Airborne Demonstration of FPGA implementation of Fast Lossless Hyperspectral Data Compression System

D. Keymeulen[1], N. Aranki[1] A. Bakhshi[2], H. Luong[1], C. Sartures[1], D. Dolman[3]
[1]Jet Propulsion Laboratory, California Institute of Technology, 4800 Oak
[2]B&A Engineering Inc., 3554 Business Park Drive, Suite A-1, Costa Mesa, CA 92626
[3]Alpha Data Inc., 3507 Ringsby Court Suite 105, Denver, CO 80216

818-354-4280
didier.keymeulen@jpl.nasa.gov

Abstract—**Efficient on-board lossless hyperspectral data compression reduces data volume in order to meet NASA and DoD limited downlink capabilities. The technique also improves signature extraction, object recognition and feature classification capabilities by providing exact reconstructed data on constrained downlink resources. At JPL a novel, adaptive and predictive technique for lossless compression of hyperspectral data was recently developed. This technique uses an adaptive filtering method and achieves a combination of low complexity and compression effectiveness that far exceeds state-of-the-art techniques currently in use. The JPL-developed 'Fast Lossless' algorithm requires no training data or other specific information about the nature of the spectral bands for a fixed instrument dynamic range. It is of low computational complexity and thus well-suited for implementation in hardware. A prototype of the compressor (and decompressor) of the algorithm is available in software, but this implementation may not meet speed and real-time requirements of some space applications. Hardware acceleration provides performance improvements of 10x-100x vs. the software implementation (about 1M samples/sec on a Pentium IV machine). This paper describes a hardware implementation of the 'Modified Fast Lossless' compression algorithm for pushbroom instruments on a Field Programmable Gate Array (FPGA). The FPGA implementation has been integrated into the Next Generation Data Capture System (NGDCS) avionics system for the Airborne Visible/ Infrared Imaging Spectrometer Next Generation (AVIRISng). The NGDCS includes two airborne hardware platforms which were flown on a Twin Otter aircraft : a National Instrument PXI and the Alpha Data Systems. The FPGA implementation targets the current state-of-the-art FPGAs (Xilinx Virtex V and VI families) and compresses one sample every clock cycle to provide a fast and practical real-time solution for Space applications**

1. INTRODUCTION

Hyperspectral images are three-dimensional data sets, where two of the dimensions are spatial and the third is spectral. A hyperspectral image can be regarded as a stack of individual images of the same spatial scene, with each such image representing the scene viewed in a narrow portion of the electromagnetic spectrum. These individual images are referred to as spectral bands. Hyperspectral images typically consist of hundreds of spectral bands; the voluminous amount of data comprising hyperspectral images makes them appealing candidates for data compression [1]. An example of a hyperspectral data cube is shown in Figure 1 using the AVIRIS instrument. (Pearl Harbor was taken by original AVIRIS).

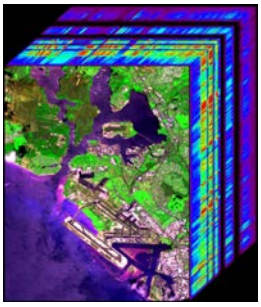

Figure 1 An example of a hyperspectral data cube for Pearl Harbor, Hawaii taken by the AVIRIS instrument

Exploiting dependencies in all three dimensions of hyperspectral data sets promises substantially more effective compression than two-dimensional approaches such as applying conventional image compression to each spectral band independently. With that in mind, the JPL Fast Lossless hyperspectral compressor was developed. It is a predictive technique that uses an adaptive filtering method and achieves a combination of low complexity and compression effectiveness that far exceeds state-of-the-art techniques currently in use [5]. This algorithm has been adapted as the standard for Lossless Multispectral & Hyperspectral Image Compression by the Consultative Committee for Space Data Systems (CCSDS) [6].

