AN 19.17

ULPI Design Guide

1 Introduction

The purpose of this document is to serve as a design and verification companion for anybody designing a system using a UTMI+ Low Pin Interface (ULPI). The intention here is to present hardware design guidelines and provide supplemental explanations of the ULPI protocol. While some sections of the ULPI specification are repeated for convenient reference, this document is a supplement to the ULPI Specification so that both documents can be used to enable successful product design.

The most common use of the ULPI interface is in systems that connect a Hi-Speed USB 2.0 transceiver to a Link.

1.1 References

- UTMI+ Low Pin Interface (ULPI) Specification, Revision 1.1, October 20th, 2004
- Universal Serial Bus Specification, Revision 2.0, April 27, 2000
- On-The-Go Supplement to the USB 2.0 Specification, Revision 1.3, December 5, 2006
- USB 2.0 Transceiver Macrocell Interface (UTMI) Specification, Version 1.02, May 27, 2000
- UTMI+ Specification, Revision 1.05, February 25, 2004

1.2 Background

The Universal Serial Bus (USB) is a serial data interface that supports data exchange between a host computer and a device. ULPI defines an interface between the Link and USB Transceiver to enable a transceiver to Serial Interface Engine (SIE), or “Link”, connection with only 12 pins. It is highly recommended that any designer become familiar with both the USB and ULPI Specifications. This application note serves as a convenient reference and design supplement to those specifications.

2 ULPI Overview

ULPI is a standards based interface designed to provide a high capacity point-to-point connection between an SoC or ASIC to a Hi-Speed USB Transceiver, as shown in Figure 2.1.

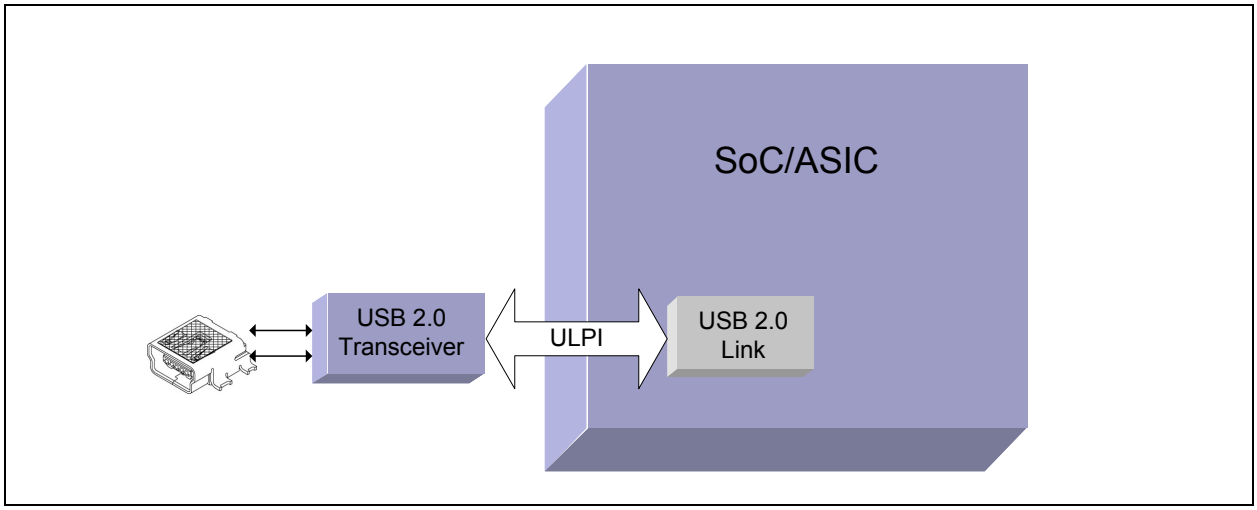

Figure 2.1 Interface Connecting a Link to a Transceiver

ULPI is a reduced pincount encapsulation and extension of the USB Transceiver Macrocell Interface (UTMI) and UTMI+ Specification. The ULPI signals are summarized here in Table 2.1 and a sample system interconnect is shown in Figure 2.2.

Table 2.1 ULPI Interface Signals Relative to the Transceiver

SIGNAL	DIRECTION	DESCRIPTION
CLK	I/O	60MHz ULPI clock. All ULPI signals are driven synchronous to the rising edge of this clock.
DATA[7:0]	I/O	8-bit bi-directional data bus. Bus ownership is determined by DIR. The Link and transceiver initiate data transfers by driving a non-zero pattern onto the data bus. ULPI defines interface timing for a single-edge data transfers with respect to rising edge of the ULPI clock.
DIR	OUT	Controls the direction of the data bus. When the transceiver has data to transfer to the Link, it drives DIR high to take ownership of the bus. When the transceiver has no data to transfer it drives DIR low and monitors the bus for commands from the Link. The transceiver will pull DIR high whenever the interface cannot accept data from the Link, such as during PLL start-up.
STP	IN	The Link asserts STP for one clock cycle to stop the data stream currently on the bus. If the Link is sending data to the transceiver, STP indicates the last byte of data was on the bus in the previous cycle.
NXT	OUT	The transceiver asserts NXT to throttle the data coming from the Link. When the Link is sending data to the transceiver, NXT indicates when the current byte has been accepted by the transceiver. The Link places the next byte on the data bus in the clock cycle following the assertion of NXT.

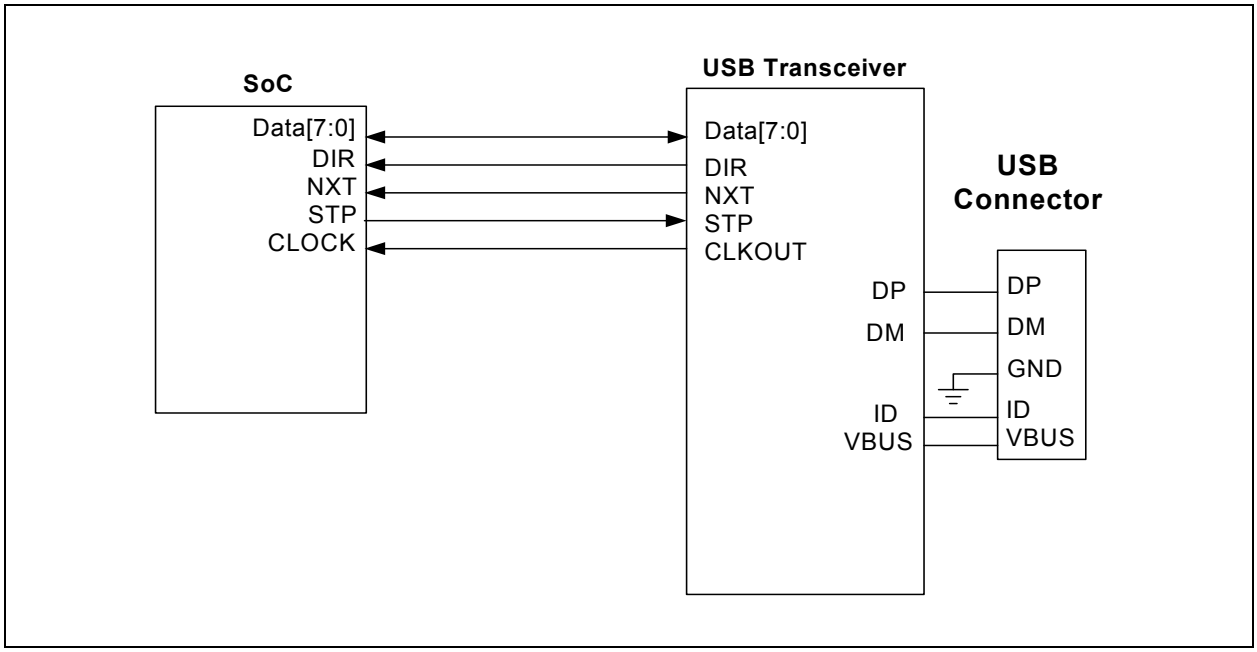

Figure 2.2 ULPI Interface Between SoC and USB Transceiver (Output Clock Mode)

