

### **Table of Contents**

Introduction																												
Design Description												 															. 2	
Conclusion												 															12	
Appendix A – Design Files																												
List of Changes	• •	·	•	 ·	·	•	•	 •	·	 •	·	 •	•	•	 ·	·	• •	·	•	•	·	•	•	 •	·	•	12	

### Introduction

The SmartFusion<sup>®</sup> customizable system-on-chip (cSoC) FPGA devices integrate FPGA technology with hardened ARM<sup>®</sup> Cortex<sup>TM</sup>-M3 processor based microcontroller subsystem (MSS) and programmable high-performance analog blocks built on a low power flash semiconductor process. The MSS consists of hardened blocks such as 100MHz ARM Cortex-M3 processor, peripheral DMA (PDMA), embedded nonvolatile memory (eNVM), embedded SRAM (eSRAM), embedded FlashROM (eFROM), external memory controller (EMC), watchdog timer, the Philips Inter-Integrated Circuit (I<sup>2</sup>C), serial peripheral interface (SPI), 10/100 Ethernet controller, real-time counter (RTC), GPIO block, fabric interface controller (FIC), in-application programming (IAP) and system registers. The programmable analog block contains analog computing engine (ACE) and analog front-end (AFE) consisting of ADCs, DACs, active bipolar prescalers (ABPS), comparators, current monitors and temperature monitors.

This application note describes how to use the FIC to maximize data transfer between FPGA fabric logic and MSS peripherals. The FIC connects the FPGA fabric logic to MSS. It performs an AHB to AHB or AHB to APB bridging functions between the AHB bus matrix and AHB or APB bus, in the FPGA fabric. The FIC is hard block and hence consumes no logic in the FPGA fabric. It provides two bus interfaces between MSS and FPGA fabric. The first is mastered by MSS and has slaves in the FPGA fabric and the second has a master in the FPGA fabric and slaves in the MSS.

Refer to the *SmartFusion Microcontroller Subsystem User's Guide* for more details on fabric interface controller.



### **Overview**

This application note explains how to use FIC to connect the FPGA fabric logic to the MSS peripherals and how to maximize the data transfer between MSS peripherals and FPGA fabric logic. Figure 1 shows the interface between MSS and FPGA fabric.

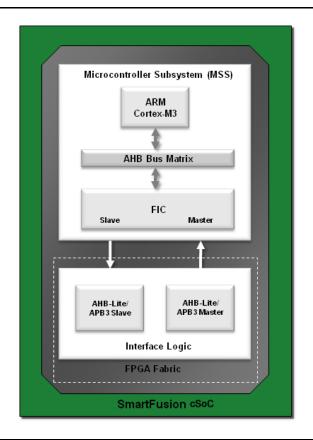


Figure 1 • FPGA Fabric to FIC Connection

## **Design Description**

The FPGA fabric logic can be connected to the MSS using AHB or APB bus interface. The FPGA fabric interface controller acts as AHB to AHB-Lite or AHB to APB bridge to interface the AHB communication matrix with FPGA fabric logic.

Refer to *Connecting User Logic to the SmartFusion Microcontroller Subsystem* application note for more details on how to connect the user logic in the FPGA fabric to microcontroller subsystem.

In some scenarios, the FPGA fabric logic needs to access the MSS peripherals like eSRAM or external memory controller (EMC) with very high throughput. In such cases the FPGA fabric logic should be connected to the FIC using AHB-Lite interface. Figure 2 on page 3 shows the FIC configuration with Interface type as AHB-Lite. The FIC now acts as AHB to AHB-Lite bridge without any bus conversion, except for signal mapping. Whereas, in APB interface, the bus conversion from AHB to APB and vice versa are needs to be done.

The FIC configurator provides the options to configure AHB-Lite as Master Interface or Slave Interface or both as shown in Figure 2 on page 3. The AHB-Lite configuration provides 'Use Bypass Mode' option to enable or disable the address and data pipelining between FPGA fabric logic and MSS.

In bypass mode (non-pipelined mode/"Use Bypass Mode" option checked), it is possible to achieve zero wait state access between FPGA master and zero-wait state capable MSS slave if there is no other master accessing that slave.



However, in Non-pipelined mode, the setup time requirement of FIC interface is bigger, which may lower overall frequency of operation. Also in Non-pipelined mode, the frequency ratio between MSS clock and FPGA fabric clock is restricted to 1:1.

