Technical Notes on using Analog Devices' DSP components and development tools
Contact our technical support by phone: (800) ANALOG-D or e-mail: dsp.support@analog.com
Or visit our on-line resources http://www.analog.com/dsp and http://www.analog.com/dsp/EZAnswers

Connecting the AD74111 Mono Audio Codec Evaluation Board to the ADSP-21161N SHARC® EZ-KIT Lite™ Board

Contributed by Mazlum Adas *October 21, 2003*

Introduction

This application note describes, setting up the AD74111 codec to communicate with the ADSP-21161 SHARC® DSP. The DSP receives the serial data from the ADC and sends it back to the DAC. An example code will delivered with this application note to show how to configure the codec and transfer the data.

Overview of the codec

The AD74111 is a front-end-codec for general-purpose audio and video applications. It provides a mono input channel and a mono output channel.

The codec supports sample rates from 8 kHz up to 48 kHz and word length from 16 to 24 bit. Also the codec provides a programmable ADC gain and on-chip volume control for the DAC channel.

The configuration of the AD74111 is done with SPORT 3, also for the data transmission the SPORTs (SPORT 1 and SPORT 3) are used.

Building the hardware

Figure 1 shows the connection between the AD74111 evaluation board and the ADSP-21161 EZ-KIT Lite™ board.

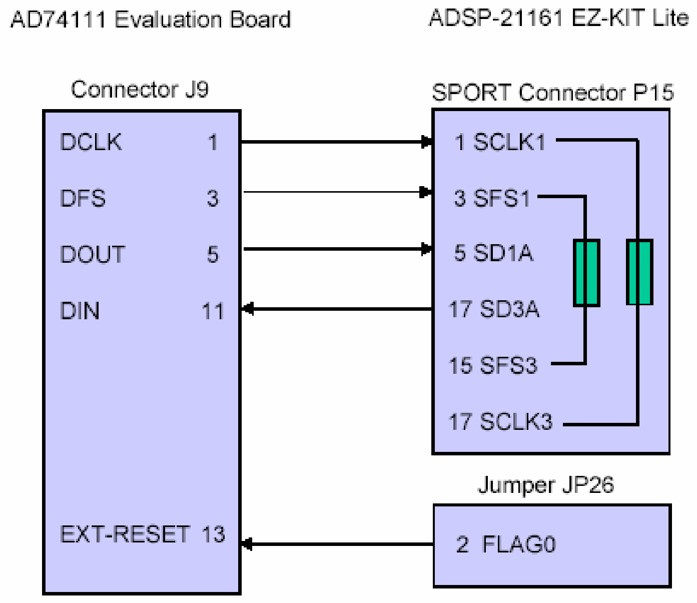

Figure 1: Connection of the AD74111 Evaluation Board and the ADSP-21161N EZ-KIT Lite

On the ADSP-21161N EZ-KIT Lite both frame syncs (SFS1 and SFS3) and the SPORT clocks (SCLK1 and SCLK3) are tied together with Zero-Ohm Resistors, so both SPORTs have the same frame sync and clock. It is necessary to have a good ground connection between the two boards. Also short cables should be used to reduce noise on the signals.

Jumper Settings on the ADSP-21161N EZ-KIT Lite Board

For the connection of the ADSP-21161N EZ-KIT Lite to the AD74111 Evaluation Board the relevant jumpers should be set as follows in Table 1. The other jumpers can be left as default.

Jumper	***Description***	***State***
JP4	FLAG 0 enable	OFF
JP5	FLAG 1 enable	OFF
JP26	Push-button enable FLAG 0	OFF
JP27	Push-button enable FLAG1	OFF
JP19	DSP ID	ON: 1-2, 3-4, 5-6
JP20	Boot Mode	ON: 3-4 OFF: 1-2, 5-6
JP21	Clock Mode	ON: 5-6 OFF: 1-2, 3-4

Table 1: ADSP-21161N EZ-KIT Lite jumper’s settings

Jumper Settings on the AD74111 Evaluation Board

The following table (Table 2) shows the jumper settings for the AD74111 Evaluation board.

