:0] | determines active read/write data strobe signals | determines active data strobe(s) | |
| WIDTH | selects DMA bus width: 8 or 16 bits | selects DMA bus width: 8 or 16 bits | |

**Table 49. Control bits for Generic DMA transfers** *...continued*

| Control bits | Description | | Reference |
|---|---|---|---|
| | GDMA read/write (opcode = 00h/01h) | MDMA (master) read/write (opcode = 06h/07h) | |
| **DMA Hardware register** | | | |
| ENDIAN[1:0] | determines whether data is to be byte swapped or normal; applicable only in 16-bit mode | determines whether data is to be byte swapped or normal; applicable only in 16-bit mode | Table 58 |
| EOT_POL | selects polarity of the EOT signal | input EOT is not used | |
| MASTER | set to logic 0 (slave) | set to logic 1 (master) | |
| ACK_POL, DREQ_POL, WRITE_POL, READ_POL | selects polarity of DMA handshake signals | selects polarity of DMA handshake signals | |

**Table 50. Control bits for IDE-specified DMA transfers**

| Control bits | Description | Reference |
|---|---|---|
| | MDMA read/write (opcode = 06h/07h) | |
| **DMA Configuration register** | | |
| ATA_MODE | set to logic 1 (ATA transfer) | Table 56 |
| DMA_MODE[1:0] | selects MDMA mode; timing are ATA(PI) compatible | |
| PIO_MODE[2:0] | selects PIO mode; timing are ATA(PI) compatible | |
| **DMA Hardware register** | | |
| MASTER | set to logic 0 | Table 58 |

**Remark:** The DMA bus defaults to 3-state, until a DMA command is executed. All the other control signals are not 3-stated.

### 9.4.1 DMA Command register (address: 30h)

The DMA Command register is a 1-byte register (for bit allocation, see Table 51) that initiates all DMA transfer activity on the DMA controller. The register is write-only: reading it will return FFh.

**Remark:** The DMA bus will be in 3-state, until a DMA command is executed.

**Table 51. DMA Command register: bit allocation**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **Symbol** | DMA_CMD[7:0] | | | | | | | |
| **Reset** | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| **Bus reset** | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| **Access** | W | W | W | W | W | W | W | W |

**Table 52. DMA Command register: bit description**

| Bit | Symbol | Description |
|---|---|---|
| 7 to 0 | DMA_CMD[7:0] | DMA command code, see Table 53.<br>PIO read or write that started using the DMA Command register only performs a 16-bit transfer. |

**Table 53. DMA commands**

| Code | Name | Description |
|---|---|---|
| 00h | GDMA Read | **Generic DMA IN token transfer (slave mode only)**: Data is transferred from the external DMA bus to the internal buffer. Strobe: DIOW by the external DMA controller. |
| 01h | GDMA Write | **Generic DMA OUT token transfer (slave mode only)**: Data is transferred from the internal buffer to the external DMA bus. Strobe: DIOR by the external DMA controller. |
| 02h to 05h | - | reserved |
| 06h | MDMA Read | **Multi-word DMA Read**: Data is transferred from the external DMA bus to the internal buffer. |
| 07h | MDMA Write | **Multi-word DMA Write**: Data is transferred from the internal buffer to the external DMA bus. |
| 0Ah | Read 1F0 | **Read at address 1F0h**: Initiates a PIO read cycle from Task File 1F0. Before issuing this command, the task file byte count must be programmed at address 1F4h (LSByte) and 1F5h (MSByte). |
| 0Bh | Poll BSY | **Poll BSY status bit for ATAPI device**: Starts repeated PIO read commands to poll the BSY status bit of the ATAPI device. When BSY = 0, polling is terminated and an interrupt is generated. The interrupt can be masked but the interrupt bit will still be set. Therefore, you can manually poll this interrupt bit. |
| 0Ch | Read Task Files | **Read Task Files**: Reads all task files. When Task File Index is set to logic 0, this command reads all registers, except 1F0h and 1F7h. If Task File Index is not logic 0, the Task register of the address set in the Task File register will be read. When the reading is completed, an interrupt is generated. The interrupt can be masked off, however, the interrupt bit will still be set. Therefore, you can manually poll this interrupt bit. |
| 0Dh | - | reserved |
| 0Eh | Validate Buffer | **Validate Buffer (for debugging only)**: Request from the microcontroller to validate the endpoint buffer, following an ATA-to-USB data transfer. |
| 0Fh | Clear Buffer | **Clear Buffer**: Request from the microcontroller to clear the endpoint buffer, after a DMA-to-USB data transfer. Logic 1 clears the TX buffer of the indexed endpoint; the RX buffer is not affected. The TX buffer is automatically cleared once data is sent on the USB bus. This bit is set only when it is necessary to forcefully clear the buffer.<br>**Remark:** If using double buffer, to clear both the buffers issue the Clear Buffer command two times, that is, set and clear this bit two times. |
| 10h | Restart | **Restart**: Request from the microcontroller to move the buffer pointers to the beginning of the endpoint FIFO. |

**Table 53. DMA commands** *…continued*

| Code | Name | Description |
|---|---|---|
| 11h | Reset DMA | **Reset DMA**: Initializes the DMA core to its power-on reset state.<br>**Remark:** When the DMA core is reset during the Reset DMA command, the DREQ, DACK, DIOW and DIOR handshake pins will temporarily be asserted. This can confuse the external DMA controller. To prevent this, start the external DMA controller **only after** the DMA reset. |
| 12h | MDMA stop | **MDMA stop**: This command immediately stops the MDMA data transfer. This is applicable for commands 06h and 07h only. |
| 13h | GDMA stop | **GDMA stop**: This command stops the GDMA data transfer. Any data in the OUT endpoint that is not transferred by the DMA will remain in the buffer. The FIFO data for the IN endpoint will be written to the endpoint buffer. An interrupt bit will be set to indicate the completion of the DMA Stop command.<br>**Remark:** For the DMA OUT transfer, if the DMA Burst Counter register is programmed to some value, for example 512 bytes, and if a GDMA Stop command is issued in the middle of a transfer, the transfer will continue until the end of the burst size (512 bytes). Issuing a GDMA Stop command does not allow the ISP1583 to stop in the middle of the burst. It can only be stopped in between bursts. |
| 14h to 20h | - | reserved |
| 21h | Read Task File register 1F1h | **Read Task File register 1F1h**: When reading is completed, an interrupt is generated. |
| 22h | Read Task File register 1F2h | **Read Task File register 1F2h**: When reading is completed, an interrupt is generated. |
| 23h | Read Task File register 1F3h | **Read Task File register 1F3h**: When reading is completed, an interrupt is generated. |
| 24h | Read Task File register 1F4h | **Read Task File register 1F4h**: When reading is completed, an interrupt is generated. |
| 25h | Read Task File register 1F5h | **Read Task File register 1F5h**: When reading is completed, an interrupt is generated. |
| 26h | Read Task File register 1F6h | **Read Task File register 1F6h**: When reading is completed, an interrupt is generated. |
| 27h | Read Task File register 3F6h | **Read Task File register 3F6h**: When reading is completed, an interrupt is generated. |
| 28h | Read Task File register 3F7h | **Read Task File register 3F7h**: When reading is completed, an interrupt is generated. |
| 29h to FFh | - | reserved |

### 9.4.2 DMA Transfer Counter register (address: 34h)

This 4-byte register sets up the total byte count for a DMA transfer (DMACR). It indicates the remaining number of bytes left for transfer. The bit allocation is given in Table 54.

**For IN endpoint —** Because there is a FIFO in the ISP1583 DMA controller, some data may remain in the FIFO during the DMA transfer. The maximum FIFO size is 8 bytes, and the maximum delay time for data to be shifted to endpoint buffer is 60 ns.

**For OUT endpoint —** Data will not be cleared from the endpoint buffer, until all the data is read from the DMA FIFO.

If the DMA counter is disabled in the DMA transfer, it will still decrement and rollover when it reaches zero.

**Table 54. DMA Transfer Counter register: bit allocation**

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Symbol | DMACR4 = DMACR[31:24] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| **Bit** | **23** | **22** | **21** | **20** | **19** | **18** | **17** | **16** |
| Symbol | DMACR3 = DMACR[23:16] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| **Bit** | **15** | **14** | **13** | **12** | **11** | **10** | **9** | **8** |
| Symbol | DMACR2 = DMACR[15:8] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| **Bit** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| Symbol | DMACR1 = DMACR[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**Table 55. DMA Transfer Counter register: bit description**

| Bit | Symbol | Description |
|---|---|---|
| 31 to 24 | DMACR4 = DMACR[31:24] | DMA transfer counter byte 4 (MSByte) |
| 23 to 16 | DMACR3 = DMACR[23:16] | DMA transfer counter byte 3 |
| 15 to 8 | DMACR2 = DMACR[15:8] | DMA transfer counter byte 2 |
| 7 to 0 | DMACR1 = DMACR[7:0] | DMA transfer counter byte 1 (LSByte) |

### 9.4.3 DMA Configuration register (address: 38h)

This register defines the DMA configuration for GDMA mode. The DMA Configuration register consists of 2 bytes. The bit allocation is given in Table 56.

**Table 56. DMA Configuration register: bit allocation**

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Symbol | reserved | | ATA_MODE | DMA_MODE[1:0] | | PIO_MODE[2:0] | | |
| Reset | - | - | 0 | 0 | 0 | 0 | 0 | 0 |
| Bus reset | - | - | 0 | 0 | 0 | 0 | 0 | 0 |
| Access | - | - | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Symbol | DIS_ XFER_CNT | reserved | | | MODE[1:0] | | reserved | WIDTH |
| Reset | 0 | - | - | - | 0 | 0 | - | 1 |
| Bus reset | 0 | - | - | - | 0 | 0 | - | 1 |
| Access | R/W | - | - | - | R/W | R/W | - | R/W |

**Table 57. DMA Configuration register: bit description**

| Bit | Symbol | Description[1] |
|---|---|---|
| 15 to 14 | - | reserved |
| 13 | ATA_MODE | **ATA Mode**: Mode selection of the DMA core.<br>**0 —** Configures the DMA core for non-ATA mode. Used when issuing DMA commands 00h and 01h.<br>**1 —** Configures the DMA core for ATA or MDMA mode. Used when issuing DMA commands 02h to 07h, 0Ah and 0Ch; also used when directly accessing Task File registers. |
| 12 to 11 | DMA_MODE [1:0] | **DMA Mode**: These bits affect the timing for MDMA mode.<br>**00 —** MDMA mode 0: ATA(PI) compatible timing<br>**01 —** MDMA mode 1: ATA(PI) compatible timing<br>**10 —** MDMA mode 2: ATA(PI) compatible timing<br>**11 —** MDMA mode 3: enables the DMA Strobe Timing register (see Table 78 and Table 79) for non-standard strobe durations; only used in MDMA mode |
| 10 to 8 | PIO_MODE [2:0][2] | **PIO Mode**: These bits affect the PIO timing.<br>**000 to 100 —** PIO mode 0 to 4: ATA(PI) compatible timing<br>**101 to 111 —** reserved |
| 7 | DIS_XFER_ CNT | **Disable Transfer Count**: Logic 1 disables the DMA Transfer Counter (see Table 54). The transfer counter can be disabled only in GDMA slave mode; in master mode the counter is always enabled. |
| 6 to 4 | - | reserved |
| 3 to 2 | MODE[1:0] | **Mode**: These bits only affect GDMA (slave) and MDMA (master) handshake signals.<br>**00 —** DIOR (master) or DIOW (slave): strobes data from the DMA bus into the ISP1583; DIOW (master) or DIOR (slave): puts data from the ISP1583 on the DMA bus.<br>**01 —** DIOR (master) or DACK (slave): strobes data from the DMA bus into the ISP1583; DACK (master) or DIOR (slave): puts data from the ISP1583 on the DMA bus.<br>**10 —** DACK (master and slave): strobes data from the DMA bus into the ISP1583 and also puts data from the ISP1583 on the DMA bus.<br>**11 —** reserved |
| 1 | - | reserved |
| 0 | WIDTH | **Width**: This bit selects the DMA bus width for GDMA (slave) and MDMA (master).<br>**0 —** 8-bit data bus<br>**1 —** 16-bit data bus |

[1] The DREQ pin will be driven only after performing a write access to the DMA Configuration register (that is, after configuring the DMA Configuration register).

[2] PIO read or write that started using the DMA Command register only performs 16-bit transfer.

### 9.4.4 DMA Hardware register (address: 3Ch)

The DMA Hardware register consists of 1 byte. The bit allocation is shown in Table 58.

This register determines the polarity of bus control signals (EOT, DACK, DREQ, DIOR and DIOW) and DMA mode (master or slave). It also controls whether the upper and lower parts of the data bus are swapped (bits ENDIAN[1:0]), for modes GDMA (slave) and MDMA (master) only.

**Table 58. DMA Hardware register: bit allocation**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Symbol | ENDIAN[1:0] | | EOT_POL | MASTER | ACK_POL | DREQ_POL | WRITE_POL | READ_POL |
| Reset | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| Bus reset | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**Table 59. DMA Hardware register: bit description**

| Bit | Symbol | Description |
|---|---|---|
| 7 to 6 | ENDIAN[1:0] | **Endian**: These bits determine whether the data bus is swapped between the internal RAM and the DMA bus. This only applies for modes GDMA (slave) and MDMA (master).<br>**00 —** Normal data representation; 16-bit bus: MSByte on DATA[15:8] and LSByte on DATA[7:0].<br>**01 —** Swapped data representation; 16-bit bus: MSByte on DATA[7:0] and LSByte on DATA[15:8].<br>**10 —** reserved<br>**11 —** reserved<br>**Remark:** While operating with the 8-bit data bus, bits ENDIAN[1:0] must always be set to logic 00. |
| 5 | EOT_POL | **EOT Polarity**: Selects the polarity of the End-Of-Transfer input; used in GDMA slave mode only.<br>**0 —** EOT is active LOW<br>**1 —** EOT is active HIGH |
| 4 | MASTER | **Master or Slave Selection**: Selects DMA master or slave mode.<br>**0 —** GDMA slave mode<br>**1 —** MDMA master mode |
| 3 | ACK_POL | **Acknowledgment Polarity**: Selects the DMA acknowledgment polarity.<br>**0 —** DACK is active LOW<br>**1 —** DACK is active HIGH |

Table 59. DMA Hardware register: bit description *...continued*

| Bit | Symbol | Description |
|---|---|---|
| 2 | DREQ_POL | **DREQ Polarity**: Selects the DMA request polarity.<br>**0 —** DREQ is active LOW<br>**1 —** DREQ is active HIGH |
| 1 | WRITE_POL | **Write Polarity**: Selects the DIOW strobe polarity.<br>**0 —** DIOW is active LOW<br>**1 —** DIOW is active HIGH |
| 0 | READ_POL | **Read Polarity**: Selects the DIOR strobe polarity.<br>**0 —** DIOR is active LOW<br>**1 —** DIOR is active HIGH |

### 9.4.5 Task File registers (addresses: 40h to 4Fh)

These registers allow direct access to the internal registers of an ATAPI peripheral using PIO mode. The supported Task File registers and their functions are shown in Table 60. The correct peripheral register is automatically addressed using pins CS1_N, CS0_N, DA2, MODE0/DA1 and BUS_CONF/DA0 (see Table 61).

Table 60. Task File register functions

| Task file | ATA function | ATAPI function |
|---|---|---|
| 1F0 | data (16-bit) | data (16-bit) |
| 1F1 | error/feature | error/feature |
| 1F2 | sector count | interrupt reason |
| 1F3 | sector number/LBA[7:0] | reserved |
| 1F4 | cylinder low/LBA[15:8] | cylinder low |
| 1F5 | cylinder high/LBA[23:16] | cylinder high |
| 1F6 | drive/head/LBA[27:24] | drive select |
| 1F7 | command | status/command |
| 3F6 | alternate status/command | alternate status/command |
| 3F7 | drive address | reserved |

Table 61. ATAPI peripheral register addressing

| Task file | CS1_N | CS0_N | DA2 | MODE0/DA1 | BUS_CONF/DA0 |
|---|---|---|---|---|---|
| 1F0 | HIGH | LOW | LOW | LOW | LOW |
| 1F1 | HIGH | LOW | LOW | LOW | HIGH |
| 1F2 | HIGH | LOW | LOW | HIGH | LOW |
| 1F3 | HIGH | LOW | LOW | HIGH | HIGH |
| 1F4 | HIGH | LOW | HIGH | LOW | LOW |
| 1F5 | HIGH | LOW | HIGH | LOW | HIGH |
| 1F6 | HIGH | LOW | HIGH | HIGH | LOW |
| 1F7 | HIGH | LOW | HIGH | HIGH | HIGH |
| 3F6 | LOW | HIGH | HIGH | HIGH | LOW |
| 3F7 | LOW | HIGH | HIGH | HIGH | HIGH |

In 8-bit bus mode, 16-bit Task File register 1F0 requires two consecutive write/read accesses before the proper PIO write/read is generated on the IDE interface. The first byte is always the lower byte (LSByte). Other Task File registers can directly be accessed.

Writing to Task File registers can be done in any order, except for the Task File register 1F7, which must be written last.

