# **Engineer-to-Engineer Note**



Technical notes on using Analog Devices DSPs, processors and development tools Visit our Web resources http://www.analog.com/ee-notes and http://www.analog.com/processors or e-mail processor.support@analog.com or processor.tools.support@analog.com for technical support.

## Interfacing Blackfin® Processors to Winbond W25X16 SPI Flash Devices

Contributed by Sunish Oturkar and Joe B.

Rev 1 – January 2, 2008

## Introduction

This EE-Note describes how to interface to, program, and boot from a Winbond W25X16 SPI flash memory device using the Blackfin® processor SPI peripheral.

The code referenced throughout this EE-Note was developed using the VisualDSP++® 4.5 (June 2007 Update) development tools and revision 2.2 of the ADSP-BF537 EZ-KIT Lite® evaluation platform.

While the discussion throughout this document focuses on the ADSP-BF537 processor, the concepts can be applied to all Blackfin processors, so long as proper care is given to the pins used in the SPI interface to the Winbond flash device and the port multiplexing is properly initialized, where applicable.

## **Connecting the SPI Device**

For the interface between the Blackfin processor SPI port and the Winbond flash device, refer to Table 1.

The chip-select (/CS) must be connected to SPI\_SSEL1 (PF10) on the ADSP-BF537 processor in order to utilize the master SPI boot mode. This pin should have an external pull-up resistor to ensure that the chip-select is disabled during power-up. Refer to the "System Reset and Booting" chapter of the ADSP-BF537 Blackfin Processor Hardware Reference<sup>[1]</sup> for details of the boot process.

The write-protect (/WP) and hold-off (/HOLD) features are not used by this driver, so they must be pulled high to ensure that the feature is not enabled during operation.

Winbond W25X16 Pin #. Name	ADSP-BF537 Pin Name
1. /CS	SPI_SSEL1 (PF10)
2. MISO	MISO (PF12)
3. /WP	NC - Pull-Up to $V_{DDEXT}$
4. GND	GND
5. MOSI	MOSI (PF11)
6. SCK	SCK (PF13)
7. /HOLD	NC - Pull-Up to $V_{DDEXT}$
8. VCC	V <sub>DDEXT</sub>

Table 1. W25X16 to BF534/6/7 interface

## **Implementing the W25X16 Driver**

The associated file includes a VisualDSP++ project,

BF534\_6\_7\_SPI\_Programmer\_Driver\_W25X16, which was developed for use by the VisualDSP++ flash programmer plug-in.

Copyright 2008, Analog Devices, Inc. All rights reserved. Analog Devices assumes no responsibility for customer product design or the use or application of customers' products or for any infringements of patents or rights of others which may result from Analog Devices assistance. All trademarks and logos are property of their respective holders. Information furnished by Analog Devices applications and development tools engineers is believed to be accurate and reliable, however no responsibility is assumed by Analog Devices regarding technical accuracy and topicality of the content provided in Analog Devices Engineer-to-Engineer Notes.



#### **Setting Up the Flash Programmer**

The following instructions describe how to use the VisualDSP++ flash programmer plug-in utility to program the device.

- Open the project file (.dpj):
   File -> Open -> Project
- 2. Build the project to produce the executable file (.dxe):

Project -> Build Project

- Open the Flash Programmer Plug-In: Tools -> Flash Programmer...
- 4. Load the executable into the Flash Programmer

Driver Tab -> Browse

Navigate to:

BF534\_6\_7\_SPI\_Programmer\_Driver\_W25X
16\Debug

Select and open:

BF534\_6\_7\_SPI\_Programmer\_Driver\_W25X
16.dxe

Click Load Driver.

5. Verify that the driver loaded successfully. The Message Center should display "Success: Driver Loaded", as shown in Figure 1.

	Sector	Start Offset	End Offset	% u	1
Driver file:		0x00000000	0x00000FFF	222	1
\Debug\BF534_6_7_SPI_Programmer_Driver_W25X16.dxe		0x00001000	0x00001FFF	???	
	2	0x00002000	0x00002FFF	???	
Load Driver Browse	3	0x00003000	0x00003FFF	???	
Flash information	4	0x00004000	0x00004FFF	???	
Manufacturer code: 0x0	5	0x00005000	0x00005FFF	???	
Device code: 0x0	6	0x00006000	0x00006FFF	???	
Part description: Winbond W25X16VSSIG	7	0x00007000	0x00007FFF	???	
Company: FRROR: Unknown variable or symbo	8	0x00008000	0x00008FFF	???	
company: Electric character character of symbol	9	0x00009000	0x00009FFF	???	1
Driver information	Message	center:			
Build date: ERROR: Unknown variable or symbo Version: ERROR: Unknown variable or symbo	Success: Driver loaded.				
Name: ADSP-BF534/6/7 SPI Flash Driver driver					
	<			3	9

Figure 1. Loading the driver into the flash programmer plug-in

#### **Using the Flash Programmer**

There are two ways to use the flash programmer utility to program a Winbond W25X16 flash device.