Link	*Description*	*State*
LK5	Selecting of the on board crystal oscillator	B
LK6	Disabling of the MCLK divider	A
LK7	MCLK/2 is not reference clock source for DCLK	OFF
LK8	Selecting as master device	A
LK11	External reset signal	B

Table 2: AD74111 Evaluation board jumper's settings

The other jumpers can be left in the default position.

Software flow

The software flow for the data transmission between the ADSP-21161N SHARC DSP and the AD74111 is described in the following figure (Figure 2):

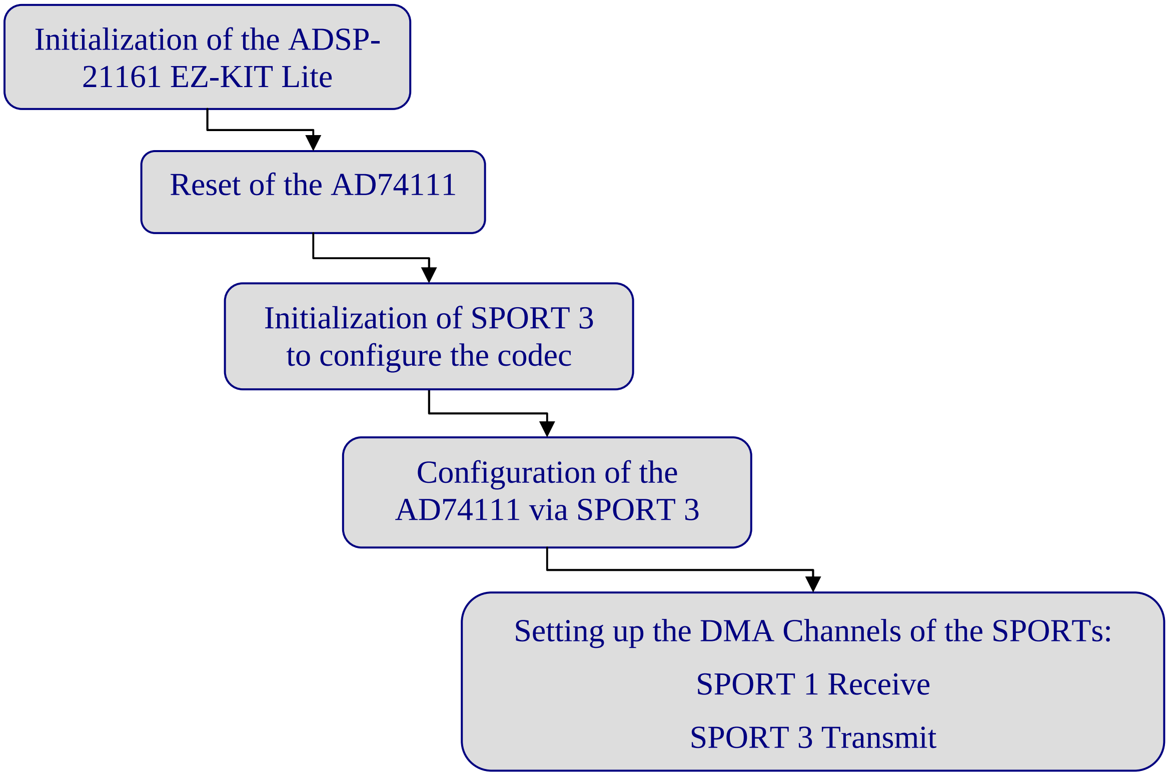
Initialization of the ADSP-21161 EZ-KIT Lite
Reset of the AD74111
Initialization of SPORT 3 to configure the codec
Configuration of the AD74111 via SPORT 3
Setting up the DMA Channels of the SPORTs:
SPORT 1 Receive
SPORT 3 Transmit

Figure 2: Data transmission flow diagram

First the AD74111 must be reset to put it into the correct state. Then the internal control registers of the codec are programmed via the SPORT 3 of the ADSP-21161. After this is done, the DMA channels of the DSP will be set up to receive data and transmit data.

The DSP receives the serial data from the codec with SPORT 1 channel A. The data is stored in the internal memory and send with SPORT 3 to the codec back. The user can modify the received data in the internal memory before sending it back.

The input data is provided to the codec over the ADC In connector on the evaluation board. The output data goes from the DAC Out connector to a loudspeaker.