3 ULPI Hardware Design

This section covers general hardware design guidelines for a ULPI interface. For hardware guidelines relative to specific SMSC USB Transceiver products, SMSC provides application notes for each package type and product line. These application notes are available through your local SMSC FAE. It is highly recommended that design reference materials from the SoC/ASIC vendor be used in conjunction with technical documentation offered by SMSC.

3.1 Physical Interconnect

ULPI interface timing is defined in the ULPI specification and summarized in Section 5.1. To meet the ULPI timing specification and insure robust ULPI interface design, three key system elements that contribute to the ULPI timing budget must be considered:

- USB Transceiver
- Printed Circuit Board Design
- SoC

The datasheet for each SMSC USB ULPI Transceiver specifies its ULPI Interface Timing. For all SMSC products, the ULPI timing of the transceiver meets or exceeds the requirements defined in the ULPI specification.

The Printed Circuit Board should be designed based on the SoC design guidelines provided by the SoC vendor. The main PCB considerations for all ULPI designs are:

- Controlled Impedance Traces (50-60 ohms)
- Trace Length
- Trace-to-Trace Skew

A conservative rule for ULPI trace length is to limit each trace to a total length of less than 3 inches, but this parameter is totally dependent on the specific SoC being used. Even SoCs from the same vendor which use the same Link IP are subject to differences, such as pin routing and muxing, which affect ULPI timing . The CLK signal is the clock reference to which all ULPI signals and logic are referenced. Ideally, ULPI traces should be routed to within 0.5 inches of CLKOUT.

To increase confidence in the ULPI PCB design, signal integrity analysis should be performed with the goal of maximizing signal quality. Accurate simulation can expose signal integrity issues early in the design cycle, and reduce development time and expense.

A series termination resistor placed near the clock source has been shown to be beneficial in some ULPI applications, but this is best confirmed through simulation. If possible, it is recommended that designs include at least a zero ohm CLK source series termination resistor in the case that termination is found to be necessary after the PCB has been built up.

3.2 Design for Test and Verification

When verifying timing or when trying to diagnose a problem on a USB interface, a logic analyzer is often required to quickly accomplish these tasks. To facilitate this effort, it is highly recommended that all ULPI signals be brought to a test point. Ideally, signals should be brought out to a Tektronix D-Max or Agilent Soft Touch low capacitance logic analyzer probe interface. However, for most applications, a more practical expectation is that all ULPI signals should simply be accessible through either a test point or a via accessible from the top or bottom side of the PCB.

3.3 Other Design Considerations

As with all high-speed digital circuits, robust designs should isolate the ULPI signals from noise from adjacent signals and power supplies.

4 ULPI Start-Up and Resets

4.1 Power-On Reset (POR)

Figure 4.1 shows the SMSC Transceiver start-up sequence from the time that the ULPI I/O power supply is stable (T0) to the time that the Transceiver completes its POR (T2). When the USB Transceiver is in reset, the DIR pin will be driven high until the 60 MHz output clock is stable. In the case of ULPI Input Clock Mode, DIR will stay high until the internal PLL is locked and stable. As soon as the Transceiver and the Link are both out of reset, the Transceiver sends an RXCMD to the Link.

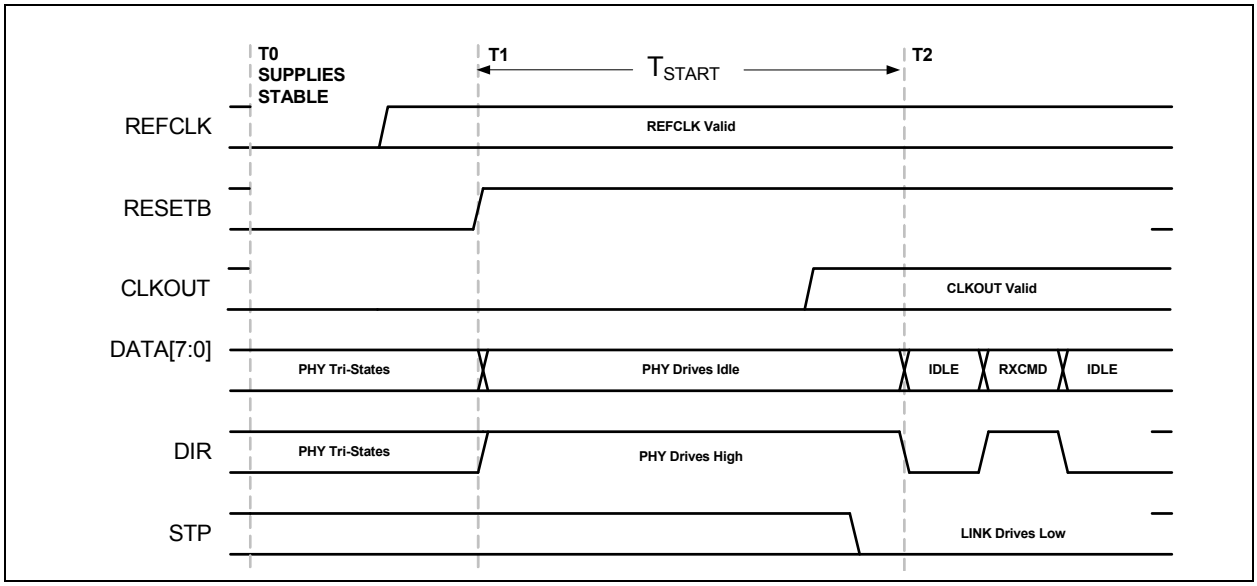

Figure 4.1 Power-On Reset Start-up Timing

4.2 Hardware Reset (RESETB)

RESETB denotes an active low hardware reset input to the USB Transceiver. Asserting RESETB will cause the 60 MHz Output Clock to stop and DIR to be driven high. Internally, the USB Transceiver will exit a hardware reset in the same way that it exits a POR, as shown in Figure 4.1. The PLL will re-lock and the ULPI registers are returned to their default states.

4.3 Software Reset

The USB Transceiver can be reset by setting bit 5 of the Function Control Register. While the USB Transceiver is in reset, DIR will be driven high. Since the 60MHz clock from the PLL stays in lock during reset, the 60 MHz Ouput Clock (CLKOUT) will continue to run.

