

---

# SmartFusion cSoC: USB Interface Using Fabric

---

## Table of Contents

---

Introduction . . . . .	1
Design Example Overview . . . . .	1
Appendix A – Design Files . . . . .	9
Conclusion . . . . .	9
List of Changes . . . . .	10

---

## Introduction

The SmartFusion<sup>®</sup> customizable system-on-chip (cSoC) contains a hard embedded microcontroller subsystem (MSS), programmable analog circuitry, and FPGA fabric, consisting of logic tiles, static random access memory (SRAM), and phase-locked loops (PLLs). The MSS consists of a 100 MHz ARM<sup>®</sup> Cortex<sup>™</sup>-M3 processor, advanced high-performance bus (AHB) matrix, system registers, Ethernet MAC, DMA engine, real-time counter (RTC), embedded nonvolatile memory (eNVM), embedded SRAM (eSRAM), fabric interface controller (FIC), the Philips Inter-Integrated Circuit (I<sup>2</sup>C), serial peripheral interface (SPI), and external memory controller (EMC).

This application note describes the capability of SmartFusion cSoC devices to interface with the USB host devices using the FPGA fabric. This design example can be used as a reference when you need to interface the SmartFusion cSoC device with a USB host device (usually a PC). This design example demonstrates the USB device functionality on SmartFusion cSoC devices using a USB device chipset from Future Technology Devices International Ltd (FTDI) and Cypress Semiconductor Corp.

This application note explains how to interface a SmartFusion cSoC device with the FT2232H Mini-Module from FTDI and with USBee EX 2.0 Experimenter's board from USBee (Cypress USB device chipset). A basic understanding of the SmartFusion design flow is assumed. Refer to [Using UART with a SmartFusion cSoC - Libero SoC and SoftConsole Flow Tutorial](#) to understand the SmartFusion design flow.

## Design Example Overview

This design example demonstrates the capability of the SmartFusion cSoC devices to interface with the USB host devices using the FPGA fabric on the A2F500 SmartFusion Development Kit Board. This design example explains the two different solutions for interfacing USB devices with SmartFusion cSoC devices as mentioned below:

1. Solution 1: Interfacing FT2232H Mini-Module from FTDI
2. Solution 2: Interfacing USBee EX 2.0 Experimenter's board from USBee (Cypress USB device chipset)

The FT2232H Mini-Module is interfaced with the SmartFusion cSoC device using SPI mode and has achieved 40 Mbps throughput. The USBee EX 2.0 Experimenter's board is interfaced with the SmartFusion cSoC device using fabric general purpose input/output (GPIOs) and has achieved 96 Mbps throughput. [Figure 1 on page 2](#) shows a typical USB device interface with the USB host (laptop).

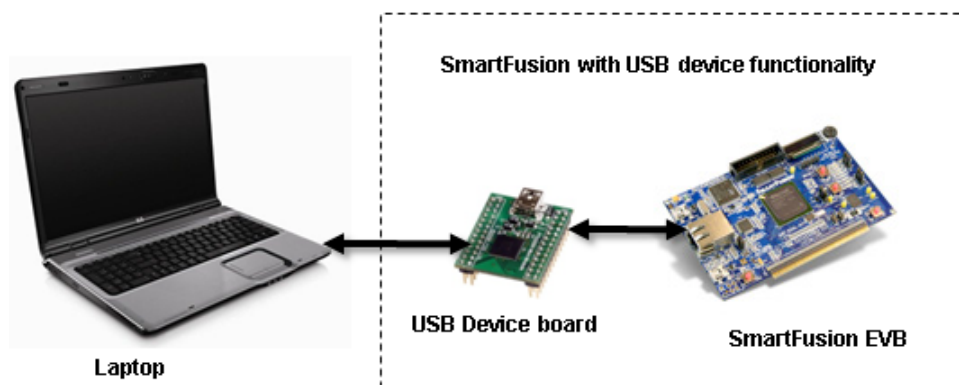


Figure 1 • Typical USB Device Interface

### Solution 1: Using the FT2232H Mini-Module

The FT2232H is a USB 2.0 high speed (480 Mbps) to UART/multi-protocol synchronous serial engine (MPSSE) IC. The device features two interfaces that can be configured for asynchronous or synchronous serial or parallel First in, First out (FIFO) interfaces. The two channels can also be independently configured to use an MPSSE engine. This allows the two ports of the FT2232H to operate independently as UART/Bit-Bang ports or MPSSE engines used to emulate JTAG, SPI, I<sup>2</sup>C, Bit-Bang, or other synchronous serial modes. Refer to the *FT2232H Mini-Module* datasheet that is available at [www.ftdichip.com](http://www.ftdichip.com) for more information.