Reset of the codec

To reset the codec in the right way is very important. The state of the DIN (data in) signal during the reset decides if the codec is in slave or in master mode. If DIN is high while the codec is reset, the codec will operate in slave mode. In slave mode DCLK and DFS are provided externally to the codec.

If DIN is low during reset the master mode will be selected. In master mode DCLK and DFS are generated from the codec. The reset timing is shown in Figure 3.

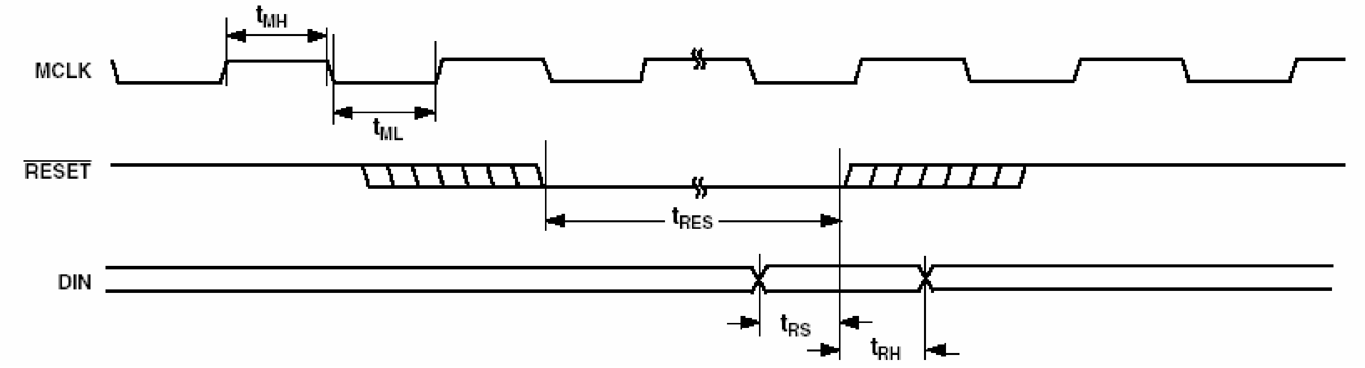

Figure 3: Reset timing of the codec

In this application Flag 0 is used to generate the reset signal for the codec. It is important to meet the timing (t_{RES}, t_{RS} and t_{RH}) to have defined reset provided for the codec.

The reset procedure takes 3072 MCLK (64ms) periods. It is not possible to write to the codec until this time is finished.

Figure 4 shows the reset timing of the codec taken with the scope.

Channel 1: DCLK

Channel 2: DFS

Channel 3: External Reset

Channel 4: DIN.

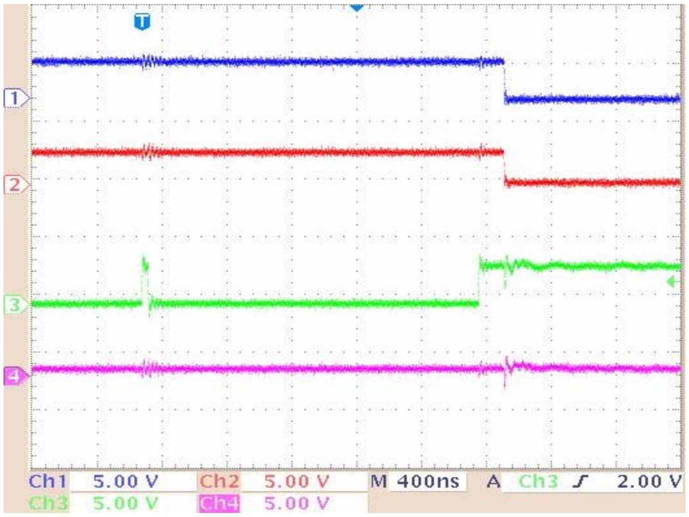

Figure 4: Reset timing of the AD74111

Modes of the AD74111 codec

There are two different modes of the AD74111 codec, mixed mode and data mode.

Mixed mode:

This is the default mode of the codec after power up or reset. In this mode the control registers of the codec can be written and read back. Figure 5 shows the timing of the mixed mode.

