Unit / Module Description:	SMT384 - Quad 125MSPS module
Unit / Module Number:	SMT384
Document Issue Number:	2
Issue Date:	15/12/2010
Original Author:	Philippe Robert

Application Note for SMT384

Sundance Multiprocessor Technology Ltd, Chiltern House, Waterside, Chesham, Bucks. HP5 1PS.

Revision History

Issue	Changes Made	Date	Initials
1	Original Version released	14/12/2010	PhSR
2	FPGA firmware update. Now adds link USB-Flash(SMT372T)	15/12/2010	PhSR

Table of Contents

1 Introduction

The purpose of this application note is to show how an SMT384 can be used when coupled with an EVP6472 platform.

The EVP6472 platform is composed of an SMT111 (single TIM carrier module) and an SMT372T, which is a multi-core DSP and FPGA processor board based on two of TI's TMS320C6472 DSPs and one Xilinx Virtex5 FX30T FPGA (-1 speed grade).

Such system comes with an example code for the FPGA (Virtex 5 on SMT372T) and the DSPs (C6472).

The SMT384 is a quad 125-MSPS ADC module that can plug on top of the SMT372T. Converters are 14-bit. The board has got a reference circuitry that can be used to lock the on-board VCXO (245.76 MHz).

The FPGA firmware has been compiled with ISE12.2 and the DSP code has been compiled with Code Composer 4.1.2. The XDS560 is used as an emulator so Code Composer can connect to the multi-core DSPs.

DSPs (SMT372T) are programmed via JTAG (XDS560 emulator). The FPGA (SMT372T) can be programmed either by JTAG (Xilinx programming cable – USB) or by booting from flash memory.

2 Related Documents

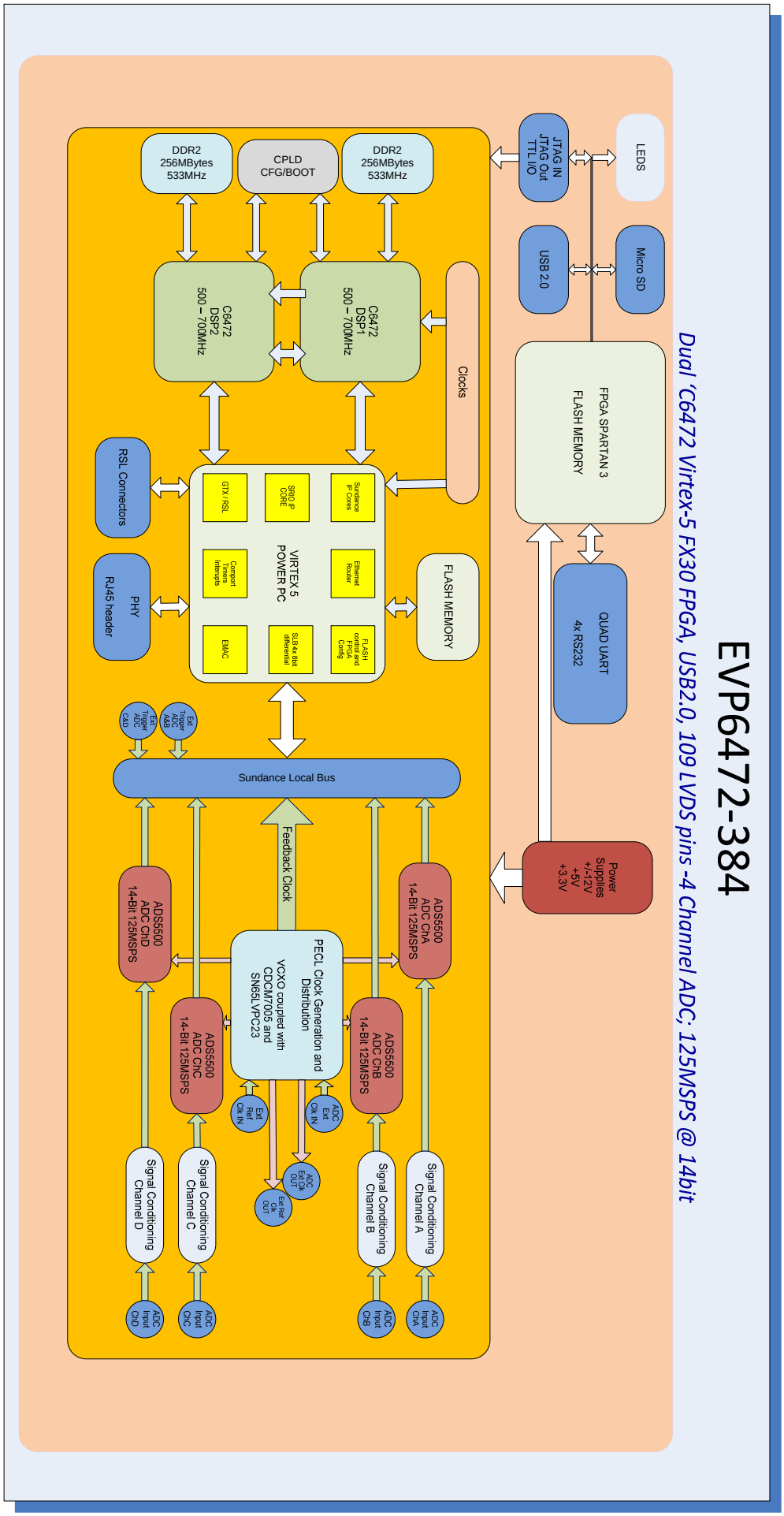
We encourage you to take a look at all the documents listed below in order to have a good understanding of the architecture of all modules:

