DSP 6701 Board

User Guide

Rev 1.01

Corporate Headquarters:
eInfochips Ltd.
11 A/B Chandra Colony,
Ellisbridge,
Ahmedabad 380 006
INDIA
Tel +91-79-2656 3705
Fax +91-79-2656 0722

USA Office:
eInfochips, Inc.
4655 Old Ironsides Drive,
Suite 385, Santa Clara,
CA 95054,
USA
Tel +1-408-496-1882
Fax +1-801-650-1480

Confidentiality Notice

CONTENTS

FIGURES

1. DOCUMENT DETAILS

1.1. Document History

Version	Date	Author	Description
Draft 0.01	15/06/2009	Amitkumar Solanki	Initial draft created
Draft 0.02	16/06/2009	Amitkumar Solanki	Updated after review comments by Pratik
Draft 0.03	17/06/2009	Rizwan	Hardware details updated
Draft 0.04	26/06/2009	Rizwan	Section 2.4 added for the clock setting of the DAC_OUT clock
Draft 0.05	30/06/2009	Rizwan	Section 2.1 added for the Basic Board Details
Rev 1.00	03/07/2009	Rizwan	Document Base lined
Rev 1.01	17/07/2009	Rizwan	Specification table added.

1.2. Acronyms

Acronym	Description
ADC	Analog to Digital Converter
CPU	Central Processing Unit
DAC	Digital to Analog Converter
DSP	Digital Signal Processor
FIFO	First In First Out
FPGA	Field Programmable Gate Array
IRAM	Internal Random Access Memory
SDRAM	Synchronous Dynamic Random Access Memory
SBSRAM	Synchronous Burst Static Random Access Memory

2. INTRODUCTION

DSP C6701 system board is a floating point DSP C6701 based board for digital processing of analog signal acquired through an ADC EVM and then generating the processed analog output through a DAC EVM board. The DSP board seamlessly interfaces to ADC EVM and DAC EVM for high speed signal acquisition using Virtex IV FPGA.

Input analog signal is acquired into the DSP using ADC clock, wherein the digital data is processed and analog output is generated using the DSP clock as the master clock. ADC sampling clock is rated up to 105MHz and DSP clock is set at 117MHz, resulting in a reasonably high rate of analog signal acquisition.

The selected DSP and FPGA can be replaced easily with Rad tolerant devices for space grade applications.

Specifications :

Board Dimensions	8.500 inch x 4.910 inch
DSP	Texas Instrument's **SM320C6701** Floating Point DSP, Clock Frequency 116.9678 MHz (Max frequency Supported: 140MHz)
Memory	Program Memory : **2MB (1Mbit x 16)** NOR Flash,Spansion's S29AL016J70TFI010 System Memory : **8MB (2Mbit x 32)** Micron's MT48LC2M32B2TG-7 IT:G
FPGA	Xilinx **virtex-4 FPGA XC4VLX25-10SF363C**, with **Xilinx Platform EEPROM of 4MB** , XCF08P-FSG48C for configuration of FPGA
Oscillator	OSC1 :ASV-29.4912MHZ-EJ-T of 29.4912MHz(Frequency selected to match baud rate accurately)
Power Input	5.0V to 12V DC, 4W Power
SDRAM Clock Speed	1/2 CPU Clock Frequency (58.48 MHz)
SBS Interface	1/2 CPU Clock Frequency (58.48 MHz) & CPU Clock Speed
GPIO	4- General Purpose Output of FPGA J18 4- General Purpose Input of FPGA J17

Debugging	JTAG interface to SM320C6701 DSP (IEEE 1149.1)
FPGA bit file Download	JTAG interface to FPGA daisy chained with Platform EPROM
Asynch Interface	Provision of Asynchronous interface between FPGA and DSP
Operating Temperature	0 degC to +85 degC
Logic Analyzer Test Port	J2- DAC2 LVDS signal Test Connector J16- DAC1 CMOS signal Test Connector J22- ADC Signal Test Connector