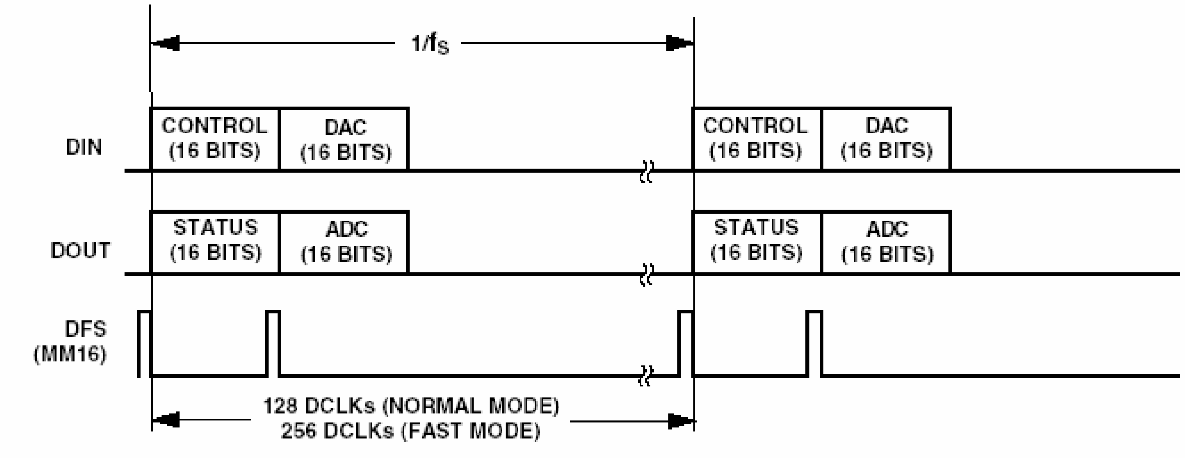

Figure 5: 16 Bit Mixed mode timing (word length = 16 Bit)

The control words must be send in the data slots. Also DAC information can be send to the codec. The codec delivers the status and ADC words which can be seen on the DOUT signal.

Data mode:

In this mode only data is sent on the DIN and DOUT pins. The user can switch to this mode, if no more configuration of the codec is required. Figure 6 shows the timing of the data mode. The only way to go back to mixed mode is, to perform a hardware reset.

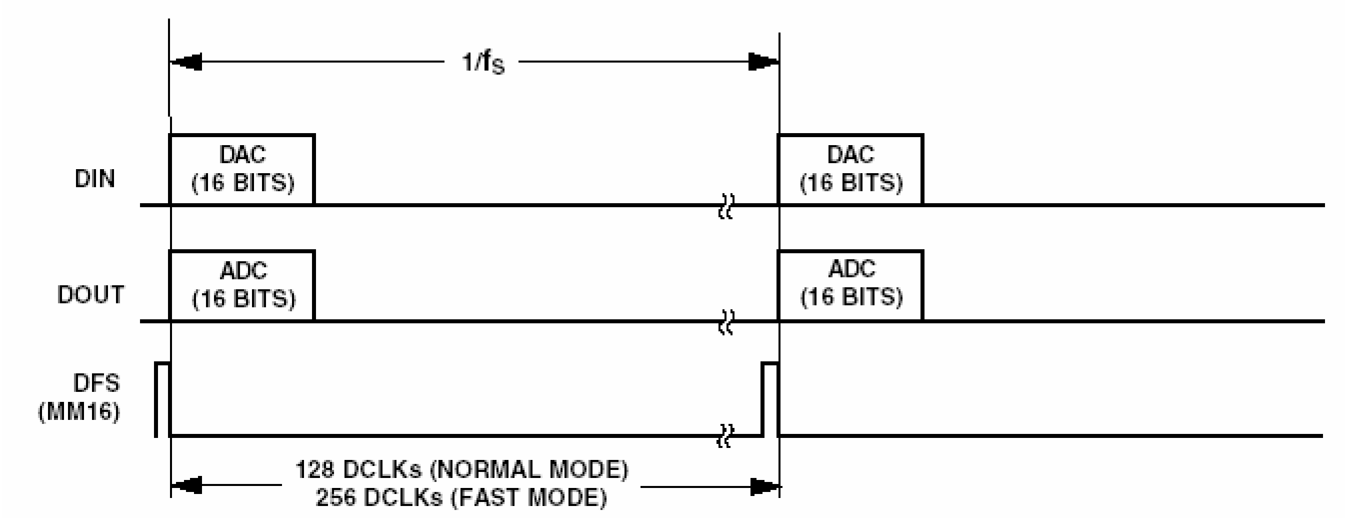

Figure 6: 16 Bit Data mode (word length = 16 Bit)