5 Timing and Protocol

The ULPI specification defines the “Synchronous” mode, the Low Power “Asynchronous” Mode, and several other optional asynchronous modes. While SMSC USB transceivers support all asynchronous modes defined by the ULPI specification, this application note focuses on Synchronous mode. For more information about non-Syncrhonous modes, consult the product datasheet and the ULPI Specification.

The Synchronous ULPI interface references a 60MHz clock which can be sourced by the USB Transceiver or by the SoC. All ULPI transceivers must be capable of providing a 60MHz output clock and may optionally be able to accept an external 60MHz clock. All of the latest SMSC USB Transceivers support both the 60 MHz Output Clock Mode and Input Clock Mode, as defined by the ULPI Specification.

5.1 ULPI Timing

All ULPI data and control signals are referenced to the rising edge of a 60 MHz reference clock. Figure 5.1 and Figure 5.2 are taken from the ULPI Specification and define the ULPI interface timing for the USB Transceiver. It is expected that any SoC that integrates a link meets the setup and hold times defined here for the USB Transceiver.

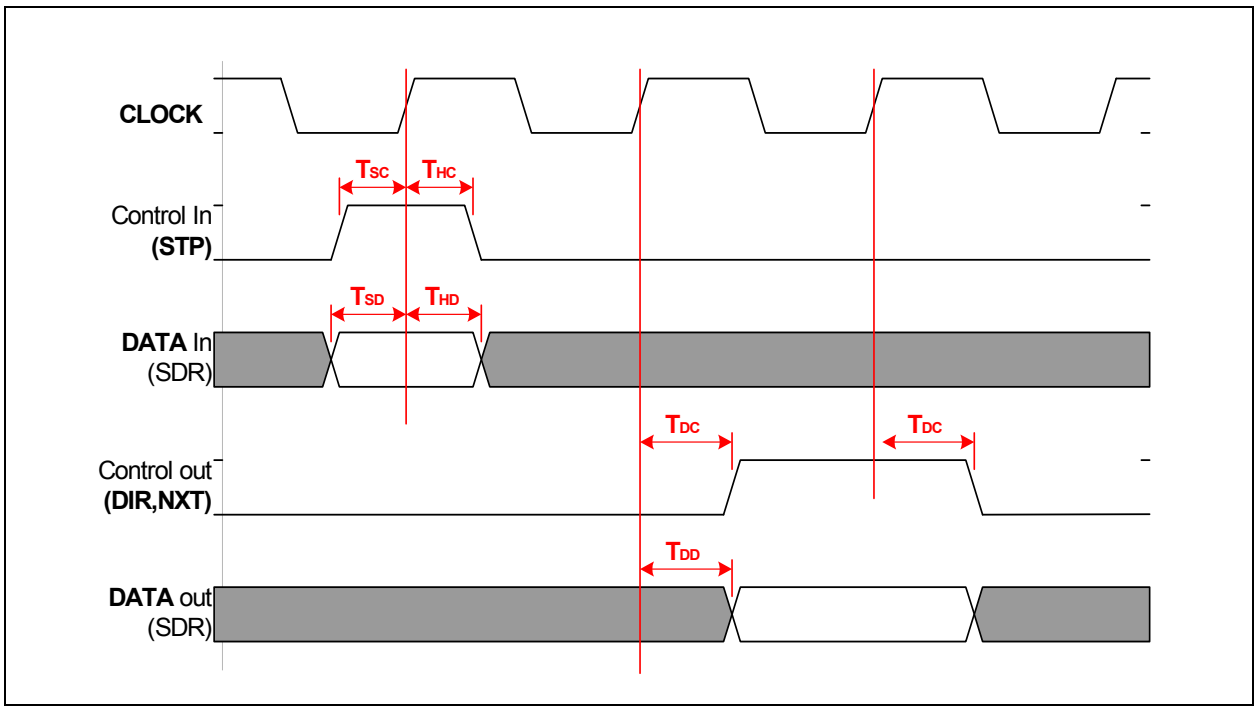

Figure 5.1 Timing Diagram for synchronous ULPI operation.

Parameter	Symbol	Min	Max	Units
Output clock				
Setup time (control in, 8-bit data in)	T_{SC}, T_{SD}		6.0	ns
Hold time (control in, 8-bit data in)	T_{HC}, T_{HD}	0.0		ns
Output delay (control out, 8-bit data out)	T_{DC}, T_{DD}		9.0	ns
Setup time (4-bit data in) (optional)	T_{SDD}		3.0	ns
Hold time (4-bit data in) (optional)	T_{HDD}	-0.8		ns
Output delay (4-bit data out) (optional)	T_{DDD}		4.0	ns
Input clock (optional)				
Setup time (control in, 8-bit data in)	T_{SC}, T_{SD}		3.0	ns
Hold time (control in, 8-bit data in)	T_{HC}, T_{HD}	1.5		ns
Output delay (control out, 8-bit data out)	T_{DC}, T_{DD}		6.0	ns
Setup time (4-bit data in)	T_{SDD}		2.5	ns
Hold time (4-bit data in)	T_{HDD}	0.8		ns
Output delay (4-bit data out)	T_{DDD}		3.5	ns

Figure 5.2 ULPI Timing Specification

5.1.1 ULPI Output Clock Mode

ULPI Output Clock Mode support is required by the ULPI specification and is the most common ULPI clock mode. In this mode, the USB Transceiver generates the 60MHz ULPI clock used by the Link, as shown in Figure 5.3.

