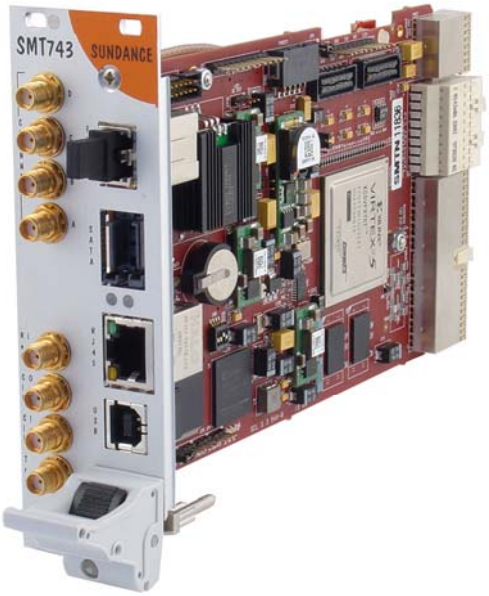
APPLICATION NOTE 1

Application Note - SMT743 System.

SMT743



SUNDANCE MULTIPROCESSOR TECHNOLOGY LTD.

Document Title	Application Note - SMT743 System.doc				
Date	23/05/2011	Revision	1	Page	1 of 8

Date	Comments / Changes	Author	Revision
23/05/11	Original Document completed	PhSR	1

System Description.

The SMT743 is composed of an SMT700 (FX70T-3 FPGA) carrier board coupled with an SMT943.

The carrier board implements a PXIe core (4 Lanes) that is responsible for transferring samples between the host and the card. Thanks to the Sundance DMA engine, speeds between 600 and 700 Mbytes/s can be achieved for a transfer size of 16Mbytes.

In this application note, ADCs and DACs have got their sampling rates set to 122.88MHz. There are 2 ADC channels coded on 14 bits and 2 DAC channels coded on 16bits.

We are aiming here at transferring ADC samples of both channels without loss from the card to the host. Transfers are 16-MByte long (that's 4MSamples from ADCA and 4MSamples from ADCB). Samples collected at the host are then sent back to the board, routed to the DAC channels this time.

In order to achieve no-loss transfers, the speed on the Express Bus should be at least:

2 bytes x 2 channels x 122.88MHz = ***491.52Mbytes/s***

This applies to both directions. This is something that the Sundance 4-Lane PCIe core can achieve.