Configuration of the codec with SPORT 3

The ADSP-21161 sends the control words to the AD74111 to set it up. There are six control registers of the codec which can be programmed.

The register commands of the AD74111 look like in Table 3.

Bits	***Description***
15	Read or Write (0 = read, 1 = write)
14:11	Codec Register Address
10	Reserved bit, always 0
9:0	Data field

Table 3: Register commands of the AD74111

The configuration words of the registers are listed in Table 4:

Register	*Value*	*Description*
CR A	0x807C	ADC, DAC, ADC input and Reference Amplifier on
CR B	0x8800	First, Second and Third MCLK divider on
CR C	0x9000	ADC/DAC word width = 16 bit
CR D	0x9801	Data mode, DSP mode = 16 bit, Master
CR E	0xA000	ADC mute and DAC mute = normal, ADC gain = 0 dB
CR G	0xB000	DAC volume = 0 dBFS

Table 4: Configuration of the Control Registers of the AD74111

Control register F cannot be written, this is a read-only register. The control register D must be sent as the last word, because in this application the codec will be set in data mode. After switching to data mode, as mentioned in the chapter before, the codec cannot be configured any more.

Initialization of the SPORT 3 for codec configuration

The SPORT 3 is used to configure the codec. The definition of the bits which are set in the SPORT Control Register SPCTL3 are shown in Table 5.

Bit Name	*Bit definition*	*Description*
DDIR	Data direction control	SPORT 3 is set as transmitter
SLEN	Serial data word length	Data word length is 16 bits
SPEN_A	SPORT channel A enable	Enabling of the SPORT 3 channel A
SDEN_A	SPORT transmit DMA channel A	Enabling of the DMA on SPORT 3 channel A
FSR	Frame sync required	Data words are send on each frame sync

Table 5: Bit definition in SPCTL3

As the frame sync looks like in Figure 5 the SPORT could send after each frame sync the data words, but the control words must be send in the data slots. So, after each control word a dummy word is send to the codec (0x5555). The scope plot looks like in Figure 7.

Channel 1: DCLK

Channel 2: DSF

Channel 3: DOUT

Channel 4: DIN

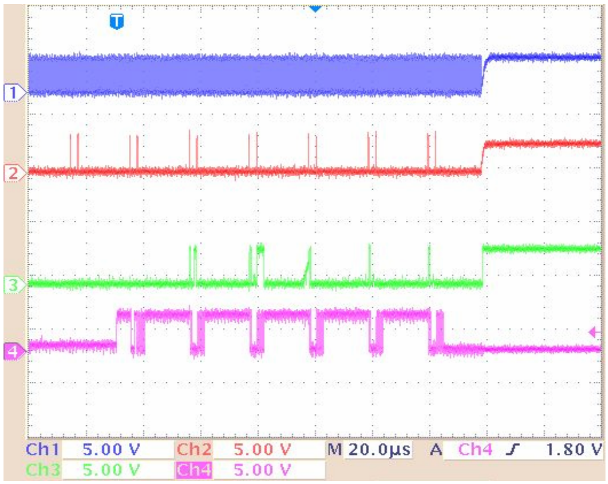

Figure 7: Sending the control words and the dummy words to the codec after latching a frame sync

After sending the control words the codec will be switched to the data mode.

Setting up the I/O Processor for chained SPORT DMA

The serial data transmission is done with the SPORTs 1 and 3 using chained DMA (direct memory access). Chained DMA is a automation of DMA transfers. After the completion of one DMA another DMA will be set off without any core intervention. The parameters of the DMA are stored in data buffers in the internal memory. This buffers are called Transfer Control Block (TCB). The I/O Processor loads the DMA channel parameters automatically from the TCBs when chaining is enabled.