3. HARDWARE DETAILS

3.1 Basic Board Details

1. Board Dimension : 8.500 inch x 4.910 inch
2. Peripheral Details
    a. CPU : Floating Point DSP **SM320C6701GLPW14** , also available in the Rad tolerant part
    b. Max Clock Frequency Supported by DSP : **140 MHz**
    c. Clock Frequency on the SM320C6701 DSP Board : **117.9648 MHz in 4x mode**. Oscillator used on the board is of **29.4912 MHz**
    d. System Memory : **8MB** (2Mbit x 32), Micron's MT48LC2M32B2TG-7 IT:G TR with 5.5ns CAS latency 2ns setup and 1ns hold time. On SM320C6701 DSP Board clock speed of the SDRAM is **117.9648 / 2 = 58.9842MHz**
    e. Program Memory : **2MB NOR Flash** (1Mbit x 16) Spansion's S29AL016J70TFI010
    f. FPGA : Xilinx virtex-4 FPGA **XC4VLX25-10SF363C** working at CPU clock speed of **117.9648 MHz,** with Xilinx Platform EEPROM of 4MB , **XCF08P-FSG48C** for configuration of FPGA.
3. Power Supply Requirement
    a. Power in through 2.5mm DC Jack from 5.0V to 12V DC, 4W Power
    b. On board generated power supply : +1.2V, +3.3V, +1.9V, +2.5V each designed to provide 2A current
4. Major Connector Interface :
    a. 40 pin connector for ADC Board EVM Interface with 40 pin test Pin
    b. Connector (40pin & 60pin-LVDS) for DAC EVM Board with test pin
    c. Spare Pin of FPGA for general purpose input and output

3.2 Interface Details

1. DSP
    a. SM320C6701, Floating Point DSP (also available in the Rad tolerant part), rated for Max. 140 MHz and tested up to 120MHz

2. Memory
    a. Micron MT48LC2M32B2TG-7 :G TR SDRAM , 8MB (2M x 32bit)
    b. Spansion™ S29AL016J70TFI010 NOR Flash 2MB (1Mx 16bit)

3. FPGA
    a. Xilinx Virtex-4 FPGA XC4LVX25-105F363C
    b. Xilinx Platform EEPROM XCF08P-FSG484Cfor Bit File loading

4. Connectors
    a. 40 pin dual raw right angle connector J8 for the ADC board
    b. 40 pin dual raw connector J23 for DAC1(CMOS-Single ended) 60 pin dual raw connector J21 for DAC2(Law voltage differential).
    c. 40 pin dual raw connector J22 for testing ADC logic signal into the logic analyzer

d. 40 pin dual raw connector J16 for testing DAC1(CMOS-Single ended) logic signal in to the logic analyzer, 60 pin dual raw connector J3 for testing DAC2(Law voltage differential) logic signals in to the logic analyzer.
e. DSP JTAG connector J9 for, connecting the DSP c6701 to CCS for debugging
f. FPGA JTAG connector J15 for bit file loading in to the FPGA and programming the platform flash
g. General purpose input (GPI) pin connector J17. GPI Connected to the FPGA
h. General purpose output (GPO) pin connector J18. GPO Connected to the FPGA

5. SMA Connector Details for the High frequency Clocks
   a. J1: Right angle SMA connector for the External clock input for the DSP
   b. J4,J5: Vertical SMA Connector for the LVDS Clock out to the DAC Board
   c. J12: Vertical SMA Connector for the CMOS clock out to the DAC Board
   d. J21: Right angle SMA Connector for the input to the ADC Board.

6. Switches
   a. Clock mode switch SW1
   b. Boot mode selection switch
   c. FPGA boot mode selection switch

7. Jumper
   a. J3 Clock in selection Jumper
   b. J6 General purpose input pin the CPU
   c. J7 Little Endian and Big Endian selection Jumper
   d. J11 Boundary scan selction Jumper
   e. J13 DSP Sync/Aync mode select jumper
   f. J14 FPGA Master input clock selection details
   g. J15 HSWAPEN selection of the FPGA

3.3 Switch Connection Details

1. DSP Boot mode selection switch SW2

Boot Mode Selection Switch (SWITCH ON = 1)						
1	**2**	**3**	**4**	**5**	**Memory at Adress 0**	**Boot Process**
BOOTMODE4	**BOOTMODE3**	**BOOTMODE2**	**BOOTMODE1**	**BOOTMODE0**		
0	0	1	0	1	Internal	None
0	0	1	1	1	Internal	Host boot HPI
0	1	1	0	1	Internal	8-bit ROM
1	0	1	0	1	Internal	16-bit ROM (Default)
1	1	1	0	1	Internal	32-bit ROM

2. Clock Mode and PLL selection switch SW1

Clock Mode

Clock Mode selection Switch (Switch ON=1)		
5	4	CLKOUT1 Description
CLKMOD1	CLKMOD0	
0	0	1 x f(CLKIN)
0	1	Reserved
1	0	Reserved
1	1	4 x f(CLKIN) (Default)

PLL Mode

PLL Frequency Select (Swtich ON=1)			
3	2	1	CLKOUT1 Frequency Range (MHz)
PLLFREQ3	PLLFREQ2	PLLFREQ1	
0	0	0	50 – 140
0	0	1	65 – 167
0	1	0	130 - 167 (Default)

