



Revision history

<i>Rev</i>	<i>Date</i>	<i>Description</i>
2.0	June 2004	Added errata 2. Changed "Device Controller" to "Peripheral Controller"
1.0	April 2002	Errata on empty packet with workaround

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



Errata 1: During the DMA transfer, empty packets are sometimes sent for an IN token.

1.1 Problem Description

In the DMA mode, for an IN token sometimes an empty packet generated and sent to the host because an empty packet is written to the buffer when the following two events occur at the same time:

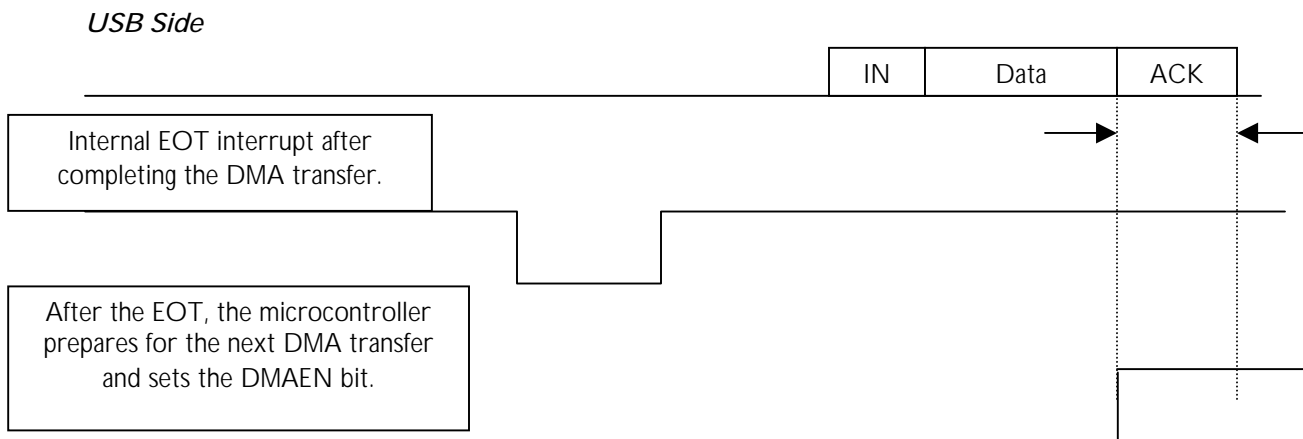
- The ISP1181B receives an ACK from the Host Controller and
- The microcontroller sets the DMAEN bit in the DMA Configuration register.

When these two events occur at the same time, an empty packet is automatically written to the internal buffer, which in turn will be sent out during the next IN token.

In a normal application, the scenario will be as follows:

- 1) The microcontroller sets the DMA counter, disables the endpoint interrupts (EOT interrupt enabled), and enables the DMA transfer by setting the DMAEN bit.
- 2) The DMA transfer starts (writing to the IN endpoint) and once the DMA transfers the number of bytes as programmed in the DMA Counter register, an internal EOT interrupt is generated.
- 3) The microcontroller prepares itself for the next DMA transfer by setting the DMAEN bit. During this time, the host might receive the last packet of the previous DMA transfer and ACKs.

The events are given in the following diagram:



The preceding events occur inside the ISP1181B. The dotted lines indicate the coincidence of these events.

1.2 Implication

The implication is serious in certain applications in which the empty packet is considered as an error condition or termination of the transfer. For example, in certain mass storage implementations, the empty packet is considered as an error condition and the Host Controller will try to send a command to rectify the error, which cannot be seen by the firmware. This is because the microcontroller has already set the DMAEN bit, activating the DMA transfer and has also disabled endpoint interrupts to reduce the overhead. In view of this, whatever the Host Controller sends for error recovery will not be seen and the system stops responding.

1.3 Workaround

To avoid the coincidence of these two events, the firmware must check whether the corresponding DMA IN endpoint is empty (in the case of double buffering, check whether both the buffers are empty) before setting the DMAEN bit in the DMA Configuration register. This ensures that the data from the IN endpoint is already sent to the Host Controller, and the ACK from the Host Controller is received well before the DMAEN bit is set.

1.4 Status

The firmware workaround is a proven one with little overhead on the CPU.

Errata 2: Data corruption during write operation to the ISP1181B Peripheral Controller.

2.1 Problem Description

After a data write operation, the ISP1181B Peripheral Controller requires a 132 ns (min) delay before a write assertion can be issued for other devices. This must be fulfilled even after CS_N is deasserted (See Figure 1 and Figure 2).