In this design example, the FT2232H Mini-Module is configured in SPI mode of the MPSSE engine using the host application (refer to "[Appendix A – Design Files](#)" on page 9) and interfaced with the SPI interface on the SmartFusion cSoC device. The host application (the FTDI USB driver, the D2XX driver, and the software) is used to access the FT2232H on the Mini-Module through a DLL. Refer to the *Interfacing FT2232H Hi-Speed Devices to SPI Bus* application note and programming examples using the D2XX drivers and DLL available at [www.ftdichip.com](http://www.ftdichip.com) to understand how to interface and configure the FT2232H Mini-Module to read and write the data from a host PC. [Figure 2 on page 3](#) shows the FT2232H Mini-Module interface with the SmartFusion cSoC.

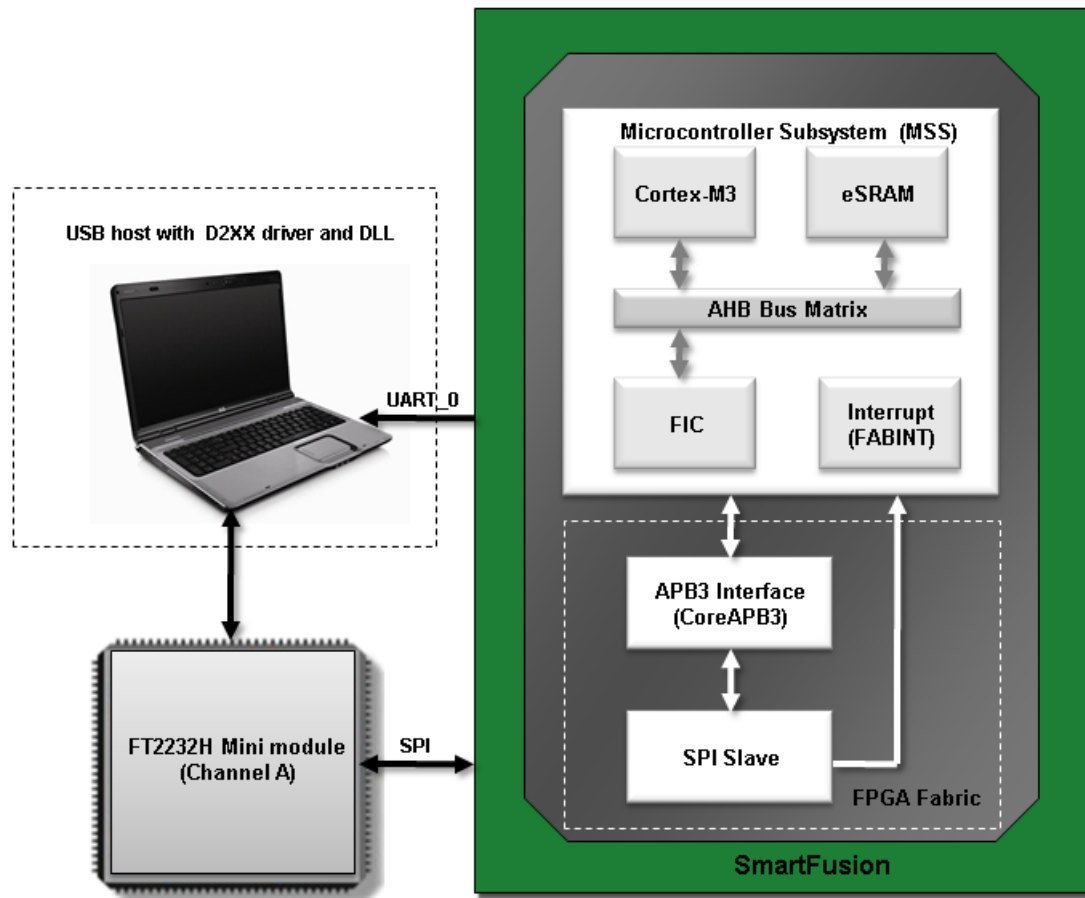


Figure 2 • FT2232H Mini-Module Interface with SmartFusion and USB Host (Laptop)

The host application reads the data from a file and sends in chunks of 255 bytes to the FT2232H Mini-Module. The SPI slave in the SmartFusion fabric collects the 32-bit data from the FT2232H Mini-Module and generates an interrupt to the MSS. Then the MSS reads the 32-bit data and stores in the eSRAM. Once the MSS has stored the complete data in the eSRAM, it sends back all the data to the host PC via *UART\_0* in the MSS. The received data is stored in the host application folder with a file name of *read\_file.txt* to check the data integrity. The maximum data rate measured at FT2232H Mini-Module and the SmartFusion interface using the Logic Analyzer is 40 Mbps.

### SmartDesign Implementation

The design example consists of the MSS and CoreAPB3 IP and SPI slave block with a custom-made APB interface in the FPGA fabric. CoreAPB3 connects the SPI slave with the FIC in the MSS. The SPI slave logic in the fabric collects the 32-bit data from the FT2232H Mini-Module and generates an interrupt. This interrupt signal is connected to a dedicated fabric interrupt, FABINT, available in the MSS.