**Table 62. Task File 1F0 register (address: 40h): bit allocation**

*CS1_N = HIGH, CS0_N = LOW, DA2 = LOW, MODE0/DA1 = LOW, BUS_CONF/DA0 = LOW.*

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Symbol | data (ATA or ATAPI) | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**Table 63. Task File 1F1 register (address: 48h): bit allocation**

*CS1_N = HIGH, CS0_N = LOW, DA2 = LOW, MODE0/DA1 = LOW, BUS_CONF/DA0 = HIGH.*

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Symbol | error/feature (ATA or ATAPI) | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**Table 64. Task File 1F2 register (address: 49h): bit allocation**

*CS1_N = HIGH, CS0_N = LOW, DA2 = LOW, MODE0/DA1 = HIGH, BUS_CONF/DA0 = LOW.*

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Symbol | sector count (ATA) or interrupt reason (ATAPI) | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**Table 65. Task File 1F3 register (address: 4Ah): bit allocation**

*CS1_N = HIGH, CS0_N = LOW, DA2 = LOW, MODE0/DA1 = HIGH, BUS_CONF/DA0 = HIGH.*

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Symbol | sector number/LBA[7:0] (ATA), reserved (ATAPI) | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**Table 66. Task File 1F4 register (address: 4Bh): bit allocation**

*CS1_N = HIGH, CS0_N = LOW, DA2 = HIGH, MODE0/DA1 = LOW, BUS_CONF/DA0 = LOW.*

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Symbol | cylinder low/LBA[15:8] (ATA) or cylinder low (ATAPI) | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**Table 67. Task File 1F5 register (address: 4Ch): bit allocation**

*CS1_N = HIGH, CS0_N = LOW, DA2 = HIGH, MODE0/DA1 = LOW, BUS_CONF/DA0 = HIGH.*

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Symbol | cylinder high/LBA[23:16] (ATA) or cylinder high (ATAPI) | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**Table 68. Task File 1F6 register (address: 4Dh): bit allocation**

*CS1_N = HIGH, CS0_N = LOW, DA2 = HIGH, MODE0/DA1 = HIGH, BUS_CONF/DA0 = LOW.*

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Symbol | drive/head/LBA[27:24] (ATA) or drive (ATAPI) | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**Table 69. Task File 1F7 register (address: 44h): bit allocation**

*CS1_N = HIGH, CS0_N = LOW, DA2 = HIGH, MODE0/DA1 = HIGH, BUS_CONF/DA0 = HIGH.*

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Symbol | command (ATA) or status[1]/command (ATAPI) | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Access | W | W | W | W | W | W | W | W |

[1] Task File register 1F7 is a write-only register; a read will return FFh.

**Table 70. Task File 3F6 register (address: 4Eh): bit allocation**

*CS1_N = LOW, CS0_N = HIGH, DA2 = HIGH, MODE0/DA1 = HIGH, BUS_CONF/DA0 = LOW.*

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Symbol | alternate status/command (ATA or ATAPI) | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**Table 71. Task File 3F7 register (address: 4Fh): bit allocation**

*CS1_N = LOW, CS0_N = HIGH, DA2 = HIGH, MODE0/DA1 = HIGH, BUS_CONF/DA0 = HIGH.*

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Symbol | drive address (ATA) or reserved (ATAPI) | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

### 9.4.6 DMA Interrupt Reason register (address: 50h)

This 2-byte register shows the source(s) of DMA interrupt. Each bit is refreshed after a DMA command is executed. An interrupt source is cleared by writing logic 1 to the corresponding bit. On detecting the interrupt, the external microprocessor must read the DMA Interrupt Reason register and mask it with the corresponding bits in the DMA Interrupt Enable register to determine the source of the interrupt.

The bit allocation is given in Table 72.

**Table 72. DMA Interrupt Reason register: bit allocation**

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Symbol | TEST3 | reserved | | GDMA_ STOP | EXT_EOT | INT_EOT | INTRQ_ PENDING | DMA_ XFER_OK |
| Reset | 0 | - | - | 0 | 0 | 0 | 0 | 0 |
| Bus reset | 0 | - | - | 0 | 0 | 0 | 0 | 0 |
| Access | R | - | - | R/W | R/W | R/W | R/W | R/W |
| **Bit** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| Symbol | reserved | | | READ_1F0 | BSY_ DONE | TF_RD_ DONE | CMD_ INTRQ_OK | reserved |
| Reset | - | - | - | 0 | 0 | 0 | 0 | - |
| Bus reset | - | - | - | 0 | 0 | 0 | 0 | - |
| Access | - | - | - | R/W | R/W | R/W | R/W | - |

**Table 73. DMA Interrupt Reason register: bit description**

| Bit | Symbol | Description |
|---|---|---|
| 15 | TEST3 | This bit is set when a DMA transfer for a packet (OUT transfer) terminates before the whole packet is transferred. This bit is a status bit, and the corresponding mask bit of this register is always 0. Writing any value other than 0 has no effect. |
| 14 to 13 | - | reserved |
| 12 | GDMA_STOP | **GDMA Stop**: When the GDMA_STOP command is issued to DMA Command registers, it means the DMA transfer has successfully terminated. |
| 11 | EXT_EOT | **External EOT**: Logic 1 indicates that an external EOT is detected. This is applicable only in GDMA slave mode. |
| 10 | INT_EOT | **Internal EOT**: Logic 1 indicates that an internal EOT is detected; see Table 74. |
| 9 | INTRQ_ PENDING | **Interrupt Pending**: Logic 1 indicates that a pending interrupt was detected on pin INTRQ. |
| 8 | DMA_XFER_OK | **DMA Transfer OK**: Logic 1 indicates that the DMA transfer is completed (DMA Transfer Counter has become zero). This bit is only used in GDMA (slave) mode and MDMA (master) mode. |
| 7 to 5 | - | reserved |
| 4 | READ_1F0 | **Read 1F0**: Logic 1 indicates that the 1F0 FIFO contains unread data and the microcontroller can start reading data. |
| 3 | BSY_DONE | **Busy Done**: Logic 1 indicates that the BSY status bit has become zero and polling has been stopped. |

Table 73. DMA Interrupt Reason register: bit description *...continued*

| Bit | Symbol | Description |
|---|---|---|
| 2 | TF_RD_DONE | **Task File Read Done**: Logic 1 indicates that the Read Task Files command has been completed. |
| 1 | CMD_INTRQ_OK | **Command Interrupt OK**: Logic 1 indicates that all bytes from the FIFO have been transferred (DMA Transfer Count zero) and an interrupt on pin INTRQ was detected. |
| 0 | - | reserved |

Table 74. Internal EOT-functional relation with DMA_XFER_OK bit

| INT_EOT | DMA_XFER_OK | Description |
|---|---|---|
| 1 | 0 | During the DMA transfer, there is a premature termination with short packet. |
| 1 | 1 | DMA transfer is completed with short packet and the DMA transfer counter has reached 0. |
| 0 | 1 | DMA transfer is completed without any short packet and the DMA transfer counter has reached 0. |

### 9.4.7 DMA Interrupt Enable register (address: 54h)

This 2-byte register controls the interrupt generation of the source bits in the DMA Interrupt Reason register (see Table 72). The bit allocation is given in Table 75. The bit description is given in Table 73.

Logic 1 enables the interrupt generation. After a bus reset, interrupt generation is disabled, with values turning to logic 0.

Table 75. DMA Interrupt Enable register: bit allocation

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Symbol | TEST4 | reserved | | IE_GDMA_STOP | IE_EXT_EOT | IE_INT_EOT | IE_INTRQ_PENDING | IE_DMA_XFER_OK |
| Reset | 0 | - | - | 0 | 0 | 0 | 0 | 0 |
| Bus reset | 0 | - | - | 0 | 0 | 0 | 0 | 0 |
| Access | R | - | - | R/W | R/W | R/W | R/W | R/W |
| **Bit** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| Symbol | reserved | | | IE_READ_1F0 | IE_BSY_DONE | IE_TF_RD_DONE | IE_CMD_INTRQ_OK | reserved |
| Reset | - | - | - | 0 | 0 | 0 | 0 | - |
| Bus reset | - | - | - | 0 | 0 | 0 | 0 | - |
| Access | - | - | - | R/W | R/W | R/W | R/W | - |

### 9.4.8 DMA Endpoint register (address: 58h)

This 1-byte register selects a USB endpoint FIFO as a source or destination for DMA transfers. The bit allocation is given in Table 76.

**Table 76. DMA Endpoint register: bit allocation**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Symbol | reserved | | | | EPIDX[2:0] | | | DMADIR |
| Reset | - | - | - | - | 0 | 0 | 0 | 0 |
| Bus reset | - | - | - | - | 0 | 0 | 0 | 0 |
| Access | - | - | - | - | R/W | R/W | R/W | R/W |

**Table 77. DMA Endpoint register: bit description**

| Bit | Symbol | Description |
|---|---|---|
| 7 to 4 | - | reserved |
| 3 to 1 | EPIDX[2:0] | **Endpoint Index**: selects the indicated endpoint for DMA access |
| 0 | DMADIR | **DMA Direction**:<br>**0 —** Selects the RX/OUT FIFO for DMA read transfers<br>**1 —** Selects the TX/IN FIFO for DMA write transfers |

The DMA Endpoint register must not reference the endpoint that is indexed by the Endpoint Index register (2Ch) at any time. Doing so will result in data corruption. Therefore, if the DMA Endpoint register is unused, point it to an unused endpoint. If the DMA Endpoint register, however, is pointed to an active endpoint, the firmware must not reference the same endpoint on the Endpoint Index register.

### 9.4.9 DMA Strobe Timing register (address: 60h)

This 1-byte register controls the strobe timing for MDMA mode, when bits DMA_MODE[1:0] in the DMA Configuration register have been set to 03h.

The bit allocation is given in Table 78.

**Table 78. DMA Strobe Timing register: bit allocation**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Symbol | reserved | | | DMA_STROBE_CNT[4:0] | | | | |
| Reset | - | - | - | 1 | 1 | 1 | 1 | 1 |
| Bus reset | - | - | - | 1 | 1 | 1 | 1 | 1 |
| Access | - | - | - | R/W | R/W | R/W | R/W | R/W |

**Table 79. DMA Strobe Timing register: bit description**

| Bit | Symbol | Description |
|---|---|---|
| 7 to 5 | - | reserved |
| 4 to 0 | DMA_STROBE_CNT[4:0] | **DMA Strobe Count**: These bits select the strobe duration for DMA_MODE = 03h (see Table 56). The strobe duration is (N + 1) cycles[1], with N representing the value of DMA_STROBE_CNT (see Figure 17). |

[1] The cycle duration indicates the internal clock cycle (33.3 ns/cycle).

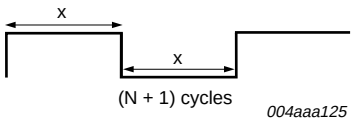



Fig 17. Programmable strobe timing

### 9.4.10 DMA Burst Counter register (address: 64h)

Table 80 shows the bit allocation of the 2-byte register.

Table 80. DMA Burst Counter register: bit allocation

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Symbol | reserved | | | BURSTCOUNTER[12:8] | | | | |
| Reset | - | - | - | 0 | 0 | 0 | 0 | 0 |
| Bus reset | - | - | - | 0 | 0 | 0 | 0 | 0 |
| Access | - | - | - | R/W | R/W | R/W | R/W | R/W |
| **Bit** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| Symbol | BURSTCOUNTER[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| Bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

Table 81. DMA Burst Counter register: bit description

| Bit | Symbol | Description |
|---|---|---|
| 15 to 13 | - | reserved |
| 12 to 0 | BURSTCOUNTER [12:0] | **Burst Counter**: This register defines the burst length. The counter must be programmed to be a multiple of two in 16-bit mode. The value of the burst counter must be programmed so that the burst counter is a factor of the buffer size.<br>It is used to determine the assertion and deassertion of DREQ. |

## 9.5 General registers

### 9.5.1 Interrupt register (address: 18h)

The Interrupt register consists of 4 bytes. The bit allocation is given in Table 82.

When a bit is set in the Interrupt register, it indicates that the hardware condition for an interrupt has occurred. When the Interrupt register content is nonzero, the INT output will be asserted corresponding to the Interrupt Enable register. On detecting the interrupt, the external microprocessor must read the Interrupt register and mask it with the corresponding bits in the Interrupt Enable register to determine the source of the interrupt.

Each endpoint buffer has a dedicated interrupt bit (EPnTX, EPnRX). In addition, various bus states can generate an interrupt: resume, suspend, pseudo SOF, SOF and bus reset. The DMA controller only has one interrupt bit: the source for a DMA interrupt is shown in the DMA Interrupt Reason register (see Table 72 and Table 73).

Each interrupt bit can individually be cleared by writing logic 1. The DMA Interrupt bit can be cleared by writing logic 1 to the related interrupt source bit in the DMA Interrupt Reason register, followed by writing logic 1 to the DMA bit of the Interrupt register.

**Table 82. Interrupt register: bit allocation**

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Symbol | reserved | | | | | | EP7TX | EP7RX |
| Reset | - | - | - | - | - | - | 0 | 0 |
| Bus reset | - | - | - | - | - | - | 0 | 0 |
| Access | - | - | - | - | - | - | R/W | R/W |
| **Bit** | **23** | **22** | **21** | **20** | **19** | **18** | **17** | **16** |
| Symbol | EP6TX | EP6RX | EP5TX | EP5RX | EP4TX | EP4RX | EP3TX | EP3RX |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| **Bit** | **15** | **14** | **13** | **12** | **11** | **10** | **9** | **8** |
| Symbol | EP2TX | EP2RX | EP1TX | EP1RX | EP0TX | EP0RX | reserved | EP0SETUP |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | - | 0 |
| Bus reset | 0 | 0 | 0 | 0 | 0 | 0 | - | 0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | - | R/W |
| **Bit** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| Symbol | VBUS | DMA | HS_STAT | RESUME | SUSP | PSOF | SOF | BRESET |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**Table 83. Interrupt register: bit description**

| Bit | Symbol | Description |
|---|---|---|
| 31 to 26 | - | reserved |
| 25 | EP7TX | logic 1 indicates the endpoint 7 TX buffer as interrupt source |
| 24 | EP7RX | logic 1 indicates the endpoint 7 RX buffer as interrupt source |
| 23 | EP6TX | logic 1 indicates the endpoint 6 TX buffer as interrupt source |
| 22 | EP6RX | logic 1 indicates the endpoint 6 RX buffer as interrupt source |
| 21 | EP5TX | logic 1 indicates the endpoint 5 TX buffer as interrupt source |
| 20 | EP5RX | logic 1 indicates the endpoint 5 RX buffer as interrupt source |
| 19 | EP4TX | logic 1 indicates the endpoint 4 TX buffer as interrupt source |
| 18 | EP4RX | logic 1 indicates the endpoint 4 RX buffer as interrupt source |
| 17 | EP3TX | logic 1 indicates the endpoint 3 TX buffer as interrupt source |
| 16 | EP3RX | logic 1 indicates the endpoint 3 RX buffer as interrupt source |
| 15 | EP2TX | logic 1 indicates the endpoint 2 TX buffer as interrupt source |
| 14 | EP2RX | logic 1 indicates the endpoint 2 RX buffer as interrupt source |
| 13 | EP1TX | logic 1 indicates the endpoint 1 TX buffer as interrupt source |
| 12 | EP1RX | logic 1 indicates the endpoint 1 RX buffer as interrupt source |
| 11 | EP0TX | logic 1 indicates the endpoint 0 data TX buffer as interrupt source |

Table 83. Interrupt register: bit description …continued

| Bit | Symbol | Description |
|---|---|---|
| 10 | EP0RX | logic 1 indicates the endpoint 0 data RX buffer as interrupt source |
| 9 | - | reserved |
| 8 | EP0SETUP | logic 1 indicates that a SETUP token was received on endpoint 0 |
| 7 | VBUS | logic 1 indicates a transition from LOW to HIGH on $V_{BUS}$ |
| 6 | DMA | **DMA status**: Logic 1 indicates a change in the DMA Interrupt Reason register. |
| 5 | HS_STAT | **High-Speed Status**: Logic 1 indicates a change from full-speed to high-speed mode (HS connection). This bit is not set, when the system goes into full-speed suspend. |
| 4 | RESUME | **Resume Status**: Logic 1 indicates that a status change from suspend to resume (active) was detected. |
| 3 | SUSP | **Suspend Status**: Logic 1 indicates that a status change from active to suspend was detected on the bus. |
| 2 | PSOF | **Pseudo SOF Interrupt**: Logic 1 indicates that a pseudo SOF or µSOF was received. Pseudo SOF is an internally generated clock signal (full-speed: 1 ms period, high-speed: 125 µs period) that is not synchronized to the USB bus SOF or µSOF. |
| 1 | SOF | **SOF Interrupt**: Logic 1 indicates that a SOF or µSOF was received. |
| 0 | BRESET | **Bus Reset**: Logic 1 indicates that a USB bus reset was detected. When bit OTG in the OTG register is set, BRESET will not be set, instead, this interrupt bit will report SE0 on DP and DM for 2 ms. |

### 9.5.2 Chip ID register (address: 70h)

This read-only register contains the chip identification and hardware version numbers. The firmware must check this information to determine functions and features supported. The register contains 3 bytes, and the bit allocation is shown in Table 84.

Table 84. Chip ID register: bit allocation

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Symbol | CHIPID[15:8] | | | | | | | |
| Reset | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 |
| Bus reset | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 |
| Access | R | R | R | R | R | R | R | R |
| **Bit** | **15** | **14** | **13** | **12** | **11** | **10** | **9** | **8** |
| Symbol | CHIPID[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| Bus reset | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| Access | R | R | R | R | R | R | R | R |
| **Bit** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| Symbol | VERSION[7:0] | | | | | | | |
| Reset | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| Bus reset | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| Access | R | R | R | R | R | R | R | R |

Table 85. Chip ID register: bit description

| Bit | Symbol | Description |
|---|---|---|
| 23 to 16 | CHIPID[15:8] | **Chip ID**: lower byte (15h) |
| 15 to 8 | CHIPID[7:0] | **Chip ID**: upper byte (82h) |
| 7 to 0 | VERSION[7:0] | **Version**: version number (30h) |

### 9.5.3 Frame Number register (address: 74h)

This read-only register contains the frame number of the last successfully received Start-Of-Frame (SOF). The register contains 2 bytes, and the bit allocation is given in Table 86. In case of 8-bit access, the register content is returned lower byte first.