Current NASA and DoD hyperspectral imagers either avoid compression or make use of only limited lossless image compression techniques during transmission. For example, the current state-of-the-practice is to use the Universal Source Encoder for Space (USES) chip [3]. USES implements a lossless compressor standardized by the Consultative Committee for Space Data Systems (CCSDS), which is based on the Rice algorithm [4], and has a multispectral mode, extending its operation to 3D data sets. The USES chip performance achieves limited compression effectiveness compared to other existing techniques, but has the advantage of being currently available in a radiation hardened form. Future instruments with more sensors and a much larger number of spectral bands will collect enormous volumes of data that will far outstrip the current ability to transmit it back to Earth (data rates for some instruments can go to several hundreds of Gbits/s). This gives rise to the need for efficient on-board hyperspectral data compression. Software solutions have limited throughput performance and are power hungry. Dedicated hardware solutions are

highly desirable, taking load off the main processor while providing a power efficient solution at the same time. Application-Specific Integrated Circuits (ASICs) implementations are power- and area-efficient, but they lack flexibility for post-launch modifications and repair, they are not scalable and cannot be configured to efficiently match specific mission needs and requirements. Field Programmable Gate Arrays (FPGAs) are programmable and offer a low cost and flexible solution compared to traditional (ASICs). We have previously described the whisk broom and push broom mode of the Fast Lossless (FL) algorithms and its FPGA implementations on a XILINX VIRTEX-4 LX25 [7]. In this paper we describe our latest implementation of the push broom mode of Fast Lossless algorithm targeting push-broom-type hyperspectral imagers used mostly for space applications, specifically designed for XILINX VIRTEX-5 FPGA. VIRTEX-5 is being qualified for space as many of its modules are radiation hard by design limiting the need to use fault tolerance strategies such as triple module redundancy (TMR). The paper discusses its implementation on a National Instrument PXIe and Alpha-Data systems. The PXIe includes a Virtex-5 SX50T-FPGA board and controller with an Intel quad core i7. The Alpha-Data system includes a Virtex-6 LX240T FPGA board and an Intel Xeon Processor. The integration of both computer and FPGA systems with the AVIRISng instrument is done through a Camera Link interface with a data rate of 60Mbytes/seconds. Finally the paper presents the results of the compression during flight test in South California on a TwinOtter aircraft. The implementation has a throughput of 38 Msamples/sec, scalable, and utilizes less than 40% of the FPGA resources.

2. FPGA IMPLEMENTATION

The push broom mode of the Fast Lossless algorithm was implemented and integrated into a reconfigurable system prototype for a spacecraft payload requiring high communication throughput. The reconfigurable system takes advantage of high-density SRAM-based FPGAs to accommodate the on-board computer resulting in efficient hardware architecture in terms of power, area, and speed. The implementation used two banks of 256 M-Bytes of synchronous dynamic random-access memory (SDRAM) with an 800 M-Bytes bandwidth to easily emulate the instrument and the spacecraft data bus.

The architecture of push broom mode of the Fast Lossless compression algorithm is shown in Figure 2. The implementation works on 32 frames of hyperspectral data at a time. Raw imagery data is stored as a three dimensional cube (for example high:Y=32, wide:X=640, bands:Z=427,). Each pixel of the hyperspectral cube can accommodate up to 13 bits depending on the resolution of the sensory data.

The current implementation targets the XILINX VIRTEX-5 SX50T FPGA, and assumes a BIP (Band interleave by Pixel) data format, but can be extended to BIL (Band interleaved by Line) and multi stream data format. The basic blocks of the implementation are:

- LOCAL MEAN is an Accumulator and 2x16 bits Shift Register that is used to store the sum of upper spatial pixels from the previous line and the previous spectral pixel from the previous band Shift Register is used to store and shift the last four accumulator results.
- DIFFERENCE block consists of three identical subtract modules that are used to subtract the local mean values from the previous spatial and spectral pixels
- WEIGHT block is made up of three, of length Z (number of spectral bands) by up to 16 bits FIFOs that is used to adjust the input to the multiplier. Values of weight are re-calculated for every new spectral row (Z). On power up all the weight are initialized to a default value.
- MULTIPLIER block includes three VIRTEX-5, 18x18 multiplier primitives. Multipliers are used to multiply the output of Difference block with their adjusted Weight values.
- ESTIMATE block consists of an Accumulator and a Comparator. The Accumulator is used to sum the multiplier values. The Estimate is adjusted and clipped depending on the accumulated result.
- DELTA block subtracts the value of current pixel data from the Estimate to adjust the WEIGHT and also used as an input to the ENCODER.
- ENCODER is made up of a Comparator, of length Z (number of spectral bands) by up to 16 bits FIFO, a Look-Up Table and miscellaneous circuitry. Output of the Encoder is used to determine the width of data to be packed.
- PACKER includes VIRTEX-5 distributed RAM and multiplexers. Distributed RAM is used for Look-Up Tables to adjust the compressed data into the final packed data word. Compressed data is packed and outputted as 128-bits words using GOLOMB codes. It is used to interface to 128 bits DRAM.

