S2C
FPGA PROTOTYPING SOLUTIONS

Overview

- S2C's ProtoBridge C-API software enabled us to accomplish rapid transmission of massive amounts of data from PC to our FPGA prototype, as well as quick access to the FPGA's memory buffer using both simple Read/Write and DMA modes.

- The large library of peripheral daughter cards allowed us to complete verification of multiple projects using the same FPGA board and thus protected our investment.

- The mature commercial products with a large user base proved reliable in our projects.

Using S2C's rapid FPGA-based prototyping system and ProtoBridge's C-API software, Penstar Technology was able to implement its Demux Design on FPGA in order to transmit large amounts of test stimuli, and to access the memory buffer from PC via C-API functions. "We were really happy because we not only saved on hardware and software/driver development costs but also reduced the overall design cycle," said Yu Tao, CTO of Penstar Technology.

Challenge

"A big portion of our products is in the digital TV set-top boxes market and the market window is short. The ability to launch our SoC early in the market is critical to our success.

About Penstar

Penstar Technology is a leading fabless IC design company and technology innovator in China. The company is focused on the design, development, manufacture, and marketing of advanced SoC products, microcontrollers, HDTV and mobile TV decoder and demodulator ICs, LCD driver ICs, and related hardware, software products and services. Penstar Technology was among the first companies to develop a HDTV decoder chip based on China's Audio and Video Coding Standard (AVS).

The TS streams which needed to be tested were huge. and how to transmit the data and fully verify all parts of our design became a big challenge," said Yu Tao. "FPGA prototyping was needed, but to design a complex FPGA prototyping board by ourselves is a challenge, and we also needed to transmit massive amounts of test data via PCIe from PC at high speed. We did not want to waste our engineering time on building the FPGA board nor on building the driver and software to allow fast communication from PC to FPGA. As a result, we decided to look for off-the-shelf solutions."
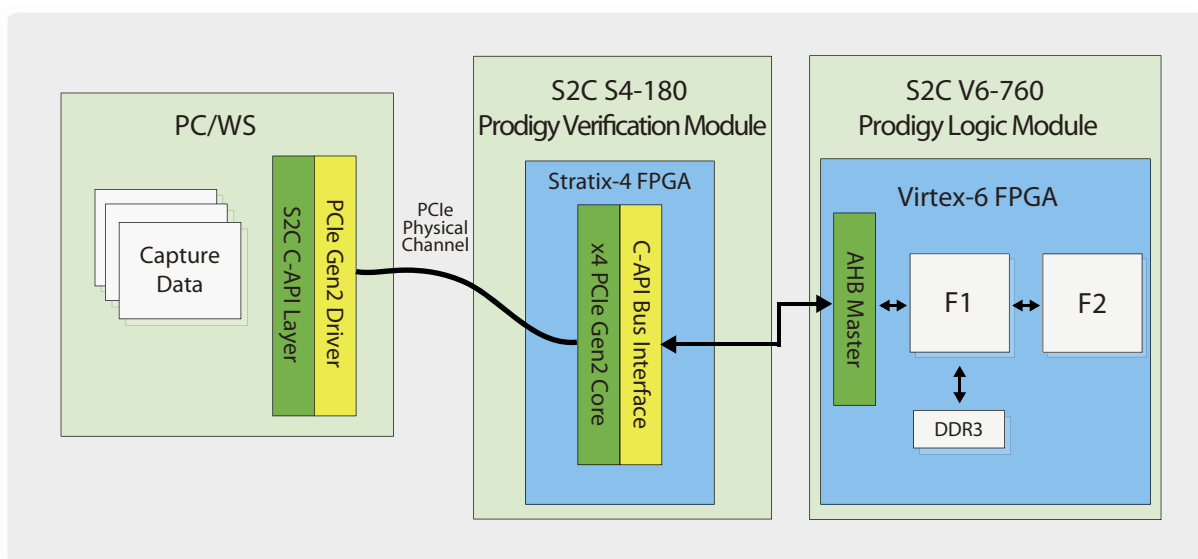

Figure 1: Penstar's Verification Setup

Solution

"We still remember the first discussion with S2C about its Prodigy Logic Modules (a multiple FPGA platform with a large library of peripheral daughter cards) and the ProtoBridge Software (C-API functions that allow the fast transmission of massive amounts of data between FPGA and PC through a PCIe Channel). The S2C demo left us with a deep impression and prompted us to come up with smart means to verify our designs. We decided to adopt the S2C Dual V6 TAI Logic Module to host our design and use the ProtoBridge Software to transmit the huge TS stream and access the data from memory," said Yu Tao. The diagram in Figure 1 shows the verification environment set up for Penstar's Demux project.

