AN10070

Interfacing ISP176x to the PXA300 processor series

Rev. 01 — 19 October 2007 **Application note**

Document information

Info	Content
Keywords	isp1760; isp1761; host controller; otg; usb; peripheral controller; on-the-go
Abstract	This document provides information on how to interface the ISP1760/1 (ISP176x) to the PXA300 processor series. **Remark**: The ISP176x denotes the ISP1760 and ISP1761 Hi-Speed Universal Serial Bus controllers, and any future derivatives.

founded by Philips

Revision history

Rev	Date	Description
01	20071019	First release.

Contact information

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

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

Remark: The ISP176x denotes the ISP1760 and ISP1761 Hi-Speed Universal Serial Bus controllers, and any future derivatives.

1. Introduction

This document provides information on how to interface the ISP1760/1 (ISP176x) to the PXA300 (Monahans) processor series.

The ISP176x must be treated as a Variable Latency I/O (VLIO) device and interfaced to the Data Flash Interface (DFI) of the PXA300 processor series. The PXA300 processor series has a multiplexed address and data bus. Therefore, external address latches must be used because the ISP176x has separate address and data buses.

This document also explains how to connect interrupts lines between the ISP176x and the PXA300 processor series. This allows the ISP176x to interrupt the PXA300 processor series; instead of allowing the PXA300 processor series to poll the Interrupt register to find out what has happened.

This document also briefly describes registers in the PXA300 processor series that are responsible for the performance between the PXA300 processor series and the ISP176x.

2. Interface signals for the ISP176x

Table 1 shows the necessary signals to interface with the ISP176x.

Table 1. Required interface signals

Pin name	Description
RD_N	Read signal active LOW
WR_N	Write signal active LOW
DATA[31:0]	Data bus
A[17:1]	Address bus
CS_N	Chip select active LOW
DC_IRQ (ISP1761 only)	Peripheral controller interrupt
HC_IRQ (ISP1761/ISP1760)	Host controller interrupt

3. Multiplexed address and data bus of the PXA300 processor series

The data flash interface of the PXA300 processor series has a 16-bit multiplexed address and data bus. The design of the PXA300 processor series factored in peripheral components that do not have a multiplexed address and data interface. The PXA300 processor series has two signals, Latch Lower Address (LLA) and Latch Upper Address (LUA), specifically for components that do not use the multiplexed address and data bus. These two signals are used with external address latches to de-multiplex the address and data.

The PXA300 processor series also has an addressing mode called low-order addressing. There are four separate address lines (DF_ADDR[3:0]) for this purpose. Address bits 3 to 0 are always provided in low-order address mode. Fig 1shows that DF_ADDR[3:0] can permanently be connected as the lower address bits to the ISP176x.

If the processor used has a separate address and data bus, the ISP176x can be interfaced without the need for external glue logic.