The MSS is configured with FIC, clock conditioning circuit (CCC), GPIOs, and a UART. The CCC generates an 80 MHz clock and acts as the clock source. The FIC is configured to use the master interface with interface types of advanced microcontroller bus architecture (AMBA) and advanced peripheral bus (APB3). The GPIOs in the MSS are configured as input and used to handle the flow control in the data transfer from the SPI slave to the MSS. The *UART\_0* is configured for data transfer from eSRAM to the host PC.

CoreAPB3 acts as a bridge between the MSS and SPI block. It provides AMBA3 APB3 fabric supporting up to 16 APB slaves. This design example uses one slave slot (Slot 0) to interface with the SPI slave block and is configured with the direct addressing mode. Refer to the [CoreAPB3 Handbook](#) for more information on CoreAPB3 IP.

Figure 3 illustrates the SmartDesign of the USB interface design example.

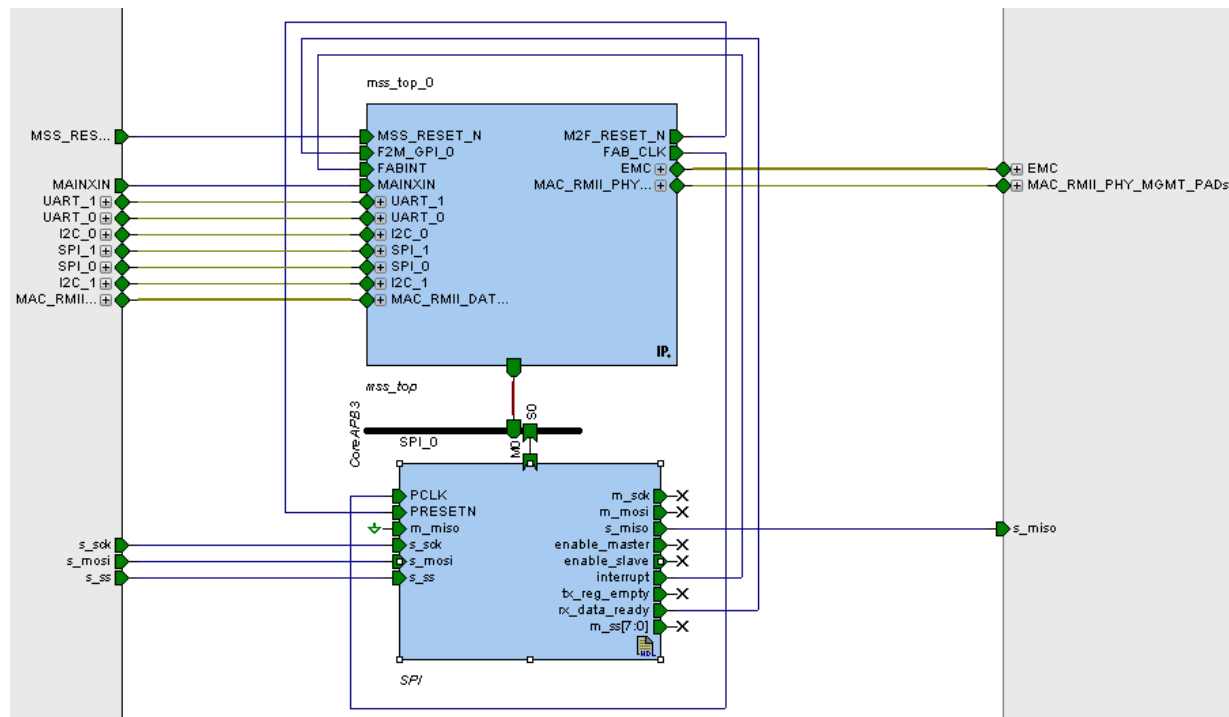


Figure 3 • SmartDesign of USB Interface Example Design (FT2232H Mini-Module)

The Libero® system-on-chip (SoC) software project is provided in the design files for this design example (refer to "Appendix A – Design Files" on page 9).

## Running the Design

Install the appropriate drivers (D2XX drivers for FT2232 chip), which can be downloaded from the [www.ftdichip.com](http://www.ftdichip.com) website. Refer to the FT2232H datasheet and application notes that are available at [www.ftdichip.com](http://www.ftdichip.com) for more information on the driver installation, electrical connections and connector pin numbers.

Connect the FT2232H Mini-Module to the A2F500 SmartFusion Development Kit Board using FPGA fabric I/Os. The I/Os are available on the mixed signal header or on the Direct C header (J22 on the Evaluation Kit Board and J13 on the Development Kit Board). Refer to the [SmartFusion Development Kit User's Guide](#) and the [SmartFusion Evaluation Kit User's Guide](#) for more information on the development and evaluation kit boards. Table 1 shows the pin mapping between the FT2232H Mini-Module and the Direct C header (J13) on the A2F500 SmartFusion Development Kit Board.