3. FPGA MODE Select Switch SW3

FPGA Boot Mode Select Switch (Swtich ON = 1)					
3	2	1	Configuration Mode	DATA width	CCLK Direction
MODE2	MODE1	MODE0			
0	0	0	Master Serial	1	Output
1	1	1	Slave Serial	1	Input
0	1	1	Master SelectMAP	8	Output
1	1	0	Slave SelectMAP8	8	Input
0	0	1	Slave SelectMAP32	32	Input
1	0	1	JTAG Mode only	1	NA

3.4 Jumper setting

1. J3 – Clock Select
   Jumper to select the Clock input to the DSP c6701 from external clock to internal clock if jumper is between,
   1-2 → Internal Clock Source

2-3 → External Clock Source

2. J6- General Purpose Input of the DSP C6701
   Jumper to test the GPI of the DSP. Two GPI pin of DSP connected to this Jumper. This pin are TINP1 and TINP0. This jumper can also be used to give input to the timer of the DSP. If jumper is not sorted to any pin than both this pin are in logic high state. If Jumper is between,

   1-2 → TINP1 in Logic Zero state
   2-3 → TINP0 in Logic Zero state

3. J7- Little Endian Or Big endian selection Jumper
   This jumper is to select DSP mode Little Endian or Big Endian if jumper is between ,

   1-2 → Little Endian
   2-3 → Big Endian
4. J11 – Boundary scan Jumper
   Jumper is provided to pull the DSP_EMU[0:1] to ground. Both this pin are High but is required than it can be pulled low. If Jumper is between,

   1-2 → DSP EMU1 is Logic low
   2-3 → DSP EMU0 is Logic low
5. J13 – DSP sync/Async Mode selection Jumper
   Jumper for selecting whether FPGA To use in Synchronous mode or Asynchronous mode to communicate with DSP. If jumper is not populated that it is in the Asynchronous mode. This setting basically depends upon the FPGA bit file used in the FPGA, for current bit file used, If jumper is between

   2-3 → Synchronous mode select
6. J14 - FPGA Master Input select Jumper
   Jumper to select the master input clock of the FPGA. If jumper is not used then, it will select DSP. This setting basically depends upon the FPGA bit file used in the FPGA, for current bit file used, If jumper is between ,

   2-3 → FPGA Master input Clock from the DSP
7. J10 – HSWAP Pin Pull down jumper
   FPGA pull up Disable/Enable Pin HSWAP pin can be pull up and pull down with the help of this Jumper. If the jumper is between ,
   1-2 → I/O pull-up resistors disabled
   2-3 → I/O pull-up resistors enabled