- SMT384,
- SMT372T,
- SMT111,
- EVP6472,
- EVP6472.com website,
- Sundance SLB mezzanines,
- How to assemble an EVP6472,
- Using the EVP6472 with Code Composer4 (Video),
- SMT6002 – Flash Utility for FPGA modules.

3 Block Diagram

The block diagram shows the carrier board (***SMT111***), the SLB base (***SMT372T***) and the SLB mezzanine modules (***SMT384***).

Below is shown a block diagram of the entire system.


EVP6472-384
Dual 'C6472 Virtex-5 FX30 FPGA, USB2.0, 109 LVDS pins -4 Channel ADC; 125MSPS @ 14bit
LEDS
JTAG IN
JTAG Out
TTL I/O
Micro SD
USB 2.0
FPGA SPARTAN 3
FLASH MEMORY
QUAD UART
4x RS232
Power Supplies
+/-12V
+5V
+3.3V
DDR2
256MBytes
533MHz
CPLD
CFG/BOOT
DDR2
256MBytes
533MHz
C6472
DSP2
500 – 700MHz
C6472
DSP1
500 – 700MHz
Clocks
RSL Connectors
PHY
RJ45 header
VIRTEX 5
POWER PC
GTX / RSL
SRIO IP CORE
Sundance IP Cores
Compact Timers Interrupts
Ethernet Router
EMAC
SL8.4x 4bit ethernet hub
FLASH control and FPGA Config
FLASH MEMORY
Ext Trigger ADC A&B
Ext Trigger ADC C&D
Sundance Local Bus
Feedback Clock
PECL Clock Generation and Distribution
VCXO coupled with CDCM7005 and SN65LVPC23
ADS5500 ADC ChA 14-Bit 125MSPS
ADS5500 ADC ChB 14-Bit 125MSPS
ADS5500 ADC ChC 14-Bit 125MSPS
ADS5500 ADC ChD 14-Bit 125MSPS
Ext Ref Clk IN
ADC Ext Clk IN
ADC Ext Clk OUT
Ext Ref Clk OUT
Signal Conditioning Channel A
Signal Conditioning Channel B
Signal Conditioning Channel C
Signal Conditioning Channel D
ADC Input ChA
ADC Input ChB
ADC Input ChC
ADC Input ChD

4 Software Setup.

Code Composer from Texas Instrument needs to be installed (version 4.1.2 is the version on which the code has been developed) and Xilinx ISE (version 12.2 is the version that has been used to develop the firmware).

SMT6002 might also be required if it is intended to program the flash memory on the SMT372T via USB (SMT111).

5 Hardware Setup

The SMT372T is mounted on the SMT111 to make the EVP6472 – you can refer to the following video for more details: Sundance SLB mezzanines.

Prior to adding the SMT384 to the system, double check that the SLB IO voltages are set to 3.3V. They are jumpers ***JP1 and JP2*** on the SMT372T and they should be on ***position 1-2***.