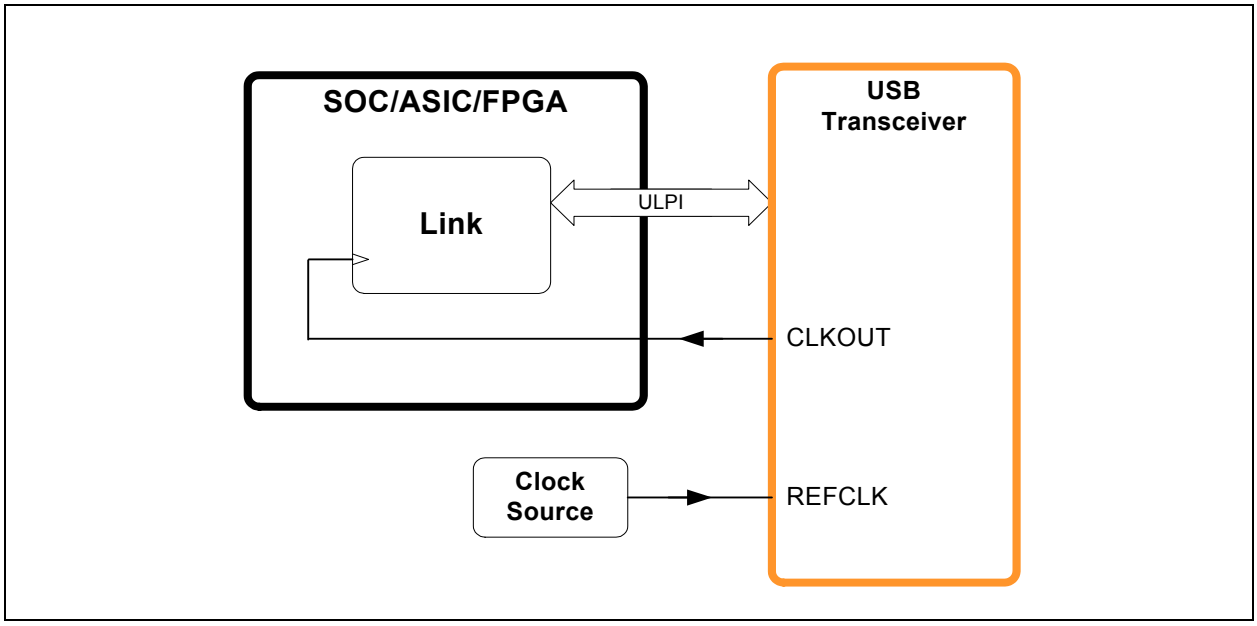

Figure 5.3 ULPI Output Clock System Diagram

5.1.2 ULPI Input Clock Mode

Most SMSC USB transceivers may also be used with a 60MHz clock source, as shown in Figure 5.4. When using ULPI Input Clock Mode, the Link must supply the 60MHz ULPI clock to the USB Transceiver.

Unlike some USB Trasnsceiver vendors who advertise support for this mode, SMSC fully meets the timing parameters required by the ULPI Specification.

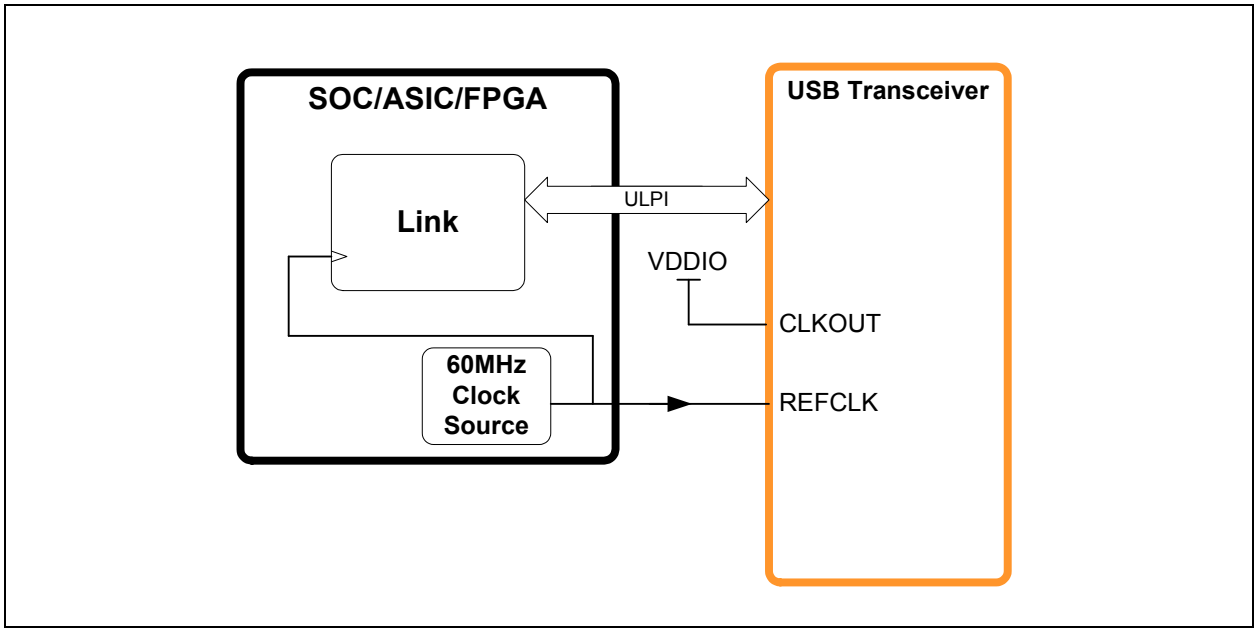

Figure 5.4 ULPI Input Clock System Diagram

5.2 ULPI Command Bytes

In addition to sending transmit and receive data across the data bus, ULPI defines Transmit Command byte that is originated by the Link and a Receive Command byte that is originated by the Transceiver.

5.2.1 Transmit Command (TX CMD)

A command from the Link begins a ULPI transfer from the Link to the USB transceiver. The TX CMD byte informs the transceiver of the type of transaction that is being initiated by the link. Table 5.1 presents all of the valid TX CMDs defined by the ULPI specification as well as the associated Packet IDs (PIDs) that are defined by the USB specification.

Table 5.1 TX CMD Encoding

TX CMD	CMD[7:6]	PAYLOAD[5:0]	HEX	COMMENTS
IDLE	00b	00_0000b	0x00	Bus idle condition
TX NO PID	01b	00_0000b	0x40	Next byte is data to transmit
TX PID OUT	01b	00_0001b	0x41	Host-to-device transaction
TX PID IN	01b	00_1001b	0x49	Device-to-host transaction
TX PID SOF	01b	00_0101b	0x45	SOF Packet Marker
TX PID SETUP	01b	00_1101b	0x4D	SETUP Token

Table 5.1 TX CMD Encoding (continued)

TX CMD	CMD[7:6]	PAYLOAD[5:0]	HEX	COMMENTS
TX PID DATA0	01b	00_0011b	0x43	Data Packet Even
TX PID DATA1	01b	00_1011b	0x4B	Data Packet Odd
TX PID ISO	01b	00_0111b	0x47	Data Packet HS Isochronous
TX PID DATA SPLIT	01b	00_1111b	0x4F	Data Packet HS Split
TX PID ACK	01b	00_0010b	0x42	ACK
TX PID NAK	01b	00_1010b	0x4A	NAK
TX PID STALL	01b	00_1110b	0x4E	STALL
TX PID NYET	01b	00_0110b	0x46	NYET
TX PID PRE	01b	00_1100b	0x4C	PRE- Send Preamble to FS hub LS device
TX PID HS SPLIT	01b	00_1000b	0x48	SPLIT - HS Split Transaction
TX PID HS PING	01b	00_0100b	0x44	PING - HS Flow Control Probe
REGISTER WRITE	10b	{reg_addr}		Immediate Register Write
REGISTER READ	11b	{reg_addr}		Immediate Register Read
REGISTER WRITE	10b	10_1111b		Extended Register Write. Address in next byte.
REGISTER READ	11b	10_1111b		Extended Register Read. Address in next byte.

5.2.1.1 Write a Register

The Link initiates a write to an immediate register in the transceiver by first sending the TX CMD byte that is described in Table 5.1. Table 5.2 provides a description of the events represented in Figure 5.5.

Table 5.2 Register Write Time Events

TIME TICK	DESCRIPTION
T1	Link drives TX CMD (Immediate Register Write).
T2	Transceiver asserts NXT to indicate that TX CMDTX CMD has been latched by the PHY.
T3	Link drives the data to be written to the register.

Table 5.2 Register Write Time Events (continued)

TIME TICK	DESCRIPTION
T4	Transceiver latches data.
T5	Transceiver latches STP. This completes the transaction.