<u> Confi</u>	guring MSS_FIC_0 (MSS_FIC - 1.0.101)
Configure	ation
ſ	Clocks Configuration
	MSS Clock Frequency 75.000
	Fabric Clock Frequency 75.000
	Interface Configuration
	Interface Type
	Use Bypass Mode
	Use Master Interface 📃
	Use Slave Interface
Help	OK Cancel

#### Figure 2 • FIC Configuration with AHB-Lite Interface

In Pipelined mode (**Use Bypass Mode** option unchecked/default mode), the interface between AHB bus matrix and FPGA has registered signals that reduce setup requirements. This may improve overall system frequency but these registers introduce a bubble in AHB transaction pipe. It results in inserting a wait-state for each transaction even if Master and Slave are capable of zero-wait state access.

You have to analyze the critical paths between the FIC and the logic in the FPGA fabric, when the **Use Bypass Mode** is enabled. You need to make sure that all the timing requirements have been met between FIC and FPGA fabric logic.

The following sections analyze the write data throughput in Bypass mode and Pipelined mode with example design.

### Fabric Master Accessing eSRAM with Pipelined Mode

The example design consists of AHB master in FPGA fabric that writes 32-bit data to 256 locations of eSRAM starting from the address 0x20007FFC. The AHB master logic is connected to the slave interface of the FIC with Pipelined mode. Figure 3 on page 4 shows the FIC configuration with Interface type as AHB-Lite slave and the unchecked **Use Bypass Mode** option.



As mentioned earlier, it is not possible to achieve zero wait transaction due to registered signals in FIC interface. HREADYOUT signal goes low for every transaction as shown in Figure 4 on page 5.

🗟 Configuring MSS_FIC_0 (MSS_FIC - 1.0.101)
Configuration
-Clocks Configuration
MSS Clock Frequency 80.000
Fabric Clock Frequency 80.000
Interface Configuration
Interface Type
Use Bypass Mode 🗌
Use Master Interface
Use Slave Interface 🛛 🗹
Help   OK Cancel

Figure 3 • FIC Configuration for Pipelined Mode



Þ \* Now: 100 us Delta: 2 Layout Simulate 2149109 ps to 2422397 ps **×** 63 ××× 00000000 Ð e 100 us 🔶 📃 📰 📰 🕅 🖓 🕜 🖓 🤂 -62.49 ns 0000000  $\square$ ď 12.498 ns • ď € + ۱ \* 1 **8**. **8** Help Window -100000 ns 2237.442 ns 2299,932 ns 2224,944 ns **Σ** 5<sup>00</sup> 5<sup>00</sup> Layout 🖬 Objects | 🌼 Processes | 🏨 Library | 🛃 sim | 🧱 Wave Now Cursor 5 Cursor 6 Cursor 7 Tools sim:/testbench Add Wave וּך בי 40 Compile Simulate ±ŋ M ModelSim ACTEL 6.5d â ·\*) (\*) Delta: View ~\$» 100 us ď Ξġ . . . . e E

It takes multiple clock cycles to complete a single write. Figure 4 shows the AHB master write transactions for single data whereas, Figure 5 on page 6 shows write transactions for 256 data.

Figure 4 • AHB Master Write Transactions for Single Data



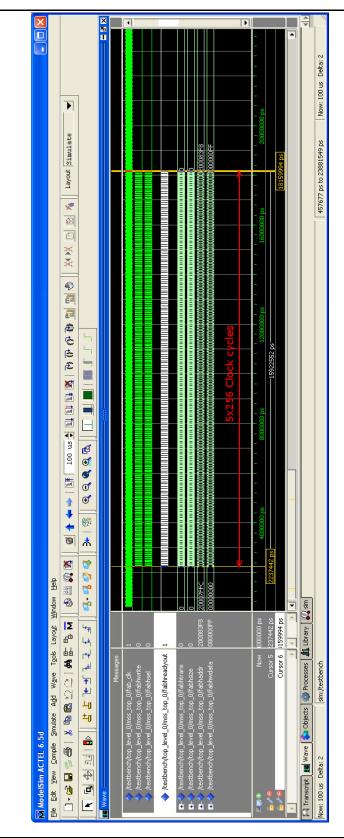


Figure 5 • AHB Master Write Transactions for 256 Data



Figure 6 and Figure 7 show the timing paths between MSS and fabric logic. These paths are in FAB (fabric clock) domain.

