AN10043

Embedded Systems Design with the ISP176x

Rev. 02 — 03 May 2005 Application note

Document information

Info	Content
Keywords	isp1760. isp1761, usb, universal serial bus, host controller, peripheral controller, otg controller, on-the-go, otg, interface device
Abstract	This document explains the factors determining ISP176x's performance. **Note**: The ISP176x denotes the ISP1760 and ISP1761 Hi-Speed Universal Serial Bus controllers, and any future derivative.

Revision history

Rev	Date	Description
02	20050503	Second release. Updated Section 1, Section 2, Section 3 (last paragraph), Section 3.1 and Section 3.3,.
01	20040907	First release.

Contact information

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

For sales office addresses, please send an email to: sales.addresses@www.semiconductors.philips.com

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

1. Introduction

The ISP1761 is a Hi-Speed Universal Serial Bus (USB) On-The-Go (OTG) Controller integrated with a Host Controller and a Peripheral Controller. The ISP1760 is a Hi-Speed USB Host Controller.

The ISP176x has a generic asynchronous processor interface, allowing direct connectivity to most of the processors present in the market today. The ISP176x is memory-mapped; adding 64 kB of address space to the existing system's memory. The CPU interface is non-multiplexed asynchronous, that is, the address bus and the data bus are separate, and no system clock input is necessary.

2. Maximum bandwidth considerations

The ISP176x specifications define a typical Host Controller access cycle time of approximately 40 ns in Programmed Input/Output (PIO) mode using CS_N for the dedicated access validation, and 51 ns in Direct Memory Access (DMA) mode. These values are different for Peripheral Controller access, that is, when Port 1 is configured as a Peripheral Controller. For details, refer to the ISP1761 data sheet.

The ISP176x can be defined as a 'Variable Latency' resource. The settings of the ISP176x RD_N and WR_N access cycle time are based on PXA255 internal MEMCLK = 99.5 MHz. This determines a resolution of about 10 ns for the RD_N and WR_N signals active time or for cycle-time adjustments.

The typical cycle-time may be increased to approximately 50 ns in PIO mode and 60 ns in DMA mode because of the 10 ns increment required, determined by the propagation delay through the address and data buffers on the extension bus, which is estimated in the range of few ns.

3. Typical DMA data transfer

Study of CPU use during various USB data transfers shows that the DMA use is important in reducing CPU loading, improving the overall system performance.

Fig 1 shows the timing diagram of a typical read data transfer performed by the system DMA from the ISP176x internal memory. In Fig 1, the master DMA of the system is waiting for the DREQ signal from the ISP176x slave DMA that will initiate the DMA transfer. Waiting for the ISP176x DREQ is optional. The PXA255 can be alternatively configured to start the DMA transfer whenever its DMA is enabled, if it is confirmed that the ISP176x is ready for data transfer.

In this example, DREQ0 is active HIGH, DACK0\ is active LOW; and other control signals CS\, WE\ and OE\ (READ\) are active LOW.

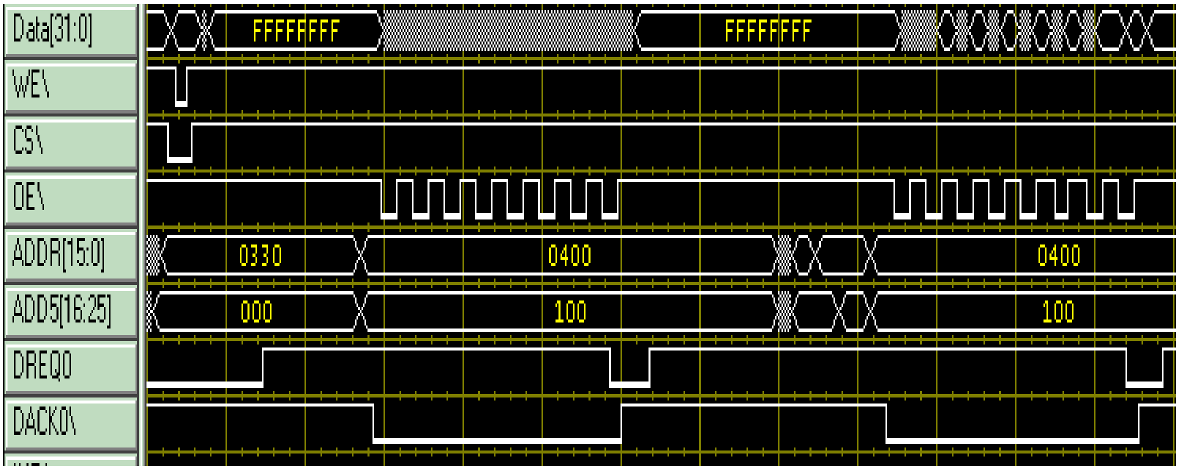


Fig 1. Read data transfer performed by the system DMA from the ISP176x internal memory.