Table 1 • Connection Details of FT2232H Mini-Module to SmartFusion

S. No	FT2232H Mini-Module Pin	SmartFusion Development Kit Board Direct C Header Pins
1	SK (ADBUSH0, pin 7 of CN2)	1 (FPGA I/O J19)
2	DO (ADBUSH1, pin 10 of CN2)	5 (FPGA I/O J21)
3	DI (ADBUSH2, pin 9 of CN2)	7 (FPGA I/O J22)
4	CS (ADBUSH3, pin 12 of CN2)	3 (FPGA I/O J20)

Connect Pin 2 of CN2 and Pin 2 of CN3 to any ground point of the A2F500 Development Kit Board. The FT2232H Mini-Module allows configuration with both USB bus-powered and USB self-powered designs. Ensure that the FT2232H is configured in the USB bus-powered only. These connections can be referred from the datasheet of the FT2232H mini module.

Figure 4 shows the interface of the FT2232H Mini-Module with the Direct C header on the SmartFusion Development Kit Board.

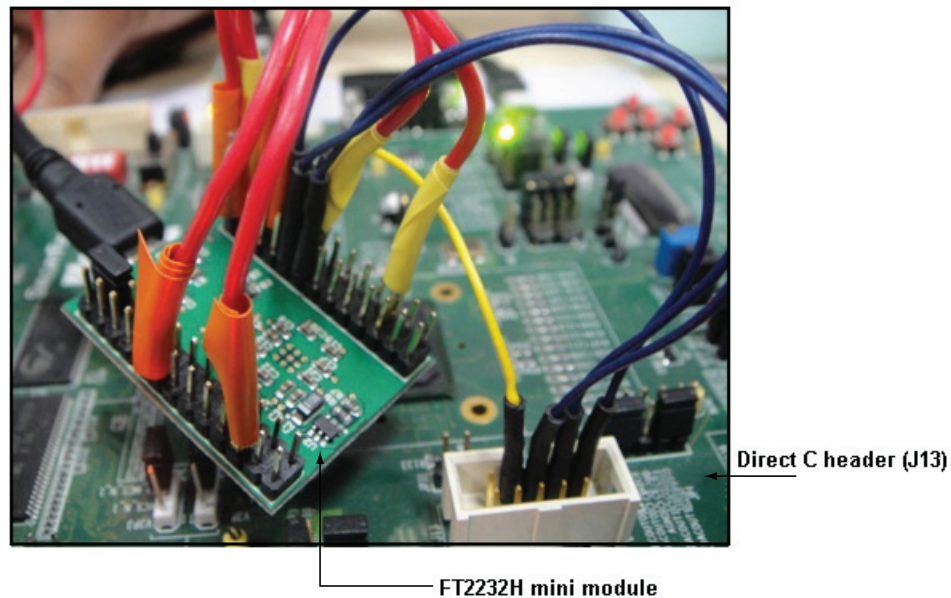


Figure 4 • FT2232H Mini-Module with Direct C Header on SmartFusion Development Kit Board

Program the A2F500 Development Kit Board with the generated or provided \*.STP file (refer to "Appendix A – Design Files" on page 9) using FlashPro and then power cycle the board.

Invoke the SoftConsole IDE, double-click **Write Application Code** under **Develop Firmware** in the Libero SoC design flow window (refer to "Appendix A – Design Files" on page 9) and launch the debugger. Then run the host application on the host PC.

## Solution 2: Using USBee EX 2.0 Experimenter's Board

The USBee EX 2.0 Experimenter's Board is a USB to GPIO converter based on the Cypress chipset. It has a mini-B USB connector on one side and signal pin header on the other side that consists of an 8-bit data bus, a read/write signal, and a clock signal. Using the libraries and source code provided, you can do byte-wide reads and writes to these signals. The USBee EX 2.0 board operates at USB 2.0 High Speed (480 Mbps) or Full-Speed (12 Mbps) and supports the two modes of data transfers: Bidirectional mode and High-speed mode. Refer to the *USBee EX 2.0 Tool Builder Project and Users Guide* available at [www.usbee.com](http://www.usbee.com) for more information.

In this design example, the USBee EX 2.0 Experimenter's Board is connected to the A2F500 SmartFusion Development Kit Board using the mixed signal header on the board. Figure 5 shows the USBee EX 2.0 Experimenter's Board interface with a SmartFusion cSoC device.