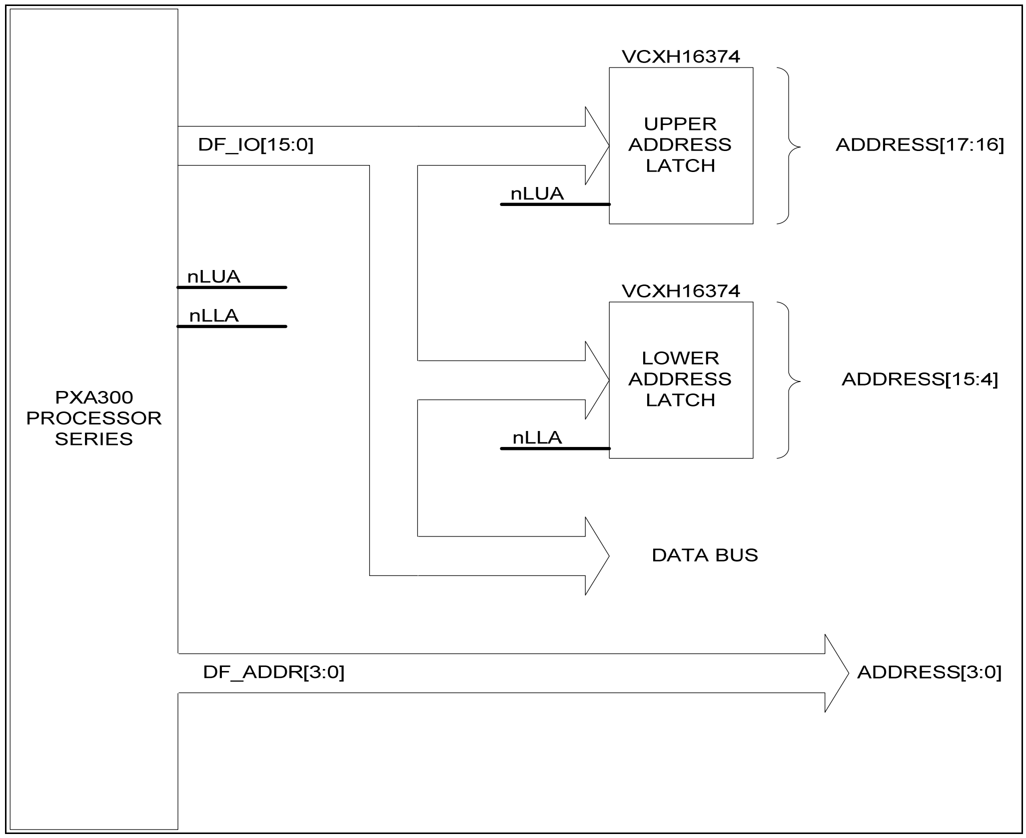
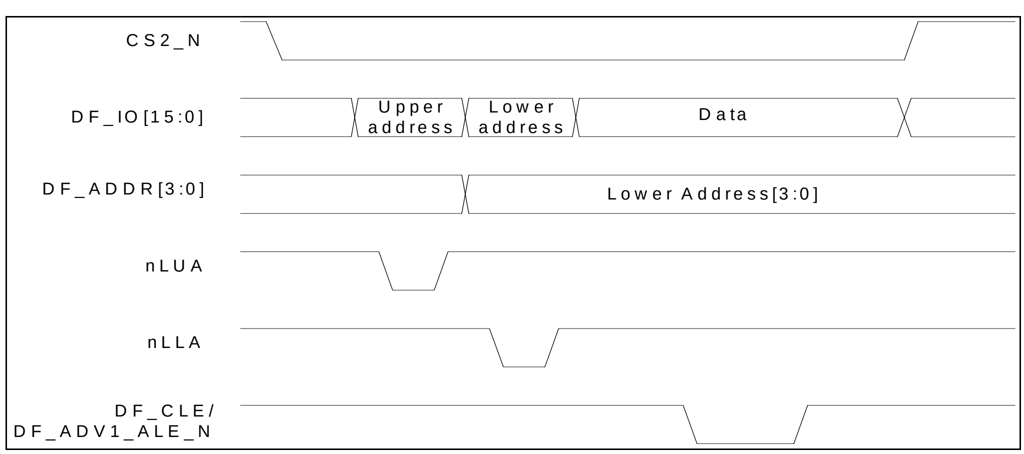


Fig 1. De-multiplexing address and data

Fig 2 shows a timing diagram on how LUA and LLA will be strobed to allow external address latches to latch the address during the addressing phase.



For details on signals, see Table 2

Fig 2. Address latch and data phase timing

4. Connecting the ISP176x to the PXA300 processor series

Table 2 shows the interface signals used by the PXA300 processor series to communicate with the ISP176x.

Table 2. Interface signals used by the PXA300 processor series

Pin name	Description
CS2_N	Chip select active LOW
DF_IO[15 : 0]	Data bus and address bus
DF_ADDR[3:0]	Low-order address signals
DF_CLE	Read signal active LOW
DF_ADV1_ALE_N	Write signal active LOW
GPIOx	Peripheral controller interrupt
GPIOy	Host controller interrupt

Fig 3 shows a connection block diagram on how the PXA300 processor series must be connected to the ISP176x.