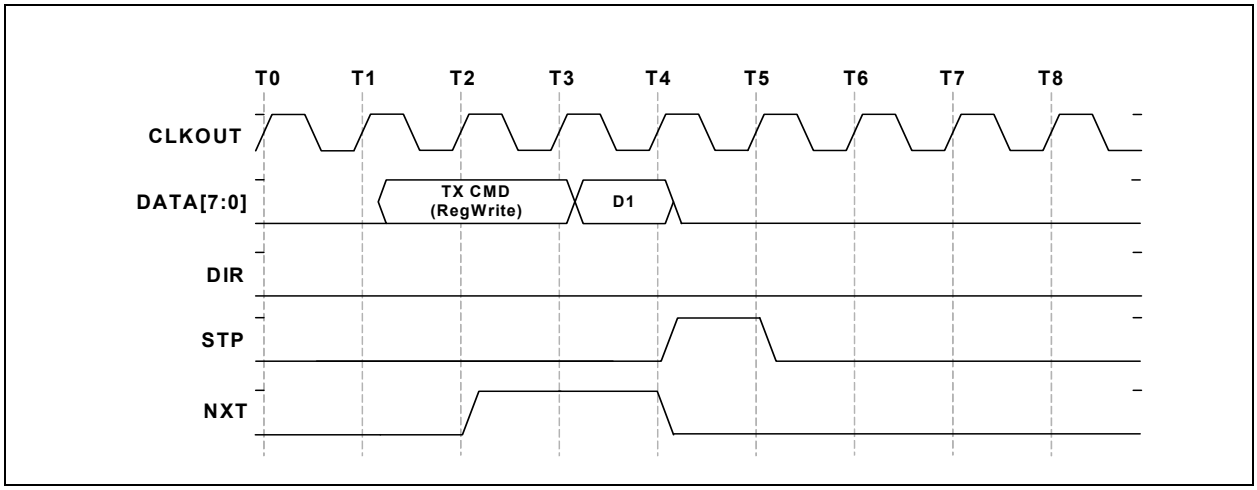

Figure 5.5 Link Writes to Immediate Register

Figure 5.6 is a logic analyzer screen shot that shows a register write operation.

0x96 = 10 010110b

CMD[7:6] = 10b (Immediate Register Write)

Payload[5:0] = 010110b (0x16h)

This transaction is an Immediate Register Write to address 0x16h (Scratch Register).

The DATA packet “D1” written to address 0x16h is value 0xAFh.

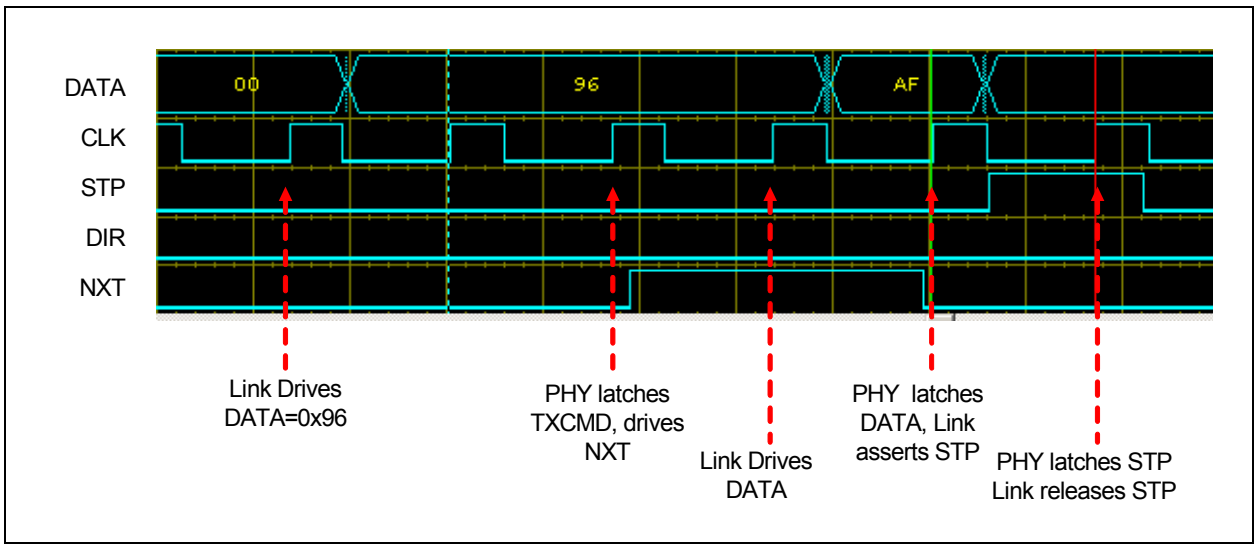

Figure 5.6 Logic Analyzer - Link Writes to Immediate Register

5.2.1.2 Reading a Register

The Link initiates reading an immediate register by first sending the TX CMD byte that is described in Table 5.1. Table 5.3 provides a description of the events represented in Figure 5.7.

Table 5.3 Register Read Time Events

TIME TICK	DESCRIPTION
T2	Link drives TX CMD (Immediate Register Read).
T3	Transceiver asserts NXT to indicate that TX CMD has been latched by the PHY.
T4	Link releases the bus and the transceiver drives DIR.
T5	Transceiver drives the requested data.
T6	Link latches the received data.

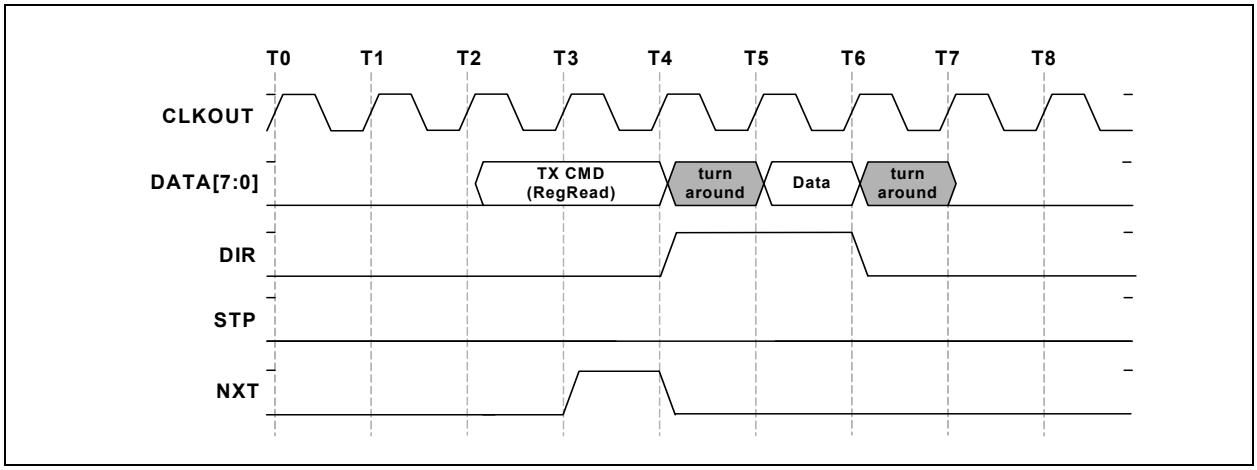

Figure 5.7 Link Reads Immediate Register

Figure 5.8 is a logic analyzer screen shot that shows a register write operation.

