

Technical Notes on using Analog Devices' DSP components and development tools

Phone: (800) ANALOG-D, FAX: (781) 461-3010, EMAIL: dsp.support@analog.com, FTP: <ftp.analog.com>, WEB: www.analog.com/dsp

Copyright 1999, 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 the technical accuracy of the content provided in all Analog Devices' Engineer-to-Engineer Notes.

Setting Up Streams with the VisualDSP Debugger - Release 4.x

This procedure is based on one of the examples that comes with the VisualDSP 4.0 software (SPORTs TXIRQ Example - can be found in *C:\Program Files\Analog Devices\VisualDSP \21k\examples\SPORTS*). This program is a variation on this example which illustrates single word transfers with interrupts and simply passes data through the serial ports. The DSP receives a value in SPORT0, then transmits it through SPORT0. The main part of the program is an infinite loop that waits for the SPORT0 receive interrupt to occur. The interrupt service routine performs all data processing (in this example, transferring data from the SPORT receive register to the SPORT transmit register).

For simulation purposes, disk files will be used as inputs and outputs to the serial port. The file, *serin.dat* provides the input for the serial port. The debugger creates a file specified by the user (arbitrarily named *serout.dat* for this example procedure) and stores the output of the SPORT in this file.

SPORT Example:

This RXIRQ example is similar to TXIRQ code shown in section 10.11.2 "Single-Word Transfers with Interrupts" of the ADSP-2106X SHARC User's Manual. It illustrates the use of interrupts to perform single word transfers.

Procedure for setting up this example in the Debugger

1. Use the IDE to open and build the RXIRQ.DPJ project.
2. Start the Debugger (select the debug toolbar button in the IDE or open the Debugger manually through the OS) and switch session to an ADSP-21062 session if necessary.
3. Load your executable into the Debugger via the *File->Load* pull down menu if necessary.
4. Choose pull down menu item *SETTINGS->STREAMS*. This will bring up the Streams dialog box. Make the following selections:
 - a. In the upper left corner of the dialog box, select Source Debug Target and ensure that it is set to **ADSP-21062 ADSP-2106X Simulation Debug Target Component** and then set Device to **TX0** (using SPORT 0).
 - b. Next, fill in the destination file associated with SPORT 0. In the lower right corner of the streams

dialog box, check the "File" checkbox and browse to the SPORT project directory and CREATE a file named "**serout.dat**" (or whatever name you choose for the output). by typing this file name into the File Name field within the "Open a Destination File" dialog box.

- c. Make sure that the format is set to **Hexadecimal**.
 - d. Click on **Connect**. These steps have configured the output stream.
 - e. Check the **ACTIVE** Tab in the streams dialog box to make certain that this stream is indeed setup properly as an active stream. Once this is verified, select the **NEW** Tab in the streams dialog box to continue the streams setup for the input stream.
 - f. Now, for the input stream, go back to Source (File). Browse to select the file "*serin.dat*" which should be in the SPORT project directory.
 - g. Ensure that the format is set to **Hexadecimal**.
 - h. In the upper right corner of the dialog box, select Destination Debug Target. Target and ensure that it is set to **ADSP-21062 ADSP-2106X Simulation Debug Target Component** and then set Device to **RX0** (again using SPORT 0).
 - i. Click on **Connect**. These steps have configured the input stream.
 - j. Check the **ACTIVE** streams to make sure your selections were set correctly. You should see two lines indicating two separate streams were set up.
 - k. Click on **OK** to close the Streams Dialog Box.
5. Run the program.
 6. In a few seconds, you should see an dialog box come up that says:

****SPORT 0 input file hit EOF: Click OK.**

Another dialog box will inform you: Target Halted at User Request. Click **OK**

****Note:** This means that the Debugger has reached the end of the file. Check the contents of your output file to determine if the simulation executed properly. You should see the same values in *serout.dat* as those in *serin.dat* with the addition of some zeros which were used kick off the interrupt process in the source code for simplicity.