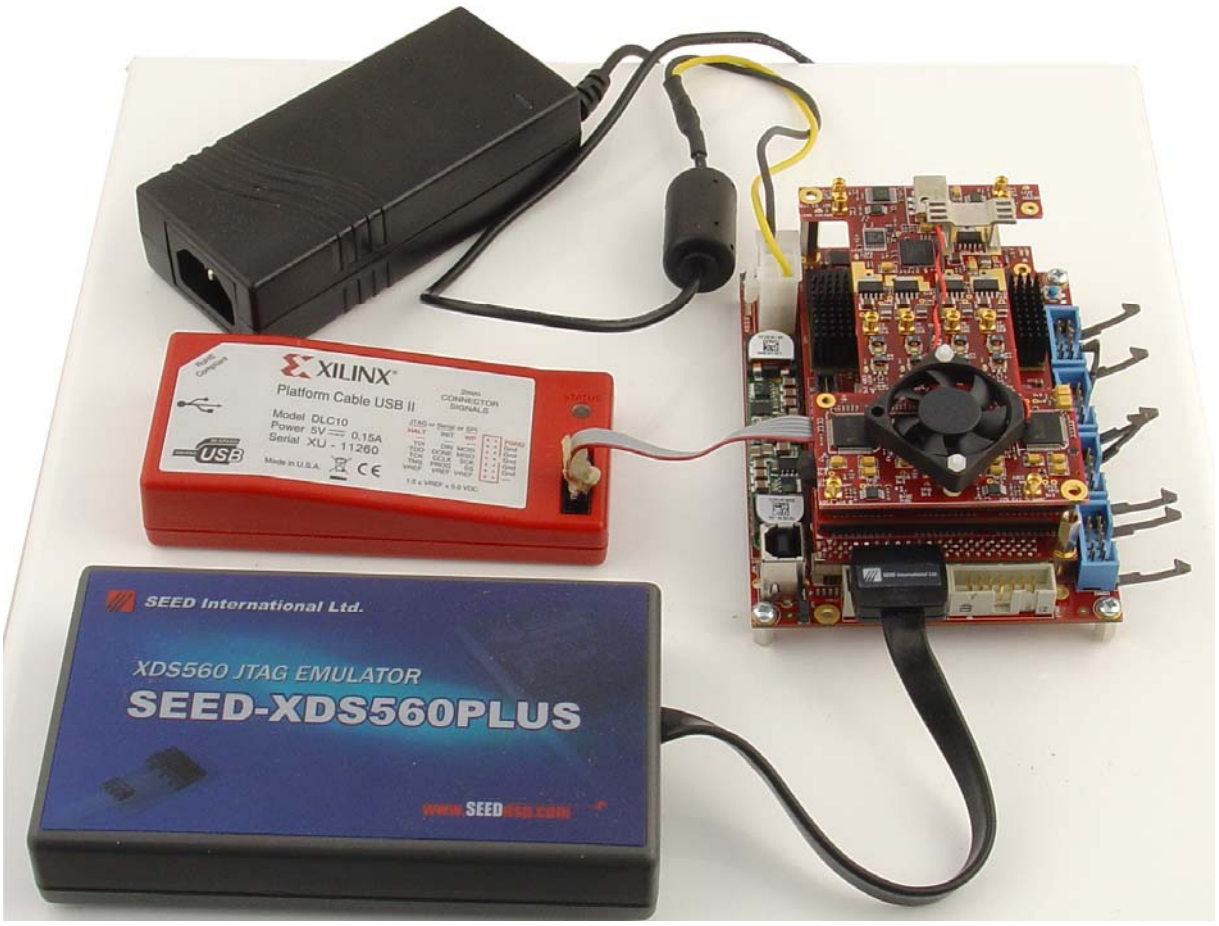
Two SLB Extenders are required – one of top of each other in order to leave a gap big enough for the DSP heatsinks.

Switches (SW1) on SMT372T should ***1=OFF, 2=OFF, 3=ON and 4=OFF***. This setting allows the flash to be accessed from the USB interface, via comport3. When the SMT6002 (Flash Utility for FPGA modules) tool has completed loading the new custom bit-stream, re-power the board and the FPGA should be configured with the newly loaded firmware. Because the FPGA configures itself from flash and is not managed by a CPLD as in other Sundance TIM's, there can only be one firmware loaded at a time, in position 0. For further information on the SMT6002, see the SMT6002 Help.

The XDS560 should be connected to JTAG1 on the carrier board. The Xilinx programming cable should be connected to JP4 on the SMT372T. The system demonstration runs on the hardware using Code Composer 4 as a debugging tool to load and run the application into DSP A.

A 12-Volt power supply is also required on JP2 (SMT111).