As shown in the diagram, DREQ0 is deasserted during the last RD_N or WR_N pulse at the end of each burst, according to the burst length specified in the ISP176x DMA Configuration register. On detecting the DACK0\ deassertion, DREQ0 is again asserted, as long as the DMA transfer counter programmed in the ISP176x DMA Configuration register is not completed.

Remark: The system's DMA master transfer must be programmed according to the ISP176x DMA settings.

Data transfer can also be executed as a memory-to-memory DMA data transfer because the ISP176x internal memory is mapped as an additional memory in the system's addressing space. The DMA transfer may be performed even when the ISP176x is in PIO mode. Therefore, its internal slave DMA may not necessarily be programmed. This mode allows using a better cycle time—PIO cycle time = 40 ns versus DMA cycle time = 51 ns—and also eliminates the DREQ assertion time (82 ns).

3.1 Maximum bandwidth in microprocessor DMA—ISP176x DMA case

The maximum data transfer rate that can be achieved by accessing the ISP176x on the system's extension bus in DMA mode, according to the timing diagram in Fig 1, will be approximately 42 MB/s.

Consider that:

ISP176x DMA access cycle time: 51 ns.

Increment because of address and data buffers propagation time: 10 ns.

DREQ assertion time: 82 ns.

Estimated DACK latency time: 200 ns (implementation dependent).

Therefore, the time required to perform an 8-cycle burst (32 B) is:

8 x (50 + 10) ns + 82 ns + 200 ns = 762 ns.

This means a maximum bandwidth of about 42 MB/s[1], for a continuous DMA data transfer, considering the time necessary to read the data from the system's SDRAM,

[1] Estimated value assuming that the DMA transfer is continuous and completely occupies the extension bus, and the DACK latency time is 200 ns.

which is a part of the DACK latency time specified earlier. The time required to write the respective data into the system's SDRAM is not included.

3.2 Maximum bandwidth in microprocessor DMA—ISP176x PIO case

As mentioned in the preceding section, a DMA data transfer can also be executed without programming the ISP176x slave DMA controller. So, the ISP176x is accessed in PIO mode. In this case, CS\ must be generated instead of DACK0. The ISP176x will not assert IRQ at the end of the data transfer because it is configured in PIO mode. Instead an internal processor interrupt should be generated when DMA is completed. This example will determine the bandwidth gain.

In this case, consider that:

ISP176x PIO access time: 40 ns.

Address and data buffers propagation time: 10 ns.

Estimated DACK latency time: 200 ns.

Therefore, the time required to perform an 8-cycle burst (32 B) will be:

8 x (40 + 10) ns + 200 ns = 600 ns.

This determines a maximum bandwidth of approximately 53 MB/s[2], a sensible improvement from the example in Section 3.1.

A different perspective on data transfer is the time left for other tasks to access the system extension bus. Consider a continuous 20 MB/s data transfer. This will occupy about 476 ms of one second, in the case of Section 3.1. It, however, will take only about 377 ms of one second in this case, that is, when the ISP176x in normal PIO mode is accessed by the system DMA.

This means that although the ISP176x slave DMA may be necessary in certain configurations, such as when the host processor requires the DREQ signal, it is a better solution to use the microprocessor DMA—ISP176x PIO configuration, whenever possible.

3.3 Factors determining the maximum bandwidth

The maximum bandwidth for the ISP176x access is determined by:

- The priority allocation for the system DMA channel, performing the data transfers between the system memory and the ISP176x memory. This will influence the DACK latency time, which directly affects the maximum achievable bandwidth.
- Other data transfers that are simultaneously in progress in the system. These affect the DACK latency, according to priority allocation.
- The system DMA burst length; a longer burst length will improve the data transfer rate.
- The system's resolution for the ISP176x access time settings, depending on processor's architecture.
- Any glue logic or buffer will add propagation delay to the cycle-time specified in the ISP176x data sheet, with an increment determined by the access time setting resolution.

[2] Estimated value assuming that the DMA transfer is continuous and completely occupies the extension bus.