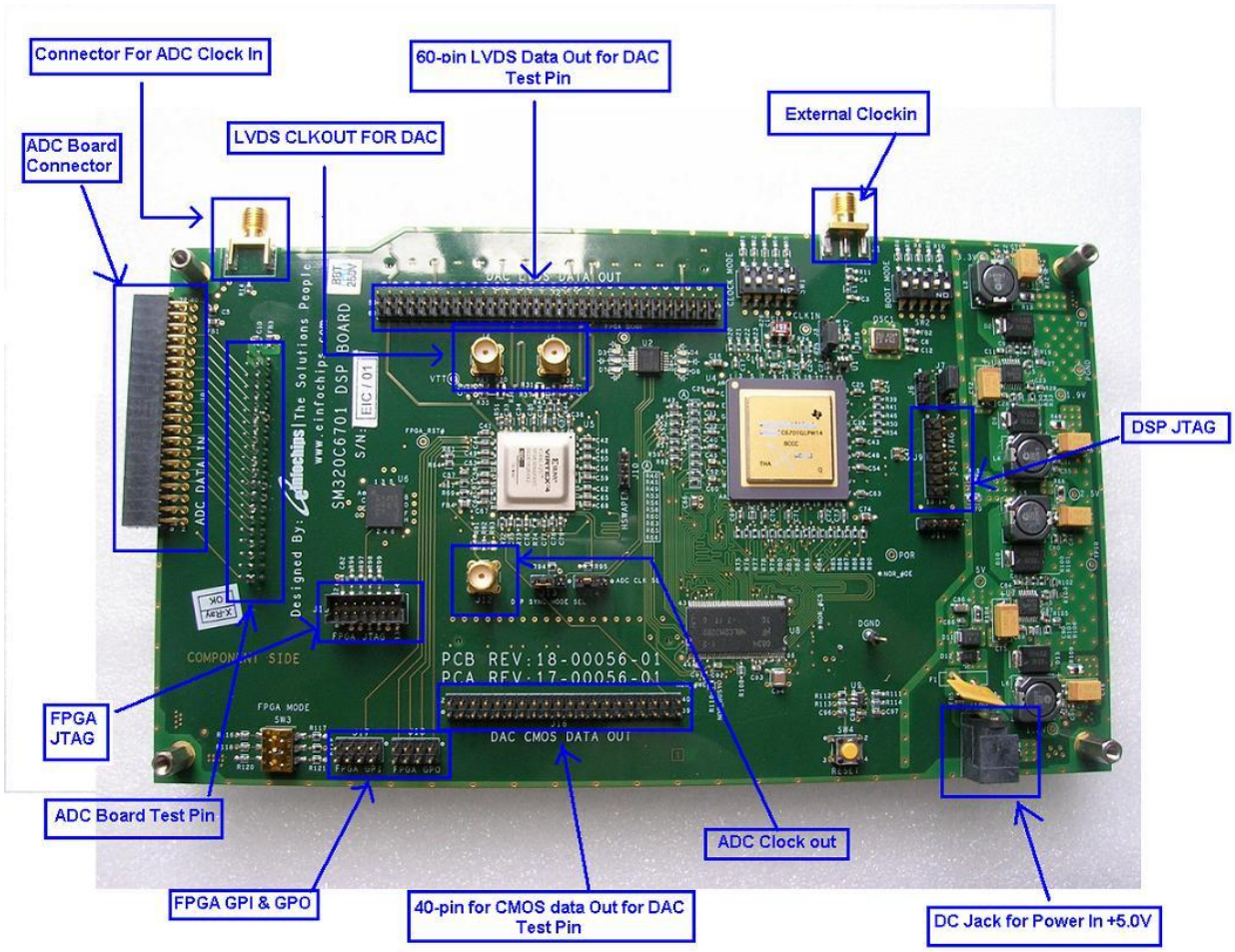

Figure 1: DSP Board Connector Details

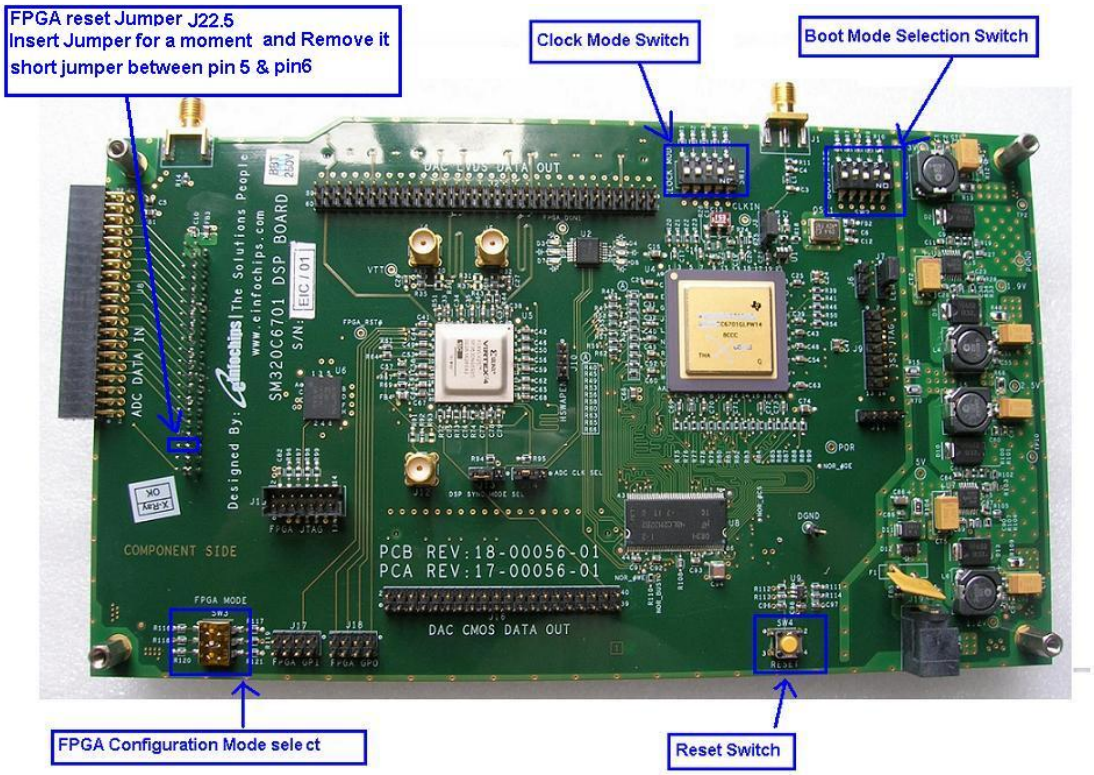

Figure 2: DSP Board Switch Details

3.5 DSP Board Interconnection with ADC Board and DAC Board

Please refer to figure 3 for the complete system connection with ADC and DAC with DSP Board. The details of the ADC board and DAC Board can be found the respective user guide for the same.