The basic concept of the data flow for the FPGA implementation (Fig. 2) is as follows. Initially raw imagery data of each spectral (Z) row is streamed one row at a time into the “FIFO to store Current Z and Upper Y”. This buffer is configured as 2Z by up to 13 bits wide shift register. Two rows of length Z are stored in “FIFO to store Current Z and Upper Y”. New data is shifted in as pixel data is compressed.

Upper spatial pixels X+1, Y-1 and previous spatial pixel (X-1,Y) are needed to compress the current pixel data. For the first pixel (x=0, y=0) of each band, actual values are fed through the compressor and packed without compression. Current pixel data being compressed is also shifted out in parallel to the external RAM to produce an efficient pipeline.

Compression of one pixel of the data happens once every FPGA clock cycle. Compressed data is fed to the PACKER module which packs the compressed data into 128 bits data words. Each 128 bits data word may contain several pixels of data. Also, a compressed pixel may fall into two 128-bit data words boundaries, which in turn will be decoded by the decompression algorithm accordingly.

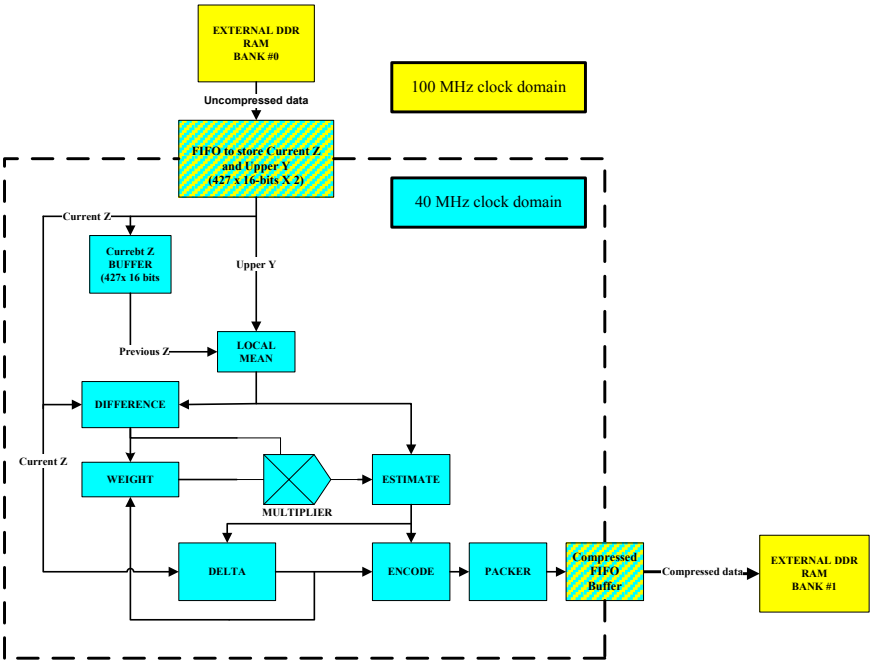

Figure 2. Block diagram of the FPGA implementation of Fast Lossless Compressor. The external DDR SDRAM is used to emulate instrument and spacecraft data buses.

3. FPGA Performance and Resources Utilization

FPGA utilization and speed

Our FPGA implementation was benchmarked on the XILINX VIRTEX-5 SX50T device and ported to a National Instruments (NI) PXIe-7962R prototype board (Figures 3 and 4). Note that all the interfaces to the external Double Data Rate (DDR) RAM are running in 100 MHz clock domain while the actual compression algorithm is running in the 40 MHz clock domain.