- If DREQ from the ISP176x slave DMA is required to trigger the DMA transfer, the DREQ latency time (82 ns) is added to the burst length time.
- The typical ISP176x cycle time will increase in systems in which the ISP176x CPU interface is powered using 1.8 V, according to the ISP176x data sheet specifications.

4. Extension bus bandwidth usage

Certain types of permanent data transfers in a system will continuously occupy part of the system bus bandwidth. Some of the most common permanent data transfers that take place on the system extension bus are:

- The LCD refresh or the display memory periodic update.
- The system’s memory refresh.
- The processor’s instructions and data fetching from the system memory.

Usually the LCD DMA has the highest priority in systems to avoid display flicker. Consider a display of 640 x 480 x 8 bpp, with a refresh rate of 60 Hz, the resulting bandwidth continuously occupied is approximately 18 MB/s.

For a burst length of 4 and 100 ns, when refresh is performed from the system SDRAM, the LCD refresh may occupy 10% to 30% of the total bus bandwidth, depending on the type and speed of the memory used in the system and the LCD size.

For details on the calculations, refer to the Intel application note *Bandwidth Calculations for the SA-1110 LCD Displays*.

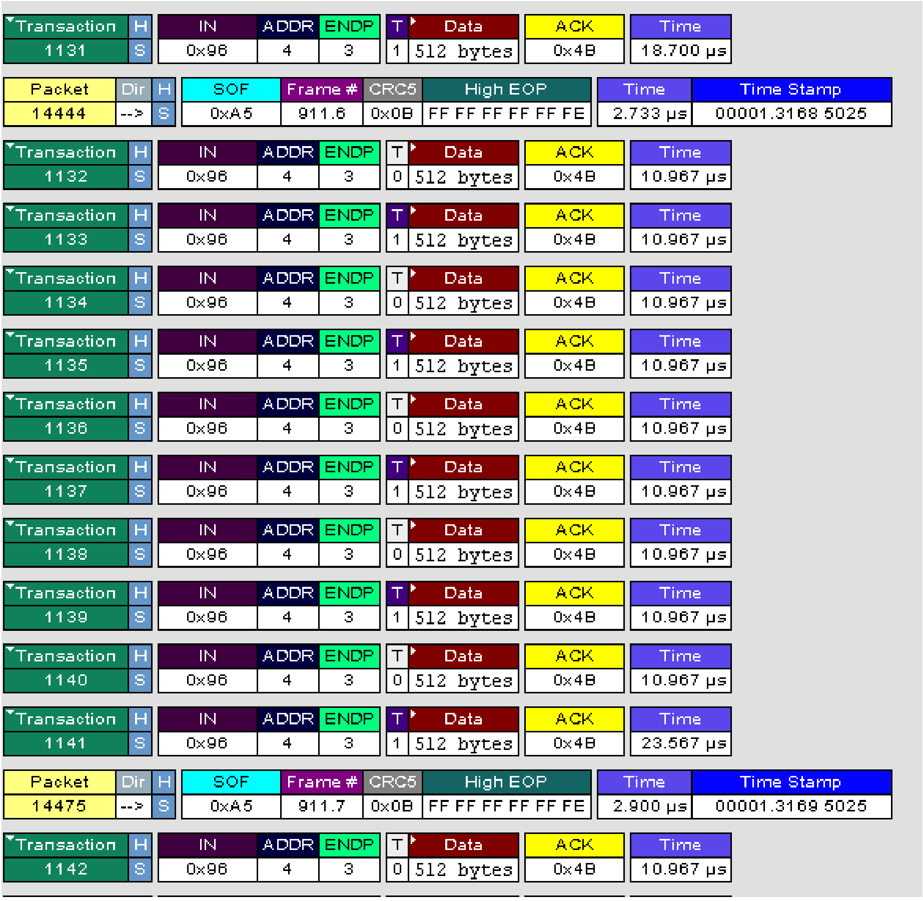
5. Confirming the ISP176x EHCI bandwidth capability

As shown in earlier sections, the CPU power or MIPS specifications do not directly influence the ISP176x USB bandwidth capability because DMA will normally be used for low processor usage. The bandwidth mostly depends on the items listed in Section 3.3.

A simple test shows that the Hi-Speed USB Host Controller architecture of the ISP176x has no intrinsic bandwidth limitation as compared to any typical Hi-Speed USB PCI Host Controller.