User Guide of the ADC Board ADS 5424 EVM Board:
http://focus.ti.com/general/docs/lit/getliterature.tsp?literatureNumber=slwu020b&fileType=pdf
User Guide of the ADC Board DAC 5675 AEVM Board:
http://focus.ti.com/lit/ug/slau080a/slau080a.pdf

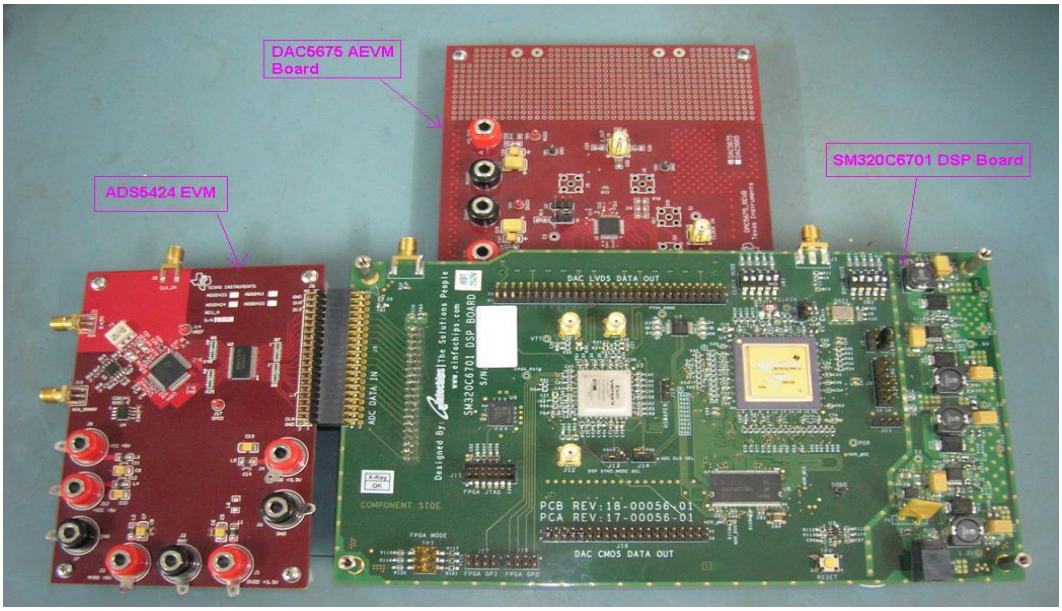

Figure 3 : DSP Board interconnection with ADC and DAC board

3.6 DAC OUTPUT Clock selection Setting

Table : Information of Populated and Non-Populated Components for DAC output clock

CLOCK CONFIGURATION	COMPONENTS INSTALLED	COMPONENTS NOT INSTALLED
Single Ended (Default)-DAC1	J12, J4, J5, R92, R91	R93, R36, R30, R35, R29 R31, R37, R32, R38
Differential (LVDS)-DAC2	J12, J4, J5 R30, R31, R35,R38	R36, R37, R93 R32, R29
Differential (CMOS)-DAC2	J12, J4, J5 R32, R29, R36,R37	R30, R31 R35, R38

3.7 Test Pint Details

Following are the through hole test points available on the SM320C6701 DSP Board

Table: List of the Test Points

#	TP No.	Description
1	TP1	VCC_3V3
2	TP2	BP_5V- 1
3	TP3	Clock In (Input Clock to DSP)
4	TP4	FPGA_DONE
5	TP5	PGND
6	TP6	DAC_CLKA_400MHz
7	TP7	+1.9V
8	TP8	FPGA_RST#
9	TP9	+2.5V
10	TP10	BP_5V- 2
11	TP11	VCC_5V
12	TP12	DGND
13	TP13	+1.2V
14	TP14	POR#

4. FPGA ADDRESS SPACE

FPGA is connected to DSP on CE -0 (SBSRAM interface) and CE -2 (Async interface) memory spaces. In case FPGA is configured to operate in SBSRAM interface, its base address will be 0h'0x00400000 and otherwise its base address will be 0h'0x02000000

4.1 FPGA Memory Map

The FPGA memory map to access FPGA resources is tabulated as follows:

	FPGA Memory Address
Configuration Registers	0x000000
ADC FIFO	0x040000
DAC1 FIFO	0x0C0000
DAC2 FIFO	0x080000

To access the ADC FIFO, add ADC FIFO address offset to FPGA base address.