Elle Edit View Actions Tools Window Help								- 8	5 2
6 <b>6 6 7</b> × 22 <b>&gt;</b>	• ×	2 📝 m 190 📴 🌬 🌭 🔭	1 🗗 🔤 🚛 🛫 🕒						
<u> </u>	From	n <b>*</b>	То *						_
MAX				1	Apply Filt	er St	ore Filter	Reset Filter	er
<ul> <li>Register to Register</li> <li>External Setup</li> </ul>		Source Pin	Sink Pin	Delay (ns)	Slack (ns)	Arrival (ns)	Required (ns)	Setup (ns)	
Clock to Output     Register to Asynchronous	1	mss_top_0/MSS_ADLIB_INST/U_CORE:GLB	fic_master_trans_0/HADDR_TEMP[14]/U1:D	9.396	4.381	13.466	17.847	0.409	
External Recovery Asynchronous to Register	2	mss_top_0/MSS_ADLIB_INST/U_CORE:GLB	fic_master_trans_0/HADDR_TEMP[15]/U1:D	8.995	4.773	13.065	17.838	0.409	
mss_fabric_interface_clock to mss_ccc_gla1	3	mss_top_0/MSS_ADLIB_INST/U_CORE:GLB	fic_master_trans_0/HADDR_TEMP[10]/U1:D	8.530	5.247	12.600	17.847	0.409	
	• 4	mss_top_0/MSS_ADLIB_INST/U_CORE:GLB	fic_master_trans_0/HADDR_TEMP[30]/U1:D	8.502	5.275	12.572	17.847	0.409	
	5	mss_top_0/MSS_ADLIB_INST/U_CORE:GLB	fic_master_trans_0/HADDR_TEMP[7]/U1:D	8.366	5.402	12.436	17.838	0.409	
10	6	mss_top_0/MSS_ADLIB_INST/U_CORE:GLB	fic_master_trans_0/HADDR_TEMP[11]/U1:D	8.294	5.474	12.364	17.838	0.409	
5	7	mss_top_0/MSS_ADLIB_INST/U_CORE:GLB	fic_master_trans_0/HADDR_TEMP[9]/U1:D	8.179	5.585	12.249	17.834	0.409	
	8	mss_top_0/MSS_ADLIB_INST/U_CORE:GLB	fic_master_trans_0/HADDR_TEMP[4]/U1:D	8.127	5.641	12.197	17.838	0.409	
	9 9	mss_top_0/MSS_ADLIB_INST/U_CORE:GLB	fic_master_trans_0/HADDR_TEMP[3]/U1:D	8.026	5.751	12.096	17.847	0.409	

Figure 6 • Maximum Delay Analysis View

Eile Edit View Actions Iools Window Help	1			al					- 1	ē
	<u> </u>	2 <u>3</u> m m m m m m m m m m m m m m m m m m m	M (19 19 19 19 19 19 19 19 19 19 19 19 19 1	<u></u>						
	Fror	n *		To *						
min						Apply Filter	Stor	e Filter 🛛 🖡	Reset Filt	lte
mss_ccc_gla1     Register to Register					Delay	Slack,	Arrival	Required	Hold	Т
External Hold Clock to Output		Source Pin	Sink	Pin	(ns)	(ns) 🛆	(ns)	(ns)	(ns)	
Register to Asynchronous	Ľ	mss_top_0/MSS_ADLIB_INST/U_CORE:GLB	fic_master_trans_0/ahb_s	tates[0]:D	4.036	2.736	7.148	4.412	0.000	
External Removal     Asynchronous to Register	2	mss_top_0/MSS_ADLIB_INST/U_CORE:GLB	fic_master_trans_0/HTRA	NS[1]/U1:D	4.386	3.094	7.498	4.404	0.000	J
	3	mss_top_0/MSS_ADLIB_INST/U_CORE:GLB	fic_master_trans_0/HADD	R_TEMP[27]/U1:D	4.472	3.180	7.584	4.404	0.000	J
mss_poikt to mss_ccc_gla1	4	mss_top_0/MSS_ADLIB_INST/U_CORE:GLB	fic_master_trans_0/HADE	R_TEMP[23]/U1:D	4.803	3.505	7.915	4.410	0.000	j
•	5	mss_top_0/MSS_ADLIB_INST/U_CORE:GLB	fic_master_trans_0/HADD	R_TEMP[19]/U1:D	4.897	3.593	8.009	4.416	0.000	J
10	6	mss_top_0/MSS_ADLIB_INST/U_CORE:GLB	fic_master_trans_0/HADE	R_TEMP[26]/U1:D	4.917	3.629	8.029	4.400	0.000	ĵ
	7	mss_top_0/MSS_ADLIB_INST/U_CORE:GLB	fic_master_trans_0/HADE	R_TEMP[25]/U1:D	4.973	3.675	8.085	4.410	0.000	J
8 4		Details for path From: mss_top_0/MSS_ADLIB_INST/U_C( To: fic_master_trans_0/ahb_states[0]:D	DRE:GLB							
		Pin Name	Туре		Net Na	me		Ce	ll Name	е