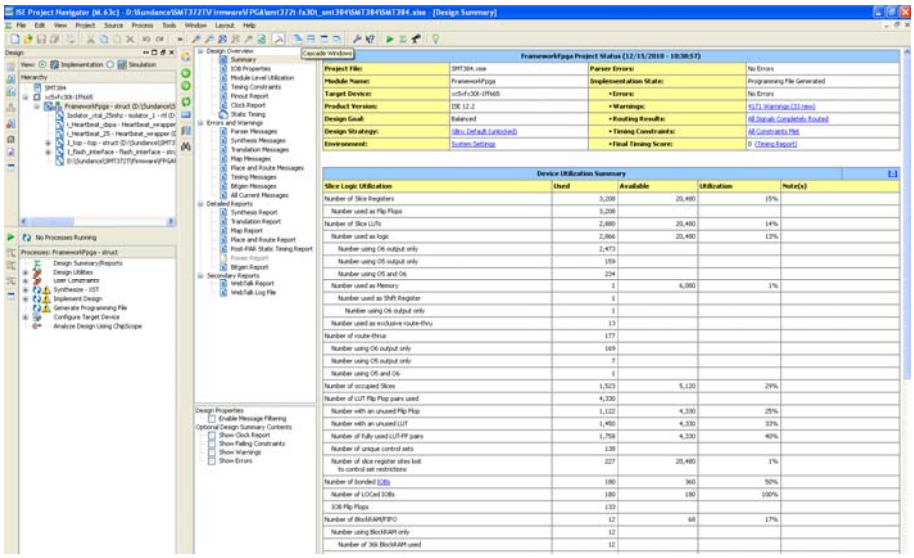
The photo below shows where connectors are:



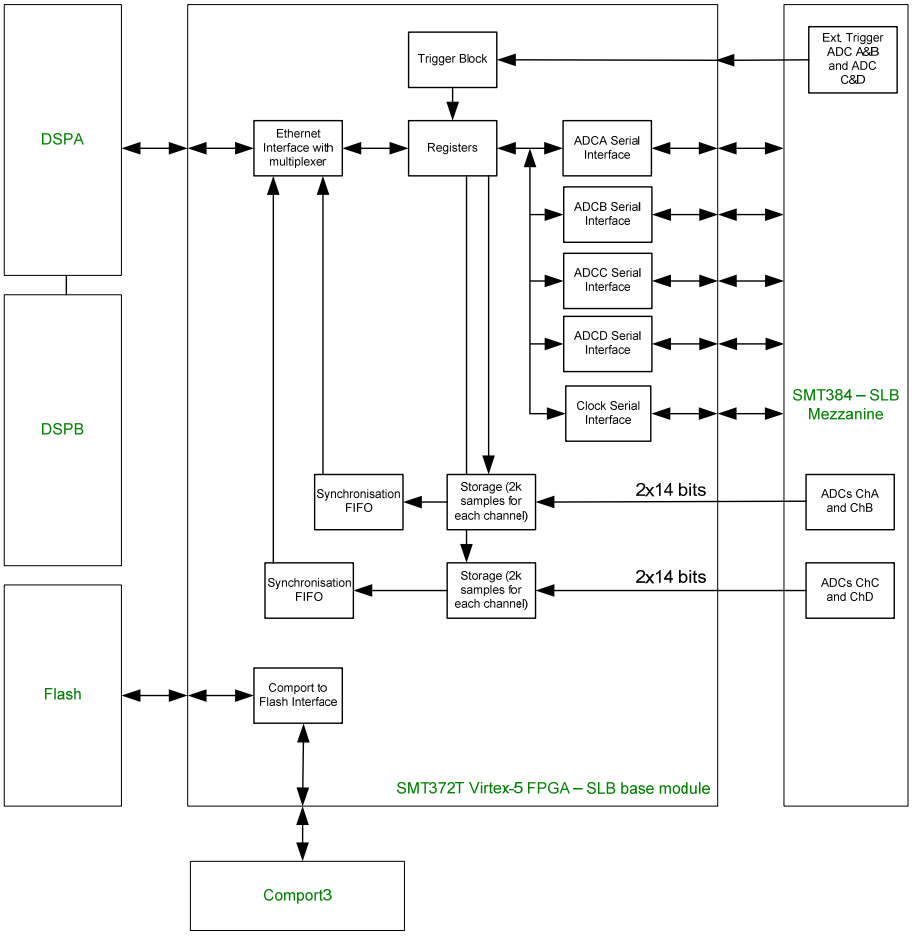
Note that it is recommended to use the hardware in a non-closed environment in order to provide enough cooling airflow. In the case where more FPGA and DSP activity is implemented, extra cooling might be required.

6 ISE project.

An ISE project is provided. It gives the source code for the standard firmware. It is called ***SMT384.xise*** and can be open with ***Project Navigator*** (ISE 12.2 or above).



The block diagram below shows how blocks are connected within the firmware:



Once the project has compiled, the bitstream created (***frameworkfpga.bit*** that can be found in the SMT384 directory) can be loaded into the FPGA using the programming cable.

Note that in order to use the USB to load the bitstream into the flash (and boot the FPGA from it), a link (Xlink) has been added to the design between Comport3 and the flash.

All options used to compile the project are saved in the project file so no alterations to the project are required.

The firmware implements all the interfaces required to drive the SMT384 as well as an Ethernet link between the FPGA and DSPA (SMT372T).