The current implementation including the packer has a critical path of 24.29 ns which dictated a maximum clock speed of 41MHz. The current implementation compresses one sample every clock cycle which results in a speed of 40M sample/sec or 25nsec per sample. The implementation has a rather low device utilization of the XILINX VIRTEX-5 SX50T as shown in the Table 1 making the total power consumption of the implementation about 700 mW.

Table 1. VIRTEX-5 SX50T Device Utilization

Device Utilization Summary (estimated values)			[-]
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slice Registers	1586	32640	4%
Number of Slice LUTs	12697	32640	38%
Number of fully used LUT-FF pairs	898	13385	6%
Number of bonded IOBs	0	480	0%
Number of Block RAM/FIFO	8	132	6%
Number of BUFG/BUFGCTRLs	1	32	3%
Number of DSP48Es	3	288	1%

Table 2. Resources available on VIRTEX-5 SX50T and radiation hardened VIRTEX-5QVFX130

Resource Available	VIRTEX-5 SX50T	VIRTEX-5QV FX130
Slice Registers (4*CLB Slices))	32,640	122,880
Slice LUTs (4*CLB Slices)	32,640	122,880
Bonded IOBs	480	960
Block RAM/FIFO (36Kbit)	132	456
DSP48E	288	384

Our FPGA implementation is easily portable to other FPGA platforms such as the space qualified radiation hardened VIRTEX-5QV FX130 (see Table 2) and to an ASIC implementation if need arises. It can also be scaled to accommodate multiple streams of data from the instrument and can be parallelized to increase the compression speed.

National Instrument environment

The current design utilizes National Instrument (NI) LabVIEW environment for integration and testing.

The FL compression Intellectual Property (IP) and all its external interfaces were designed using Verilog HDL programming language. The FL IP is then integrated to LabVIEW FPGA testbench tool using the NI Component-Level-IP (CLIP) technology [9]. The NI LabVIEW FPGA extends LabVIEW graphical development for instruments to FPGAs on NI Reconfigurable I/O hardware, such as the NI PXIe-7962 used in this project.

Two main LabVIEW virtual instruments (VI's) are created for this project, HOST VI and FPGA VI. The HOST VI is the user interface. It is mainly used to transfer uncompressed and compressed data to/from computer hard disk to PXIe-7962R DRAM chips through direct memory

access (DMA). The FPGA VI is where the compression IP is instantiated and is interfaced to external resources of the FPGA available on the PXIe-7962R hardware, such as two memory banks of 256 M-Bytes DRAM.

The DRAM is used to emulate the instrument data interface providing uncompressed data as well as the interface to spacecraft computer receiving the compressed data. The FPGA VI along with the FL IP is then synthesized; mapped and routed with the resulting bit file loaded into the VIRTEX-5 SX50T.

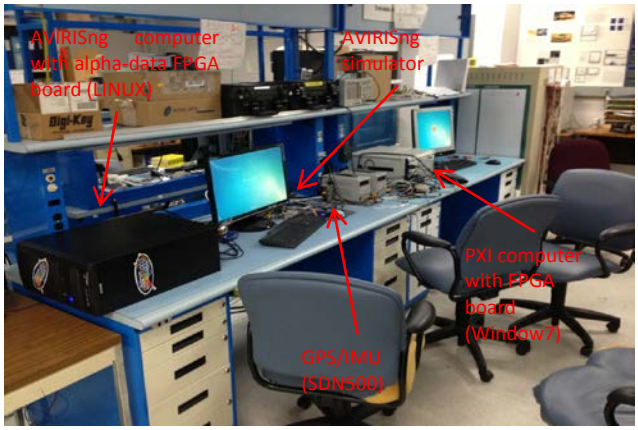

Figure 3: Lab Testbed of the Next Generation Data Capture System (NGDCS) avionics system for the Airborne Visible/ Infrared Imaging Spectrometer Next Generation (AVIRISng) with the National Instrument PXIe chassis implementing the data compression

Figure 4: PXIe-7962R FPGA Development board with VIRTEX-5 SX50T and 512 Mbytes DRAM

Compression process starts with FPGA VI reading in real-time uncompressed data (61MBytes/s) and transferring them the data to PXIe-7962 DRAM bank 0 through the Camera Link interface attached to the FPGA. At that point the FPGA VI startd compression. FPGA VI reads a data block of 32 frames at the time of uncompressed data from DRAM bank 0 and places the resulting compressed data into DRAM bank 1 at each 40MHz clock cycle. At the end of compression of a data block, it proceeds with the next data blocks. Then the FPGA VI signals the HOST VI that it has completed compression of 32 frames and DMA the uncompressed data to the HOST through the PXIe chassis backplane. HOST VI then saves the compressed data into a file on the host computer hard disk and proceeds with the next 32 compressed frames.