Table 86. Frame Number register: bit allocation

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Symbol | reserved | | MICROSOF[2:0] | | | SOFR[10:8] | | |
| Reset | - | - | 0 | 0 | 0 | 0 | 0 | 0 |
| Bus reset | - | - | 0 | 0 | 0 | 0 | 0 | 0 |
| Access | - | - | R | R | R | R | R | R |
| **Bit** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| Symbol | SOFR[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Access | R | R | R | R | R | R | R | R |

Table 87. Frame Number register: bit description

| Bit | Symbol | Description |
|---|---|---|
| 15 to 14 | - | reserved |
| 13 to 11 | MICROSOF[2:0] | microframe number |
| 10 to 0 | SOFR[10:0] | frame number |

### 9.5.4 Scratch register (address: 78h)

This 16-bit register can be used by the firmware to save and restore information. For example, the device status before it enters the suspend state. The bit allocation is given in Table 88.

Table 88. Scratch register: bit allocation

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Symbol | SFIRH[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| **Bit** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| Symbol | SFIRL[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

Table 89. Scratch register: bit description

| Bit | Symbol | Description |
|---|---|---|
| 15 to 8 | SFIRH[7:0] | Scratch firmware information register (higher byte) |
| 7 to 0 | SFIRL[7:0] | Scratch firmware information register (lower byte) |

### 9.5.5 Unlock Device register (address: 7Ch)

To protect registers from getting corrupted when the ISP1583 goes into suspend, the write operation is disabled if bit PWRON in the Mode register is set to logic 0. In this case, when the chip resumes, the Unlock Device command must first be issued to this register before attempting to write to the rest of the registers. This is done by writing unlock code (AA37h) to this register. The bit allocation of the Unlock Device register is given in Table 90.

Table 90. Unlock Device register: bit allocation

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Symbol | ULCODE[15:8] = AAh | | | | | | | |
| Reset | not applicable | | | | | | | |
| Bus reset | not applicable | | | | | | | |
| Access | W | W | W | W | W | W | W | W |
| **Bit** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| Symbol | ULCODE[7:0] = 37h | | | | | | | |
| Reset | not applicable | | | | | | | |
| Bus reset | not applicable | | | | | | | |
| Access | W | W | W | W | W | W | W | W |

Table 91. Unlock Device register: bit description

| Bit | Symbol | Description |
|---|---|---|
| 15 to 0 | ULCODE[15:0] | **Unlock Code**: Writing data AA37h unlocks internal registers and FIFOs for writing, following a resume. |

When bit PWRON in the Mode register is logic 1, the chip is powered. In such a case, you do not need to issue the Unlock command because the microprocessor is powered and therefore, the RW_N/RD_N, DS_N/WR_N and CS_N signals maintain their states.

When bit PWRON is logic 0, the RW_N/RD_N, DS_N/WR_N and CS_N signals are floating because the microprocessor is not powered. To protect the ISP1583 registers from being corrupted during suspend, register write is locked when the chip goes into suspend. Therefore, you need to issue the Unlock command to unlock the ISP1583 registers.

### 9.5.6 Test Mode register (address: 84h)

This 1-byte register allows the firmware to set the DP and DM pins to predetermined states for testing purposes. The bit allocation is given in Table 92.

**Remark:** Only one bit can be set at a time. Either bit FORCEHS or FORCEFS must be set to logic 1 at a time. Of the four bits PRBS, KSTATE, JSTATE and SE0_NAK only one bit must be set at a time. This must be implemented for the Hi-Speed USB logo compliance testing. To exit test mode, power cycle is required.

**Table 92. Test Mode register: bit allocation**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| **Symbol** | FORCEHS | reserved | | FORCEFS | PRBS | KSTATE | JSTATE | SE0_NAK |
| **Reset** | 0 | - | - | 0 | 0 | 0 | 0 | 0 |
| **Bus reset** | unchanged | - | - | unchanged | 0 | 0 | 0 | 0 |
| **Access** | R/W | - | - | R/W | R/W | R/W | R/W | R/W |

**Table 93. Test Mode register: bit description**

| Bit | Symbol | Description |
|---|---|---|
| 7 | FORCEHS | **Force High-Speed**: Logic 1 forces the hardware to high-speed mode only and disables the chirp detection logic. |
| 6 to 5 | - | reserved |
| 4 | FORCEFS | **Force Full-Speed**: Logic 1 forces the physical layer to full-speed mode only and disables the chirp detection logic. |
| 3 | PRBS | **Predetermined Random Pattern**: Logic 1 sets the DP and DM pins to toggle in a predetermined random pattern. |
| 2 | KSTATE | **K-State**: Logic 1 sets the DP and DM pins to the K state. |
| 1 | JSTATE | **J-State**: Logic 1 sets the DP and DM pins to the J state. |
| 0 | SE0_NAK | **SE0 NAK**: Logic 1 sets the DP and DM pins to a high-speed quiescent state. The device only responds to a valid high-speed IN token with a NAK. |

# 10. Limiting values

**Table 94. Limiting values**

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

| Symbol | Parameter | Conditions | | Min | Max | Unit |
|---|---|---|---|---|---|---|
| $V_{CC(3V3)}$ | supply voltage (3.3 V) | | | –0.5 | +4.6 | V |
| $V_{CC(I/O)}$ | input/output supply voltage | | | –0.5 | +4.6 | V |
| $V_I$ | input voltage | | [1] | –0.5 | $V_{CC(3V3)}$ + 0.5 | V |
| $I_{lu}$ | latch-up current | $V_I$ < 0 V or $V_I$ > $V_{CC(3V3)}$ | | - | 100 | mA |
| $V_{esd}$ | electrostatic discharge voltage | $I_{LI}$ < 1 µA | | | | |
| | | pins DP, DM, $V_{BUS}$, AGND and DGND | | –4000 | +4000 | V |
| | | other pins | | –2000 | +2000 | V |
| $T_{stg}$ | storage temperature | | | –40 | +125 | °C |

[1] The maximum value for 5 V tolerant pins is 6 V.

# 11. Recommended operating conditions

**Table 95. Recommended operating conditions**

| Symbol | Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $V_{CC(3V3)}$ | supply voltage (3.3 V) | | 3.0 | - | 3.6 | V |
| $V_{CC(I/O)}$ | input/output supply voltage | | 1.65 | - | 3.6 | V |
| $V_I$ | input voltage | $V_{CC(3V3)}$ = 3.3 V | 0 | - | $V_{CC(I/O)}$ | V |
| $V_{IA(I/O)}$ | input voltage on analog I/O pins | on pins DP and DM | 0 | - | 3.6 | V |
| $V_{(pu)OD}$ | open-drain pull-up voltage | | 0 | - | $V_{CC(3V3)}$ | V |
| $T_{amb}$ | ambient temperature | | –40 | - | +85 | °C |
| $T_j$ | junction temperature | | –40 | - | +125 | °C |

# 12. Static characteristics

**Table 96. Static characteristics: supply pins**

*$V_{CC(3V3)}$ = 3.3 V ± 0.3 V; $V_{GND}$ = 0 V; $T_{amb}$ = −40 °C to +85 °C; typical values at $T_{amb}$ = +25 °C; unless otherwise specified.*

| Symbol | Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| **Supply voltage** | | | | | | |
| $V_{CC(3V3)}$ | supply voltage (3.3 V) | | 3.0 | 3.3 | 3.6 | V |
| $I_{CC(3V3)(oper)}$ | operating supply current (3.3 V) | high-speed | - | 47 | 60 | mA |
| | | full-speed | - | 19 | 25 | mA |
| $I_{CC(3V3)(susp)}$ | suspend mode supply current (3.3 V) | | - | 160 | - | µA |
| **I/O pad supply voltage** | | | | | | |
| $V_{CC(I/O)}$ | input/output supply voltage | | 1.65 | 3.3 | 3.6 | V |

**Table 96. Static characteristics: supply pins** *…continued*

*$V_{CC(3V3)}$ = 3.3 V ± 0.3 V; $V_{GND}$ = 0 V; $T_{amb}$ = −40 °C to +85 °C; typical values at $T_{amb}$ = +25 °C; unless otherwise specified.*

| Symbol | Parameter | Conditions | | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|---|
| $I_{CC(I/O)}$ | supply current on pin $V_{CC(I/O)}$ | | [1] | - | 3 | - | mA |
| **Regulated supply voltage** | | | | | | | |
| $V_{CC(1V8)}$ | supply voltage (1.8 V) | with voltage converter | | 1.65 | 1.8 | 1.95 | V |

[1] $I_{CC(I/O)}$ test condition: device set up under the test mode vector and I/O is subjected to external conditions.

**Table 97. Static characteristics: digital pins**

*$V_{CC(I/O)}$ = 1.65 V to 3.6 V; $V_{GND}$ = 0 V; $T_{amb}$ = −40 °C to +85 °C; unless otherwise specified.*

| Symbol | Parameter | Conditions | | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|---|
| **Input levels** | | | | | | | |
| $V_{IL}$ | LOW-level input voltage | | | - | - | $0.3V_{CC(I/O)}$ | V |
| $V_{IH}$ | HIGH-level input voltage | | | $0.7V_{CC(I/O)}$ | - | - | V |
| **Output levels** | | | | | | | |
| $V_{OL}$ | LOW-level output voltage | $I_{OL}$ = rated drive | | - | - | $0.15V_{CC(I/O)}$ | V |
| $V_{OH}$ | HIGH-level output voltage | $I_{OH}$ = rated drive | | $0.8V_{CC(I/O)}$ | - | - | V |
| **Leakage current** | | | | | | | |
| $I_{LI}$ | input leakage current | | [1] | −5 | - | +5 | µA |

[1] This value is applicable to transistor input only. The value will be different if internal pull-up or pull-down resistors are used.

**Table 98. Static characteristics: OTG detection**

*$V_{CC(I/O)}$ = 1.65 V to 3.6 V; $V_{GND}$ = 0 V; $T_{amb}$ = −40 °C to +85 °C; unless otherwise specified.*

| Symbol | Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| **Charging and discharging resistor** | | | | | | |
| $R_{DN(VBUS)}$ | pull-down resistance on pin $V_{BUS}$ | only when bit DISCV is set in the OTG register | 680 | 800 | 1030 | Ω |
| $R_{UP(DP)}$ | pull-up resistance on pin DP | only when bit DP is set in the OTG register | 300 | 550 | 780 | Ω |
| **Comparator levels** | | | | | | |
| $V_{BVALID}$ | $V_{BUS}$ valid detection | | 2.0 | - | 4.0 | V |
| $V_{SESEND}$ | $V_{BUS}$ B-session end detection | | 0.2 | - | 0.8 | V |

**Table 99. Static characteristics: analog I/O pins DP and DM**

*$V_{CC(3V3)}$ = 3.3 V ± 0.3 V; $V_{GND}$ = 0 V; $T_{amb}$ = −40 °C to +85 °C; unless otherwise specified.*[1]

| Symbol | Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| **Input levels** | | | | | | |
| $V_{DI}$ | differential input sensitivity voltage | $\lvert V_{I(DP)} - V_{I(DM)} \rvert$ | 0.2 | - | - | V |
| $V_{CM}$ | differential common mode voltage range | includes $V_{DI}$ range | 0.8 | - | 2.5 | V |
| $V_{SE}$ | single-ended receiver threshold | | 0.8 | - | 2.0 | V |
| $V_{IL}$ | LOW-level input voltage | | - | - | 0.8 | V |
| $V_{IH}$ | HIGH-level input voltage | | 2.0 | - | - | V |

**Table 99. Static characteristics: analog I/O pins DP and DM** *…continued*

*$V_{CC(3V3)}$ = 3.3 V ± 0.3 V; $V_{GND}$ = 0 V; $T_{amb}$ = −40 °C to +85 °C; unless otherwise specified.*[1]

| Symbol | Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| Schmitt-trigger inputs | | | | | | |
| $V_{th(LH)}$ | positive-going threshold voltage | | 1.4 | - | 1.9 | V |
| $V_{th(HL)}$ | negative-going threshold voltage | | 0.9 | - | 1.5 | V |
| $V_{hys}$ | hysteresis voltage | | 0.4 | - | 0.7 | V |
| **Output levels** | | | | | | |
| $V_{OL}$ | LOW-level output voltage | $R_L$ = 1.5 kΩ to 3.6 V | - | - | 0.4 | V |
| $V_{OH}$ | HIGH-level output voltage | $R_L$ = 15 kΩ to GND | 2.8 | - | 3.6 | V |
| **Leakage current** | | | | | | |
| $I_{LZ}$ | OFF-state leakage current | 0 V < $V_I$ < 3.3 V | –10 | - | +10 | µA |
| **Capacitance** | | | | | | |
| $C_{in}$ | input capacitance | pin to GND | - | - | 10 | pF |
| **Resistance** | | | | | | |
| $Z_{DRV}$ | driver output impedance for driver which is not high-speed capable | steady-state drive | 40.5 | - | 49.5 | Ω |
| $Z_{INP}$ | input impedance exclusive of pull-up/pull-down (for low-/full-speed) | | 10 | - | - | MΩ |

[1] Pin DP is the USB positive data pin, and pin DM is the USB negative data pin.

# 13. Dynamic characteristics

**Table 100. Dynamic characteristics**

*$V_{CC(3V3)}$ = 3.3 V ± 0.3 V; $V_{GND}$ = 0 V; $T_{amb}$ = −40 °C to +85 °C; unless otherwise specified.*

| Symbol | Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| **Reset** | | | | | | |
| $t_{W(RESET\_N)}$ | external RESET_N pulse width | crystal oscillator running | 500 | - | - | µs |
| **Crystal oscillator** | | | | | | |
| $f_{XTAL1}$ | frequency on pin XTAL1 | | - | 12 | - | MHz |
| $R_S$ | series resistance | | - | - | 100 | Ω |
| $C_L$ | load capacitance | | - | 18 | - | pF |
| **External clock input** | | | | | | |
| $t_J$ | external clock jitter | | - | - | 500 | ps |
| δ | clock duty cycle | | 45 | 50 | 55 | % |
| $t_r$ | rise time | | - | - | 3 | ns |
| $t_f$ | fall time | | - | - | 3 | ns |
| $V_I$ | input voltage | | 1.65 | 1.8 | 1.95 | V |

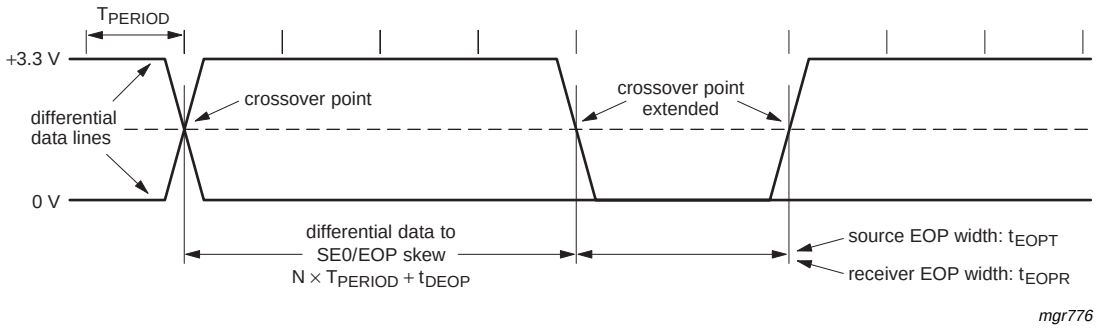
**Table 101. Dynamic characteristics: analog I/O pins DP and DM**

*$V_{CC(3V3)}$ = 3.3 V ± 0.3 V; $V_{GND}$ = 0 V; $T_{amb}$ = −40 °C to +85 °C; $C_L$ = 50 pF; $R_{PU}$ = 1.5 kΩ on DP to $V_{TERM}$; test circuit of Figure 38; unless otherwise specified.*

| Symbol | Parameter | Conditions | | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|---|
| **Driver characteristics** | | | | | | | |
| Full-speed mode | | | | | | | |
| $t_{FR}$ | rise time | $C_L$ = 50 pF; 10 % to 90 % of $\|V_{OH} - V_{OL}\|$ | | 4 | - | 20 | ns |
| $t_{FF}$ | fall time | $C_L$ = 50 pF; 90 % to 10 % of $\|V_{OH} - V_{OL}\|$ | | 4 | - | 20 | ns |
| FRFM | differential rise time/fall time matching | $t_{FR}/t_{FF}$ | [1] | 90 | - | 111.11 | % |
| $V_{CRS}$ | output signal crossover voltage | | [1][2] | 1.3 | - | 2.0 | V |
| High-speed mode | | | | | | | |
| $t_{HSR}$ | rise time (10 % to 90 %) | with captive cable | | 500 | - | - | ps |
| $t_{HSF}$ | fall time (10 % to 90 %) | with captive cable | | 500 | - | - | ps |
| **Data source timing** | | | | | | | |
| Full-speed mode | | | | | | | |
| $t_{FEOPT}$ | source SE0 interval of EOP | see Figure 18 | [2] | 160 | - | 175 | ns |
| $t_{FDEOP}$ | source jitter for differential transition to SE0 transition | see Figure 18 | [2] | −2 | - | +5 | ns |
| **Receiver timing** | | | | | | | |
| Full-speed mode | | | | | | | |
| $t_{JR1}$ | receiver jitter to next transition | see Figure 19 | [2] | −18.5 | - | +18.5 | ns |
| $t_{JR2}$ | receiver jitter for paired transitions | see Figure 19 | [2] | −9 | - | +9 | ns |
| $t_{FEOPR}$ | receiver SE0 interval of EOP | accepted as EOP; see Figure 18 | [2] | 82 | - | - | ns |
| $t_{FST}$ | width of SE0 interval during differential transition | rejected as EOP; see Figure 20 | [2] | - | - | 14 | ns |

[1] Excluding the first transition from the idle state.