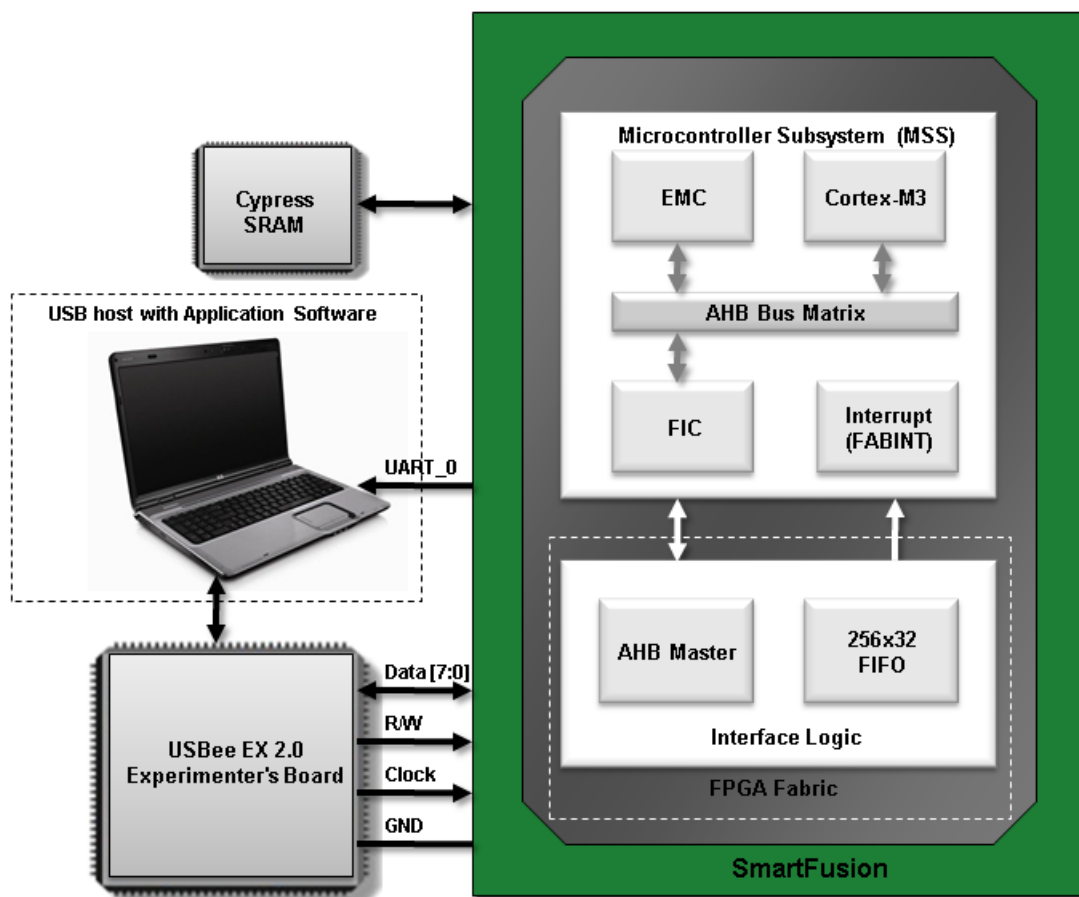


Figure 5 • USBee EX 2.0 Experimenter's Board Interface with SmartFusion and USB Host (Laptop)

The USBee EX 2.0 Experimenter's Board is interfaced with the FPGA fabric of a SmartFusion cSoC device using an 8-bit data bus, a read/write (R/W) signal, and a clock signal available on the signal pin header. The FPGA fabric has interface logic to communicate with the USBee EX 2.0 module. The interface logic consists of an AHB master interface and a 256x32 FIFO. The host application (refer to "Appendix A – Design Files" on page 9) reads the data from a file and sends the data to the USBee EX 2.0 module.

The interface logic in the FPGA fabric receives the data or transmits the data based on the R/W signal. When the R/W signal is asserted for the write transaction, the interface logic receives the 8-bit data and forms it into 32-bit, then stores it in FIFO. The interface logic reads back the 32-bit data from FIFO and sends it to the external SRAM. Once the host completes the data transfer and the MSS stores the complete data in the external SRAM, the interface logic generates an Interrupt to the MSS. The MSS then reads the complete data from external SRAM and sends back to the host PC via *UART\_0*. The received data is stored in the host application folder with a file name of *read\_file.txt* to check the data integrity. The maximum data rate measured at the USBee EX 2.0 Experimenter's Board and SmartFusion interface using Logic Analyzer is 96 Mbps.

## SmartDesign Implementation

The design example consists of MSS and interface logic with the AHB master in the FPGA fabric. The AHB master interfaces the interface logic with the FIC in the MSS. The MSS is configured with EMC, FABINT, FIC, CCC, and a UART. The CCC generates an 80 MHz clock and acts as a clock source. The FIC is configured to use the slave interface with the interface type as AHB Lite. The *UART\_0* is configured for data transfer from the external SRAM to the host PC. Figure 6 illustrates the SmartDesign of the USBee EX 2.0 module interface design example.