Figure 7 • Minimum Delay Analysis View



### Fabric Master Accessing eSRAM with Bypass Mode

The example design consists of AHB master in FPGA fabric that writes 32-bit data to 256 locations of eSRAM starting from the address 0x20007FFC. The AHB master logic is connected to the slave interface of FIC with Bypass mode. Figure 8 shows the FIC configuration with interface type as AHB-Lite slave and the **Use Bypass Mode** option selected. As mentioned, it is possible to achieve zero wait state access between the FPGA fabric logic and MSS since there is no register stage between the FPGA fabric and AHB bus matrix. In this configuration, the HREADYOUT signal from AHB bus matrix is always high since no other master is accessing eSRAM and each write takes single clock cycle.

🗟 Configuring MSS_FIC_0 (MSS_FIC - 1.0.101)
Configuration
-Clocks Configuration
MSS Clock Frequency 80.000
Fabric Clock Frequency 80.000
Interface Configuration
Interface Type AHBLite
Use Bypass Mode 🔽
Use Master Interface
Use Slave Interface 🔽
Help

Figure 8 • FIC Configuration for Bypass Mode



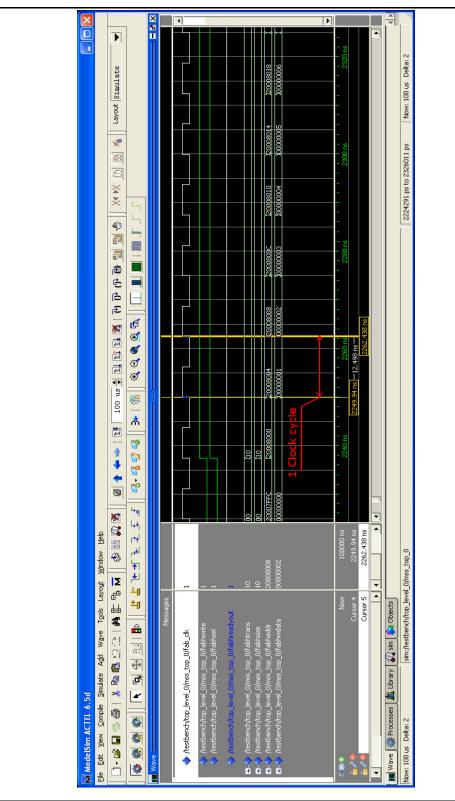
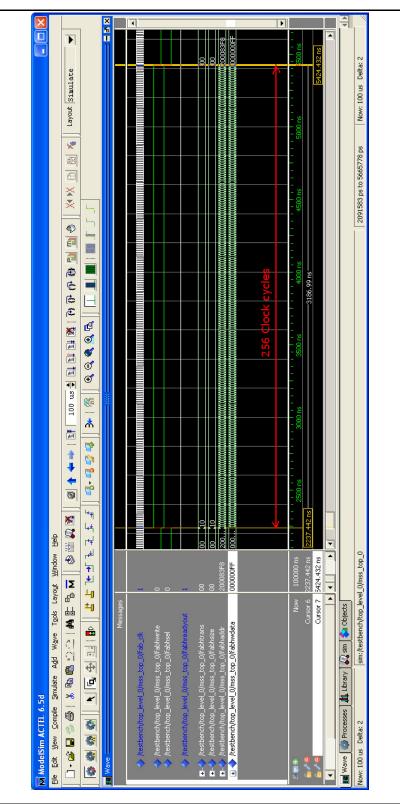


Figure 9 shows the AHB master write transactions for single data, whereas Figure 10 on page 10 shows write transactions for 256 data.

Figure 9 • AHB Master Write Transactions for Single Data





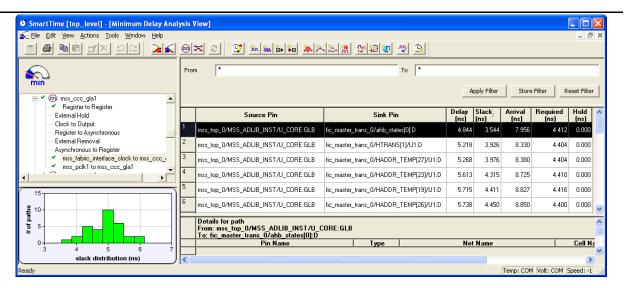
### Figure 10 • FIC Configuration for Bypass Mode

Figure 11 on page 11 and Figure 12 on page 11 show the timing paths between MSS and fabric logic. These paths are in FAB (fabric clock) domain.