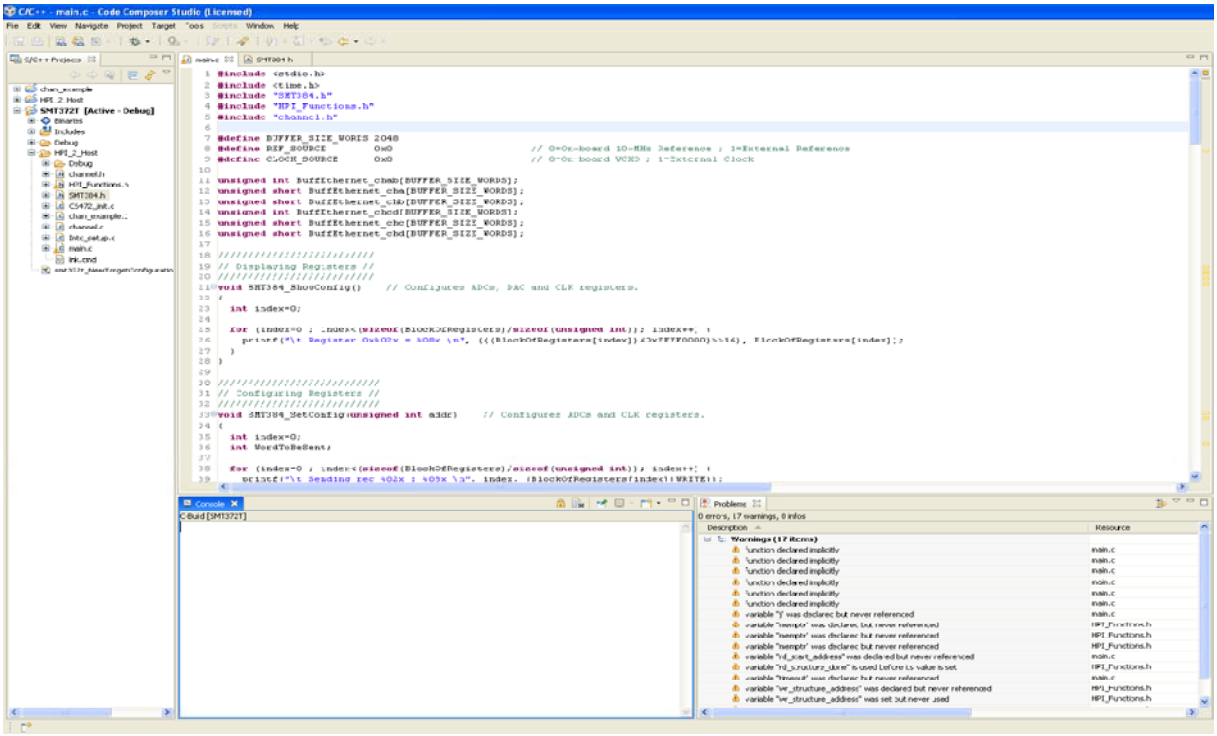
Some IODELAY blocks have been implemented on the adc data lines. They can be dynamically programmed (VARIABLE mode). Delays (decimal value between 0 and 63 – 1 step being equivalent to 78ps) are loaded by the DSP. This can be useful when the firmware is being altered and timing are changed, which means some delays might be adjusted.

7 Code Composer 4 workspace.

A Code Composer 4 workspace is provided. It gives the source code of the standard DSP software. DSPs are responsible for programming the SMT384 and collecting samples from the ADCs.

The workspace has been created with version 4.1.2 of the tool as it is the earliest version that provides enough compatibility with XDS560 emulator.

The workspace to open when prompted is ***CodeComposer4Workspace***.