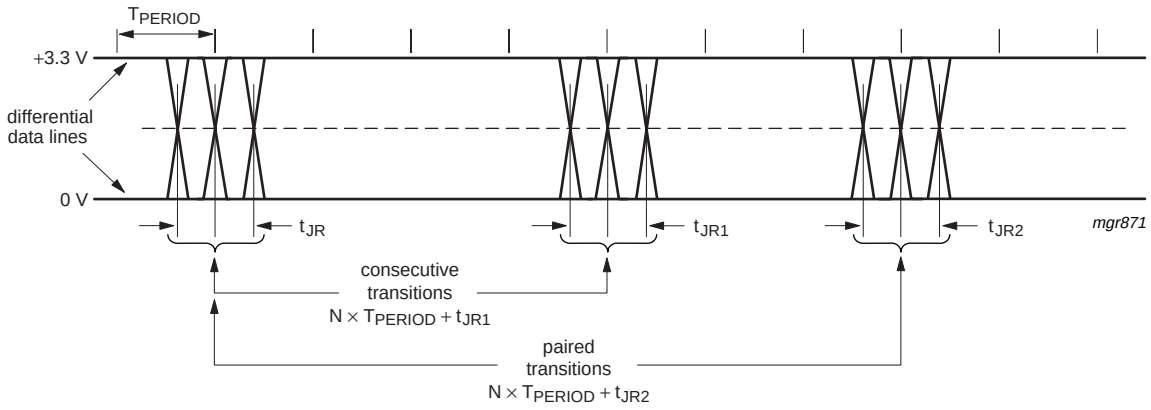
[2] Characterized only, not tested. Limits guaranteed by design.




$T_{PERIOD}$ is the bit duration corresponding to the USB data rate.

Full-speed timing symbols have a subscript prefix 'F', low-speed timing symbols have a prefix 'L'.

**Fig 18. Source differential data-to-EOP transition skew and EOP width**




$T_{PERIOD}$ is the bit duration corresponding to the USB data rate.

**Fig 19. Receiver differential data jitter**

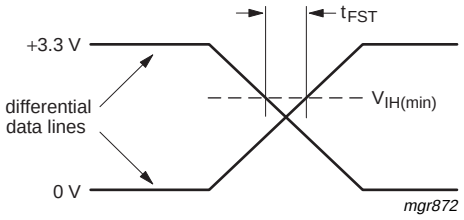



**Fig 20. Receiver SE0 width tolerance**

## 13.1 Register access timing

**Remark:** In the following subsections, RW_N/RD_N, DS_N/WR_N, READY/IORDY and ALE/A0 refer to the ISP1583 pin.

### 13.1.1 Generic processor mode

BUS_CONF/DA0 = HIGH: generic processor mode

#### 13.1.1.1 8051 mode

MODE0/DA1 = HIGH: 8051 mode; see Table 3

**Table 102. ISP1583 register access timing parameters: separate address and data buses**

*$V_{CC(I/O)}$ = 1.65 V to 3.6 V; $V_{CC(3V3)}$ = 3.3 V; $V_{GND}$ = 0 V; $T_{amb}$ = −40 °C to +85 °C; unless otherwise specified.*

| Symbol | Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| **Reading** | | | | | | |
| $t_{RLRH}$ | RW_N/RD_N LOW pulse width | | > $t_{RLDV}$ | - | - | ns |
| $t_{AVRL}$ | address set-up time before RW_N/RD_N LOW | | 0 | - | - | ns |
| $t_{RHAX}$ | address hold time after RW_N/RD_N HIGH | | 0 | - | - | ns |
| $t_{RLDV}$ | RW_N/RD_N LOW to data valid delay | | - | - | 26 | ns |
| $t_{RHDZ}$ | RW_N/RD_N HIGH to data outputs 3-state delay | | 0 | - | 15 | ns |
| $t_{RHSH}$ | RW_N/RD_N HIGH to CS_N HIGH delay | | 0 | - | - | ns |

**Table 102. ISP1583 register access timing parameters: separate address and data buses** *…continued*

*$V_{CC(I/O)}$ = 1.65 V to 3.6 V; $V_{CC(3V3)}$ = 3.3 V; $V_{GND}$ = 0 V; $T_{amb}$ = −40 °C to +85 °C; unless otherwise specified.*

| Symbol | Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $t_{SLRL}$ | CS_N LOW to RW_N/RD_N LOW delay | | 2 | - | - | ns |
| **Writing** | | | | | | |
| $t_{WLWH}$ | DS_N/WR_N LOW pulse width | | 15 | - | - | ns |
| $t_{AVWL}$ | address set-up time before DS_N/WR_N LOW | | 0 | - | - | ns |
| $t_{WHAX}$ | address hold time after DS_N/WR_N HIGH | | 0 | - | - | ns |
| $t_{DVWH}$ | data set-up time before DS_N/WR_N HIGH | | 11 | - | - | ns |
| $t_{WHDZ}$ | data hold time after DS_N/WR_N HIGH | | 5 | - | - | ns |
| $t_{WHSH}$ | DS_N/WR_N HIGH to CS_N HIGH delay | | 0 | - | - | ns |
| $t_{SLWL}$ | CS_N LOW to DS_N/WR_N LOW delay | | 2 | - | - | ns |
| **General** | | | | | | |
| $T_{cy(RW)}$ | read or write cycle time | | 50 | - | - | ns |
| $t_{RDY1}$ | READY/IORDY HIGH to RW_N/RD_N or DS_N/WR_N HIGH of the last access | | - | - | 91 | ns |

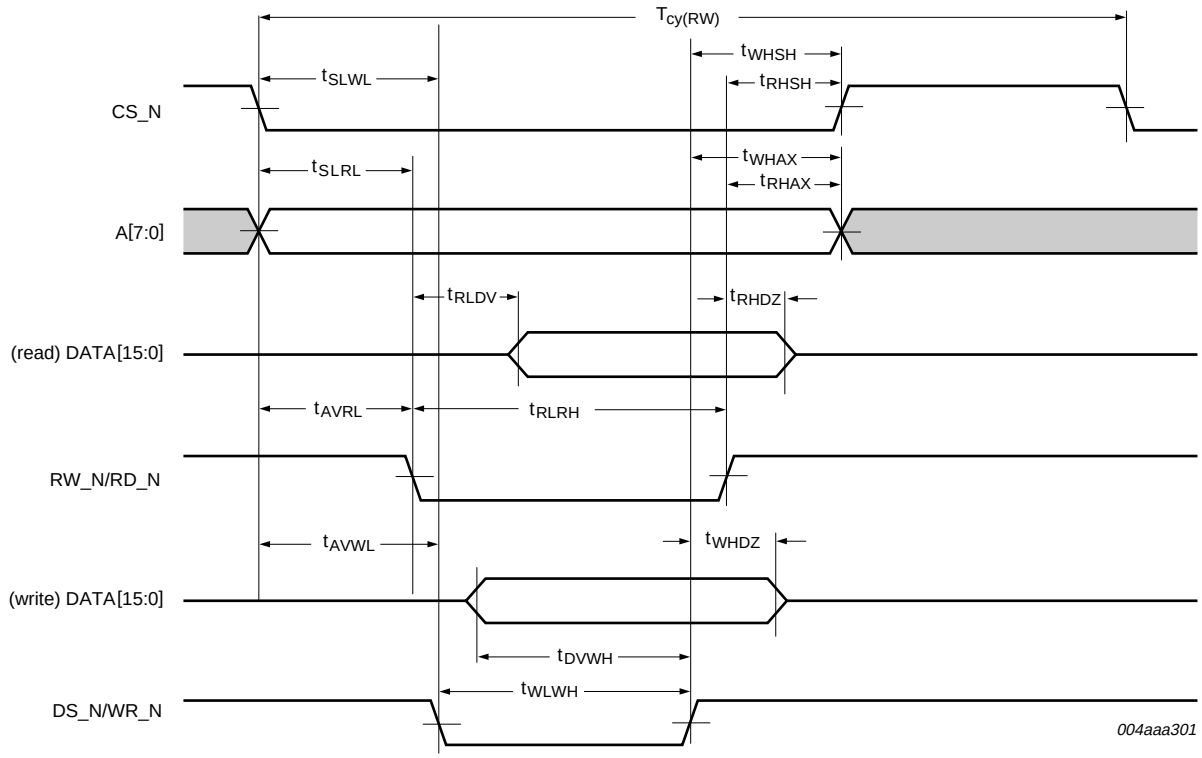



**Fig 21. ISP1583 register access timing: separate address and data buses (8051 mode)**

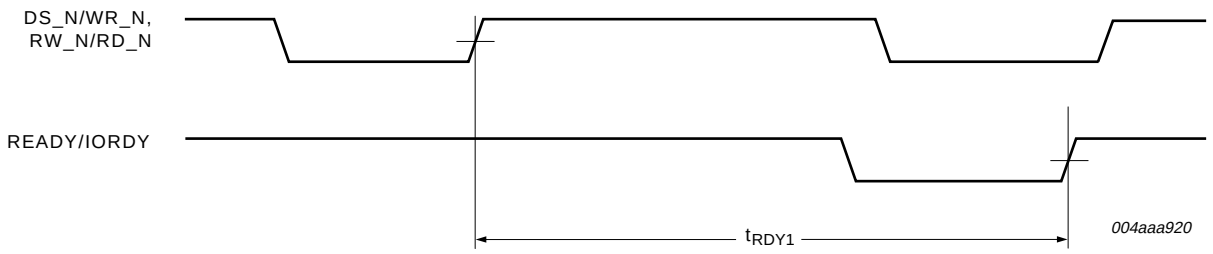



**Fig 22. ISP1583 ready signal timing**

#### 13.1.1.2 Freescale mode

MODE0/DA1 = LOW: Freescale mode; see Table 3

**Table 103. ISP1583 register access timing parameters: separate address and data buses**

*$V_{CC(I/O)}$ = 1.65 V to 3.6 V; $V_{CC(3V3)}$ = 3.3 V; $V_{GND}$ = 0 V; $T_{amb}$ = −40 °C to +85 °C; unless otherwise specified.*

| Symbol | Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| **Reading or writing** | | | | | | |
| $t_{WLWH}$ | DS_N/WR_N LOW pulse width | | 15 | - | - | ns |
| $t_{AVWL}$ | address set-up time before DS_N/WR_N LOW | | 0 | - | - | ns |
| $t_{WHAX}$ | address hold time after DS_N/WR_N HIGH | | 0 | - | - | ns |
| $t_{DVWH}$ | data set-up time before DS_N/WR_N HIGH | | 11 | - | - | ns |
| $t_{WHDZ}$ | data hold time after DS_N/WR_N HIGH | | 5 | - | - | ns |
| $t_{WHSH}$ | DS_N/WR_N HIGH to CS_N HIGH delay | | 0 | - | - | ns |
| $t_{I1VI2L}$ | RW_N/RD_N set-up time before DS_N/WR_N LOW | | 0 | - | - | ns |
| $t_{I2HI1X}$ | RW_N/RD_N hold time after DS_N/WR_N HIGH | | 0 | - | - | ns |
| **General** | | | | | | |
| $T_{cy(RW)}$ | read or write cycle time | | 50 | - | - | ns |
| $t_{RDY1}$ | READY/IORDY HIGH to DS_N/WR_N HIGH of the last access | | - | - | 91 | ns |

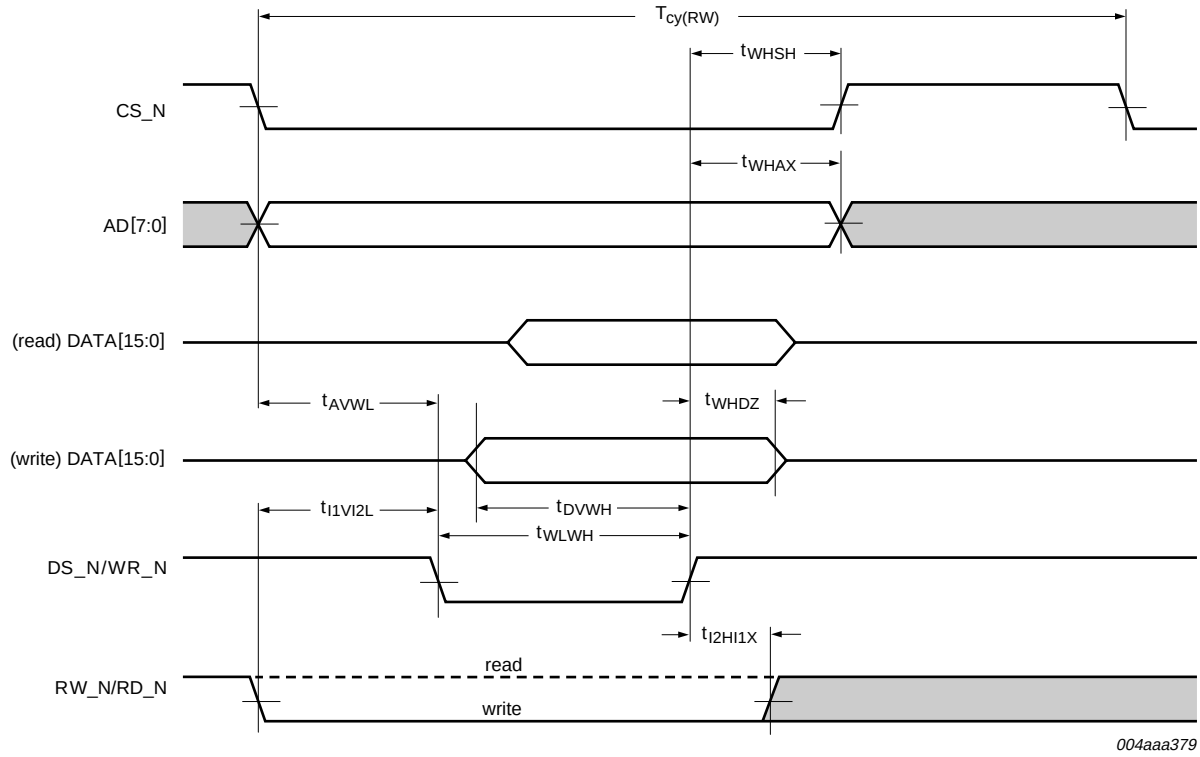



**Fig 23. ISP1583 register access timing: separate address and data buses (Freescale mode)**

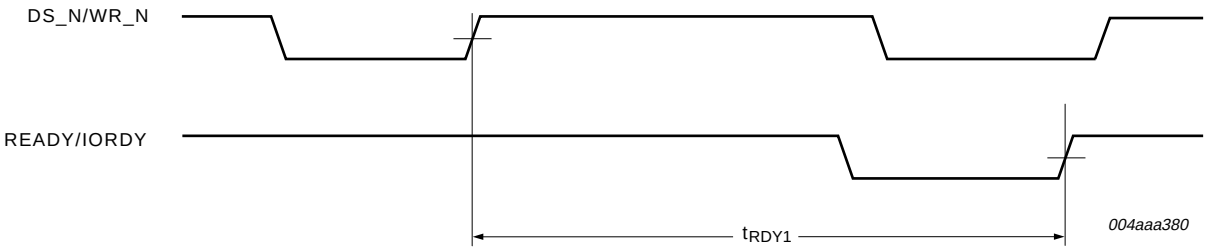
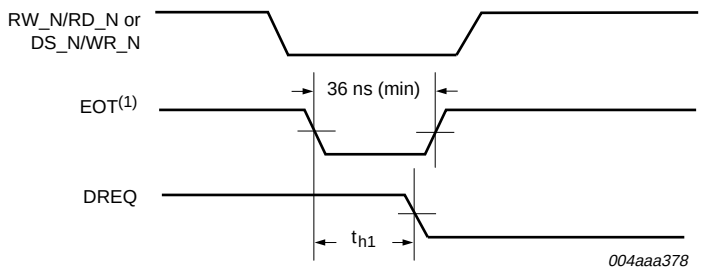



**Fig 24. ISP1583 ready signal timing**




(1) Programmable polarity: shown as active LOW.