0xD6 = 11 010110b

CMD[7:6] = 11b (Immediate Register Read)

Payload[5:0] = 010110b (0x16h)

This transaction is an Immediate Register Read from address 0x16h (Scratch Register).

The DATA packet “Data” read from address 0x16h is value 0xBAh.

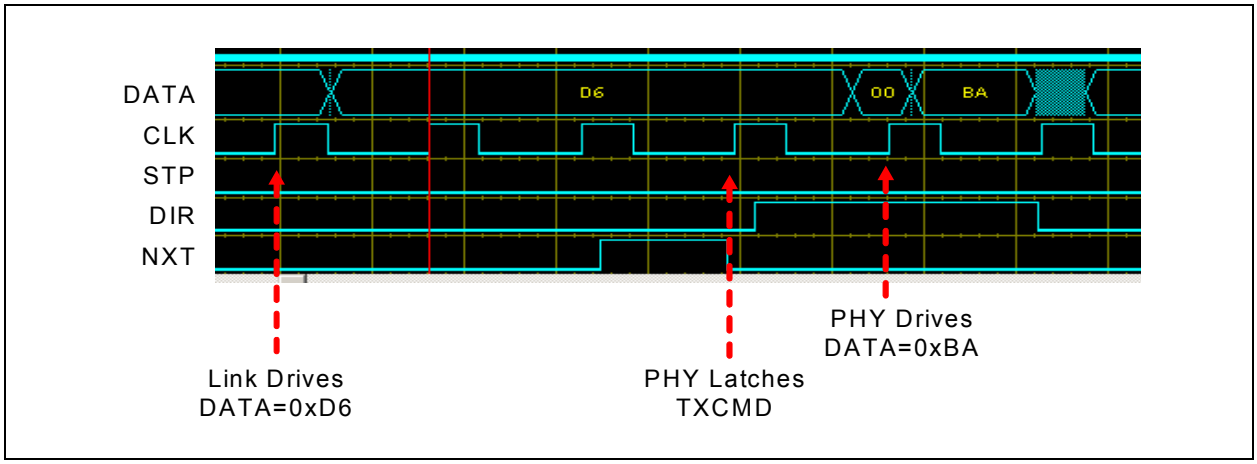

Figure 5.8 Logic Analyzer - Link Reads Immediate Register

5.2.2 Receive Command (RXCMD)

The ULPI Specification defines the Receive Command Byte (RXCMD) to send status information to the Link. The RXCMD byte is defined as shown in Figure 5.9.

Data	Name	Description				
1:0	**Linestate**	**Data[0] = Linestate[0] DP Data[1] = Linestate[1] DM**				
3:2	**Vbus State**	Value	Sess End	Sess Valid	Vbus Valid	
		00	1	0	0	
		01	0	0	0	
		10	X	1	0	
		11	X	X	1	
5:4	**RxEvent**	Value	Rxactive	RxError	Host Disc	
		00	0	0	0	
		01	1	0	0	
		11	1	1	0	
		10	X	X	1	
6	**ID**	ID==0 Host, ID==1 Device				
7	Alt Interrupt	Synchronous Interrupt Bit, Active High				

Figure 5.9 ULPI RXCMD Byte Definition

An RXCMD will be sent when one of the following conditions changes:

- USB Receive Information (LineState, RxActive, RxError)
- Interrupt Events (Hostdisconnect, Vbus, IdGnd, Carkit interrupts, RID Converter Done)

A single RXCMD and sequential (back-to-back) RXCMDs are shown in Figure 5.10. An RXCMD has a lower priority than USB receive and transmit data, but has higher priority than Register Read and Write commands.

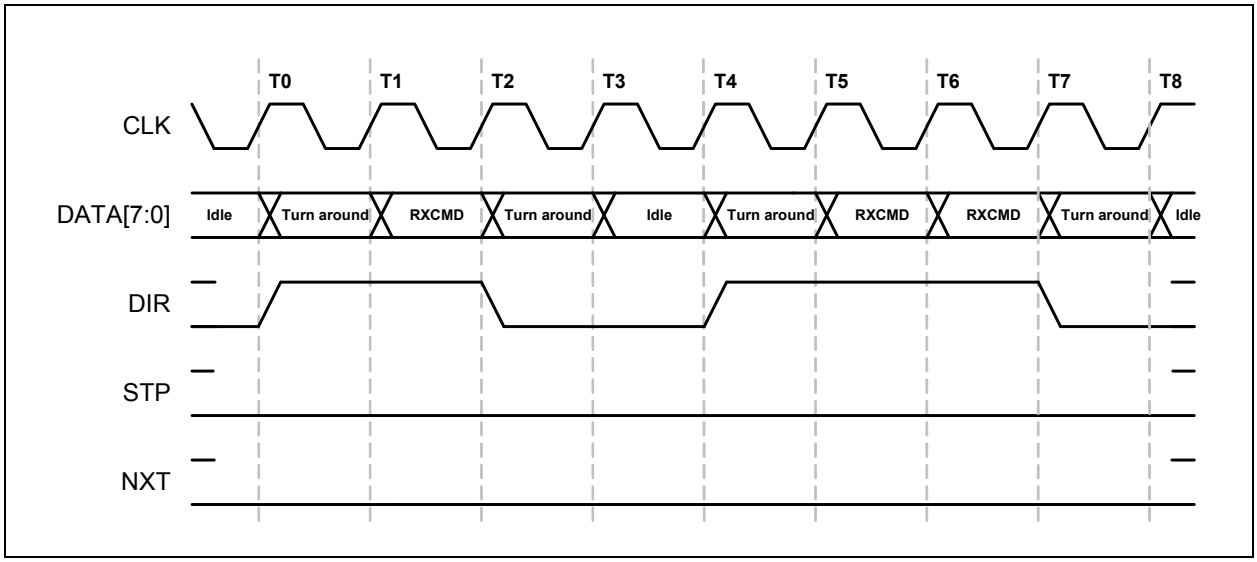

Figure 5.10 ULPI RXCMD Timing

5.3 USB Packets

Table 5.4 shows all of the valid Packet IDs (PIDs) that a USB packet can contain.

Table 5.4 PID Definitions

PID	NAME	BYTE	DESCRIPTION
TOKEN	OUT	0xE1	Host Transfer to Endpoint
	IN	0x69	Endpoint to Host Transfer
	SOF	0xA5	Start of Frame (never ACK'd)
	SETUP	0x2D	Host to function setup of control pipe. (Cannot be NAK'd.)
DATA	DATA0	0xC3	Data Packet PID Even
	DATA1	0x4B	Data Packet PID Odd
	DATA2	0x87	DATA Packet High Speed Isochronous
	MDATA	0x0F	Data Packet High Speed Split Isochronous
HANDSHAKE	ACK	0xD2	Acknowledge; Receiver has accepted error-free packet
	NAK	0x5A	Receiver cannot accept data or Transmit cannot send data
	STALL	0x1E	Endpoint has halted
	NYET	0x96	No response yet from receiver
SPECIAL	PRE	0x3C	Full speed hub pre-PID LS packet
	ERR	0x3C	Split transaction error
	SPLIT	0x78	High Speed Split Transaction
	PING	0xB4	High Speec control probe for bulk/control
	LPM	0xF0	Link Power Managment (formerly Reserved)

5.3.1 Transmit Packet

A TX CMD from the Link begins a ULPI transfer from the Link to the USB Transceiver. For a complete list of Transmit Packet IDs and how they are encoded on the ULPI bus, refer to Table 5.1.