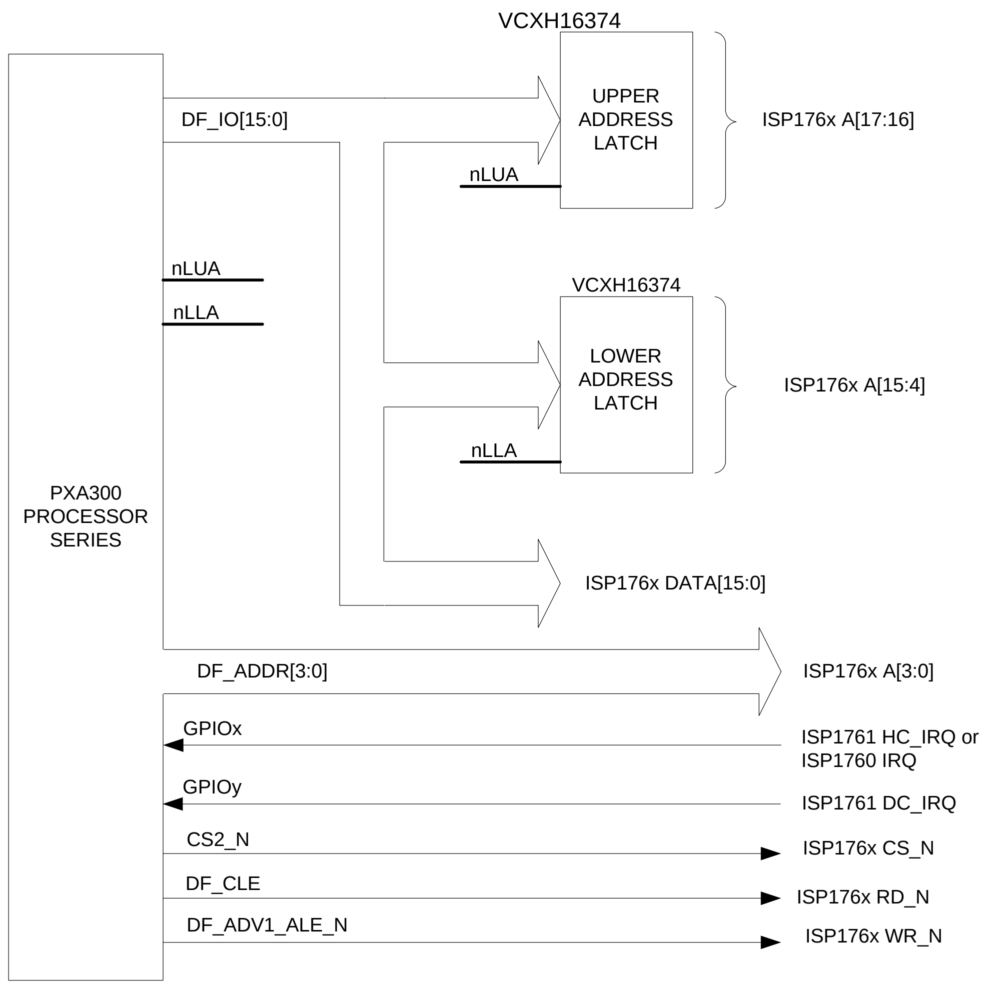


Fig 3. Connection block diagram

5. Configuring the ISP176x into 16-bit mode

Because the PXA300 processor series has an external 16-bit data bus, the ISP176x must also be configured in 16-bit mode. By default, however, the ISP176x is in 32-bit mode. By design, the DATA_BUS_WIDTH bit is located in the lower word of the HW Mode Control register (address: 0300h) of the ISP176x.