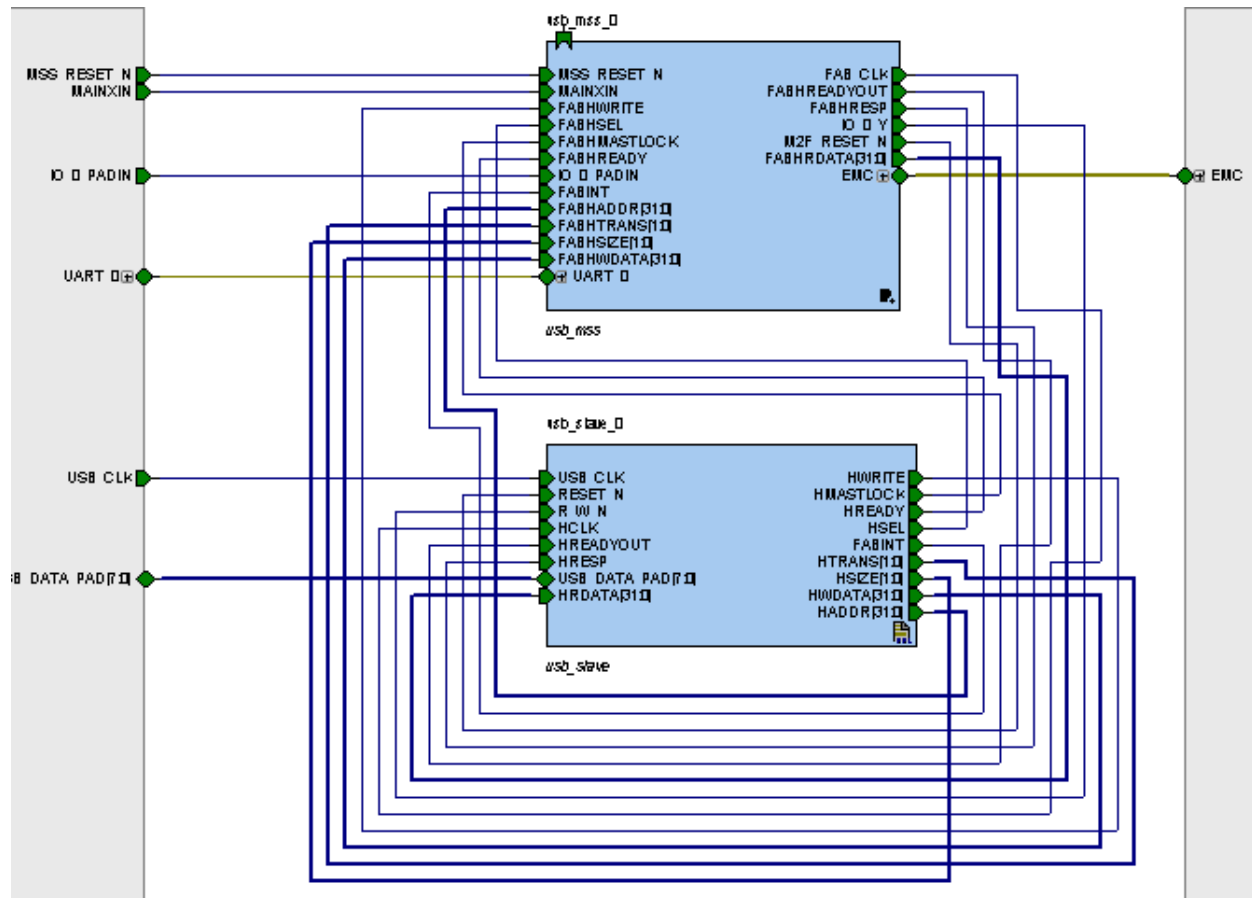


Figure 6 • SmartDesign of USBee EX 2.0 Module Interface Example Design

The Libero SoC project is provided in the design files for this design example (refer to "Appendix A – Design Files" on page 9).

## Running the Design

Install the appropriate drivers for USBee EX 2.0 Experimenter's Board, which can be downloaded from [www.usbee.com](http://www.usbee.com) website. Connect the USBee EX 2.0 Experimenter's Board to the SmartFusion Development Kit Board using the FPGA fabric I/Os. Figure 8 on page 9 shows the connector pin details of the USBee Ex 2.0 Experimenter's Board.

The I/Os are available on the mixed signal header (J21). Refer to the *SmartFusion Development Kit User's Guide* for more information on the development kit boards. Table 2 on page 8 shows the pin mapping between the USBee EX 2.0 Experimenter's Board and the mixed signal header (J21) on the SmartFusion Development Kit Board.