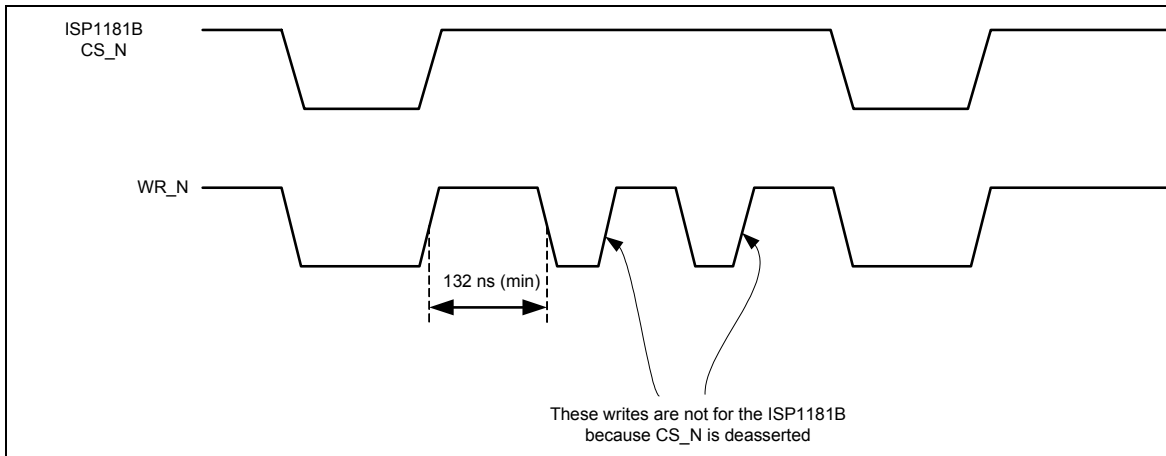


Figure 1

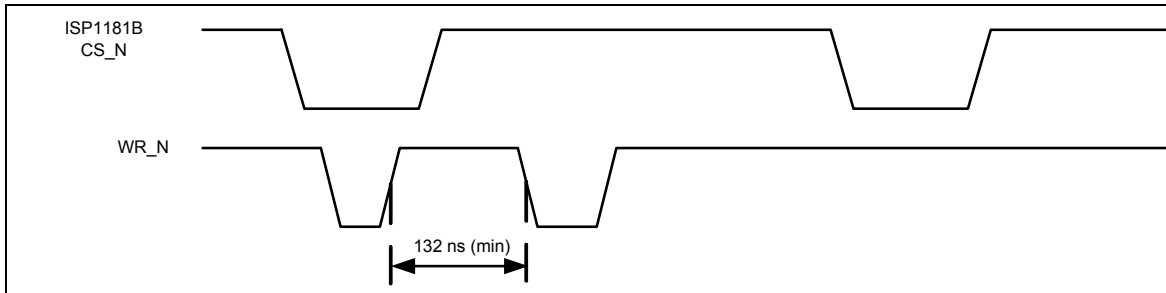


Figure 2

2.2 Implications

Will lead to data corruption if the timing requirements in Figure 1 and Figure 2 are not handled.

2.3 Workaround

Make sure that the system is able to handle the write timing requirements as given in Figure 1 and Figure 2. If the system is really fast and needs immediate write accesses to other devices, then it is better to qualify WR_N with respect to CS_N and provide the resultant write signal to the ISP1181B (see Figure 3).

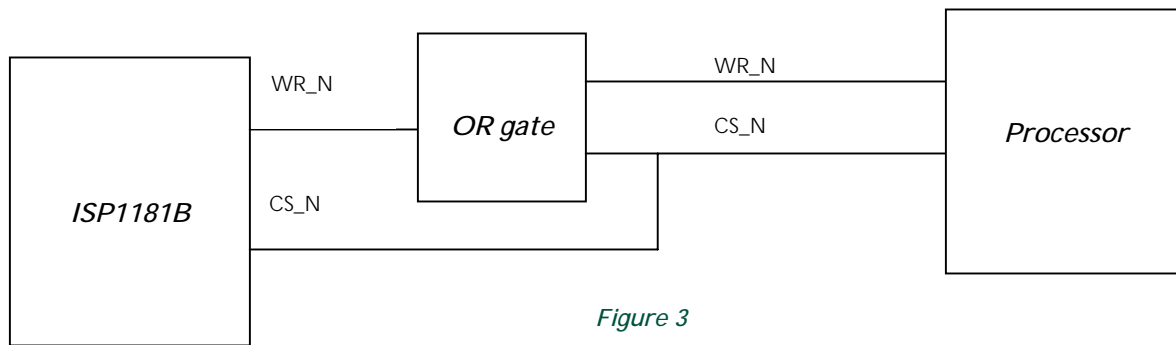


Figure 3



Note: Please take into consideration the propagation delay of the logics.

2.4 Status

No fix is planned.