For the first initialization, the ISP176x will be in 32-bit mode, which is the default setting. To set the ISP176x in 16-bit mode, data lines 31 to 16 must be externally pulled up using a 15 kΩ resistor. Therefore, when the microprocessor writes a 16-bit value to the HW Mode Control register, FFFF nnnnh gets written, where nnnn equals the respective configuration value that the microprocessor writes, and FFFF is because of external pull-up resistors.

Once this is completed, the microprocessor must know that the ISP176x is configured in 16-bit mode, and subsequent access to the ISP176xs HW Mode Control register will be performed with two 16-bit writes or reads. The microprocessor must then clear to zero the upper word of the HW Mode Control register because according to the data sheet, the upper word must be zero.

Remark: The ISP176x is capable of both 16-bit and 32-bit operations. If the external data bus of the CPU is configurable as 32-bit mode, it is recommended that you use 32-bit. The 32-bit operation allows better throughput because 4 bytes of data are read or written per strobe in 32-bit mode versus 2 bytes per strobe in 16-bit mode.

6. Common interrupt mode of the ISP1761

The ISP1761 is a Hi-Speed USB host controller and peripheral controller. COMN_INT mode feature in the ISP1761 allows both the host controller and peripheral controller interrupts to be routed to the HC_IRQ pin of the ISP1761.

The ISP1761 has two sets of registers to handle interrupt events for the host and device. Because these are two separate functions, the ISP1761 has two interrupt pins HC_IRQ and DC_IRQ.

Using COMN_INT mode in the HW Mode Control register allows saving of one interrupt pin for the PXA300 processor series. When any interrupt event occurs, the interrupt will always be routed to the HC_IRQ pin, irrespective of whether it is from the host or device.

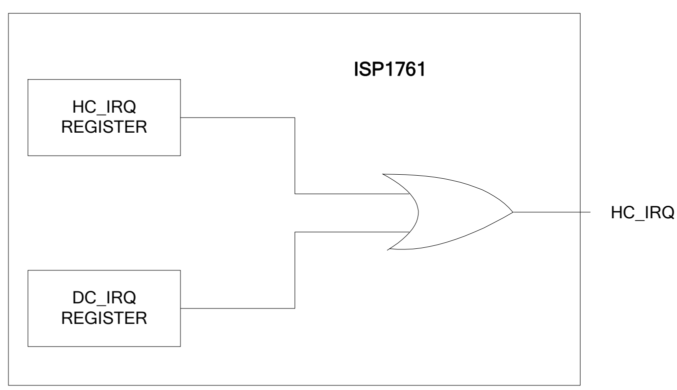


Fig 4. Simplified logic of COMN_INT

7. Configuring interrupts on the PXA300 processor series

The General-Purpose I/O (GPIO) pins of the PXA300 processor series can be used as interrupt inputs. These must be programmed to sense active LOW and HIGH transitions. This section describes the register set of the PXA300 processor series and how they relate to interrupt configurations. From a system point of view, typically, there will be a real-time operating system, and the hardware abstraction layer will abstract the register details of the PXA300 processor series. Because the operating system is present, standard API calls are used to configure and initialize interrupts of the PXA300 processor series.

7.1 Configuring the GPIO pin as an interrupt input

The GPIO Pin Direction Register x (GPDRx) controls the direction of the GPIO pin in use. Because the pin is selected to be an input interrupt pin, the direction register must be programmed according to the ISP176x interrupt polarity. It is recommended that you configure the ISP176x as level-triggered to ensure that the processor sees the interrupt before clearing it. To set or clear the GPDRx register, the PXA300 processor series provides set and clear registers called GPIO Pin Output Set Register x (GSDRx) and GPIO Pin Output Clear Register x (GCDRx), respectively.

The respective GPIO pin that is selected as an interrupt input must be configured as an input.

7.2 Falling edge transitions

The GPIO Falling-Edge Detect-Enable Register x (GFERx) is responsible to detect falling edges. There are four such registers. Each is 32-bit wide, controlling the respective 32 GPIO pins.