The TCB format for the SPORTs looks as follows in Table 6.

SPORT TCB Registers	***Description***
GPx	General Purpose Register
CPx	Chain Point Register, points to the last address (IIx) of the next TCB to jump to upon completion of this TCB
Cx	Count Register, length of source buffer
IMx	Internal Modifier Register, source buffer step size
IIx	Internal Index Register, start address of the source buffer

Table 6: TCB format of the SPORTs

Since the codec has mono input and output channels, it is necessary to setup for SPORT 1 one TCB for channel A and for the SPORT 3 one TCB for channel A.

```
.var     sp1a_tcb[5] = 0, sp1a_tcb+4, N, 1, rx1a_buf;  /* SPORT 1 receive channel a tcb */
.var     sp3a_tcb[5] = 0, sp3a_tcb+4, N, 1, tx3a_buf;  /* SPORT 3 transmit channel a tcb */
```

Listing 1: Definition of the TCBs for SPORT 1 and SPORT 3

The first DMA in the chain is started by writing the address of the TCB to the Chain Pointer Register of the DMA channel.

```
r8=sp1a_tcb+4;          // Internal Index 1 address
r9=sp3a_tcb+4;          // Internal Index 3 address

dm(CP1A)=r8;            // SPORT 1 channel A DMA chain enable
dm(CP3A)=r9;            // SPORT 3 channel A DMA chain enable
```

Listing 2: Starting of the DMAs

Listing Setting up the SPORTs

After setting up the codec and the DMAs, it is necessary to set up the SPORTs. SPORT 1 is the receiver and SPORT 3 is the transmitter. The settings in the SPORT 1 control register are as follows:

```
// SPORT 1 control register set up as a receiver in normal serial mode
R0 = SCHEN_A | SDEN_A | SLEN16 | SPEN_A | FSR;
dm(SPCTL1) = R0;
```

Listing 3: Settings in the SPORT 1 control register

Note, that the channel B is not being used, because the codec delivers mono data, therefore only the parameters of channel A need to be configured (Table 7).

Bit Name	***Bit Definitions***	***Description***
SPEN_A	SPORT enable channel A	This bit enables the channel A of SPORT 1
SLEN	Serial Word Length	These five bits defines the word length of the data transmission. The codec sends 16 bit data to the DSP, so the word length is set to 16 bit.
SDEN_A	SPORT DMA enable for channel A (receive)	The serial data is received with channel A, so this bit must be enabled.
SCHEN_A	DMA chaining enable for channel A	This bit enables the DMA chaining for channel A of SPORT 1. If one DMA is finished the next one gets started.
FSR	Frame Sync Required	This bit selects, if the SPORT 1 requires a transfer frame sync or not.

Table 7: Bit settings of the serial port 1 control register (SPCTL1)

Similarly SPORT 3 control register is configured as follows:

```
// SPORT 3 control register set up as a transmitter in normal serial mode
R0 = SCHEN_A | SDEN_A | SLEN16 | SPEN_A | DDIR | FSR;
dm(SPCTL3) = R0;
```

Listing 4: Settings in SPORT 3 control register

Also the SPORT 3 is transmitting with channel A, because the codec has a mono input. (Table 8)

Bit Name	***Bit definition***	***Description***
SPEN_A	SPORT enable channel A	This bit enables the channel A of SPORT 3.
SDEN_A	SPORT DMA enable for channel A (transmit)	This bit enables the DMA for channel A of SPORT 3.
SCHEN_A	DMA chaining enable for channel A	This bit enables the DMA chaining for channel A of SPORT 3.
DDIR	Data Direction Control	This bit must be set now, because the SPORT 3 is transmitter.
SLEN	Serial Word Length	The ADSP-21161 sends via SPORT 3 data with 16 bit length to the codec, so the word length is set to 16 bit.
FSR	Frame Sync Required	This bit selects, if the SPORT 3 requires a transfer frame sync or not.