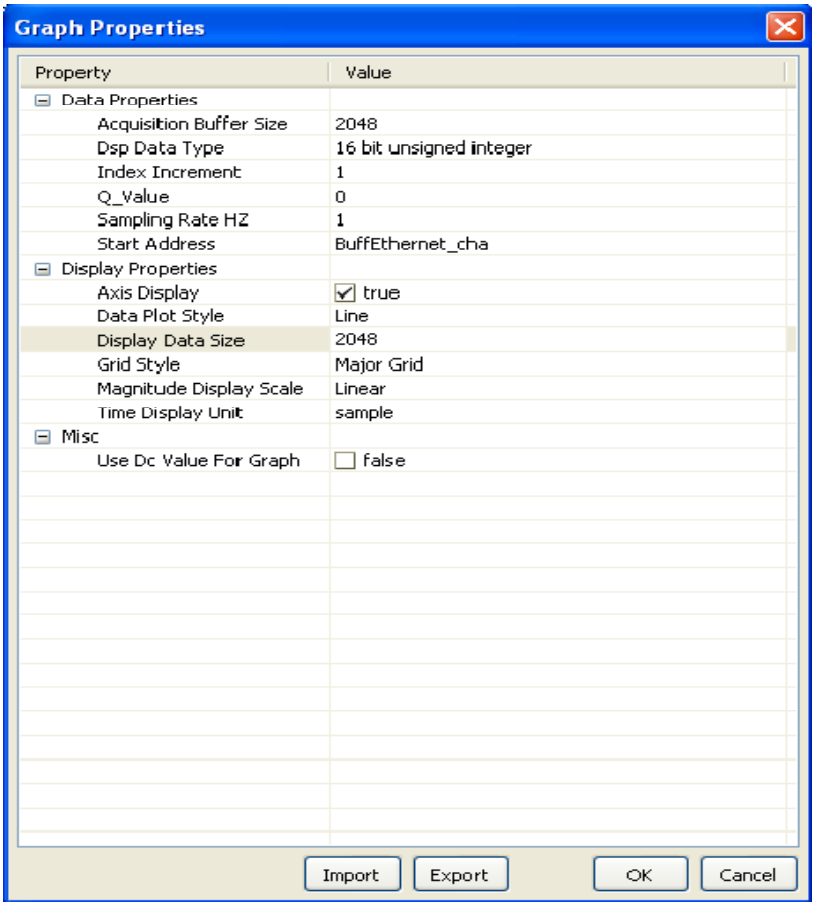
The DSP uses only one Ethernet link between for all communications (control and data) the FPGA and the DSP. In order to do so, DSPs have to have their internal pll initialised. An initialisation process is also required. Once done, you should see ***D6 and D7 (SMT372T) flashing***.

This is what is done at the very beginning of the code (main.c). SMT384 registers can then be programmed. Once clocks and adcs are programmed LED5 and LED7 should be flashing. They reflect an image of the sampling clocks of respectively ADCa&b and ADCc&d.

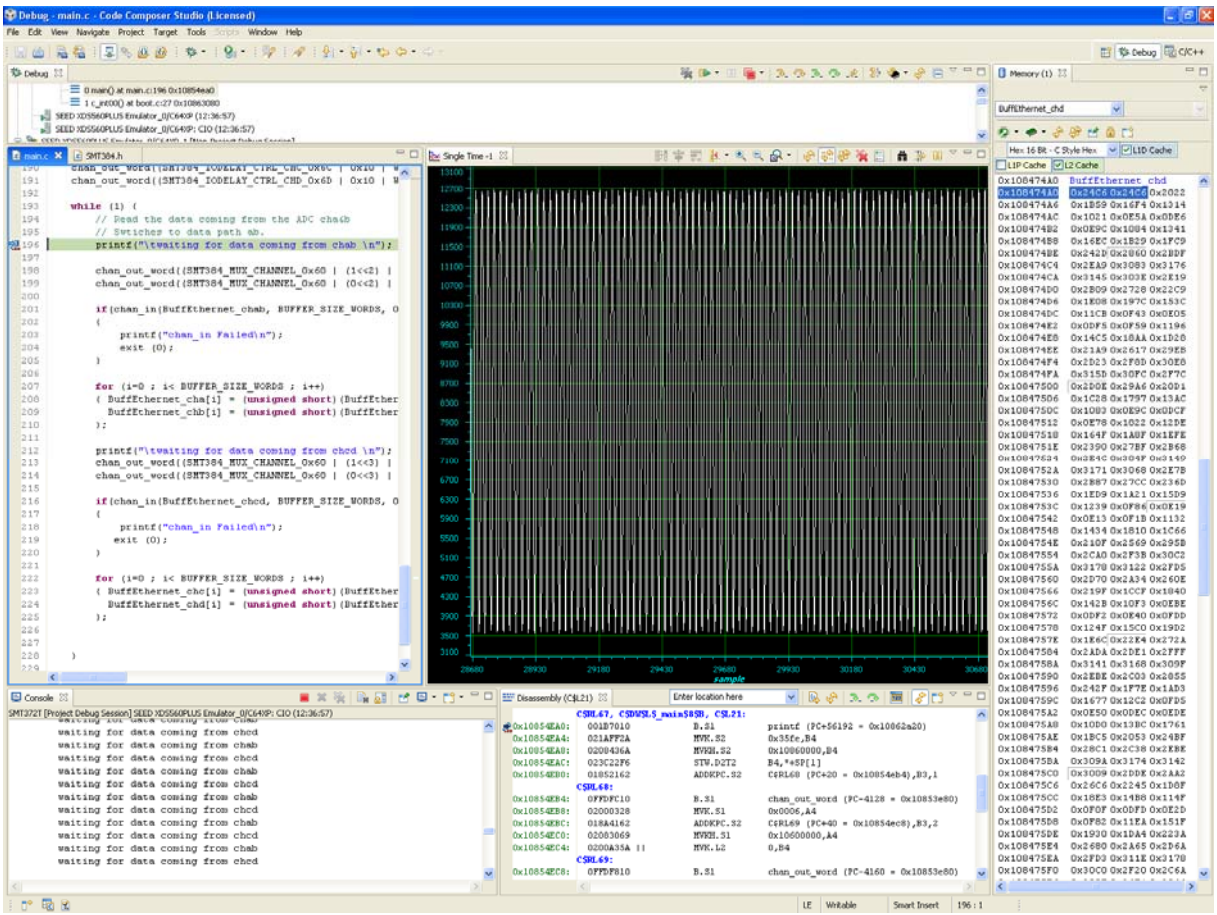
Clocks are programmed in such way that the on-board 10-MHz reference signal is used to lock the on-board VCXO (245.76MHz). ADCs are clocked at half the VCXO frequency, which is 122.88MHz. An external reference can also be used to lock the VCXO as well as an external sampling clock. In order to so do, register settings have to be changed (they can be found in smt384.h – **BlockOfRegisters**). More details of the registers can be found in the SMT384 User Manual.