Alpha-Data Environment

The data compression has also been ported from the LabVIEW FPGA environment onto a Virtex-6 LX240T on an Alpha Data Parallel Systems' PCIe ADPE-XRC-6T/LX240T-3 using Alpha Data's Camera Link SDK and PCIe bridge driver [8]. The Camera Link SDK provided a template, VHDL firmware and C++ software, design for frame acquisition. Building on the Alpha Data Camera Link SDK template design, the system shown in figure 7 was implemented. The Virtex-6 FPGA on the Alpha Data board interfaces with the hyperspectral instrument and IMU/GPS device (Figure 5 and 6). The Alpha Data FPGA board is connected to the AVIRISng instrument through a FMC Camera Link IO module using a single base Camera Link connection with a clock rate of 34MHz (952Mib/s). Alpha Data's adclinkchip_base IP is used to de-serialize the data inside the FPGA. The FPGA board is attached to the PCIe bus of a computer mother board with an Intel Xeon 8-core (2.9GHz) CPU. The bandwidth available between the Alpha Data card and host system can be up to 1.5GiB/s on a PCIe Gen2 system. The Alpha Data card also has on-board memory; the implemented FPGA design utilizes this memory to buffer up to 512MiB of frame data on the FPGA board. The PCIe bus host also has a 1TiB RAID01 array of Solid State drives to achieve a bandwidth of 800MiB/s to store, in real-time, the processed data from the AVIRISng instrument.

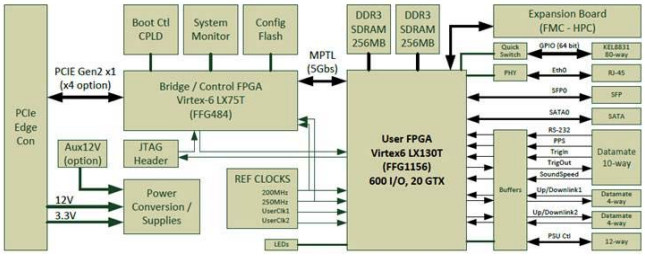

Figure 5: Alpha-data ADPE-XRC-6T family implenting the real-time data capture and synchronization as well as the data compression of the New Generation Data Capture System (NGDCS) [8]

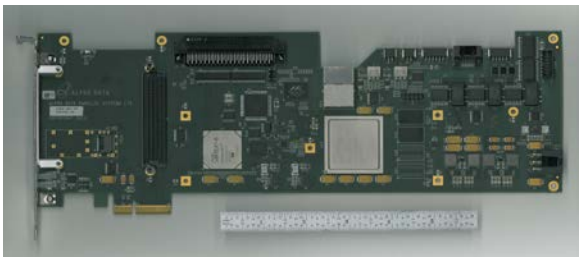

Figure 6: Alpha-data ADPE-XRC-6T hardware used in the New Generation Data Capture System (NGDCS) of AVIRISng [8].