**Remark:** EOT must be valid for 36 ns (minimum) when pins RW_N/RD_N and DS_N/WR_N are active.

**Fig 25. EOT timing in generic processor mode**

### 13.1.2 Split bus mode

#### 13.1.2.1 ALE function

##### 8051 mode

- BUS_CONF/DA0 = LOW: split bus mode
- MODE1 = LOW: ALE function
  - MODE0/DA1 = HIGH: 8051 mode; see Table 3

**Table 104. ISP1583 register access timing parameters: multiplexed address/data bus**

*$V_{CC(I/O)}$ = 1.65 V to 3.6 V; $V_{CC(3V3)}$ = 3.3 V; $V_{GND}$ = 0 V; $T_{amb}$ = −40 °C to +85 °C; unless otherwise specified.*

| Symbol | Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| **Reading** | | | | | | |
| $t_{RLRH}$ | RW_N/RD_N LOW pulse width | | > $t_{RLDV}$ | - | - | ns |
| $t_{RLDV}$ | RW_N/RD_N LOW to data valid delay | | - | - | 25 | ns |
| $t_{RHDZ}$ | RW_N/RD_N HIGH to data outputs 3-state delay | | 0 | - | 15 | ns |
| $t_{RHSH}$ | RW_N/RD_N HIGH to CS_N HIGH delay | | 0 | - | - | ns |
| $t_{LLRL}$ | ALE/A0 LOW set-up time before RW_N/RD_N LOW | | 0 | - | - | ns |
| **Writing** | | | | | | |
| $t_{WLWH}$ | DS_N/WR_N LOW pulse width | | 15 | - | - | ns |
| $t_{DVWH}$ | data set-up time before DS_N/WR_N HIGH | | 5 | - | - | ns |
| $t_{LLWL}$ | ALE/A0 LOW to DS_N/WR_N LOW delay | | 0 | - | - | ns |
| $t_{WHDZ}$ | data hold time after DS_N/WR_N HIGH | | 5 | - | - | ns |
| $t_{WHSH}$ | DS_N/WR_N HIGH to CS_N HIGH delay | | 0 | - | - | ns |
| **General** | | | | | | |
| $T_{cy(RW)}$ | read or write cycle time | | 80 | - | - | ns |
| $t_{AVLL}$ | address set-up time before ALE/A0 LOW | | 0 | - | - | ns |

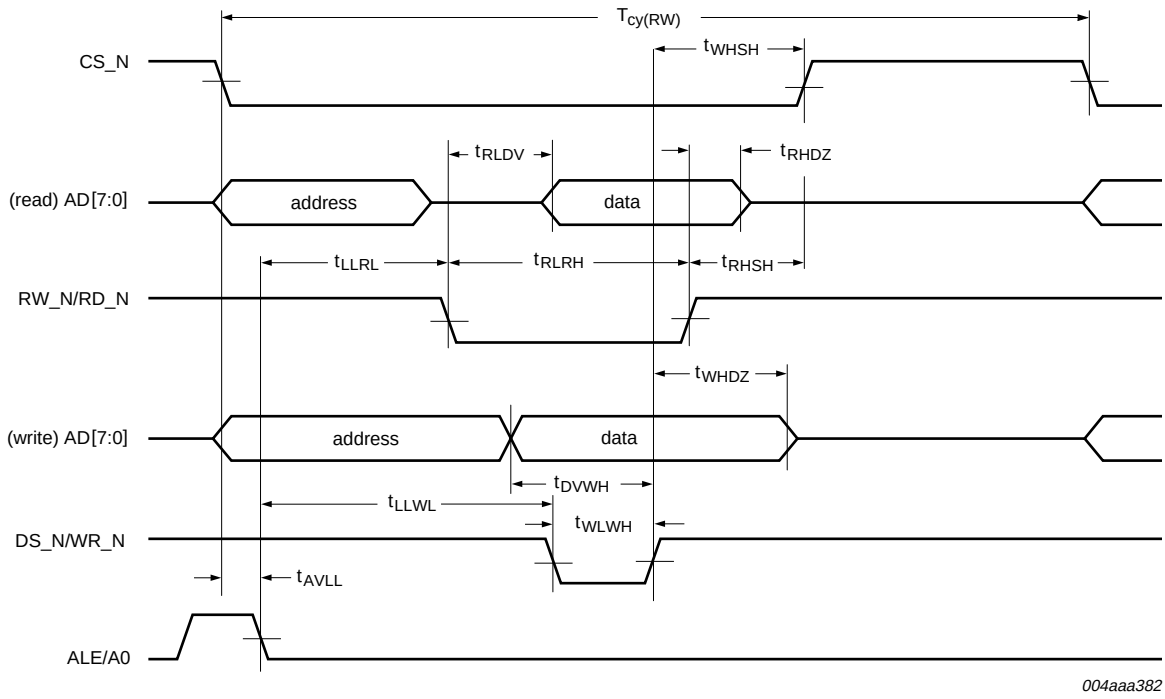



**Fig 26. ISP1583 register access timing: multiplexed address/data bus (8051 mode)**

### Freescale mode

- BUS_CONF/DA0 = LOW: split bus mode
- MODE1 = LOW: ALE function
  - MODE0/DA1 = LOW: Freescale mode; see Table 3

**Table 105. ISP1583 register access timing parameters: multiplexed address/data bus**

*$V_{CC(I/O)}$ = 1.65 V to 3.6 V; $V_{CC(3V3)}$ = 3.3 V; $V_{GND}$ = 0 V; $T_{amb}$ = −40 °C to +85 °C; unless otherwise specified.*

| Symbol | Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| **Reading or writing** | | | | | | |
| $t_{WLWH}$ | DS_N/WR_N LOW pulse width | | 15 | - | - | ns |
| $t_{DVWH}$ | data set-up time before DS_N/WR_N HIGH | | 5 | - | - | ns |
| $t_{WHDZ}$ | data hold time after DS_N/WR_N HIGH | | 5 | - | - | ns |
| $t_{WHSH}$ | DS_N/WR_N HIGH to CS_N HIGH delay | | 0 | - | - | ns |
| **General** | | | | | | |
| $T_{cy(RW)}$ | read or write cycle time | | 80 | - | - | ns |
| $t_{I1VLL}$ | RW_N/RD_N set-up time before ALE/A0 LOW | | 5 | - | - | ns |
| $t_{LLI2L}$ | ALE/A0 LOW to DS_N/WR_N LOW delay | | 5 | - | - | ns |
| $t_{I2HI1X}$ | RW_N/RD_N hold time after DS_N/WR_N HIGH | | 5 | - | - | ns |

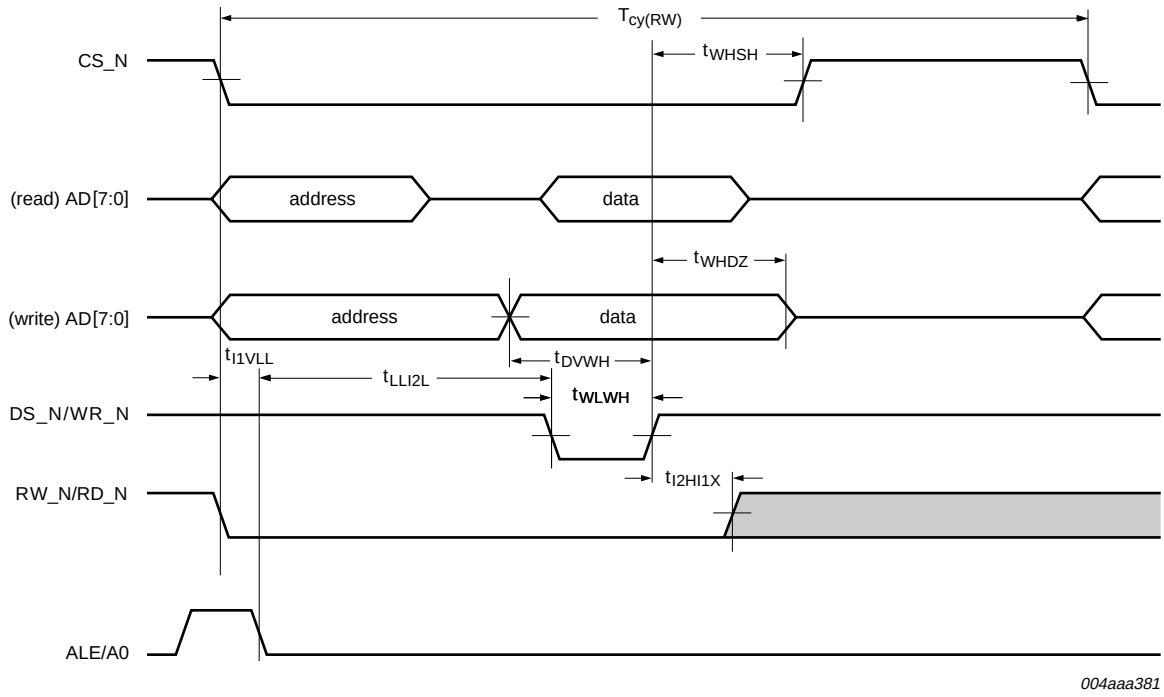



**Fig 27. ISP1583 register access timing: multiplexed address/data bus (Freescale mode)**

#### 13.1.2.2 A0 function

#### 8051 mode

- BUS_CONF/DA0 = LOW: split bus mode
- MODE1 = HIGH: A0 function
  - MODE0/DA1 = HIGH: 8051 mode; see Table 3

**Table 106. ISP1583 register access timing parameters: multiplexed address/data bus**

*$V_{CC(I/O)}$ = 1.65 V to 3.6 V; $V_{CC(3V3)}$ = 3.3 V; $V_{GND}$ = 0 V; $T_{amb}$ = −40 °C to +85 °C; unless otherwise specified.*

| Symbol | Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| **Reading** | | | | | | |
| $t_{RLDV}$ | RW_N/RD_N LOW to data valid delay | | - | - | 26 | ns |
| $t_{RHDZ}$ | RW_N/RD_N HIGH to data outputs 3-state delay | | 0 | - | 15 | ns |
| $t_{RHSH}$ | RW_N/RD_N HIGH to CS_N HIGH delay | | 0 | - | - | ns |
| $t_{RLRH}$ | RW_N/RD_N LOW pulse width | | > $t_{RLDV}$ | - | - | ns |
| $t_{WHRH}$ | DS_N/WR_N HIGH to RW_N/RD_N HIGH delay | | 40 | - | - | ns |
| **Writing** | | | | | | |
| $t_{A0WL}$ | ALE/A0 set-up time before DS_N/WR_N LOW | | 0 | - | - | ns |
| $t_{AVWH}$ | address set-up time before DS_N/WR_N HIGH | | 5 | - | - | ns |
| $t_{DVWH}$ | data set-up time before DS_N/WR_N HIGH | | 5 | - | - | ns |
| $t_{WHDZ}$ | data hold time after DS_N/WR_N HIGH | | 5 | - | - | ns |
| $t_{WHSH}$ | DS_N/WR_N HIGH to CS_N HIGH delay | | 0 | - | - | ns |
| $t_{WLWH}$ | DS_N/WR_N LOW pulse width | | 15 | - | - | ns |

**Table 106. ISP1583 register access timing parameters: multiplexed address/data bus** *...continued*

*$V_{CC(I/O)}$ = 1.65 V to 3.6 V; $V_{CC(3V3)}$ = 3.3 V; $V_{GND}$ = 0 V; $T_{amb}$ = −40 °C to +85 °C; unless otherwise specified.*

| Symbol | Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $t_{WHWH}$ | DS_N/WR_N HIGH (address) to DS_N/WR_N HIGH (data) delay | | 40 | - | - | ns |
| **General** | | | | | | |
| $T_{cy(RW)}$ | read or write cycle time | | 50 | - | - | ns |

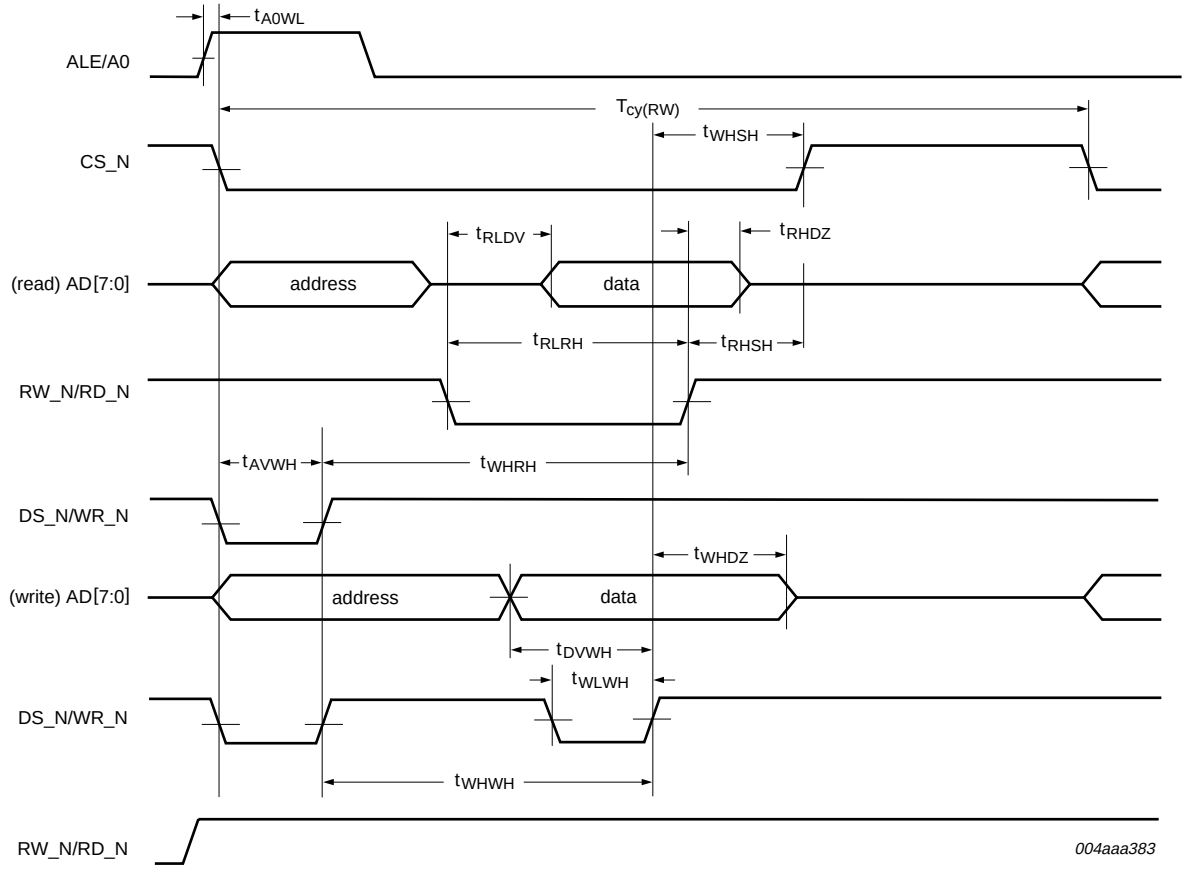



**Fig 28. ISP1583 register access timing: multiplexed address/data bus (A0 function and 8051 mode)**

### Freescale mode

- BUS_CONF/DA0 = LOW: split bus mode
- MODE1 = HIGH: A0 function
  - MODE0/DA1 = LOW: Freescale mode; see Table 3

**Table 107. ISP1583 register access timing parameters: multiplexed address/data bus**

*$V_{CC(I/O)}$ = 1.65 V to 3.6 V; $V_{CC(3V3)}$ = 3.3 V; $V_{GND}$ = 0 V; $T_{amb}$ = −40 °C to +85 °C; unless otherwise specified.*

| Symbol | Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| **Reading** | | | | | | |
| $t_{RLDV}$ | DS_N/WR_N LOW to data valid delay | | - | - | 26 | ns |
| $t_{RHDZ}$ | DS_N/WR_N HIGH to data outputs 3-state delay | | 0 | - | 15 | ns |

**Table 107. ISP1583 register access timing parameters: multiplexed address/data bus** *…continued*

*$V_{CC(I/O)}$ = 1.65 V to 3.6 V; $V_{CC(3V3)}$ = 3.3 V; $V_{GND}$ = 0 V; $T_{amb}$ = −40 °C to +85 °C; unless otherwise specified.*

| Symbol | Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $t_{RHSH}$ | DS_N/WR_N HIGH to CS_N HIGH delay | | 0 | - | - | ns |
| $t_{RLRH}$ | DS_N/WR_N LOW pulse width | | > $t_{RLDV}$ | - | - | ns |
| $t_{WHRH}$ | DS_N/WR_N HIGH (address) to DS_N/WR_N HIGH (data read) delay | | 40 | - | - | ns |
| **Writing** | | | | | | |
| $t_{A0WL}$ | ALE/A0 set-up time before DS_N/WR_N LOW | | 0 | - | - | ns |
| $t_{AVWH}$ | address set-up time before DS_N/WR_N HIGH | | 5 | - | - | ns |
| $t_{DVWH}$ | data set-up time before DS_N/WR_N HIGH | | 5 | - | - | ns |
| $t_{WHDZ}$ | data hold time after DS_N/WR_N HIGH | | 5 | - | - | ns |
| $t_{WHSH}$ | DS_N/WR_N HIGH to CS_N HIGH delay | | 0 | - | - | ns |
| $t_{WLWH}$ | DS_N/WR_N LOW pulse width | | 15 | - | - | ns |
| $t_{WHWH}$ | DS_N/WR_N HIGH (address) to DS_N/WR_N HIGH (data written) delay | | 40 | - | - | ns |
| **General** | | | | | | |
| $T_{cy(RW)}$ | read or write cycle time | | 50 | - | - | ns |
| $t_{I2HI1X}$ | RW_N/RD_N hold time after DS_N/WR_N HIGH | | 5 | - | - | ns |