#### Individual Commands

The first method is to manually program sectors by sending individual commands through the Commands tab (Figure 1).

For information on using functions from the Commands tab, refer to *VisualDSP++* Flash *Programmer API* for *Blackfin Processors*  $(EE-311)^{[2]}$  and the following topic in VisualDSP++ online Help:

- From the menu bar in VisualDSP++, choose Help -> Contents.
- 2. Expand Graphical Environment.
- 3. Expand Emulation Tools.
- 4. Expand Flash Programmer.
- 5. Select About the Flash Programmer.

A screen capture from the VisualDSP++ development tools is shown in Figure 2.

💕 VisualDSP++ 5.0 Help for Blackfi	n Processors
Hide Locate Back Forward Home	Print Options
Contents       Index       Search       Favorites         Welcome       Assistance       Index       Search       State of the search	About the Flash Programmer The Flash Programmer utility (Tools -> Flash Programmer) provides a convenient graphical interface to on-board flash memory devices. Use the Flash Programmer window to view various flash memory information, fill or erase portions of flash memory, or load a file into flash memory. Analog Devices supplies flash drivers (algorithms) for many EZ-KIT Lite evaluation systems. Additionally, the Flash Programmer allows you to create custom flash drivers for use on custom boards with a variety of processor and flash device combinations. See Also

Figure 2. Flash Programmer Help topic



#### Loader File

The second method of using the flash programmer utility is to program a loader file (.ldr) directly into the Winbond flash device (through the Programming tab as shown in Figure 1).

#### Creating an SPI-Compatible Loader File

Perform the following procedure to create the loader image that will be used to program the Winbond flash memory device:

1. Open a VisualDSP++ project:

File -> Open -> Project

- Open the Project Options dialog box:
   Project -> Project Options...
- 3. Under the Type pull-down menu, select Loader file (as shown in the top portion of Figure 3).



Figure 3. Specifying project and load options

- 4. In the project tree under the Load control (lower portion of Figure 3), click Options.
- 5. Under Boot Mode, click SPI
- 6. Under Boot Format, click Intel Hex
- 7. Verify that:
  - □ 8-Bit is selected under Output width
  - NONE is selected under the Programmable Flag pull-down menu.
  - □ Use default start address is selected
  - □ Verbose is not selected
  - □ None of the three edit fields (Initialization file, Output file, or Additional options) are populated.
- 8. Click OK.
- 9. Rebuild the project:

Project -> Rebuild Project

When the build completes, a loader file (.ldr) will be generated in the project's output directory.

#### Programming the Loader File into the Device

Once the loader file has been properly generated, can be programmed into the Winbond W25X16 flash device, as follows:

1. Load the driver's executable into the plug-in by following Steps 4 and 5 of



Setting Up the Flash Programmer on page 2.

- 2. Click the Programming tab.
- 3. Under Pre-program erase options, select Erase affected.
- 4. Under File format, select Intel Hex.
- 5. Select Verify while programming.
- 6. Click Browse.
- 7. Navigate to the output directory of the project used in the previous section and select the loader file (.ldr).
- 8. Click Program and observe the status bar updating.
- 9. Verify that the device was programmed successfully. The Message Center should display: Success: Erased sector(s). Success: Program complete

If successful, the Winbond memory device now contains the appropriate loader image for the processor to boot from by way of the master SPI boot mode available on the ADSP-BF537 Blackfin processor.

## **Booting from the W25X16**

If the Winbond memory device is properly connected to the Blackfin processor SPI peripheral, as described in Table 1, booting from the memory device is possible. Simply set the boot mode of the ADSP-BF537 Blackfin processor to b#11 (boot from SPI) and reset the processor. Refer to *ADSP-BF533 Blackfin Booting Process (EE-240)*<sup>[3]</sup> for more booting information.

## References

- [1] ADSP-BF537 Blackfin Processor Hardware Reference. Rev 2.0, May 2005. Analog Devices, Inc.
- [2] *VisualDSP*++® *Flash Programmer API for Blackfin*® *Processors (EE-311)*. Rev 1, December 2006. Analog Devices, Inc.
- [3] ADSP-BF533 Blackfin® Booting Process (EE-240). Rev 3, January 2005. Analog Devices, Inc.
- [4] *W25X16/32/64 SPI Flash Datasheet*. Rev F, March 2007. Winbond Electronics Corporation.

## **Document History**

Revision	Description
Rev 1 – January 2, 2008 by Sunish Oturkar and Joe B.	Initial Release