For Transmit Packets, the TX CMD byte is followed by the data being sent, as shown in Figure 5.11.

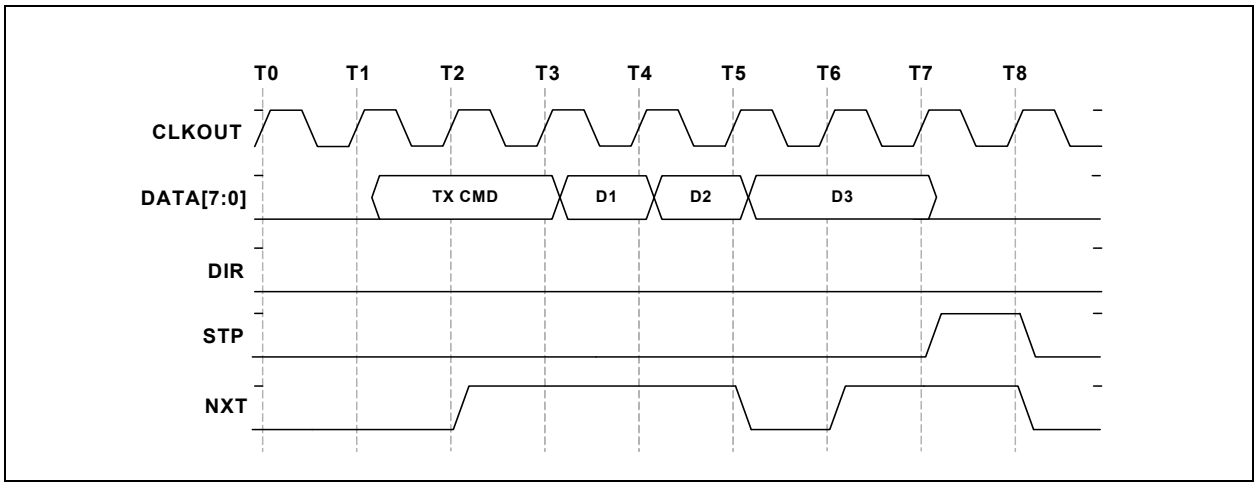

Figure 5.11 Transmitting a Packet

5.3.2 Receive Packet

The transceiver gains ownership of the data bus due to receive packet by asserting DIR when it is receiving data, as shown in Figure 5.12.

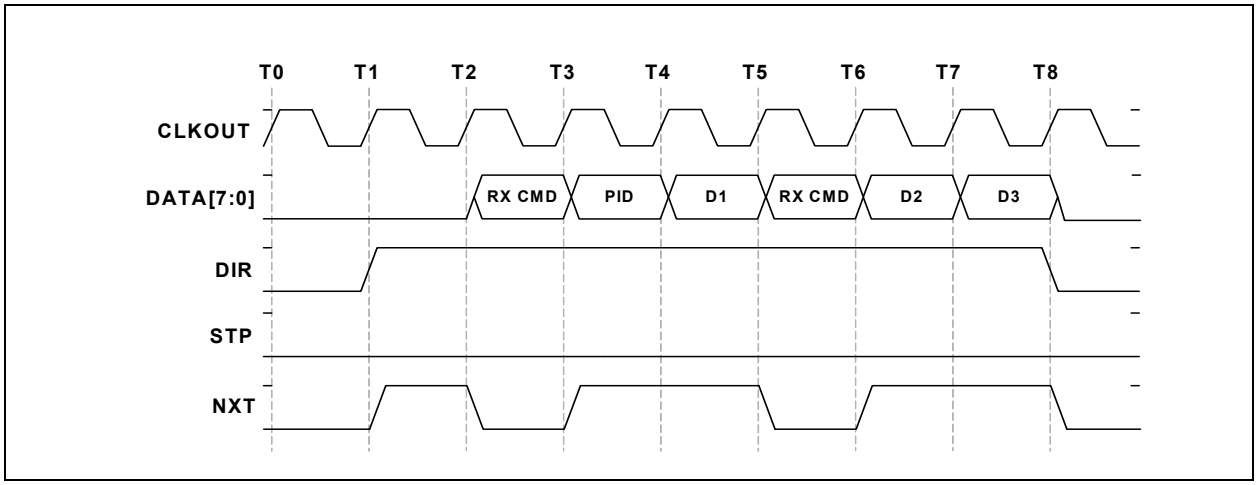

Figure 5.12 Receiving a Packet

5.4 Entering and Exiting Suspend

It is important to note that a ULPI Suspend is not the same thing as a USB Suspend. A ULPI Suspend condition is defined in the ULPI specification and provides for a low power mode for the USB Transceiver. For more information about the USB Suspend condition, refer to the USB 2.0 Specification.

To enter suspend, the SuspendM bit must be cleared in the Function Control Register, as shown in Figure 5.13.

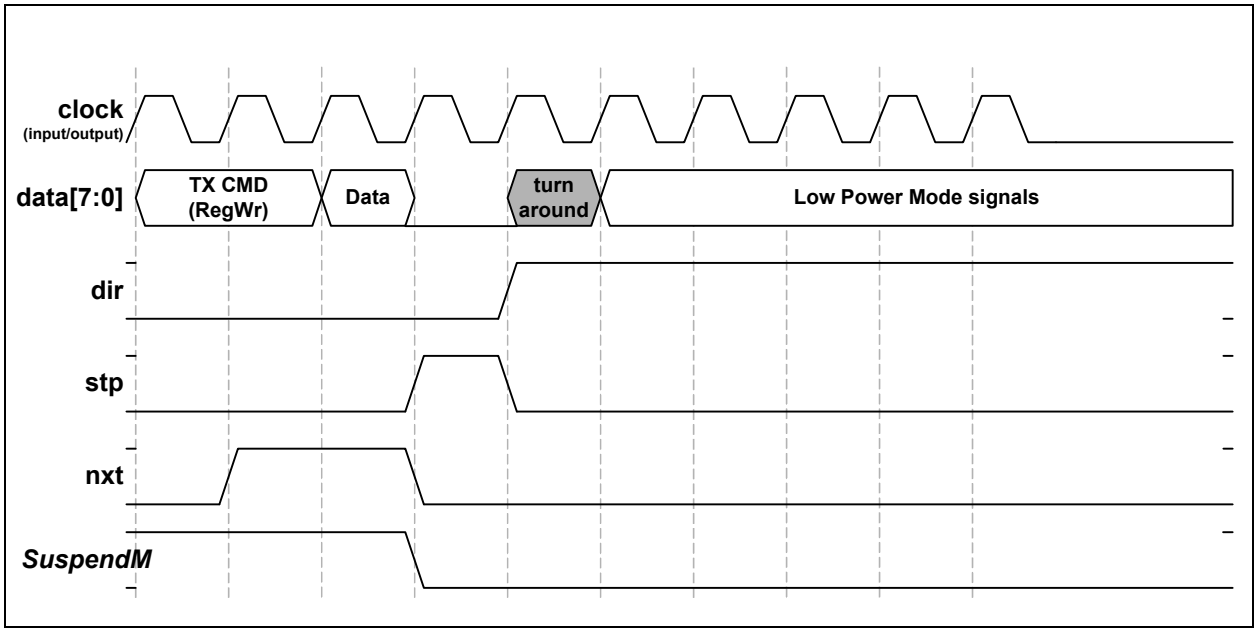

Figure 5.13 Entering Low Power Mode (Suspend)

TX CMD = 0x84 = 10 000100b

CMD[7:6] = 10b (Immediate Register Write)

Payload[5:0] = 000100b (Address 0x04h)

Data = 0x9 (SuspendM=0, Non-driving, Full Speed Transceiver enabled)

The USB Transceiver cannot wake itself from suspend. The Link must assert STP to wake up the USB Transceiver, as shown in Figure 5.14. Alternatively, the USB Transcevier may be woken up by receiving a hardware reset, as described in Section 4.2.