Table 8: Bit settings of the serial port 3 control register (SPCTL3)

To kick off the DMA chaining of the SPORTs, the DMA chain pointers must be written with the start address of the buffers.

```
/* Kick off DMA chaining of SPORT1&3 */

r8=sp1a_tcb+4;
r9=sp3a_tcb+4;

dm(CP1A)=r8;           // SPORT1 channel A DMA chain enable
dm(CP3A)=r9;           // SPORT3 channel A DMA chain enable
```

Listing 5: Starting of the DMA chaining

Interrupt service routines of the SPORTs

When the SPORT 1 receives the serial data from the ADC an interrupt will be generated. In the *Process_ _Audio_Samples* routine, the data will be read from the receive buffer of SPORT 1. After reading the data from the receive buffer, the data will then be written to the transmit buffer A of SPORT 3. Note that these data transfers are performed by the core (using the Data Address Generators - DAGs)

Once the data is in the transmit buffers, the serial data will get transmitted to the DACs.

```
Process_Audio_Samples:

R0=DM(I0,M0);                      // first value
LCNTR= N-1, DO (PC,1) UNTIL LCE;   // do the next instruction N-1 times
R0=DM(I0,M0); PM(I8,M8);           // read the data from the receive buffer a
                                   // write the data to the transmit buffer a
PM(I8,M8);                         // last value

rti;
```

Listing 6: SPORT 1 Interrupt

SPORT 3 Interrupt is being generated when data is transmitted with SPORT 3. In the interrupt service routine no special operation defined.

Figure 8 shows the data transmission between the ADSP-21161N SHARC DSP SPORTs and the AD74111 codec:

Channel 1: DCLK
Channel 2: DFS in data mode
Channel 3: DOUT
Channel 4: DIN

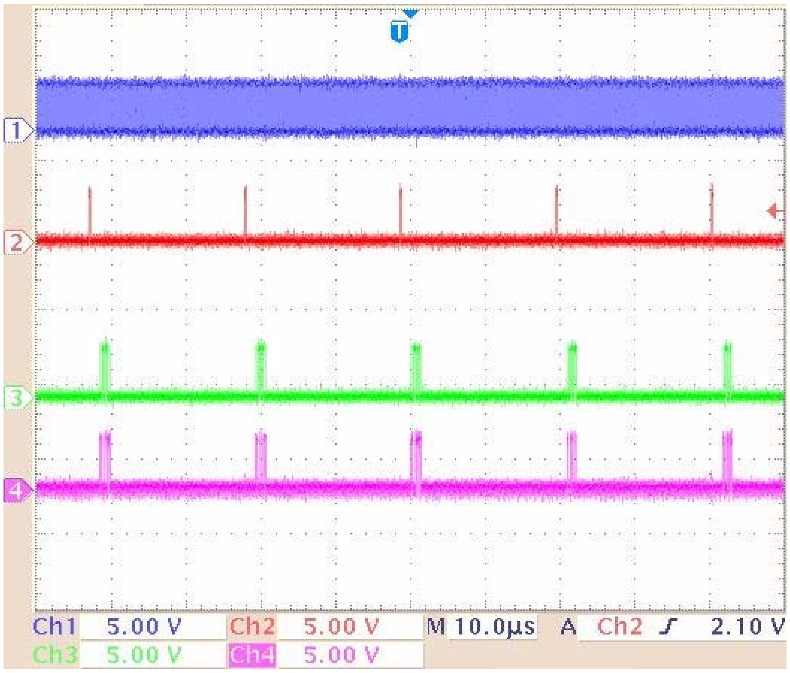

Figure 8: 16 Bit data mode (word length = 16 Bits)

References

[1] ADSP-21161 SHARC DSP Hardware Reference. Third Edition, May 2002. Analog Devices, Inc.

[2] ADSP-21161 SHARC DSP Microcomputer Datasheet. Rev. A, May 2003. Analog Devices, Inc.

[3] AD74111 Datasheet. Rev 0, 2003. Analog Devices, Inc.

[4] AD74111 Evaluation Board documentation. Rev 0, January 2003. Analog Devices, Inc.

[5] ADSP-21161N EZ-KIT Lite™ Evaluation System Manual. First Edition September 2001. Analog Devices, Inc.

[6] ADSP-21161N EZ-KIT Lite™ Schematics. Rev 2.3 July 2002. Analog Devices, Inc.

Document History

Version	Description
October 21, 2003 by Mazlum Adas	Initial Release