For example: Here FPGA base address is 0x02000000 then ADC FIFO address = FPGA base address + ADC FIFO offset

i.e. ADC FIFO address = 0x02000000 + 0x040000
= 0x02040000

4.2 FPGA Configuration Register Memory Map

The address offsets of FPGA configuration registers are as follows:

Register	**Configuration memory offset**
Configuration register	0x0000
Status register	0x0010
Test register	0x0040
Interrupt status register	0x00C0
Version register	0x00D0
General Purpose register	0x00E0
Fixed pattern register	0x00F0

To access any of the register, add the register offset to FPGA base address.

For example: To access version register, the address = FPGA base address + + Version register offset.

i.e. Version no Register address = 0x02000000 + 0x00D0
= 0x020000D0

5. CCS GEL FILE

A CCS gel file DSP6701.gel is provided with DSP 6701 board for connecting and configuring the board through CCS

- The gel file configures the DSP chip enables as following:
  1. CE 0 - FPGA SBSRAM interface
  2. CE 1 - 16-bit NOR flash
  3. CE 2 - FPGA Asynchronous interface
  4. CE 3 – 32-bit SDRAM
- For details of EMIF CE registers configuration please refer SPRUE266E: TMS320C6000 DSP External Memory Interface (EMIF) Reference Guide.

6. SOFTWARE TEST UTILITIES

For all the test utilities using FPGA, please ensure following:

- Interface of communication between FPGA and DSP is selected properly. Depending upon DSP SYNC MODE SEL jumper (J13), the SBSRAM macro in *fpga.h* should be defined, 1 for SBSRAM mode and 0 for Asynchronous mode.

- Ensure that ADC CLK SEL jumper (J14) is set to use DSP clock for writing data out of DAC1 FIFO.

- BOOT MODE switch (SW2) is set to its default value (10101b). This will select Map 1 for DSP 6701.

The following utilities are provided with DSP 6701 board:

1. NOR flash test utility
2. Digital loopback test utility
3. FPGA Interrupts test utility

6.1 NOR flash test utility

6.1.1 Objective

This utility tests read, write and erase of 2MB of NOR flash on DSP 6701 board. In addition to that, it also verifies EMIF in async mode.

6.1.2 Test Setup

Connect the DSP 6701 board to PC using a USB JTAG emulator.

6.1.3 Configuration

Before building this application please ensures that NOR_TEST macro is defined in *NORFlashUtility\src\writer.c*. By default this macro is defined.

6.1.4 Description

- This test utility initializes CE1 address space for 16-bit NOR flash with the following values for both write and read:
  - Setup = 4 CPU cycles
  - Strobe = 8 CPU cycles
  - Hold = 3 CPU cycles
- This test utility initializes a 2MB buffer and populates it with a 32-bit counter. Each sector of the NOR flash is erased sector wise.. The utility then writes the buffer to the NOR flash.

- The NOR flash is then read back and verified with the the data pattern (incremental counter) written.

6.1.5 Steps

Build the project, load and execute the binary file on the board.

Note: For CCS related help please visit http://focus.ti.com/lit/ug/spru509h/spru509h.pdf

6.1.6 Expected output

The test should pass with the following message "NOR flash written successfully" on standard output window of CCS. In case of failure the following message "NOR Flash Verify Failed @ "will be printed with corresponding fail address.

6.2 Limitations

None

6.3 Digital loopback test utility

6.3.1 Objective

This test verifies 2 paths, DSP => DAC1 and ADC => DSP using external data loopback from DAC1 output to ADC input as shown in figure 1.

6.3.2 Test Setup

Connect the test board to the DSP 6701 board, this will externally loopback DAC1 output to ADC input. Connect the DSP 6701 board to PC using a USB JTAG emulator.

6.3.3 Configuration

Before building this application please ensures the SBSRAM macro in fpga.h is properly defined. BUF_SIZE_WORDS (default 512) is the number of words written to DAC1 FIFO. BUF_SIZE_WORDS is defined in main.c. Its maximum value is 8K which is the size of the DAC1 FIFO.

6.3.4 Description

This utility enables DAC1 FIFO and ADC FIFO. The utility populates the buffer incrementally. The utility then writes BUF_SIZE_WORDS words to the DAC1 FIFO and reads 64K bytes from the ADC FIFO. This data is than dumped in to a file *ADCDump.bin*.