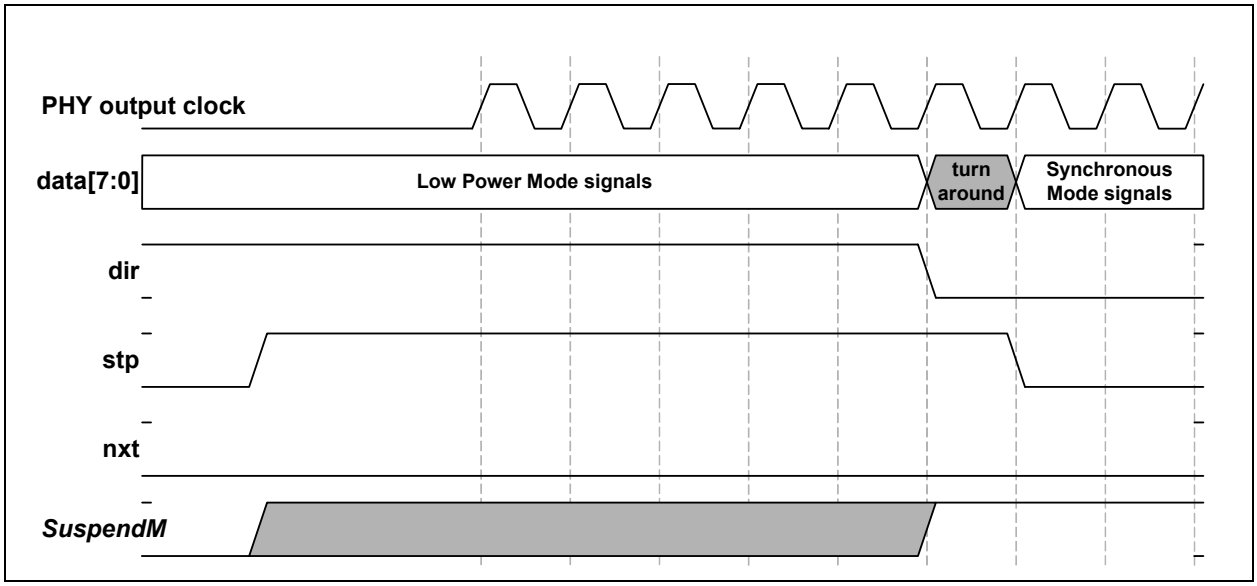

Figure 5.14 Resuming from Low Power Mode (Suspend)

6 Software Design

Because of the standards-based nature of the USB Transceiver ULPI and USB interfaces, a common driver may be used for any USB Transceiver compliant with the ULPI specification. This is commonly delivered with the SoC Board Support Package (BSP) made available from your SoC vendor.

The registers used for USB operation are detailed in the ULPI specification. The software drivers should be written to use only the standard registers. This makes it possible to maintain software compatibility with ULPI transceivers from various manufacturers. The SMSC transceiver can be designed in with little or no change required to existing software.

“Vendor-specific” registers are provided in the ULPI specification for features unique to the vendor or product. For instance, SMSC USB Transceivers offer Transmit Boost and Receiver Sensitivity Enahancment features that are configured through these vendor-specific registers.

7 ULPI Communications Issues

If problems are encountered during ULPI communication, this section offers suggestions on how to diagnose common problems.

Table 7.1 Common Symptoms and Issues

SIGNAL(S) TO MEASURE (RECOMMENDED TEST EQUIPEMENT)	EXPECTED	IF UNEXPECTED RESULT, POSSIBLE ISSUES INCLUDE:
CLKOUT (Oscilloscope)	60 MHz	1. REFCLK not oscillating at correct frequency. 2. RESETB held low. 3. VDD18/VDDIO not powered. 4. SuspendM bit cleared (HY in Suspend).
VDD18, VDDIO (Volt Meter, Oscilloscope)	VDD18 = 1.8V (Typ) VDDIO = 1.8V - 3.3V (Typ)	PHY digital core, ULPI interface not powered.
RESETB (Oscilloscope)	RESETB = 1.6V - 3.3V for synchronous ULPI function.	1. RESETB pin held low by logic external to PHY. 2. RESETB pin floating.
Start-up: ULPI Bus at falling edge of STP (Logic Analyzer)	STP held high until link comes out of reset. (See Figure 4.1) Link must not drive anything other than Idle (0x00h) prior to completing reset and driving STP low.	Link not operating per ULPI specification during start-up/reset.
Start-up: ULPI Bus when CLKOUT begins oscillating (Logic Analyzer)	CLKOUT starts oscillating, DIR transitions from high to low. (See Figure 4.1)	1. RESETB being held low. 2. VDD18/VDDIO not powered.
Start-up: ULPI Bus at Falling edge of DIR (Logic Analyzer	After PHY exits reset, and RXCMD should be sent from the PHY to the link. (See Figure 4.1).	1. RXCMD not sent. Verify RESETB=1, CLKOUT is active, and DIR. 2. RXCMD sent. Verify DATA[6] = 1 (B Device) or = 1 (A Device / Host). 3. Verify DATA[3:2] reflects the Vbus voltage state.
Start-up: ULPI Bus at Falling edge of DIR (Logic Analyzer	RXCMD following PHY POR is present, but the value of DATA[6] does not = 1 (Device) or = 0 (Host).	1. (Device) ID Pin pulled down or grounded. 2. (Host) ID pin floating or pulled up.
Start-up: ULPI Bus at Falling edge of DIR (Logic Analyzer VBUS at PHY pin (Oscilloscope)	RXCMD following PHY POR is present, but the value of DATA[3:2] does not reflect correct voltage state of VBUS.	1. USB cable not connected. 2. (Host) VBUS resistor (RVBUS) too large. Refer to specific product datasheet for details. 3. VbusValid/SessEnd comparators disabled.

8 Application Note Revision History

Table 8.1 Customer Revision History

REVISION LEVEL & DATE	SECTION/FIGURE/ENTRY	CORRECTION
Rev. 1.0 (11-26-12)	Document co-branded: Microchip logo added; modification to legal disclaimer.	
Rev. 1.0 (05-07-09)	Initial Release	