In order to gather samples, four separate buffers have been declared in the code.

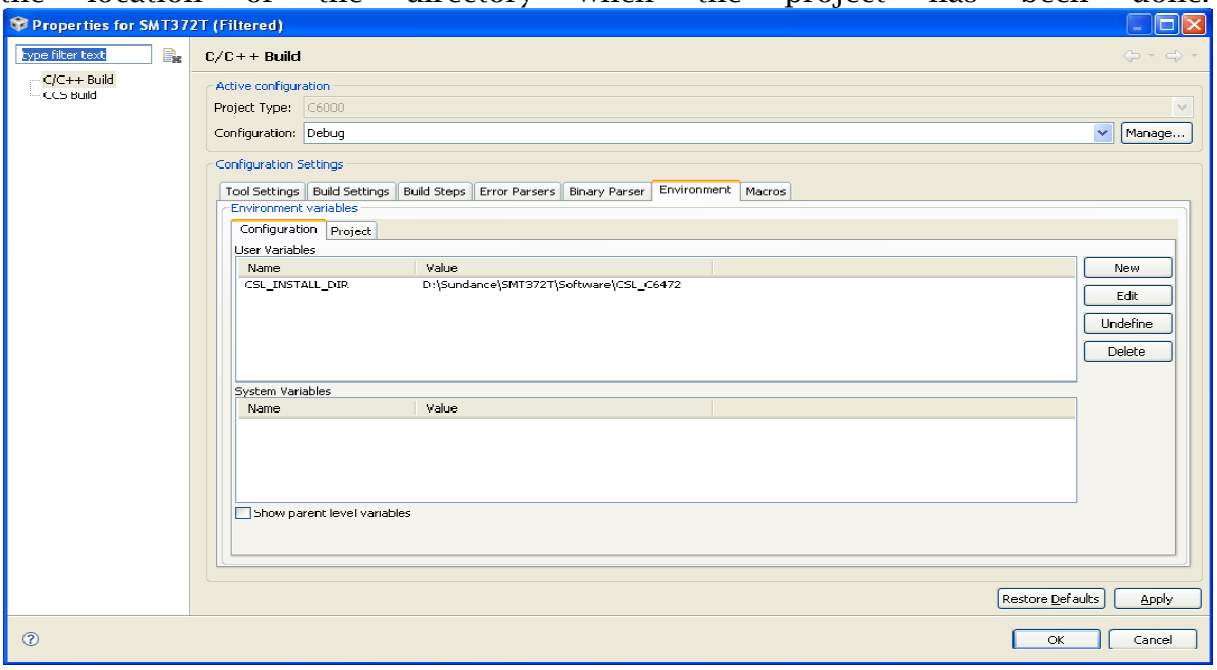
BuffEthernet_cha, ***BuffEthernet_chb***, ***BuffEthernet_chc*** and ***BuffEthernet_chd*** are all four 2k-sample buffers. They can be used directly to visualise the samples (for instance using a Memory window under Code Composer that points to one of the buffers) or to plot them using a Graph window.



The project can be run in debug mode so a pointer can be placed in the while(1) loop and each time the DSP stops, graph and memory windows are updated.