	R		<u>}</u> <u>m ma ≥+ ∞ ≫ ≻ ≥ m</u> <u>1</u>	¥22 🕂 🖓 🕒					
		Fro	om *	То 🖡					
MAX						Apply Filte	r Sto	re Filter	Reset Fil
External Setup			Source Pin	Sink Pin	Delay (ns)	Slack (ns)	Arrival (ns)	Required (ns)	Setup (ns)
Clock to Duput Register to Asynchronous External Recovery Asynchronous to Register 'mss_fabric_interface_clock to mss_coc_gla1		1	mss_top_0/MSS_ADLIB_INST/U_CORE:GLB	fic_master_trans_0/HADDR_TEMP[14]/U1:D	10.512			17.847	0.40
	_	2	mss_top_0/MSS_ADLIB_INST/U_CORE:GLB	fic_master_trans_0/HADDR_TEMP[15]/U1:D	10.111	3.657	14.181	17.838	0.40
		3	mss_top_0/MSS_ADLIB_INST/U_CORE:GLB	fic_master_trans_0/HADDR_TEMP[10]/U1:D	9.646	4.131	13.716	17.847	0.40
→ mss_pclk1 tomss_ccc_gla1 → √ (00) mss ccc gla0	-	4	mss_top_0/MSS_ADLIB_INST/U_CORE:GLB	fic_master_trans_0/HADDR_TEMP[30]/U1:D	9.618	4.159	13.688	17.847	0.40
10		5	mss_top_0/MSS_ADLIB_INST/U_CORE:GLB	fic_master_trans_0/HADDR_TEMP[7]/U1:D	9.482	4.286	13.552	17.838	0.40
22		6	mss_top_0/MSS_ADLIB_INST/U_CORE:GLB	fic_master_trans_0/HADDR_TEMP[11]/U1:D	9.410	4.358	13.480	17.838	0.40
stpe 5-		7	mss_top_0/MSS_ADLIB_INST/U_CORE:GLB	fic_master_trans_0/HADDR_TEMP[9]/U1:D	9.295	4.469	13.365	17.834	0.40
*		8	mss_top_0/MSS_ADLIB_INST/U_CORE:GLB	fic_master_trans_0/HADDR_TEMP[4]/U1:D	9.243	4.525	13.313	17.838	0.40
		9	mss_top_0/MSS_ADLIB_INST/U_CORE:GLB	fic_master_trans_0/HADDR_TEMP[3]/U1:D	9.142	4.635	13.212	17.847	0.40





#### Figure 12 • Minimum Delay Analysis View

Table 1 shows the result of the write data throughput analysis in pipelined mode and bypass mode.

Table 1 •	Summary of Write Data	Throughput Analysis	in Pipelined Mod	e and Bypass Mode
-----------	-----------------------	---------------------	------------------	-------------------

Mode	No. of clock cycles required for single data transfer	No. of clock cycles required for 256 data transfer
Pipelined	5	1280
Bypass	1	256

The result shows that the Pipelined mode takes 1024 extra clock cycles to write 32-bit data to 256 locations of eSRAM compared to the bypass mode. Libero<sup>®</sup> System-on-Chip (SoC) projects are provided in the design files attached with this design example (refer to "Appendix A – Design Files" on page 12).



### Conclusion

This application note describes the functionality of **Use Bypass Mode** in the FIC to maximize the data transfer between FPGA fabric logic and MSS peripherals. It also analyzes the write data throughput in Bypass mode and Pipelined mode with example design.

## Appendix A – Design Files

You can download the design files from the Microsemi SoC Products Group website: www.microsemi.com/soc/download/rsc/?f=A2F\_AC363\_DF.

The design file consists of Libero SoC projects. Refer to the Readme.txt file included in the design file for directory structure and description.

## **List of Changes**

The following table lists critical changes that were made in the current version of the chapter.

Revision*	Changes	Page
Revision 4 (February 2012)	Removed ".zip" extension in the Design files link (SAR 36763).	12
Revision 3 (January 2012)	Replaced "Libero IDE" with "Libero SoC" (SAR 35792)	11, 12
Revision 2 (March 2011)	Figure 1 has been replaced with a new figure (SAR 30743)	2
Revision 1 (January 2011)	Modified second row contents of Table 1.	11

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**.

© 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.