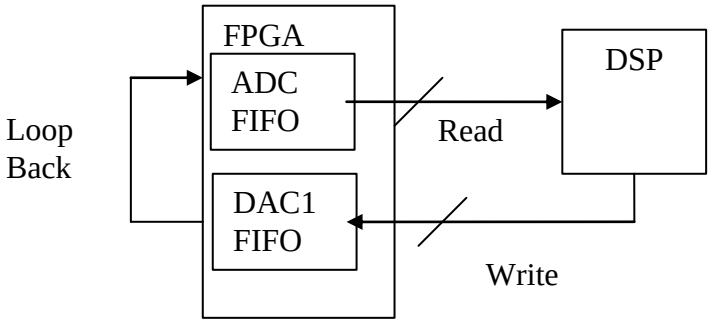

Figure 4: Digital Loopback

6.3.5 Steps

- Build the project, load and execute the binary file on the board.
- The user can open the dump file located in the "Debug" folder of the project in a hex editor and compare the data sequence written to DAC1 with the dump.

Note: For CCS related help please visit http://focus.ti.com/lit/ug/spru509h/spru509h.pdf

6.3.6 Expected output

After some unknown values, the dump should start with 0x0000 and have repeated odd values, i.e. 0x0001, 0x0003, 0x0005, etc.
The reason for this is, FPGA drives the previous 14-bits on the DAC connector until new data arrives. As the write to DAC1 FIFO is slower than the FPGA latching data on ADC connector, the upper 16-bits sequence of the 32-bits written to DAC1 FIFO is repeated multiple times in the dump

6.3.7 Limitations

None

6.4 FPGA Interrupts test utility

6.4.1 Objective

This utility can be used to test the interrupt conditions generated from FPGA.

6.4.2 Test Setup

Connect the test board to the DSP 6701 board. Connect the DSP 6701 board to PC using a USB JTAG emulator.

Note: For CCS related help please visit http://focus.ti.com/lit/ug/spru509h/spru509h.pdf

6.4.3 Configuration

Before building this application please ensures the SBSRAM macro in fpga.h is properly defined.

6.4.4 Description

Depending upon the choice of the user, the utility enables the particular interrupt in the FPGA control register and generates an interrupt condition.
The DSP catches the interrupt and executes the ISR (Interrupt Service Routine). The interrupt service routine will read the Interrupt Status Register of FPGA to identify the interrupt generated and print the appropriate message on standard output window of CCS.

Overflow and underflow is defined as under:
Overflow: FIFO overflow occurs when FIFO is full and a write to it is attempted.
Underflow: FIFO underflow occurs when an empty FIFO is read.

6.4.5 Steps

- Build the project, load and execute the binary file on the board
- The utility prompts the user for testing a particular interrupt. The user has the following (1 - 6) options:
  1. ADC underflow
  2. ADC overflow
  3. DAC1 empty
  4. DAC1 overflow
  5. DAC2 empty
  6. DAC2 overflow
- User prompt is shown in Figure 2.

Note: For CCS related help please visit http://focus.ti.com/lit/ug/spru509h/spru509h.pdf