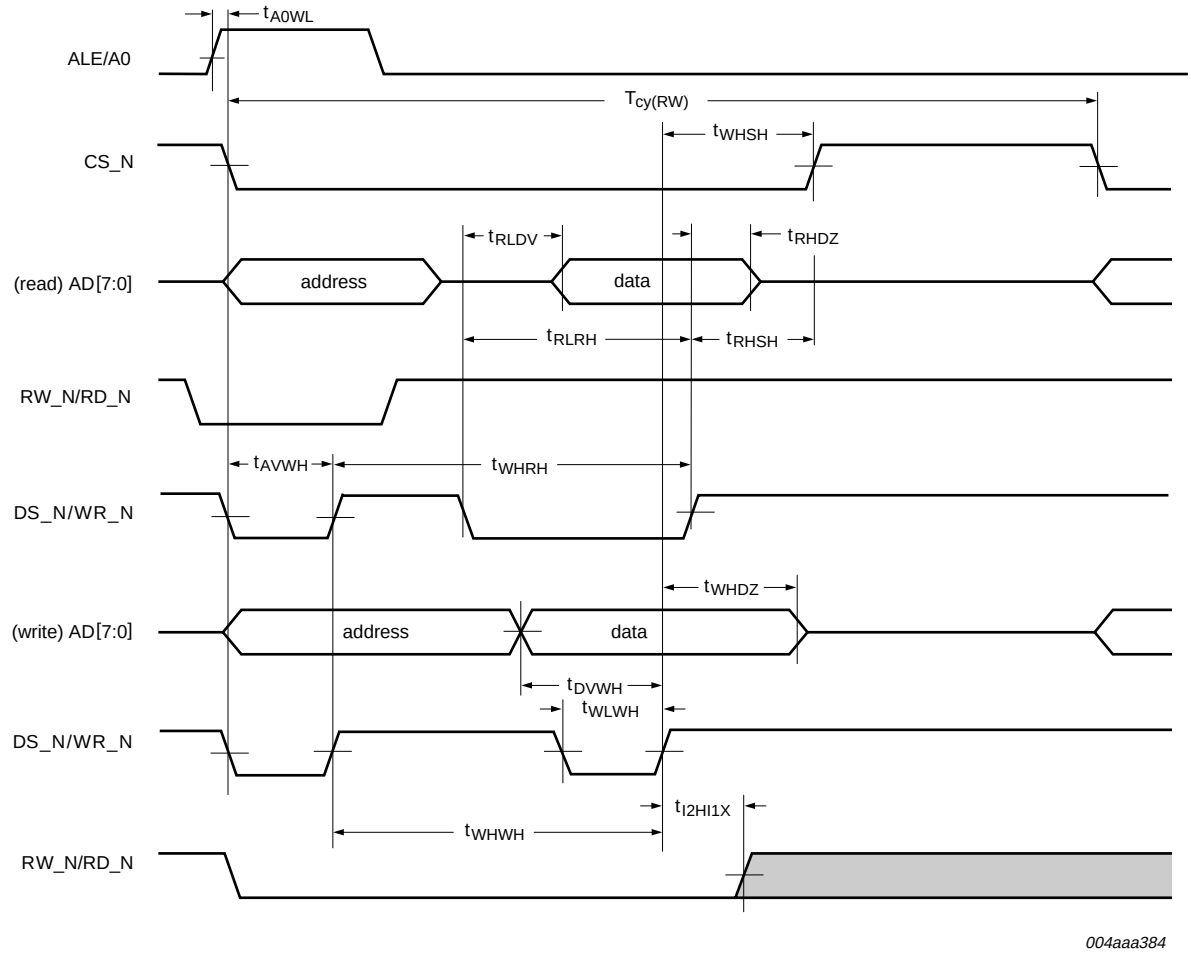
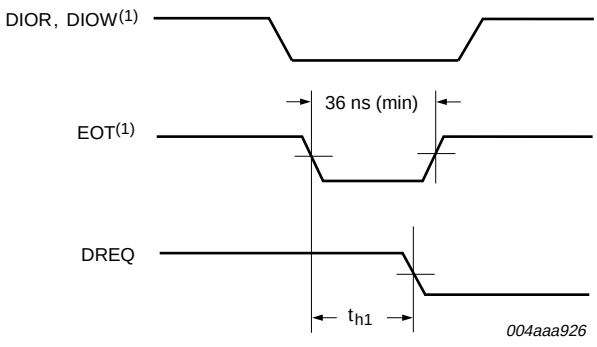



**Fig 29. ISP1583 register access timing: multiplexed address/data bus (A0 function and Freescale mode)**




(1) Programmable polarity: shown as active LOW.

**Remark:** EOT must be valid for 36 ns (minimum) when DIOR or DIOW is active.

**Fig 30. EOT timing in split bus mode**

## 13.2 DMA timing

### 13.2.1 PIO mode

**Remark:** In the following subsections, RW_N/RD_N, DS_N/WR_N, READY/IORDY and ALE/A0 refer to the ISP1583 pin.

**Table 108. PIO mode timing parameters**

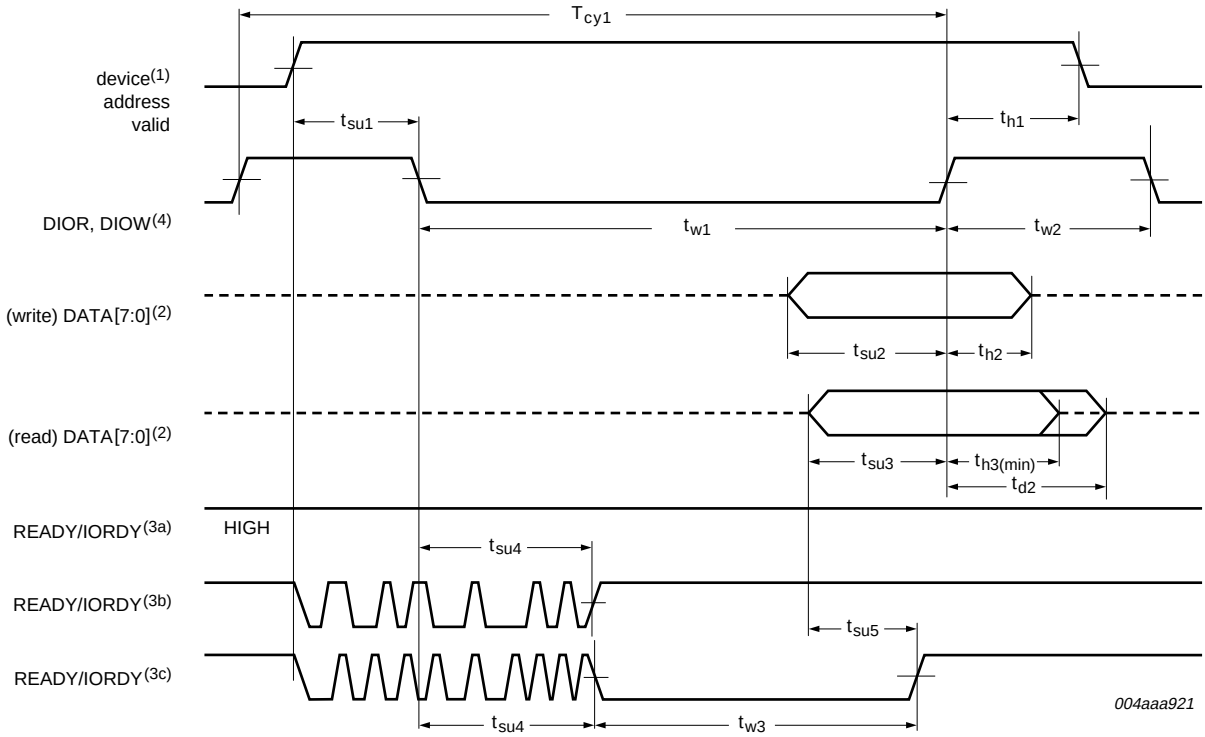
*$V_{CC(I/O)}$ = 1.65 V to 3.6 V; $V_{CC(3V3)}$ = 3.3 V; $V_{GND}$ = 0 V; $T_{amb}$ = −40 °C to +85 °C; unless otherwise specified.*

| Symbol | Parameter | Conditions | | Mode 0 | Mode 1 | Mode 2 | Mode 3 | Mode 4 | Unit |
|---|---|---|---|---|---|---|---|---|---|
| $T_{cy1(min)}$ | read or write cycle time | | [1] | 600 | 383 | 240 | 180 | 120 | ns |
| $t_{su1(min)}$ | address to DIOR or DIOW on set-up time | | | 70 | 50 | 30 | 30 | 25 | ns |
| $t_{w1(min)}$ | DIOR or DIOW pulse width | | [1] | 165 | 125 | 100 | 80 | 70 | ns |
| $t_{w2(min)}$ | DIOR/DIOW recovery time | | [1] | - | - | - | 70 | 25 | ns |
| $t_{su2(min)}$ | data set-up time before DIOW off | | | 60 | 45 | 30 | 30 | 20 | ns |
| $t_{h2(min)}$ | data hold time after DIOW off | | | 30 | 20 | 15 | 10 | 10 | ns |
| $t_{su3(min)}$ | data set-up time before DIOR on | | | 50 | 35 | 20 | 20 | 20 | ns |
| $t_{h3(min)}$ | data hold time after DIOR off | | | 5 | 5 | 5 | 5 | 5 | ns |
| $t_{d2(max)}$ | data to 3-state delay after DIOR off | | [2] | 30 | 30 | 30 | 30 | 30 | ns |
| $t_{h1(min)}$ | address hold time after DIOR or DIOW off | | | 20 | 15 | 10 | 10 | 10 | ns |
| $t_{su4(min)}$ | READY/IORDY after DIOR or DIOW on set-up time | | [3] | 35 | 35 | 35 | 35 | 35 | ns |
| $t_{su5(min)}$ | read data to READY/IORDY HIGH set-up time | | [3] | 0 | 0 | 0 | 0 | 0 | ns |
| $t_{w3(max)}$ | READY/IORDY LOW pulse width | | | 1250 | 1250 | 1250 | 1250 | 1250 | ns |

[1] $T_{cy1}$ is the total cycle time, consisting of command active time $t_{w1}$ and command recovery (inactive) time $t_{w2}$, that is, $T_{cy1} = t_{w1} + t_{w2}$. Minimum timing requirements for $T_{cy1}$, $t_{w1}$ and $t_{w2}$ must all be met. As $T_{cy1(min)}$ is greater than the sum of $t_{w1(min)}$ and $t_{w2(min)}$, a host implementation must lengthen $t_{w1}$ and/or $t_{w2}$ to ensure that $T_{cy1}$ is equal to or greater than the value reported in the IDENTIFY DEVICE data. A device implementation shall support any legal host implementation.

[2] $t_{d2}$ specifies the time after DIOR is negated, when the data bus is no longer driven by the device (3-state).

[3] If READY/IORDY is LOW at $t_{su4}$, the host waits until READY/IORDY is made HIGH before the PIO cycle is completed. In that case, $t_{su5}$ must be met for reading ($t_{su3}$ does not apply). When READY/IORDY is HIGH at $t_{su4}$, $t_{su3}$ must be met for reading ($t_{su5}$ does not apply).




(1) The device address consists of signals CS1_N, CS0_N, DA2, DA1 and DA0.

(2) The data bus width depends on the PIO access command used. The Task File register access uses 8 bits (DATA[7:0]), except for Task File register 1F0 that uses 16 bits (DATA[15:0]). DMA commands 04h and 05h also use a 16-bit data bus.

(3) The device can negate READY/IORDY to extend the PIO cycle with wait states. The host determines whether or not to extend the current cycle after $t_{su4}$, following the assertion of DIOR or DIOW. The following three cases are distinguished:

a) Device keeps READY/IORDY released (high-impedance): no wait state is generated.

b) Device negates READY/IORDY during $t_{su4}$, but re-asserts READY/IORDY before $t_{su4}$ expires: no wait state is generated.

c) Device negates READY/IORDY during $t_{su4}$ and keeps READY/IORDY negated for at least 5 ns after $t_{su4}$ expires: a wait state is generated. The cycle is completed as soon as READY/IORDY is re-asserted. For extended read cycles (DIOR asserted), the read data on lines DATAn must be valid at $t_{d1}$ before READY/IORDY is asserted.

(4) DIOR and DIOW have a programmable polarity: shown here as active LOW signals.

**Fig 31. PIO mode timing**

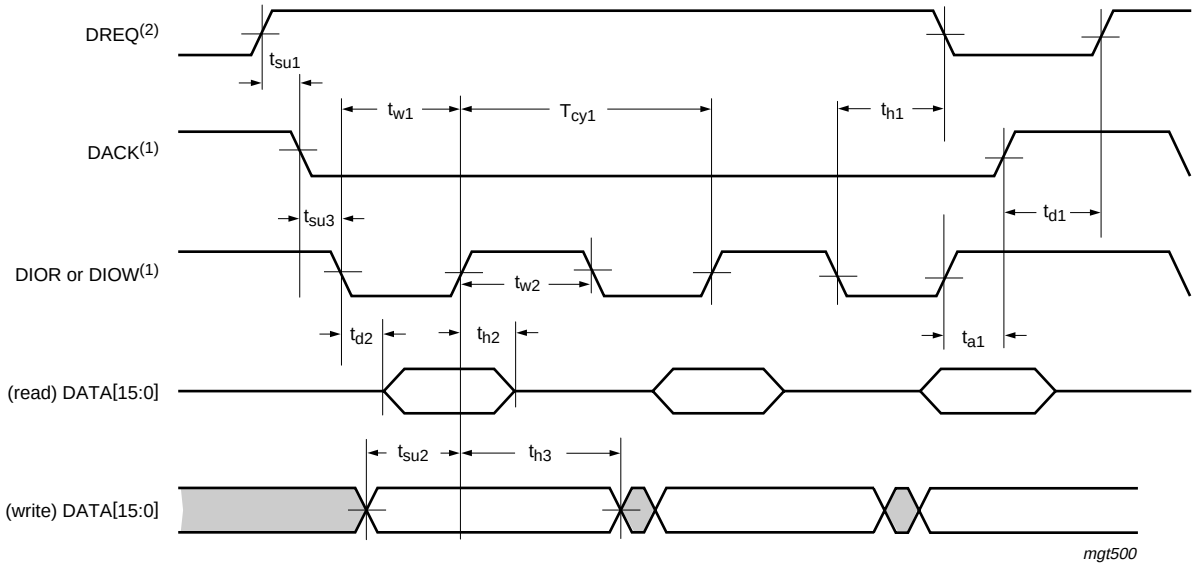
### 13.2.2 GDMA slave mode

- Bits MODE[1:0] = 00: data strobes DIOR (read) and DIOW (write); see Figure 32
- Bits MODE[1:0] = 01: data strobes DIOR (read) and DACK (write); see Figure 33
- Bits MODE[1:0] = 10: data strobes DACK (read and write); see Figure 34

**Table 109. GDMA slave mode timing parameters**

*$V_{CC(I/O)}$ = 1.65 V to 3.6 V; $V_{CC(3V3)}$ = 3.3 V; $V_{GND}$ = 0 V; $T_{amb}$ = −40 °C to +85 °C; unless otherwise specified.*

| Symbol | Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $T_{cy1}$ | read or write cycle time | | 75 | - | - | ns |
| $t_{su1}$ | DREQ set-up time before first DACK on | | 10 | - | - | ns |
| $t_{d1}$ | DREQ on delay after last strobe off | | 33.33 | - | - | ns |
| $t_{h1}$ | DREQ hold time after last strobe on | | 0 | - | 53 | ns |
| $t_{w1}$ | DIOR or DIOW pulse width | | 39 | - | 600 | ns |
| $t_{w2}$ | DIOR or DIOW recovery time | | 36 | - | - | ns |
| $t_{d2}$ | read data valid delay after strobe on | | - | - | 20 | ns |
| $t_{h2}$ | read data hold time after strobe off | | - | - | 5 | ns |
| $t_{h3}$ | write data hold time after strobe off | | 1 | - | - | ns |
| $t_{su2}$ | write data set-up time before strobe off | | 10 | - | - | ns |
| $t_{su3}$ | DACK set-up time before DIOR/DIOW assertion | | 0 | - | - | ns |
| $t_{a1}$ | DACK deassertion after DIOR/DIOW deassertion | | 0 | - | 30 | ns |



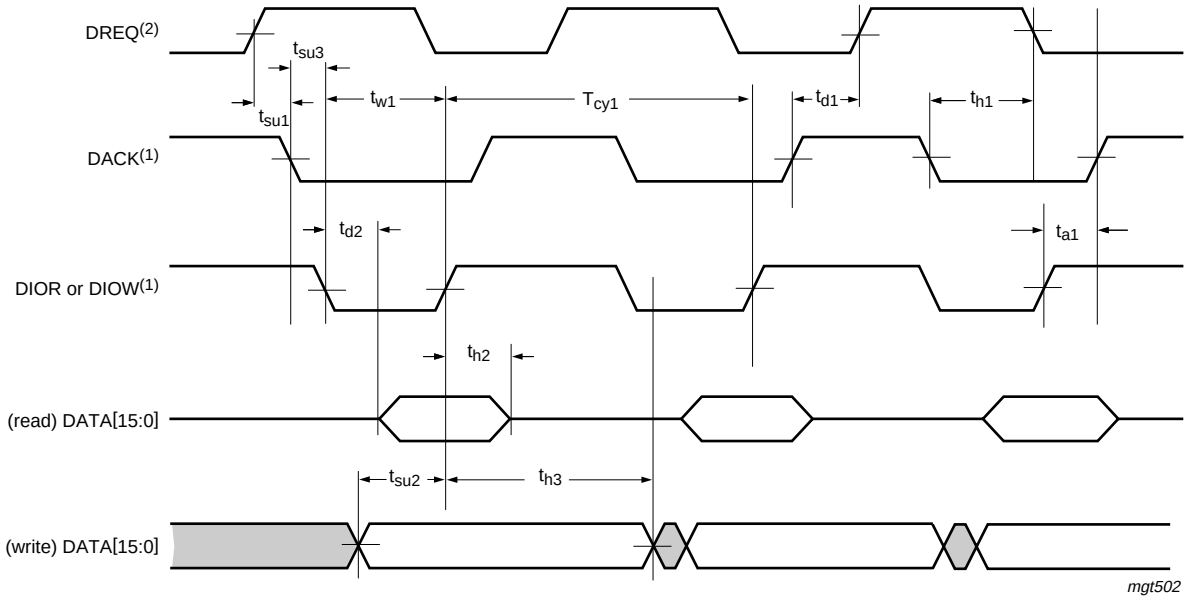

DREQ is continuously asserted, until the last transfer is done or the FIFO is full.