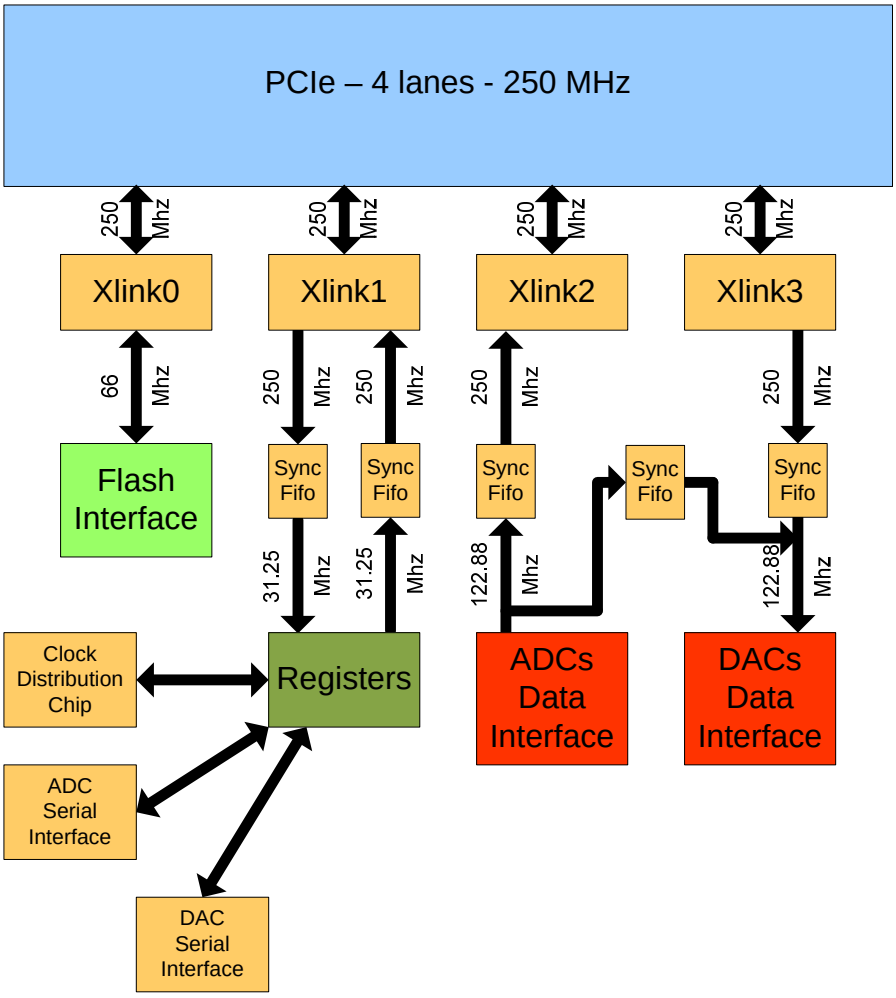
Part of the ADCs features, there is the option to set them to output a counter. ADCs can't be stopped or halted. The Host is responsible for checking all samples collected. Any missing one will be detected straight away. Once captured, samples (from ADC counter) are sent to the DACs.

When it comes to the DACs, a comparator has been implemented on the FPGA to check that all samples are there. Any missing one will be detected. The error bit is reported in a status register that the Host application polls at each cycle.

Transfer speed of each cycle is displayed in the host application.

System Block Diagram.

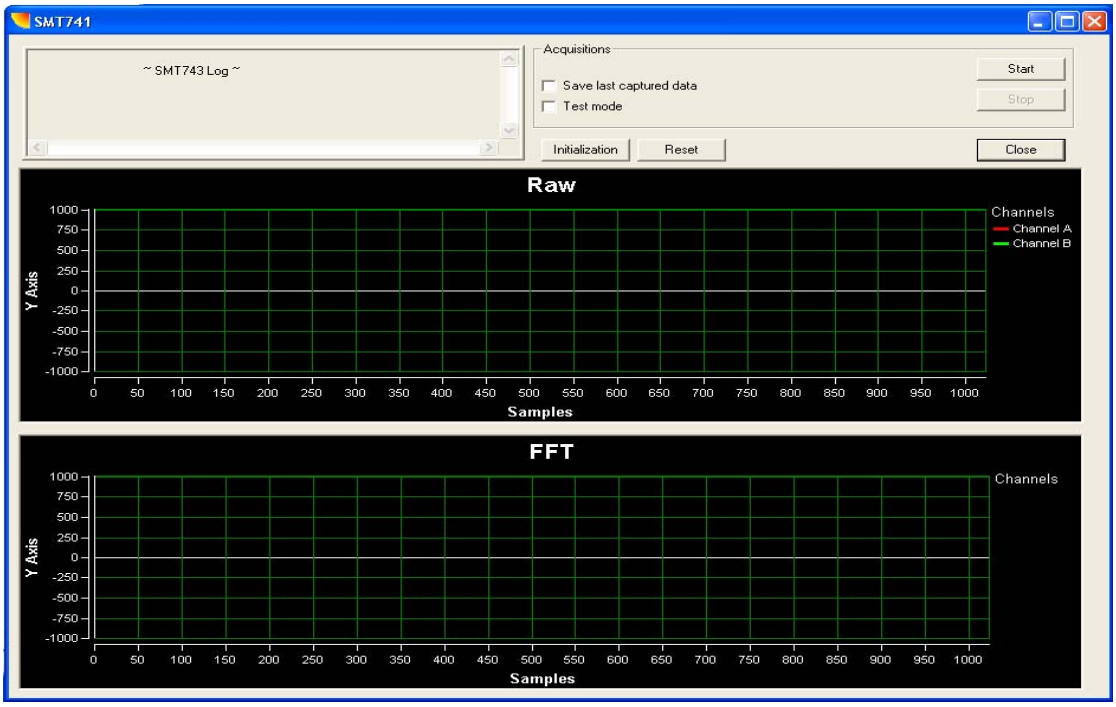
The diagram below shows the connections in the FPGA. Four Xlinks are connected to the Express core. The first one allows access to the flash where the FPGA configuration is stored and loaded into the chip at startup. The second one implements all necessary registers to configure the SMT943. The third one is where ADC samples are being collected from. And the fourth one is where DAC samples are being sent to.