"Since S2C's ProtoBridge C-API software includes a PCIe driver, PCIe implementation via a S2C Verification Module and an AHB master port supplied by S2C, setting up the design environment was straightforward. First, we hooked up our design interface to the AHB master port and mapped the design onto the 2 FPGAs on the S2C Dual V6 TAI Logic Module. Then, we wrote a C program that calls the S2C supplied C-API functions for transferring the TS stream to the Demuxblock in the FPGA. In addition, our C program could read/write and import/export the data in DDR3 memory through an easy to use GUI developed by us as shown in Figure 2" said Yu Tao.

Results

Penstar Technology was able to set up their FPGA verification environment quickly and run a large amount of tests, reducing software development time. "We were really happy because we not only saved on hardware and software/driver development costs but also reduced the overall design cycle," said Yu Tao.

"S2C's technical support team played an important role in the success of our projects and we look forward to long-term cooperation with S2C."
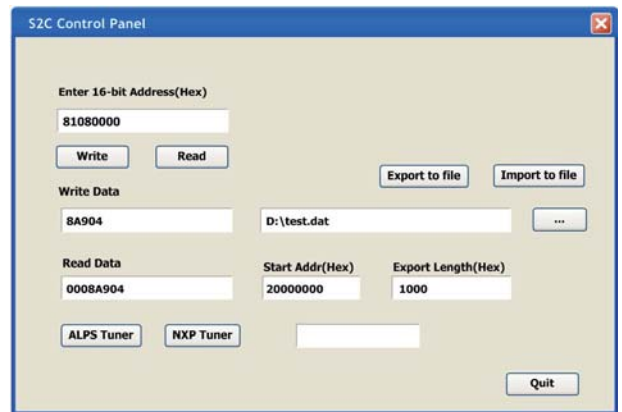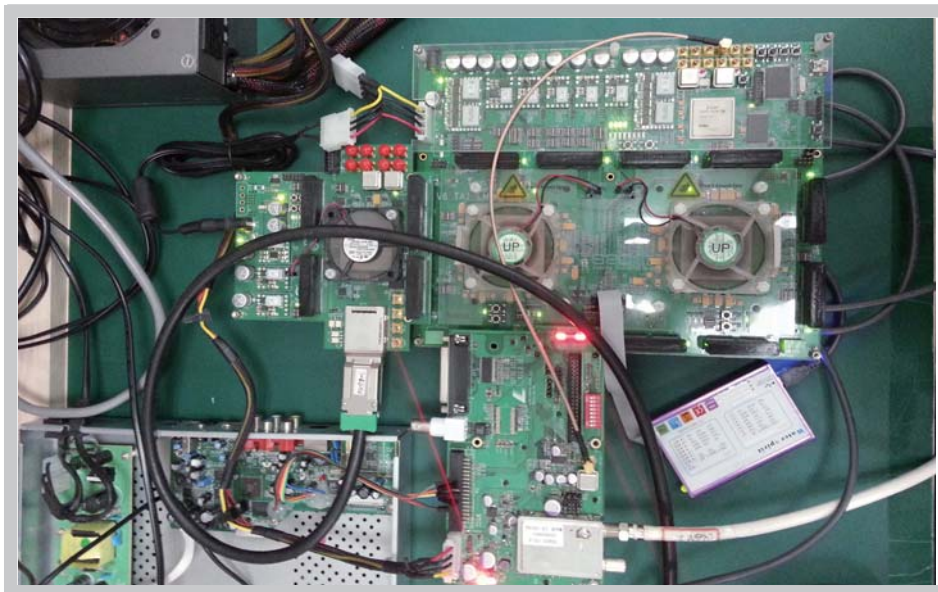
Figure 2: Memory Access GUI

San Jose | Shanghai | Beijing | Hsinchu | Seoul | Yokohama