To eliminate the possible limitations determined by the system extension bus, as described in the preceding section, and to obtain a correct estimation of the maximum Hi-Speed USB Host Controller block capabilities, the payload data is not updated during the data transfer as shown in Fig 2. This means that data transfer is performed by just initializing the PTDs and then validating the transfers, with the ‘undetermined’ data.

Fig 2 shows that the Host Controller can achieve 10 x 512 B packets in one µFrame, which is equivalent to a maximum instantaneous bandwidth of 40 MB/s. This is the same as the maximum data transfer achieved on a standard PCI Hi-Speed USB Host Controller.



```
Transaction H  IN    ADDR ENDP  T Data       ACK   Time
1131        S  0x96  4    3     1 512 bytes  0x4B  18.700 µs
Packet Dir H  SOF   Frame # CRC5  High EOP           Time      Time Stamp
14444  --> S  0xA5  911.6   0x0B  FF FF FF FF FF FE  2.733 µs  00001.3168 5025
Transaction H  IN    ADDR ENDP  T Data       ACK   Time
1132        S  0x96  4    3     0 512 bytes  0x4B  10.967 µs
Transaction H  IN    ADDR ENDP  T Data       ACK   Time
1133        S  0x96  4    3     1 512 bytes  0x4B  10.967 µs
Transaction H  IN    ADDR ENDP  T Data       ACK   Time
1134        S  0x96  4    3     0 512 bytes  0x4B  10.967 µs
Transaction H  IN    ADDR ENDP  T Data       ACK   Time
1135        S  0x96  4    3     1 512 bytes  0x4B  10.967 µs
Transaction H  IN    ADDR ENDP  T Data       ACK   Time
1136        S  0x96  4    3     0 512 bytes  0x4B  10.967 µs
Transaction H  IN    ADDR ENDP  T Data       ACK   Time
1137        S  0x96  4    3     1 512 bytes  0x4B  10.967 µs
Transaction H  IN    ADDR ENDP  T Data       ACK   Time
1138        S  0x96  4    3     0 512 bytes  0x4B  10.967 µs
Transaction H  IN    ADDR ENDP  T Data       ACK   Time
1139        S  0x96  4    3     1 512 bytes  0x4B  10.967 µs
Transaction H  IN    ADDR ENDP  T Data       ACK   Time
1140        S  0x96  4    3     0 512 bytes  0x4B  10.967 µs
Transaction H  IN    ADDR ENDP  T Data       ACK   Time
1141        S  0x96  4    3     1 512 bytes  0x4B  23.567 µs
Packet Dir H  SOF   Frame # CRC5  High EOP           Time      Time Stamp
14475  --> S  0xA5  911.7   0x0B  FF FF FF FF FF FE  2.900 µs  00001.3169 5025
Transaction H  IN    ADDR ENDP  T Data       ACK   Time
1142        S  0x96  4    3     0 512 bytes  0x4B  10.967 µs
```

Fig 2. CATC capture performed during a high-speed bulk data transfer.

The average bandwidth reported by the CATC in the ‘Time and Bus Usage Calculations’ tool is approximately 32 MB/s, again similar to a Hi-Speed USB PCI Host Controller because not all μFrames contain 10 x 512 B packets. This is because of the non-optimized USB data transfer programming, related to operating system-specific latencies.

The preceding calculations show that as long as the system can perform the necessary data transfers to or from the ISP176x internal memory, the maximum bandwidth achieved by the ISP176x is similar to that achieved by a standard PCI Hi-Speed USB Host Controller.

6. References

- Bandwidth Calculations for the SA-1110 LCD Displays
- Universal Serial Bus Specification Rev. 2.0
- ISP1761 Hi-Speed Universal Serial Bus On-The-Go controller data sheet
- ISP1760 Hi-Speed Universal Serial Bus host controller for embedded applications data sheet
- ISP176x Hi-Speed USB OTG Host Controller PCI Demo Board user manual.

7. Disclaimers

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

Right to make changes — Philips Semiconductors reserves the right to make changes in the products - including circuits, standard cells, and/or software - described or contained herein in order to improve design and/or performance. When the product is in full production (status ‘Production’), relevant changes will be communicated via a Customer Product/Process Change Notification (CPCN). Philips Semiconductors assumes no responsibility or liability for the use of any of these products, conveys no licence or title under any patent, copyright, or mask work right to these products, and makes no representations or warranties that these products are free from patent, copyright, or mask work right infringement, unless otherwise specified.

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

8. Trademarks

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

9. Contents

Date of release: 03 May 2005

Published in The Netherlands