Host Application.

Overview.

Part of the SMT7026 package, is a Microsoft C++ project for this system. It comes as a Graphical User Interface (GUI) and looks like the capture below.



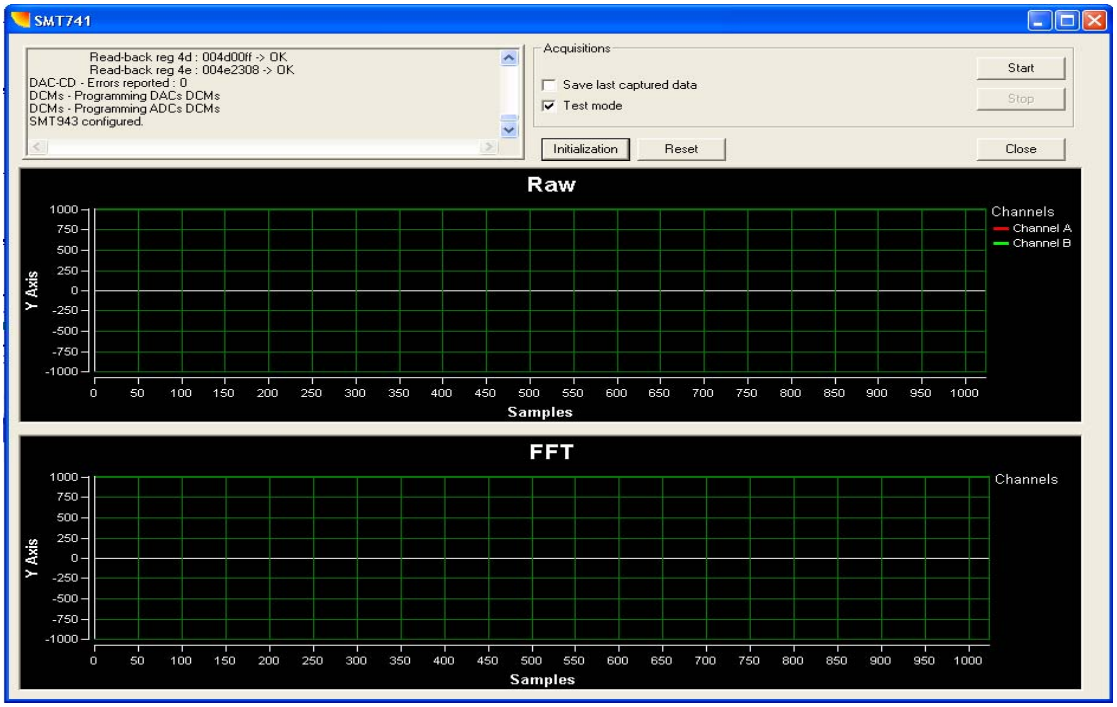
Mode selection.

