ADSP-BF526 EZ-Board™ Support Package (BSP) v1.0.0 Release Notes

Thank you for installing the ADSP-BF526 EZ-Board™ Support Package (BSP). The BSP provides software and documentation in support of the ADSP-BF526 EZ-Board™.

The EZ Board is designed for use with CrossCore® Embedded Studio (CCES) for Analog Devices Processors software development tools. The CCES development environment aids advanced application code development and debug, such as:

- Create, compile, assemble, and link application programs written in C++, C, and assembly
- Load, run, step, halt, and set breakpoints in application programs
- Read and write data and program memory
- Read and write core and peripheral registers

- Plot memory

For more details on CCES, please visit www.analog.com/cces. For more on the ADSP-BF526 EZ-Board™, please visit http://www.analog.com/en/processors-dsp/blackfin/adsp-bf526/products/BF526-EZLITE/eb.html.

The ADSP-BF526 EZ-Board™ BSP provides comprehensive software support for the ADSP-BF526 EZ-Board™. Specifically, drivers, examples and code sketches are included for the following components:

- SSM2603 Low Power Audio Codec.

The BSP also provides comprehensive examples which demonstrates the on-chip drivers and services.

The CCES Help environment provides complete hardware and software documentation.

License Checking

There are no license requirements for the ADSP-BF526 EZ-Board™ BSP.

Installation Logging

The installer does not create a log file by default. If you encounter installation issues, you can generate an installation log file by running the installer from the command prompt.

Change to the directory containing downloaded installer executable and run the following from the command prompt:

```
ADI_ADSP-BF526_EZKIT-Rel1.0.0.exe  /v"/l*v c:\temp\installer.log"
```

Support and Assistance

There are several options for contacting support:

- Submit your questions online at:

  http://www.analog.com/support

- E-mail your Processor and DSP software and development tools questions from within CrossCore Embedded Studio:

  Go to "Help->E-mail Support…". This will create a new e-mail addressed to processor.tools.support@analog.com, and will automatically attach your CrossCore Embedded Studio version information (ProductInfo.html).

- E-mail your Processors and DSP applications and processor questions to:
  - processor.support@analog.com OR
  - processor.china@analog.com (Greater China support)

- Post your questions in the Processors and DSP online technical support community in Engineer Zone at:

  http://ez.analog.com/community/dsp

Software Requirements

To build the example projects included in the ADSP-BF526 EZ-Board™ BSP, CrossCore Embedded Studio version 1.0.1 or later is required.

Test Configurations

The software versions used to test are:

CrossCore® Embedded Studio version 1.0.1 with 1.0.0 ADSP-BF526 EZ-Board™ BSP

At the time of release, the tested hardware revision is:

ADSP-BF526 EZ-Board™ PCB Revision 1.1 BOM Revision 1.4

Getting Started

Adding a Driver to a Project

When adding an ADSP-BF526 Driver to your project, the IDE will add the sources for the driver to the CCES Project folders, starting at "system". There will be a folder specific to the driver(s) or service(s) you have added under this folder.

Creating a project which includes a ADSP-BF526 driver/service

In order to create a project you should follow the instructions provided in the CrossCore Embedded Studio help. As part of the project creation, the page "Add-in selection" contains a list of all the available add-ins for the project that you are creating based on the installed products and the project's chosen processor and type. You can see the drivers in support of the ADSP-BF526 EZ-Board™ under the "Device Drivers and System Services" category. Within this catgeory you will see "ADSP-BF526 EZ-Board" which contains the drivers for the on-board peripherals (SSM2603). The on-chip peripheral drivers will be listed in "On-chip peripheral drivers" folder and the system services are listed in the "System Services" folder.

The ADSP-BF526 EZ-Board add-in generates a call to adi_initComponents(). For more information on adi_initComponents(), please refer to the CCES help section:

CrossCore® Embedded Studio 1.0.1 > Graphical Development Environment > System Configuration

Adding a ADSP-BF526 driver to an existing project

Every CrossCore Embedded Studio project contains a System Configuration file called `system.svc` which is located in the root of the project. The file is the IDE's interface for managing the various pre-written software components used in the "system" implemented by a project. Double-clicking any `system.svc` file in a navigation view opens that file in the System Configuration Utility which allows you to see the add-ins that you currently have in your project. Click on "Add..." and select the ADSP-BF526 EZ-Board Drivers add-in which is under the "Device Drivers and System Services" for the on-board SSM2603 Audio Codec driver. For adding on-chip peripherals drivers select the "On-chip peripheral drivers" and for the system services select the "System Services".

Notes:

- If the IDE detects that `adi_initComponents()` is not yet present in `main()`, it prompts you to add it and offers to insert it for you.