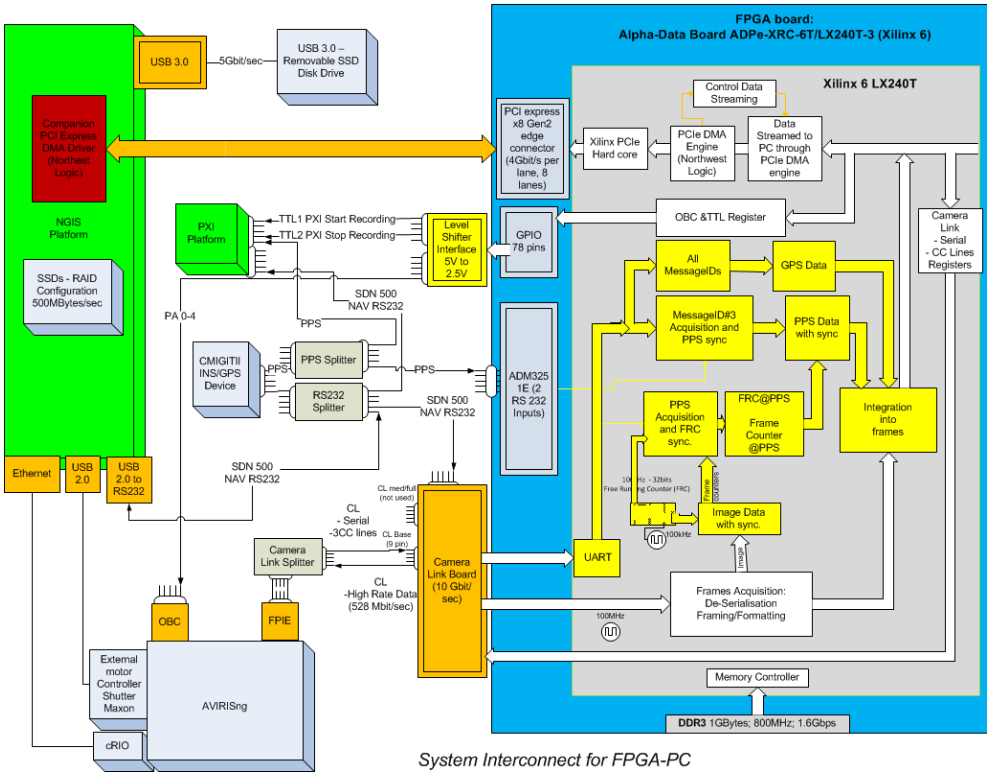

Figure 7: Architecture of the New Generation Data Capture System (NGDCS) avionics system for the Airborne Visible/ Infrared Imaging Spectrometer New Generation (AVIRISng).

Airborne Platform with AVIRISng

The PXI chassis and the Alpha-Data system are part of the Next Generation Data Capture System (NGDCS) avionics system for the Airborne Visible/ Infrared Imaging Spectrometer Next Generation (AVIRISng) (Figure 7). The NGDCS allows to acquire in real-time hyperspectral data in 481 bands with wavelengths from 381 to 2500 nanometers and 640 cross-track pixels generated at 100 frames a second. NGDCS provides synchronization data between the hypespectral image and the inertial, navigation and gps data to perform post-processing orthorectification. The NGDCS platform also is able to the data compression in real-time of the hyperspectral images.

The Software of the Next Generation Imaging Spectrometers is responsible for the operator GUI and the archive of the data on a TeraBytes Solid State Drive. The FPGA of the NGDCS platform is interfacing and processing data from the AVIRISng instrument and the INS/GPS device. The AVIRISng avionics demonstrates the new real-time on-board data compression algorithm.

The PXI with the NGDCS platform has been integrated into a twin otter aircraft during the November 2012 flight test and has performed data compression in real-time of the Moraje desert in California (Figure 8).

4. Data Compression Performance

A synthetic test image was designed to be a 'stress test' for the compressor algorithm and its hardware implementation, to trigger things that might arise very rarely for natural images. For example, the synthetic image exercises the full 13 bit input dynamic range issued by the input image as well as triggers overflow condition of predicted value which is limited to a 32bits register. The synthetic image was successfully compressed with the FPGA implementation and uncompressed with the software code to reconstruct exactly the original synthetic image.

The compression speed results for the hyper-spectral synthetic sample image is similar to hyper-spectral data. Figure 9 shows an un-calibrated AVIRIS hyper-spectral image from Hawaii with dimensions 614 x 512, with 224 spectral bands, and with 12-bits per samples [8]. For this specific image, the output was compressed to 25% of the original packed size (3.02 bits per pixel, a factor 3.98 compression).

Figure 8: Next Generation Data Capture System (NGDCS) avionics system for the Airborne Visible/ Infrared Imaging Spectrometer Next Generation (AVIRISng) deployed on a twin otter aircraft.