Data strobes: DIOR (read), DIOW (write).

(1) Programmable polarity: shown as active LOW.

(2) Programmable polarity: shown as active HIGH.

**Fig 32. GDMA slave mode timing: DIOR (master) and DIOW (slave)**



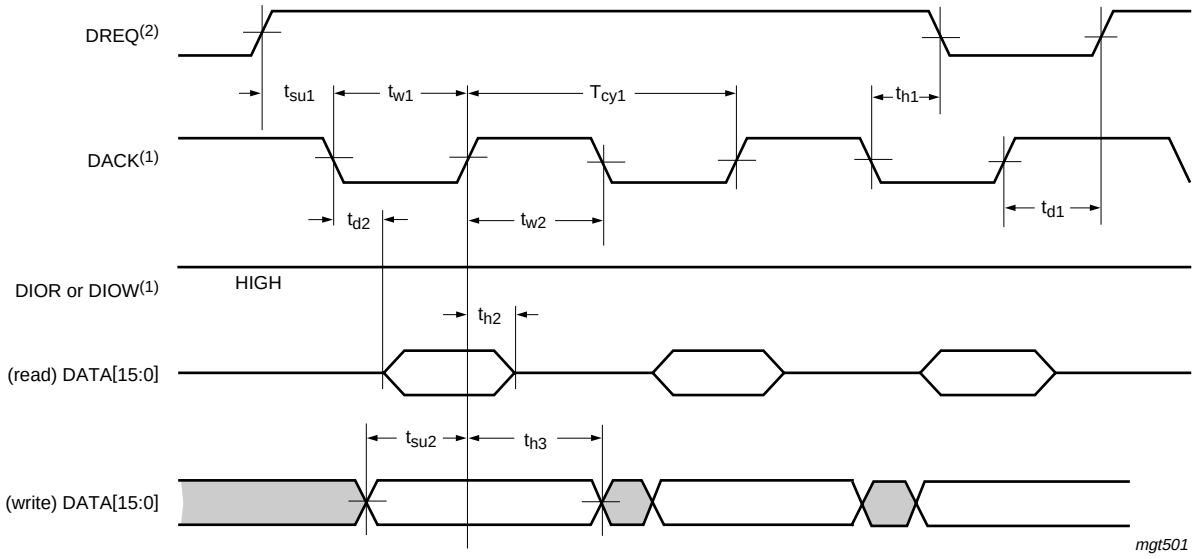

DREQ is asserted for every transfer.

Data strobes: DIOR (read), DACK (write).

(1) Programmable polarity: shown as active LOW.

(2) Programmable polarity: shown as active HIGH.

**Fig 33. GDMA slave mode timing: DIOR (master) or DACK (slave)**




DREQ is continuously asserted, until the last transfer is done or the FIFO is full.

Data strobe: DACK (read/write).

(1) Programmable polarity: shown as active LOW.

(2) Programmable polarity: shown as active HIGH.

**Fig 34. GDMA slave mode timing: DACK (master and slave)**

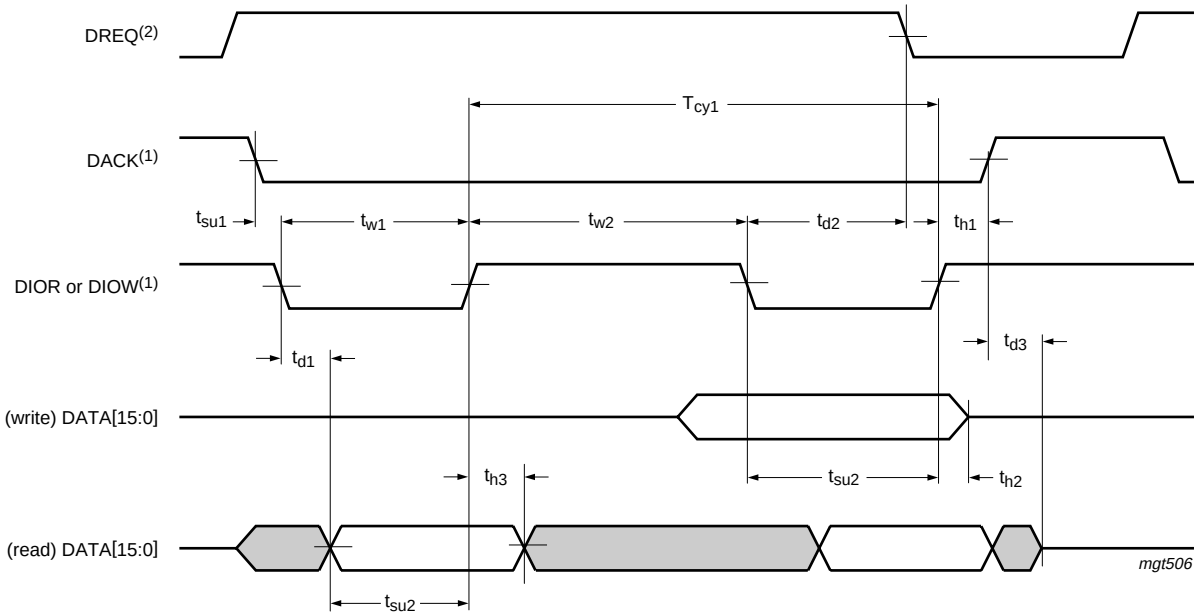
### 13.2.3 MDMA mode

**Table 110. MDMA mode timing parameters**

*$V_{CC(I/O)}$ = 1.65 V to 3.6 V; $V_{CC(3V3)}$ = 3.3 V; $V_{GND}$ = 0 V; $T_{amb}$ = −40 °C to +85 °C; unless otherwise specified.*

| Symbol | Parameter | Conditions | | Mode 0 | Mode 1 | Mode 2 | Unit |
|---|---|---|---|---|---|---|---|
| $T_{cy1(min)}$ | read/write cycle time | | [1] | 480 | 150 | 120 | ns |
| $t_{w1(min)}$ | DIOR or DIOW pulse width | | [1] | 215 | 80 | 70 | ns |
| $t_{d1(max)}$ | data valid delay after DIOR on | | | 150 | 60 | 50 | ns |
| $t_{h3(min)}$ | data hold time after DIOR off | | | 5 | 5 | 5 | ns |
| $t_{su2(min)}$ | data set-up time before DIOR or DIOW off | | | 100 | 30 | 20 | ns |
| $t_{h2(min)}$ | data hold time after DIOW off | | | 20 | 15 | 10 | ns |
| $t_{su1(min)}$ | DACK set-up time before DIOR or DIOW on | | | 0 | 0 | 0 | ns |
| $t_{h1(min)}$ | DACK hold time after DIOR or DIOW off | | | 20 | 5 | 5 | ns |
| $t_{w2(min)}$ | DIOR recovery time | | [1] | 50 | 50 | 25 | ns |
| | DIOW recovery time | | [1] | 215 | 50 | 25 | ns |
| $t_{d2(max)}$ | DIOR on to DREQ off delay | | | 120 | 40 | 35 | ns |
| | DIOW on to DREQ off delay | | | 40 | 40 | 35 | ns |
| $t_{d3(max)}$ | DACK off to data lines 3-state delay | | | 20 | 25 | 25 | ns |

[1] $T_{cy1}$ is the total cycle time, consisting of command active time $t_{w1}$ and command recovery (inactive) time $t_{w2}$, that is, $T_{cy1} = t_{w1} + t_{w2}$. Minimum timing requirements for $T_{cy1}$, $t_{w1}$ and $t_{w2}$ must all be met. As $T_{cy1(min)}$ is greater than the sum of $t_{w1(min)}$ and $t_{w2(min)}$, a host implementation must lengthen $t_{w1}$ and/or $t_{w2}$ to ensure that $T_{cy1}$ is equal to or greater than the value reported in the IDENTIFY DEVICE data. A device implementation shall support any legal host implementation.




(1) Programmable polarity: shown as active LOW.

(2) Programmable polarity: shown as active HIGH.

**Fig 35. MDMA master mode timing**

# 14. Application information

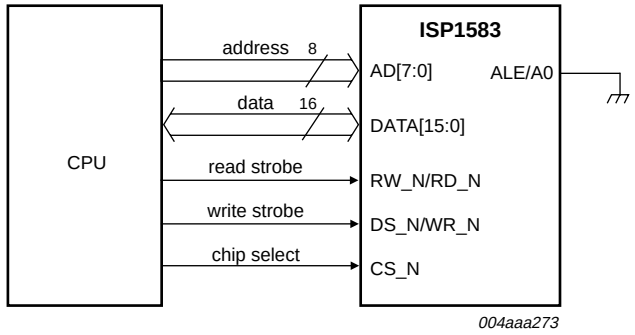



**Fig 36. Typical interface connections for generic processor mode**

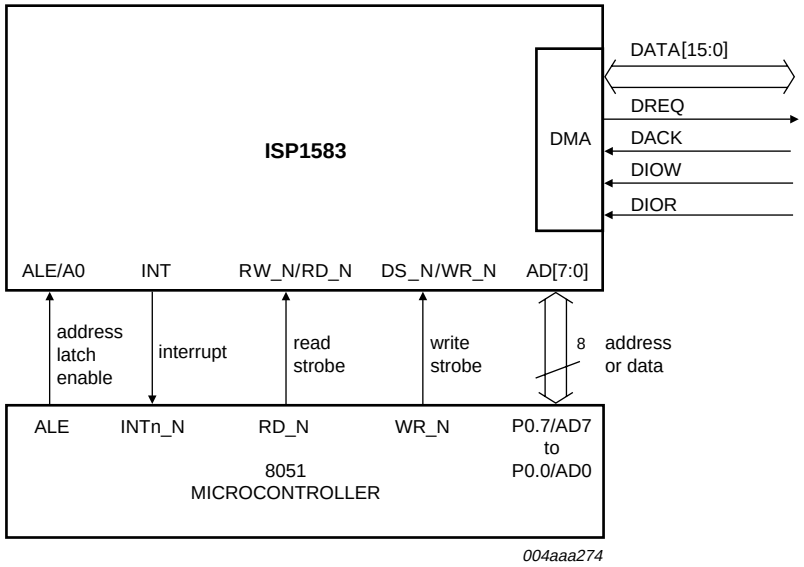
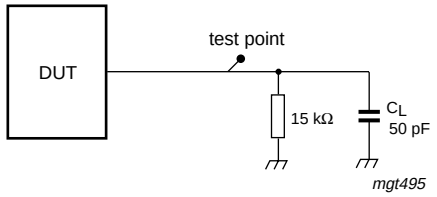



**Fig 37. Typical interface connections for split bus mode (slave mode)**

# 15. Test information

The dynamic characteristics of analog I/O ports DP and DM are determined using the circuit shown in Figure 38.



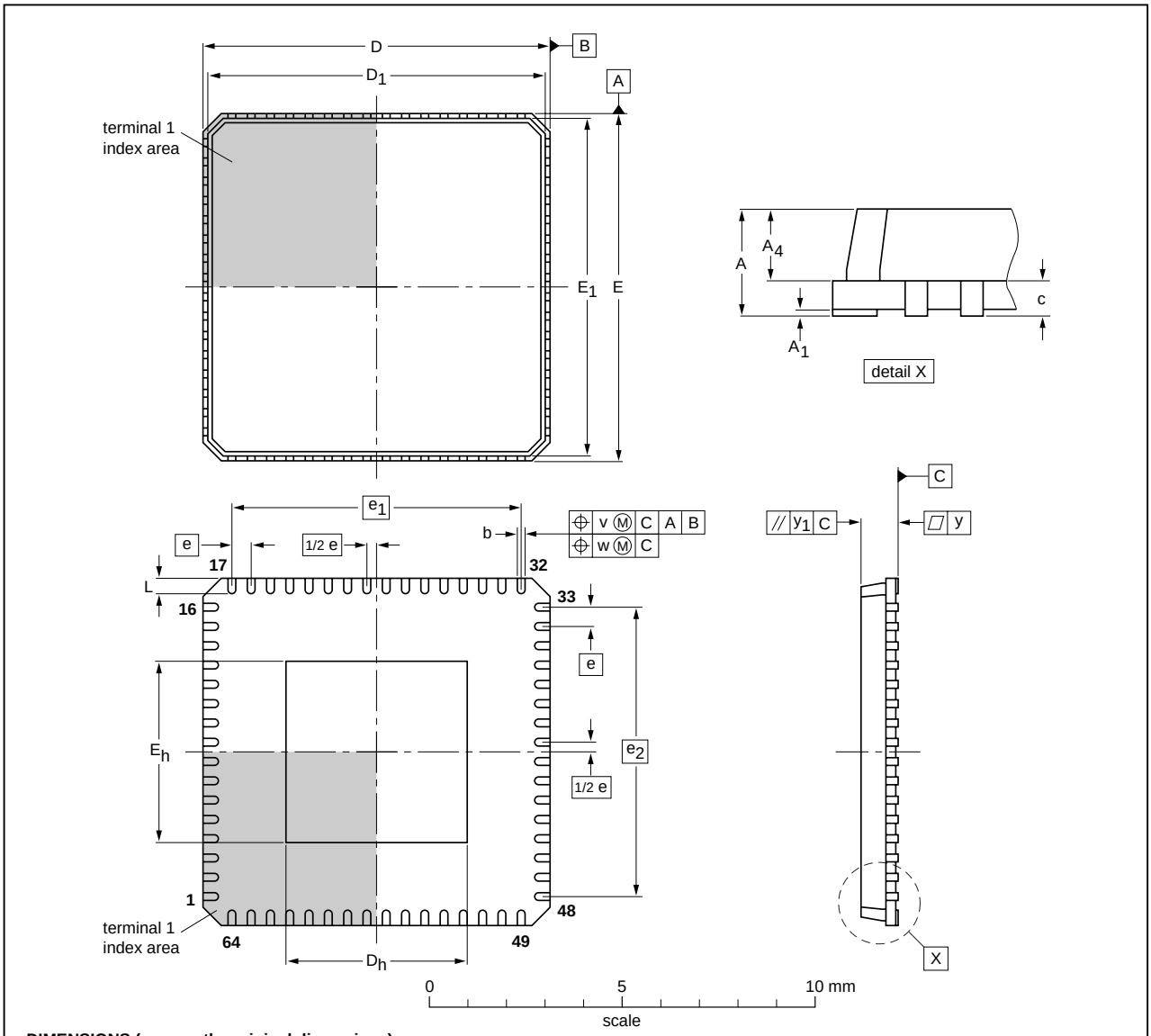

In full-speed mode, an internal 1.5 kΩ pull-up resistor is connected to pin DP.

**Fig 38. Load impedance for the DP and DM pins (full-speed mode)**

# 16. Package outline

**HVQFN64: plastic thermal enhanced very thin quad flat package; no leads; 64 terminals; body 9 x 9 x 0.85 mm**

**SOT804-1**



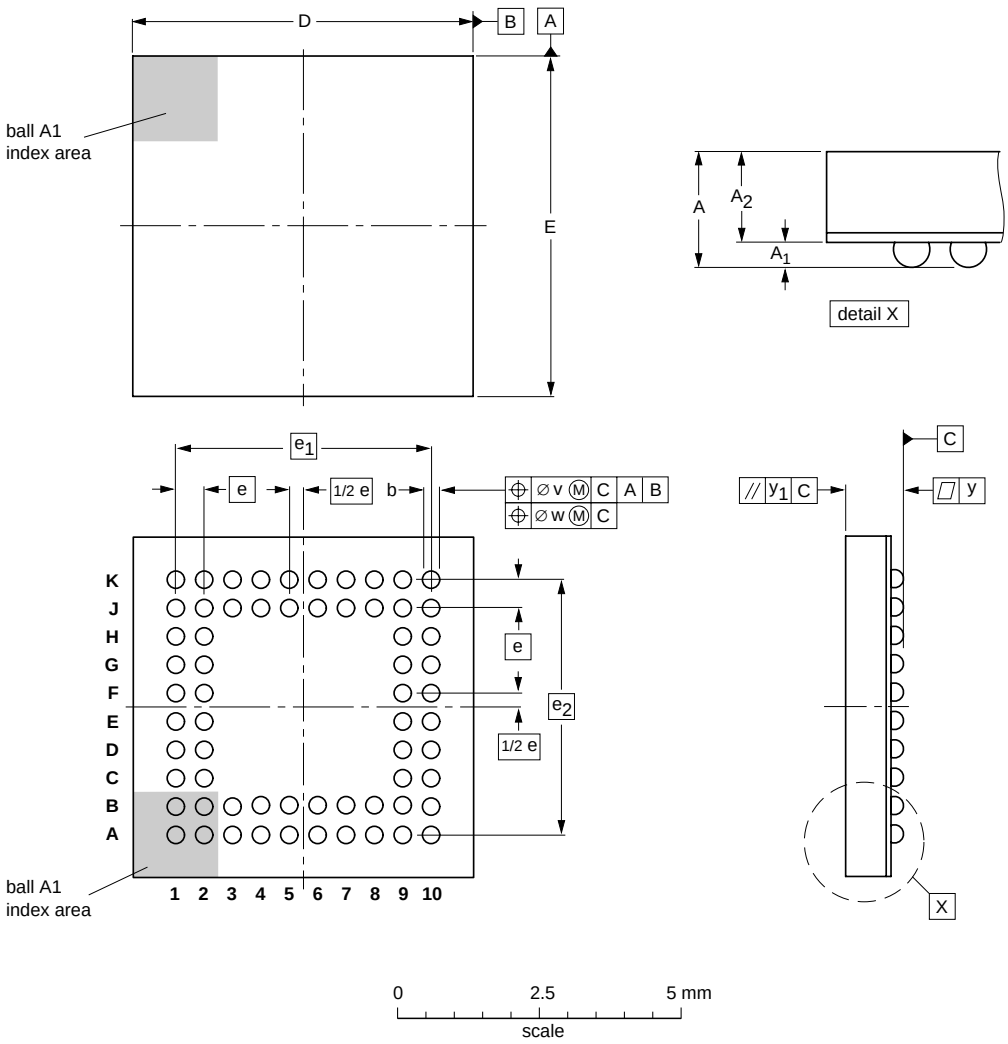

**DIMENSIONS (mm are the original dimensions)**

| UNIT | A max. | $A_1$ | $A_4$ | b | c | D | $D_1$ | $D_h$ | E | $E_1$ | $E_h$ | e | $e_1$ | $e_2$ | L | v | w | y | $y_1$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| mm | 1 | 0.05<br>0.00 | 0.80<br>0.65 | 0.30<br>0.18 | 0.2 | 9.05<br>8.95 | 8.95<br>8.55 | 4.85<br>4.55 | 9.05<br>8.95 | 8.95<br>8.55 | 4.85<br>4.55 | 0.5 | 7.5 | 7.5 | 0.5<br>0.3 | 0.1 | 0.05 | 0.05 | 0.1 |

| OUTLINE VERSION | REFERENCES | | | | EUROPEAN PROJECTION | ISSUE DATE |
|---|---|---|---|---|---|---|
| | IEC | JEDEC | JEITA | | | |
| SOT804-1 | - - - | MO-220 | - - - | | | 03-03-26 |