The GUI implements 2 modes:

- ***Normal Mode*** (Test Mode box not checked): ADCs are set to digitise their input and 16Mbytes of samples are captured and send to the DACs without loss of data. Connecting a sine wave to the ADCs will shows the same sinewave on the DACs during 16Mbytes worth of samples.
- ***Test Mode*** (Test Mode box checked): ADCs are set in their counter mode. 16Mbytes of samples are captured, checked for no loss of data (errors found are displayed on the log section) and then sent to the DACs. Before reaching the DACs, samples are checked for loss of data. Any error found will be displayed in the log section.

Initialisation.

Once the operating mode is selected, the ***Initialisation*** button can be pressed. At this stage, all necessary registers are programmed. They are mainly registers related to the SMT943, such as ADC internal registers, DAC registers, CLOCK registers, etc.

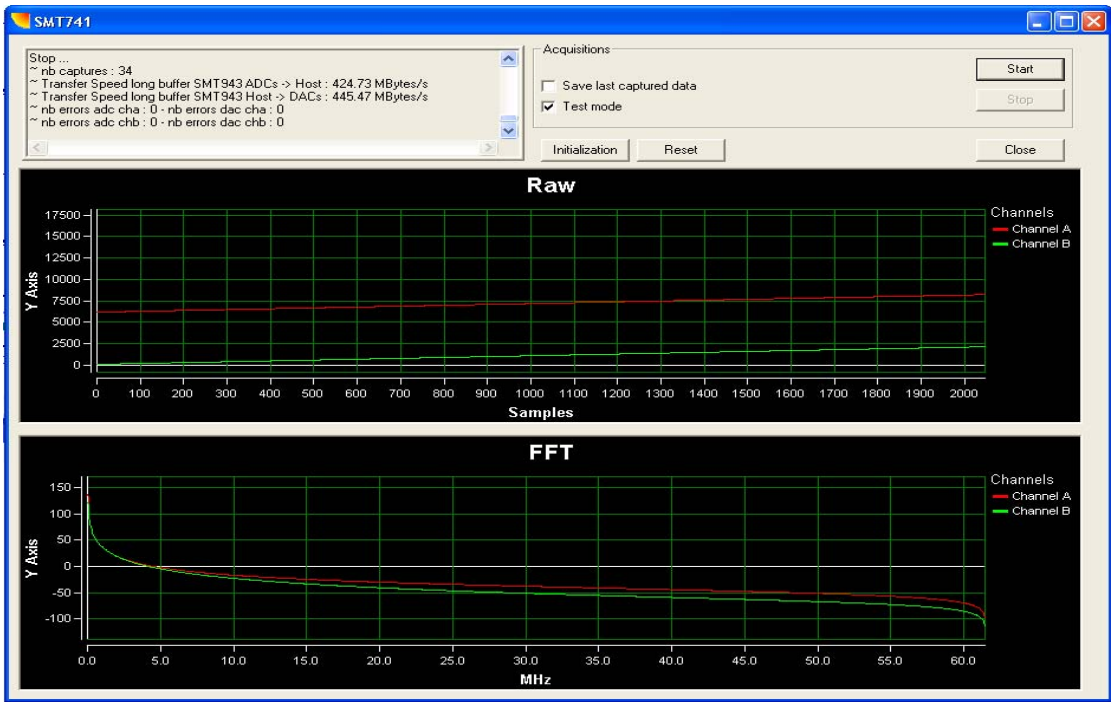


In the log section, register values are displayed as well as the result of their read-back operation.

Performing Acquisitions.

Cycles of acquisitions are started by pressing the ***Start*** button. While during acquisitions, the transfer speed is also displayed in the log file. It is to be noted that the speed figure also includes the process of setting up a DMA and the wait period until it is granted. This explains why figures a lower than the effective output rate of two channels of 14-bit ADCs (491.52Mbytes/s).

The raw window shows the first 2k samples captured and the FFT window shows the result of its FFT.



Storing into files.

The box labelled ***save last captured data*** allows dumping into a file the last 16MByte buffer captured after pressing the ***Stop*** button.
Files have got the *txt* extension and can be found where the application is running from. They are ASCII files with one sample per line (unsigned integer),