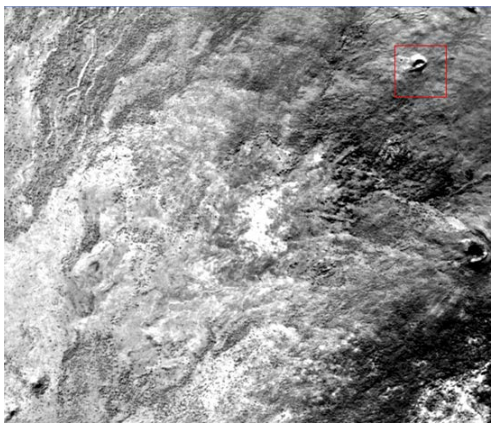

Figure 9: AVIRIS Hawaii: Band 19 of the hyper-spectral Image. The full cube has 224(bands) x 6414(width) x 512 (length), 12 bits per Sample.

The average bit rate of the 32 frames compressed data achieved was 2.38 bits/sample for the synthetic image. The 32 frames (8,744,960 samples) were compressed in 219 milliseconds which represents a compression speed of 10.69 microseconds per spectra (427 samples) (Table 3).

Table 3. Data Compression Performance on VIRTEX-5 SX50T

Test Image	compression ratio Bits/Samples	Compression Ratio	Data Rate Samples/ Sec	Data Rate Spectra/Sec
Synthetic Image (512 frames)	2.38	1 : 5.5	40M	93,516

7. CONCLUSIONS

We presented in this paper a VIRTEX-5 SX50T and Virtex-6 LX240T FPGA implementation of a novel hyper-spectral data compression algorithm, the JPL adaptive push broom mode of the Fast Lossless (FL) compressor for push-broom instruments which is being formalized as the emerging CCSDS 123.0 standard for lossless multispectral and hyper-spectral image compression [4], [5]. The data compression intellectual property (IP) implementation targets the XILINX VIRTEX-5 SX50T and Virtex-6 LX240T FPGAs and provides a throughput of 40 M-Samples per seconds. The data compression IP can also run with the same performance on a space-qualified radiation hardened XILINX VIRTEX-5QV FX130, making the use of this data compression-IP compressor practical for satellites and planet orbiting missions with hyper-spectral instruments. Future development will address multi-data stream through a parallel version of the implementation

ACKNOWLEDGMENT

The research described in this paper was carried out at the Jet Propulsion Laboratory, California Institute of Technology.

REFERENCES

[1] C. Hartzell, L. Graham, T. Tao, H. Goldberg, J. Carpena-Nunez, D. Racek, C. Taylor, C. Norton "Data System Design for a Hyperspectral Imaging Mission Concept", In IEEE Aerospace Conference 2009, 9-13 March 2009.

[2] W. Campbell, N. M. Short, "Remote Sensing Tutorial", 2004 http://www.fas.org/irp/imint/docs/rst/Sect13/Sect13_9.html

[3] J. Venbrux, J. Gambles, D. Wiseman, G. Zweigle, W. H. Miller, and P.-S. Yeh, "A VLSI Chip Set Development for Lossless Data Compression," Ninth AIAA Computing in Aerospace Conference, San Diego, California, October 19–21, 1993.

[4] CCSDS, Lossless Data Compression, Recommendation for space data system standards vol. 121.0-B-1: CCSDS, 1997. (http://public.ccsds.org)

[5] M. Klimesh, "Low-Complexity Lossless Compression of Hyperspectral Imagery via Adaptive Filtering," The Interplanetary Network Progress Report, vol. 42-163, Jet Propulsion Laboratory, Pasadena, California, pp.1–10, November 15, 2005.

[6] CCSDS, Lossless Multispectral & Hyperspectral Image Compression, Draft Recommendation for Space Data System Standards vol. 123.0-R-1: CCSDS, 2011 (http://www.ccsds.org)

[7] N. Aranki, D. Keymeulen, M. Klimesh, A. Bakhshi, "Hardware Implementation of Lossless Adaptive and Scalable Hyperspectral Data Compression for Space," NASA/ESA Conference on Adaptive Hardware and Systems, pp. 315–322, July, 2009, San Francisco, CA, USA

[8] Alpha-Data XRM-CLINK SDK User Manual, revision V2.3, Alpha data Parallel Systems, 3507 Ringsby Court Suite 105, Denver, CO 80216, 2012 (http://www.alpha-data.com)