**Fig 39. Package outline SOT804-1 (HVQFN64)**

**TFBGA64: plastic thin fine-pitch ball grid array package; 64 balls; body 6 x 6 x 0.8 mm** **SOT543-1**



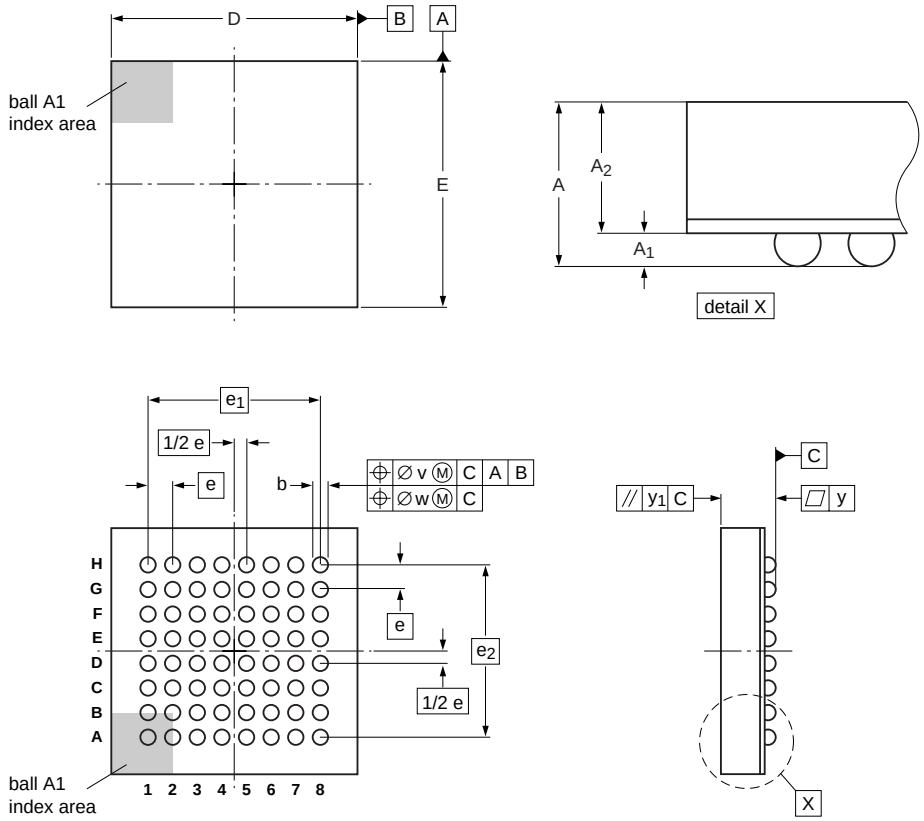

**DIMENSIONS (mm are the original dimensions)**

| UNIT | A max. | $A_1$ | $A_2$ | b | D | E | e | $e_1$ | $e_2$ | v | w | y | $y_1$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| mm | 1.1 | 0.25<br>0.15 | 0.85<br>0.75 | 0.35<br>0.25 | 6.1<br>5.9 | 6.1<br>5.9 | 0.5 | 4.5 | 4.5 | 0.15 | 0.05 | 0.08 | 0.1 |

| OUTLINE VERSION | REFERENCES | | | | EUROPEAN PROJECTION | ISSUE DATE |
|---|---|---|---|---|---|---|
| | IEC | JEDEC | JEITA | | | |
| SOT543-1 | - - - | MO-195 | - - - | | | ~~00-11-22~~<br>02-04-09 |

**Fig 40. Package outline SOT543-1 (TFBGA64)**

**TFBGA64: plastic thin fine-pitch ball grid array package; 64 balls; body 4 x 4 x 0.8 mm** **SOT969-1**




**DIMENSIONS (mm are the original dimensions)**

| UNIT | A max | $A_1$ | $A_2$ | b | D | E | e | $e_1$ | $e_2$ | v | w | y | $y_1$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| mm | 1.1 | 0.25<br>0.15 | 0.85<br>0.75 | 0.3<br>0.2 | 4.1<br>3.9 | 4.1<br>3.9 | 0.4 | 2.8 | 2.8 | 0.15 | 0.05 | 0.08 | 0.1 |

| OUTLINE VERSION | REFERENCES | | | | EUROPEAN PROJECTION | ISSUE DATE |
|---|---|---|---|---|---|---|
| | IEC | JEDEC | JEITA | | | |
| SOT969-1 | - - - | - - - | - - - | | | ~~06-09-22~~<br>06-09-27 |

**Fig 41. Package outline SOT969-1 (TFBGA64)**

# 17. Soldering of SMD packages

This text provides a very brief insight into a complex technology. A more in-depth account of soldering ICs can be found in Application Note *AN10365 "Surface mount reflow soldering description"*.

## 17.1 Introduction to soldering

Soldering is one of the most common methods through which packages are attached to Printed Circuit Boards (PCBs), to form electrical circuits. The soldered joint provides both the mechanical and the electrical connection. There is no single soldering method that is ideal for all IC packages. Wave soldering is often preferred when through-hole and Surface Mount Devices (SMDs) are mixed on one printed wiring board; however, it is not suitable for fine pitch SMDs. Reflow soldering is ideal for the small pitches and high densities that come with increased miniaturization.

## 17.2 Wave and reflow soldering

Wave soldering is a joining technology in which the joints are made by solder coming from a standing wave of liquid solder. The wave soldering process is suitable for the following:

- Through-hole components
- Leaded or leadless SMDs, which are glued to the surface of the printed circuit board

Not all SMDs can be wave soldered. Packages with solder balls, and some leadless packages which have solder lands underneath the body, cannot be wave soldered. Also, leaded SMDs with leads having a pitch smaller than ~0.6 mm cannot be wave soldered, due to an increased probability of bridging.

The reflow soldering process involves applying solder paste to a board, followed by component placement and exposure to a temperature profile. Leaded packages, packages with solder balls, and leadless packages are all reflow solderable.

Key characteristics in both wave and reflow soldering are:

- Board specifications, including the board finish, solder masks and vias
- Package footprints, including solder thieves and orientation
- The moisture sensitivity level of the packages
- Package placement
- Inspection and repair
- Lead-free soldering versus SnPb soldering

## 17.3 Wave soldering

Key characteristics in wave soldering are:

- Process issues, such as application of adhesive and flux, clinching of leads, board transport, the solder wave parameters, and the time during which components are exposed to the wave
- Solder bath specifications, including temperature and impurities

## 17.4 Reflow soldering

Key characteristics in reflow soldering are:

- Lead-free versus SnPb soldering; note that a lead-free reflow process usually leads to higher minimum peak temperatures (see Figure 42) than a SnPb process, thus reducing the process window
- Solder paste printing issues including smearing, release, and adjusting the process window for a mix of large and small components on one board
- Reflow temperature profile; this profile includes preheat, reflow (in which the board is heated to the peak temperature) and cooling down. It is imperative that the peak temperature is high enough for the solder to make reliable solder joints (a solder paste characteristic). In addition, the peak temperature must be low enough that the packages and/or boards are not damaged. The peak temperature of the package depends on package thickness and volume and is classified in accordance with Table 111 and 112

**Table 111. SnPb eutectic process (from J-STD-020C)**

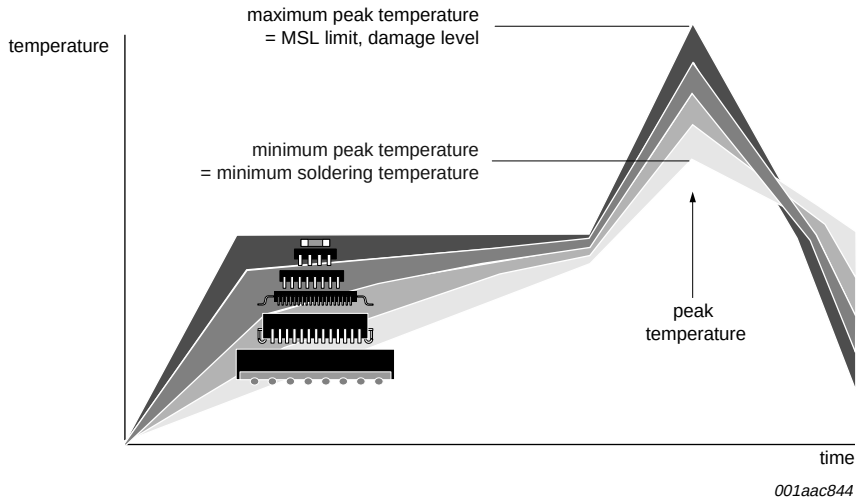
| Package thickness (mm) | Package reflow temperature (°C) | |
|---|---|---|
| | Volume ($mm^3$) | |
| | < 350 | ≥ 350 |
| < 2.5 | 235 | 220 |
| ≥ 2.5 | 220 | 220 |

**Table 112. Lead-free process (from J-STD-020C)**

| Package thickness (mm) | Package reflow temperature (°C) | | |
|---|---|---|---|
| | Volume ($mm^3$) | | |
| | < 350 | 350 to 2000 | > 2000 |
| < 1.6 | 260 | 260 | 260 |
| 1.6 to 2.5 | 260 | 250 | 245 |
| > 2.5 | 250 | 245 | 245 |

Moisture sensitivity precautions, as indicated on the packing, must be respected at all times.

Studies have shown that small packages reach higher temperatures during reflow soldering, see Figure 42.




MSL: Moisture Sensitivity Level

**Fig 42. Temperature profiles for large and small components**

For further information on temperature profiles, refer to Application Note *AN10365 "Surface mount reflow soldering description"*.

## 18. Abbreviations

**Table 113. Abbreviations**

| Acronym | Description |
|---|---|
| ACK | Acknowledgement |
| ACPI | Advanced Configuration and Power Interface |
| ASIC | Application-Specific Integrated Circuit |
| ATA | Advanced Technology Attachment |
| ATAPI | Advanced Technology Attachment Peripheral Interface |
| CRC | Cyclic Redundancy Code |
| DMA | Direct Memory Access |
| EMI | ElectroMagnetic Interference |
| ESR | Equivalent Series Resistance |
| FS | Full-Speed |
| GDMA | Generic DMA |
| HS | High-Speed |
| IDE | Integrated Development Environment |
| MDMA | Multi-word DMA |
| MMU | Memory Management Unit |
| MO | Magneto-Optical |
| NAK | Not Acknowledged |
| NRZI | Non-Return-to-Zero Inverted |
| NYET | Not Yet |

**Table 113. Abbreviations** *...continued*

| Acronym | Description |
|---|---|
| OTG | On-The-Go |
| PCB | Printed-Circuit Board |
| PHY | Physical |
| PID | Packet IDentifier |
| PIE | Parallel Interface Engine |
| PIO | Parallel Input/Output |
| PLL | Phase-Locked Loop |
| POR | Power-On Reset |
| SE0 | Single-Ended Zero |
| SIE | Serial Interface Engine |
| SRP | Session Request Protocol |
| TTL | Transistor-Transistor Logic |
| USB | Universal Serial Bus |

## 19. References

[1] Universal Serial Bus Specification Rev. 2.0

[2] On-The-Go Supplement to the USB Specification Rev. 1.3

[3] Using ISP1582/3 in a composite device application with alternate settings (AN10071)

[4] AT Attachment with Packet Interface Extension (ATA/ATAPI-4), ANSI INCITS 317-1998 (R2003)

[5] ISP1582/83 and ISP1761 clearing an IN buffer (AN10045)

# 20. Revision history

**Table 114. Revision history**

| Document ID | Release date | Data sheet status | Change notice | Supersedes |
|---|---|---|---|---|
| ISP1583_7 | 20080922 | Product data sheet | - | ISP1583_6 |
| Modifications: | • Added ISP1583ET2.<br>• Added Section 5 “Marking”.<br>• Added Table 4 “Endpoint access and programmability”.<br>• Figure 16 “Bus-powered mode”: updated the voltage range for $V_{CC(I/O)}$.<br>• Removed Section 7.9 “Clear buffer”.<br>• Table 24 “Mode register: bit allocation”: updated the bus reset value for bits CLKAON and PWRON.<br>• Table 34 “Endpoint Index register: bit allocation”: updated the reset and bus reset values for bit EP0SETUP.<br>• Table 38 “Control Function register: bit description”: updated description for bit CLBUF.<br>• Section 9.3.6 “Endpoint MaxPacketSize register (address: 04h)”: updated the first paragraph.<br>• Table 48 “Endpoint Type register: bit description”: added a remark to the DBLBUF bit description.<br>• Figure 32 “GDMA slave mode timing: DIOR (master) and DIOW (slave)”: updated.<br>• Table 53 “DMA commands”: added a remark to GDMA stop.<br>• Section 19 “References”: updated Ref. 3. | | | |
| ISP1583_6 | 20070820 | Product data sheet | - | ISP1583_5 |
| ISP1583_5 | 20070209 | Product data sheet | - | ISP1583-04 |
| ISP1583-04 (9397 750 14335) | 20050104 | Product data | 200412038 | ISP1583-03 |
| ISP1583-03 (9397 750 13461) | 20040712 | Product data | - | ISP1583-02 |
| ISP1583-02 (9397 750 12978) | 20040503 | Product data | - | ISP1583-01 |
| ISP1583-01 (9397 750 11497) | 20040225 | Preliminary data | - | - |

# 21. Legal information

## 21.1 Data sheet status

| Document status[1][2] | Product status[3] | Definition |
|---|---|---|
| Objective [short] data sheet | Development | This document contains data from the objective specification for product development. |
| Preliminary [short] data sheet | Qualification | This document contains data from the preliminary specification. |
| Product [short] data sheet | Production | This document contains the product specification. |

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

[2] The term 'short data sheet' is explained in section "Definitions".

[3] The product status of device(s) described in this document may have changed since this document was published and may differ in case of multiple devices. The latest product status information is available on the Internet at URL http://www.nxp.com.

## 21.2 Definitions

**Draft —** The document is a draft version only. The content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included herein and shall have no liability for the consequences of use of such information.

**Short data sheet —** A short data sheet is an extract from a full data sheet with the same product type number(s) and title. A short data sheet is intended for quick reference only and should not be relied upon to contain detailed and full information. For detailed and full information see the relevant full data sheet, which is available on request via the local NXP Semiconductors sales office. In case of any inconsistency or conflict with the short data sheet, the full data sheet shall prevail.

## 21.3 Disclaimers

**General —** Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information.

**Right to make changes —** NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

**Suitability for use —** NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in medical, military, aircraft, space or life support equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors accepts no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

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

**Limiting values —** Stress above one or more limiting values (as defined in the Absolute Maximum Ratings System of IEC 60134) may cause permanent damage to the device. Limiting values are stress ratings only and operation of the device at these or any other conditions above those given in the Characteristics sections of this document is not implied. Exposure to limiting values for extended periods may affect device reliability.

**Terms and conditions of sale —** NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at http://www.nxp.com/profile/terms, including those pertaining to warranty, intellectual property rights infringement and limitation of liability, unless explicitly otherwise agreed to in writing by NXP Semiconductors. In case of any inconsistency or conflict between information in this document and such terms and conditions, the latter will prevail.

**No offer to sell or license —** Nothing in this document may be interpreted or construed as an offer to sell products that is open for acceptance or the grant, conveyance or implication of any license under any copyrights, patents or other industrial or intellectual property rights.

## 21.4 Trademarks

Notice: All referenced brands, product names, service names and trademarks are the property of their respective owners.

**SoftConnect —** is a trademark of NXP B.V.

# 22. Contact information

For more information, please visit: **http://www.nxp.com**

For sales office addresses, please send an email to: **salesaddresses@nxp.com**

## 23. Tables



continued >>

## 24. Figures

## 25. Contents



continued >>

For more information, please visit: http://www.nxp.com
For sales office addresses, please send an email to: salesaddresses@nxp.com

**Date of release: 22 September 2008**
**Document identifier: ISP1583_7**