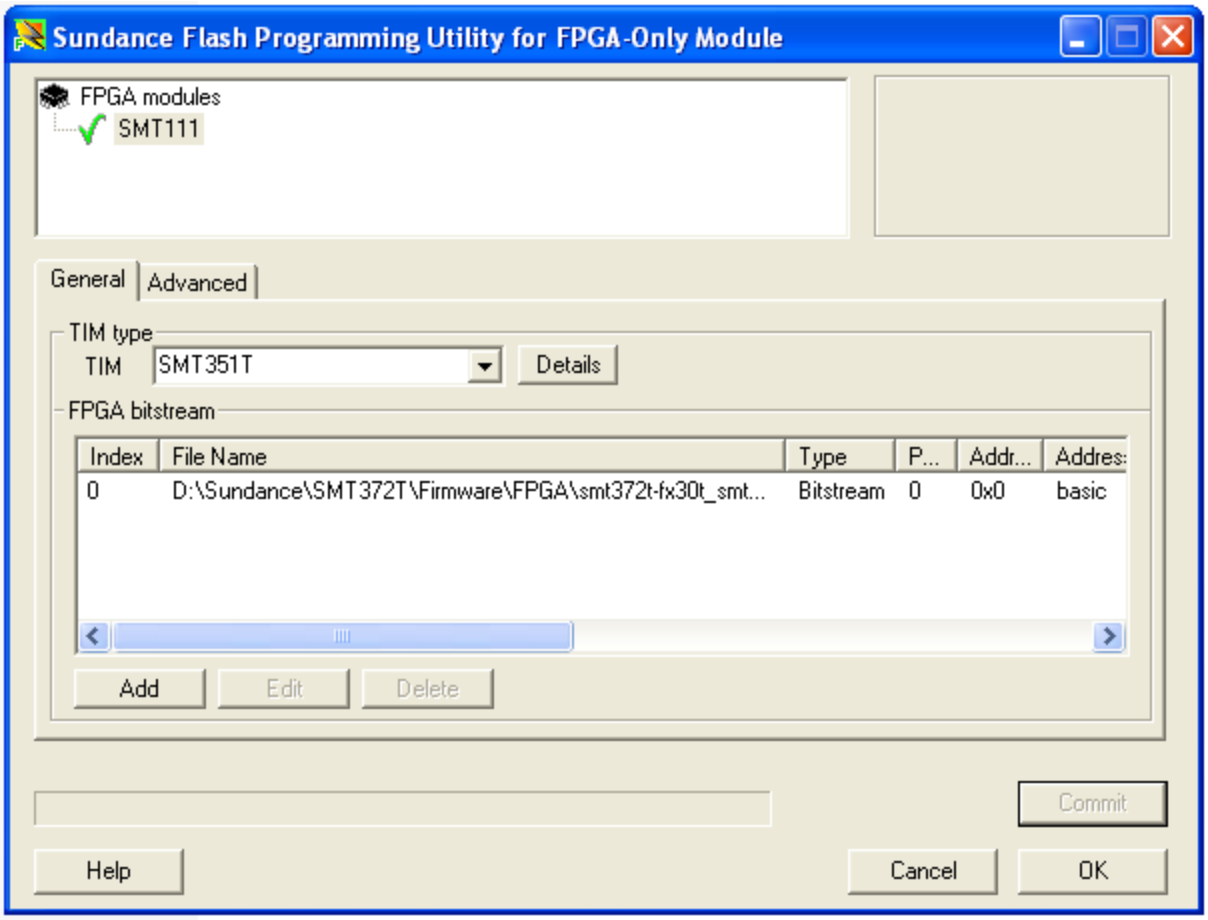
Note that one environment variable might need to be added in order for the compiler to use and find specific libraries. It is done by right-clicking the active project, then go to Build Properties, and in the environment tab, you need to add ***CSL_INSTALL_DIR*** to point to the CSL_C6472 directory. The picture below shows the location of the directory when the project has been done:



8 SMT6002 – Flash Utility for FPGA modules.

In order to program the flash (SMT372T) from the USB, simply connect a USB cable to JP4 (SMT111). The board will be detected as a Sundance USB product (SMT111 should appear as well).

The SMT6002 host application can be started. Because the flash is connected directly to the FPGA, it means it can only boot from one location. To load a new bitstream, start by *deleting* any existing and *add* the new one. Press *Comit* to start the process. Once completed, power off and on the system. It will then boot from the new bitstream just stored in flash.



Note that for some version of the Sundance software, the SMT372T can be detected as an SMT351T. This has been corrected for future release and causes no problems at all for the programming process.

In case the bitstream stored into flash is corrupted or shows any problem with link (comport3-Flash), the JTAG will have to be used to come out of the situation and re-load the bitstream provided.