Configuration

There are no ADSP-BF526 EZ-Board™ driver configuration options available in the IDE.

Interrupts

CrossCore Embedded Studio provides a coherent interrupt management mechanism which allows for the same interface to be used in RTOS and non-RTOS applications. This means that interrupt service routines in all applications must be written in C and use the adi_int interface. Any thread-safety requirements or interactions with tasks are handled by the adi_int interface. For more information on the adi_int API, in CrossCore Embedded Studio go to Help > Search and enter adi_int.

Examples of the usage of this interrupt management mechanism are the System Services and Device Drivers provided with Crosscore Embedded Studio. By using the adi_int interface, the same services and drivers can be used in all applications regardless of whether an operating system is used.

Sketches and Examples

Sketches

CrossCore Embedded Studio provides a mechanism by which small code fragments, called sketches, can be generated with parameterized input provided by the user. The resulting code can then be copied and pasted to a project. Sketches for the on-board peripherals on the ADSP-BF526 EZ-Board™ are provided in the BSP. To locate the sketches specific to the ADSP-BF526 EZ-Board™ BSP, open up the example browser (Help -> Browse Examples) and then select ADSP-BF526 EZ-Board™ product in the "Product:" pulldown. The sketches for the on-chip drivers and system services can be located by selecting the CrossCore Embedded Studio product in the "Product" pulldown.

Examples

In addition to the code sketches, the ADSP-BF526 EZ-Board™ BSP provides the following examples.

1. Parallel flash programmer interface application.
2. Serial Flash programmer interface application.

EXAMPLES FOR DRIVERS:

1. AudioLoopback example which uses on-board SSM2603 Audio Codec.
2. RotaryCounter example which uses the on-chip ADSP-BF526 Rotary encoder
3. DigiPot_TWI example which uses the on-board digital potentiometer (AD5258).
4. CharEcho_UART example to demonstrates character echo using UART.

5. SPI_flash_read example which uses SPI driver to access the on-board serial flash memory (SST25WF040).

EXAMPLES FOR SERVICES:

1. Button_LED_GPIO demonstrates the GPIO service, which toggles the led when user presses the push buttons on the board.
2. SetGetFreq example which demonstrates the power management service.
3. STDIO_CharEcho example demonstrates the STDIO service which echos the character of the key pressed by the user.
4. CTimer_Periodic example demonstrates the core timer service.
5. Timer_Callback example demonstrates the General Purpose timer service.
6. Watchdog example demonstrates the Watchdog service.
7. EachDayAlaram example demonstrates the Real Time Clock service which programs RTC to generate alarm each day.
8. SetGetDateTime example demonstrates the Real Time Clock service to set and get the time.
9. MemCopyArrayMode example demonstrates the Memory DMA service to do a memory copy using Array Mode.
10. MemCopyListMode example demonstrates the Memory DMA service to do a memory copy using List Mode.
11. MemCopyOneShot1D example demonstrates the Memory DMA service to do a memory copy using One Dimensional DMA mode.
12. MemCopyOneShot1D example demonstrates the Memory DMA service to do a memory copy using Two Dimensional DMA mode.
13. MemCopyOneToMany example demonstrates the Memory DMA service to do a memory copy from single location to many locations in the memory.

Location

In order to locate the ADSP-BF526 BSP examples and sketches, you can use the following:

*   Open CrossCore Embedded Studio's Example Browser which can be found in CrossCore Embedded Studio under Help. Select in the Product section "ADSP-BF526 EZ-Board v1.0.0" for a full list of examples and sketches.
*   Import projects located in your ADSP-BF526 EZ-Board™ BSP installation folder under `the example directoy in product installation (<ADSP_BF526_EZ_Board_installation_root>\BF526_EZ-Board\Blackfin\Examples).`

Documentation

API documentation for the drivers included in the ADSP-BF526 EZ-Board™ BSP can be found in CCES Help.

General information on the driver model can be found in CCES help under

    CrossCore® Embedded Studio 1.0.1 > System Runtime Documentation > System Services and Device Drivers

MISRA-C Support

MISRA C is a software development standard for the C programming language developed by the Motor Industry Software Reliability Association (MISRA). Its aims are to facilitate code safety, portability, and reliability in the context of embedded systems, specifically those systems programmed in ANSI C. The compiler detects violations of the MISRA rules at compile-time, link-time, and run-time.

System Services and Device Driver Thread Safety

All system services and device drivers (SSDD) use mutexes and semaphores to ensure thread-safety. If an RTOS is present then the SSDD will use the RTOS mutex and semaphores. If an RTOS is not present then the SSDD will use a non-RTOS implementation of mutexes and semaphores (spin locks).

Known issues with the ADSP-BF526 EZ-Board™ Support Package (BSP)

None