GFER0 – Controls GPIO[31:0]

GFER1 – Controls GPIO[63:32]

GFER2 – Controls GPIO[95:64]

GFER3 – Controls GPIO[127:96]

GFERx registers are set and cleared by their respective GSFERx (GPIO Bit-Wise Set Falling – Edge Detect - Enable) and GCFERx (GPIO Bit-Wise Clear Falling – Edge Detect - Enable) registers.

7.3 Rising edge transitions

The register that is responsible to detect rising edges is GRERx (GPIO Rising-Edge Detect-Enable Register x). There are four such registers. Each is 32-bit wide, controlling the respective 32 GPIO pins.

GRER0 – Controls GPIO[31:0]

GRER1 – Controls GPIO[63:32]

GRER2 – Controls GPIO[95:64]

GRER3 – Controls GPIO[127:96]

GRERx registers are set and cleared by their respective GPIO Bit-Wise Set Rising – Edge Detect - Enable (GSRERx) and GPIO Bit-Wise Clear Rising –Edge Detect - Enable (GCRERx) registers.

7.4 PXA300 processor series registers to read on interrupt

The four registers related to notifying the software about interrupt events in the PXA300 processor series are Interrupt Controller Pending registers ICPR and ICPR2, and Interrupt Controller IRQ Pending registers ICIP and ICIP2.

Interrupt Controller Pending register: Reflects the status of an interrupt event whether it is masked or unmasked.

Interrupt Controller IRQ Pending register: Reflects the status of an interrupt event only if it is unmasked.

The GPIO_X bit reflects that one of the GPIOs with edge-detection enable has been triggered.

7.5 GPIO triggered on interrupt

Software must navigate the respective GPIO Edge Detect Status Register x (GEDRx) to find out which GPIO pin has detected an event, and decide which interrupt handler to call. Registers that correspond to the respective GPIOs are:

GEDR0[31:0] Correspond to GPIO[31:0]

GEDR1[31:0] Correspond to GPIO[63:32]

GEDR2[31:0] Correspond to GPIO[95:64]

GEDR3[31:0] Correspond to GPIO[127:96]

8. Improving performance between the PXA300 series and the ISP176x

This section explains the registers in the PXA300 processor series that allow you to control the access timing of the VLIO interface, improving performance between the ISP176x and the PXA300 processor series.

DF_SCLK is the reference of all speed improvements. When timing is adjusted on the VLIO interface, the amount of time that is to be adjusted is stated in terms of the number of DF_SCLK cycles. The faster the DF_SCLK frequency, the shorter the time is for a cycle, and therefore, the wait cycle is shorter.

SMCFS[25:23]: SMCFS bits is controlled by register ACCR in the PXA300 processor series. Based on settings in the SMCFS field, various frequencies clocking Static Memory Controller (SMEMC) can be configured. DF_SCLK is then based on a division of SMEMC.

DF_CLKDIV [18:16]: This divisor is found in the MEMCLKCFG register under field DF_CLKDIV of the PXA300 processor series. At the time this document was written, the DF_SCLK maximum frequency was 52 MHz. Therefore, an example of the setting is:

SMCFS = 104 MHz

DF_CLKDIV = 2

CSADRCFGx – ALT, ALW, ADDRCONFIG

Because the PXA300 processor series is a multiplexed address and data bus interface, the addressing phase must be considered to improve system performance. Address Latch Timing (ALT) controls whether there is one D_SCLK either of set-up or hold time, or of both, for the address latching action on the LUA and LLA signals.

ALW controls the width of the LUA and LLA assertion portions. This width is determined by the number of DF_SCLK cycles.

RDNx, RDFx: These fields in MSC1 control the assertion and deassertion timing of DF_CLE and DF_ADV1_ALE_N, which are the read and write signals, respectively.

9. Legal information

9.1 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.

9.2 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 a 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 for 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.

9.3 Trademarks

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

10. Contents

Date of release: 19 October 2007
Document identifier: AN10070_1