Table 2 • SmartDesign of USBee EX 2.0 Module Interface Example Design

S. No.	USBee EX 2.0 Experimenter's Board Pin	SmartFusion Development Kit Mixed Signal Header Pins
1	CLK	35 (FPGA I/O B22)
2	R/W#	7 (FPGA I/O V1)
3	DATA 0	29 (FPGA I/O F3)
4	DATA 1	30 (FPGA I/O G4)
5	DATA 2	32 (FPGA I/O H5)
6	DATA 3	33 (FPGA I/O H6)
7	DATA 4	34 (FPGA I/O J6)
8	DATA 5	27 (FPGA I/O E3)
9	DATA 6	37 (FPGA I/O C22)
10	DATA 7	38 (FPGA I/O F1)
11	Ground (GND)	To any ground point on the board

The complete USBee EX 2.0 Tool builder source code for Visual C and Visual Basic 6.0 projects for controlling the High-Speed USBee EX 2.0 Experimenter's board is available at [www.usbee.com](http://www.usbee.com). Figure 7 shows the interface of the USBee EX 2.0 Experimenter's board with the mixed signal header (J21) on the SmartFusion Development Kit Board.

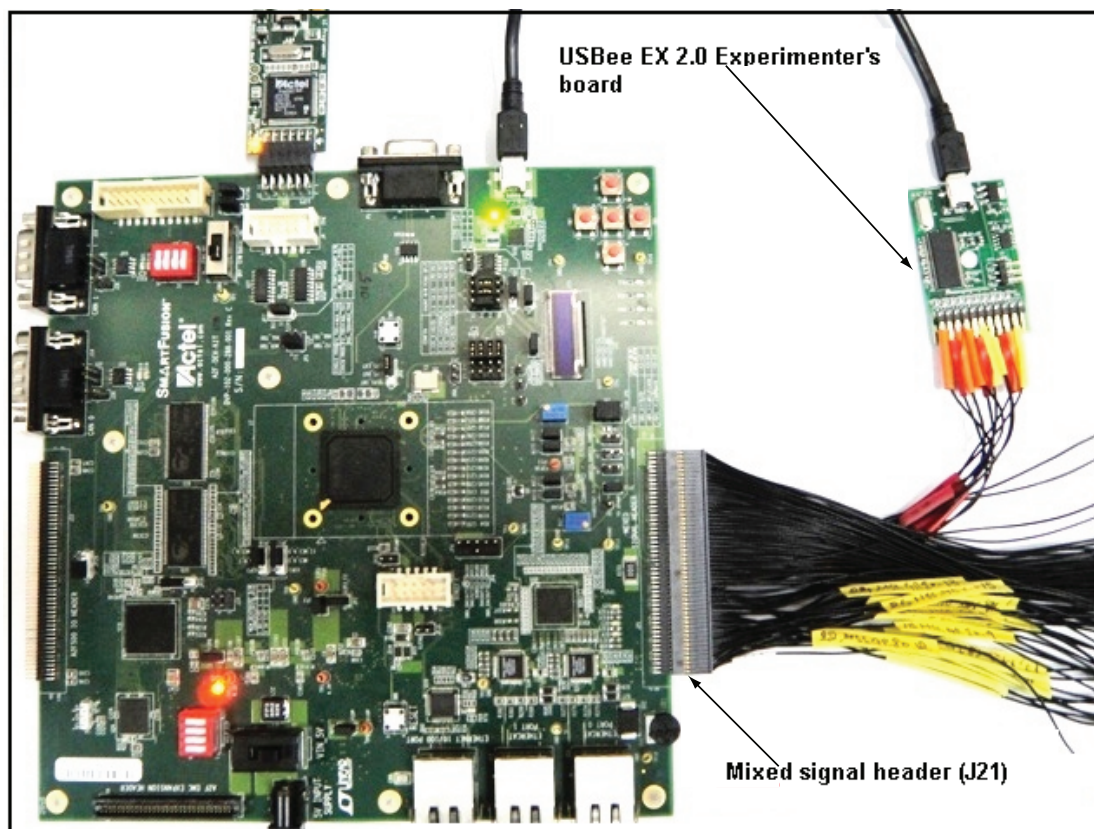


Figure 7 • USBee EX 2.0 Experimenter's Board with Mixed Signal Header (J21) on SmartFusion Development Kit Board