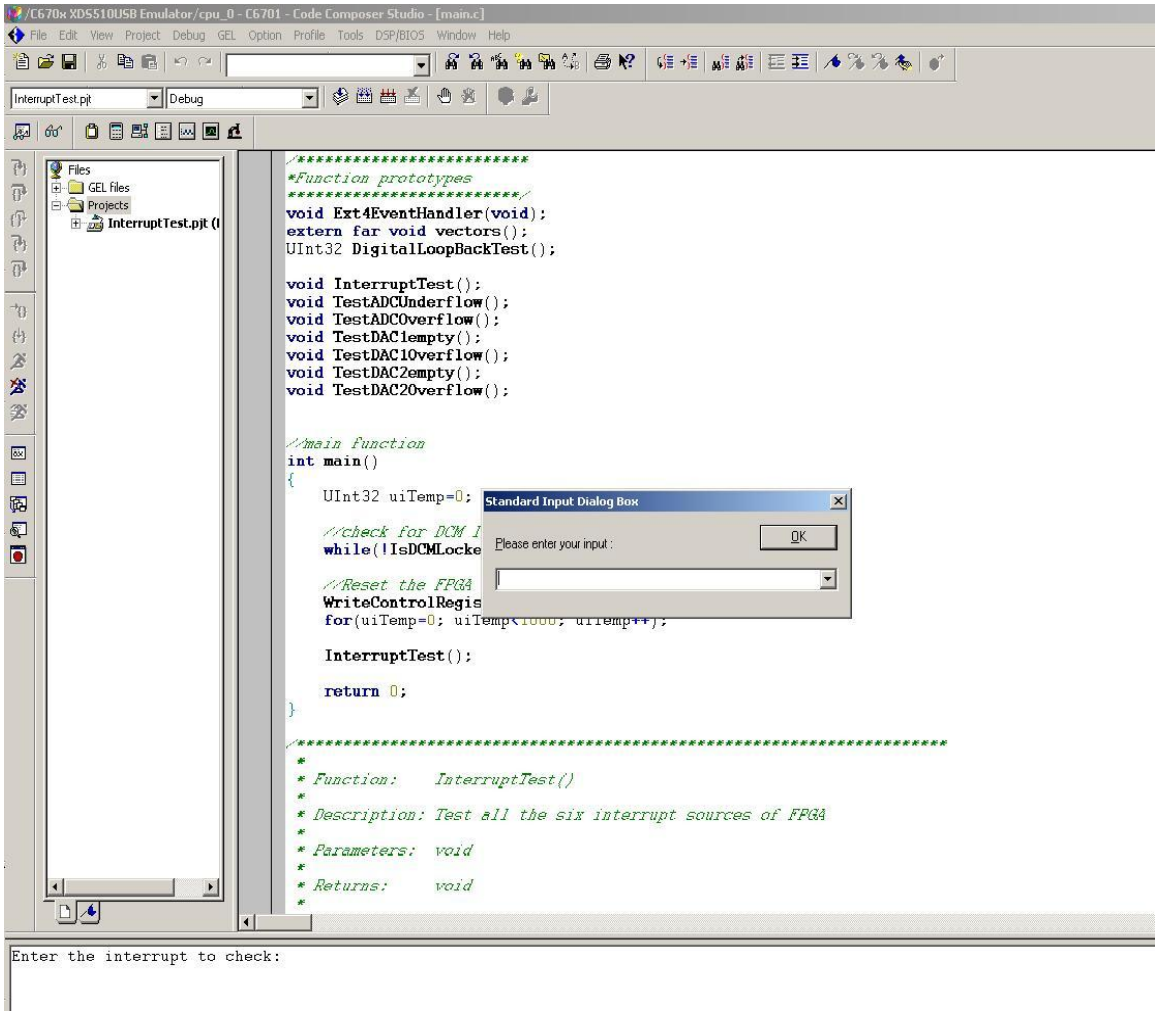

Figure 5: Interrupt testing snapshot

6.4.6 Expected output

Depending upon the user's choice a message related to that particular interrupt is printed on the standard output window of CCS.

6.4.7 Limitation

Please note that under present conditions following interrupt conditions are not going to occur:

- ADC underflow for both SBSRAM and asynchronous interface- As ADC latches data at a rate faster than the rate at which DSP reads the ADC FIFO DSP will never observe ADC underflow.
- DAC2 overflow for SBSRAM interface. – As FPGA writing rate to DAC2 FIFO is slower than DAC2 output rate in SBSRAM mode

6.5 ADC-DAC Signal Acquire/Reproduce Test

External loopback implies, Analog signal is received on external ADC board. This ADC board converts the analog data in to digital form and sends it to FPGA on sampling clock of ADC. DSP reads this data from FPGA fifo using DMA channel – 1 in 1K sample chunk. DMA channel -2 writes this data back to DAC2 fifo in 1K sample chunk. DAC should be connected to DAC port to see reproduced data at the output of external DAC board.

6.5.1 Objective

To acquire and reproduce analog signal using external ADC and DAC boas provided by TI.

6.5.2 Test Setup

Connect the ADC board and DAC board to the DSP 6701 board. Connect the DSP 6701 board to PC using a USB JTAG emulator.
5.4.3 Before building this application please ensures the SBSRAM macro in fpga.h is properly defined.

ELEMENT_COUNT (default 1024) is the number of samples written to DAC2 FIFO. ELEMENT_COUNT is defined in main.c. Its maximum value is 32K which is the size of the ADC fifo.

6.5.3 Description

In this utility DMA Channel – 1 is configured for ADC read and DMA Channel 2 is configured for writing to DAC2. On receiving frame completion interrupt from any of the channel, the other channel is enabled. For example, if ADC read channel gives interrupt, DAC2 DMA channel is enabled to write captured data to DAC2 fifo.

6.5.4 Steps

- Build the project, load and execute the binary file on the board.

Note: For CCS related help please visit http://focus.ti.com/lit/ug/spru509h/spru509h.pdf

6.5.5 Expected output

Give a known i/p (sine wave) to ADC board input. Check o/p on DAC board. You should be able to see reproduction of i/p singal with some delay due to process in DSP and FPGA.

6.5.6 Limitations

None