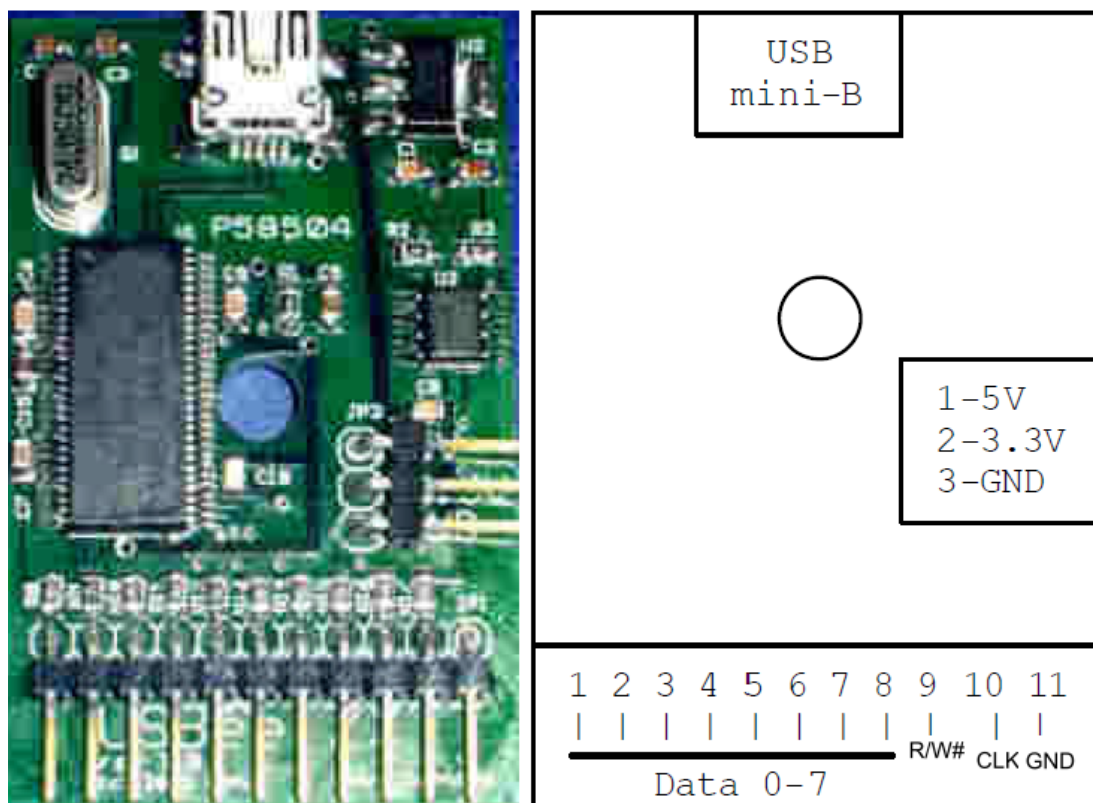


Figure 8 • Connector Pin Details of the USBee Ex 2.0 Experimenter's Board

Program the SmartFusion Evaluation Kit Board or the Development Kit Board with the generated or provided \*.STP file (refer to "Appendix A – Design Files") using FlashPro and then power cycle the board.

Invoke SoftConsole IDE, double-click **Write Application Code** under **Develop Firmware** in the Libero SoC design flow window (refer to "Appendix A – Design Files"), and launch the debugger. Then run the host application on the host PC.

## Appendix A – Design Files

You can download the design files from the SoC Products group website:

[www.microsemi.com/soc/download/rsc/?f=A2F\\_AC356\\_DF](http://www.microsemi.com/soc/download/rsc/?f=A2F_AC356_DF).

The design file consists of a Libero SoC project, SoftConsole software project, and the host application. Refer to the Readme.txt file included in the design file for directory structure and description.

## Conclusion

This application note describes the capability of the SmartFusion cSoC devices to interface with the USB host devices using the FPGA fabric on the SmartFusion Evaluation Kit Board and SmartFusion Development Kit Board with two different solutions:

Solution 1: Interfacing the FT2232H Mini-Module from FTDI

Solution 2: Interfacing the USBee EX 2.0 Experimenter's board from the USBee (Cypress USB device chipset)

The FT2232H Mini-Module is interfaced with a SmartFusion cSoC device using SPI mode and has achieved 40 Mbps throughput. The USBee EX 2.0 Experimenter's Board is interfaced with a SmartFusion cSoC device using the fabric GPIOs and has achieved 96 Mbps throughput.

## List of Changes

The following table lists critical changes that were made in each revision of the document.

Revision*	Changes	Page
Revision 2 (February 2012)	Modified the section " <a href="#">Design Example Overview</a> " (SAR 36680).	1
	Added <a href="#">Figure 8</a> (SAR 36680).	9
	Removed ".zip" extension in the link (SAR 36763).	9
Revision 1 (January 2012)	Modified the section " <a href="#">Design Example Overview</a> " (SAR 36052).	1

*Note: \*The revision number is located in the part number after the hyphen. The part number is displayed at the bottom of the last page of the document. The digits following the slash indicate the month and year of publication.*



**Microsemi Corporate Headquarters**  
One Enterprise, Aliso Viejo CA 92656 USA  
Within the USA: +1 (949) 380-6100  
Sales: +1 (949) 380-6136  
Fax: +1 (949) 215-4996

Microsemi Corporation (NASDAQ: MSCC) offers a comprehensive portfolio of semiconductor solutions for: aerospace, defense and security; enterprise and communications; and industrial and alternative energy markets. Products include high-performance, high-reliability analog and RF devices, mixed signal and RF integrated circuits, customizable SoCs, FPGAs, and complete subsystems. Microsemi is headquartered in Aliso Viejo, Calif. Learn more at [www.microsemi.com](http://www.microsemi.com).

© 2012 Microsemi Corporation. All rights reserved. Microsemi and the Microsemi logo are trademarks of Microsemi Corporation. All other trademarks and service marks are the property of their respective owners.