

# Reconfigurable Computing As an Enabling Technology for Single-Photon-Counting Laser Altimetry

Wesley Powell, Edward Hicks, and Maxime Pinchinat  
National Aeronautics and Space Administration  
Goddard Space Flight Center  
Microelectronics and Signal Processing Branch  
Greenbelt, MD 20771  
301-286-2823  
Wesley.A.Powell@nasa.gov

Philip Dabney, Jan McGarry  
National Aeronautics and Space Administration  
Goddard Space Flight Center  
Laser Remote Sensing Branch  
Greenbelt, MD 20771  
301-614-5858  
Philip.W.Dabney@nasa.gov

Paul Murray  
SEAKR Engineering  
Centennial, CO 80111  
303-790-8499  
paul@seakr.com

*Abstract* - Single-photon-counting laser altimetry is a new measurement technique offering significant advantages in vertical resolution, reducing instrument size, mass, and power, and reducing laser complexity as compared to analog or threshold detection laser altimetry techniques. However, these improvements come at the cost of a dramatically increased requirement for onboard real-time data processing.

Reconfigurable computing has been shown to offer considerable performance advantages in performing this processing. These advantages have been demonstrated on the Multi-KiloHertz Micro-Laser Altimeter (MMLA), an aircraft based single-photon-counting laser altimeter developed by NASA Goddard Space Flight Center with several potential spaceflight applications.

This paper describes how reconfigurable computing technology was employed to perform MMLA data processing in real-time under realistic operating constraints, along with the results observed. This paper also expands on these prior results to identify concepts for using reconfigurable computing to enable spaceflight single-photon-counting laser altimeter instruments.

## TABLE OF CONTENTS

1. Introduction.....	1
2. MMLA Instrument.....	2
3. MMLA Reconfigurable Computing Application ...	5
4. Spaceflight Instrument Application .....	6
5. SEAKR Re-Configurable Computer (RCC) .....	7
6. Spaceflight RC Application.....	11
7. Conclusions.....	11
References .....	11

## 1. INTRODUCTION

Laser altimeters are remote sensing instruments that have been used to measure land, snow, and ice elevation, as well as the height of vegetation and man-made structures. In general, these instruments function by firing a pulse of laser energy at the target surface, and measuring the time-of-flight (TOF) required for the energy to reflect off the surface and return to the instrument (see Figure 2.2). The range to the target is calculated using the TOF and the speed of light in the relevant medium. Using knowledge of the instrument attitude and GPS information, the location of the target reflector is geolocated in standard cartographic coordinates.

There are currently several methods for measuring this TOF, each with varying capabilities for extracting surface information from the returned signal. One method is to digitize the returned signal with a high-speed analog-to-digital converter. Alternatively, the returned signal can be compared to a threshold to obtain pulse widths that correspond to detected surfaces and structures. In another method, the leading edges of returned signal are detected. Figure 1.1 illustrates the different laser ranging methods and their abilities to accurately detect the height of multiple vegetation canopy structures.

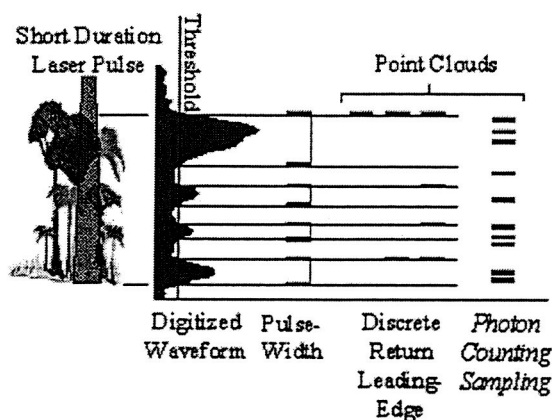


Figure 1.1 - Comparison of Laser Ranging Methods

The last method ranging method shown is Photon Counting. Single-photon-counting laser altimetry is a NASA patented technique [1,2] that has been shown to have superior capabilities for ranging to and detecting surface scatterers while requiring a minimum of laser energy. This technique uses a photon counting sampling method, whereby the laser fires at a very high rate (several kilohertz) and the TOF for multiple discrete photons are measured, converted to range and/or height, and then assembled statistically to recover valid signal events that represent the detected surface. The expected existence of and the ability to handle background photo-electron noise is fundamental to this approach. Non-signal events are uncorrelated random events, whereas the actual target ranges are relatively well correlated in range space as shown in Figure 1.2. The core enabling innovation in this approach is the patented statistical signal extraction algorithm around which the primary effort described in this paper is focused.

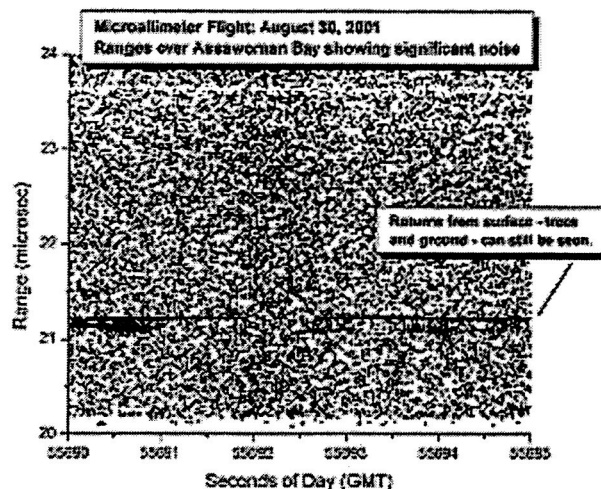


Figure 1.2 - Worst case example of raw TOF data showing the presence of correlated signal in high noise background (flying above dense clouds at daylight.).

Single-photon-counting laser altimetry demonstrates its advantages in developing high spatial resolution digital topographic databases, monitoring biomass and vegetation canopy heights, sea and lake levels, ice sheet thickness, etc. This technique also allows reductions in size, mass, and power the use of lower power, less complex passively Q-switched lasers for lower altitude airborne applications.

To demonstrate this laser altimetry technique, NASA Goddard Space Flight Center has developed the Multi-Kilohertz Microlaser Altimeter (MMLA). This aircraft based instrument has undergone a series of test flights that have successfully validated the overall instrument concept and demonstrated the science value of the data obtained.

## 2. MMLA INSTRUMENT

The MMLA is an airborne single-photon-counting laser altimeter developed at GSFC under the Instrument Incubator Program (IIP) for the purpose of demonstrating single-photon-counting laser altimetry. The Multi-KiloHertz Microlaser altimeter is so named because it uses a very compact, low-energy, sub-nanosecond pulse, solid-state Micro Laser as its source and relatively small (typically 10 to 20 cm in diameter) telescopes. The unique characteristics of this instrument are: (1) its use of a very low output energy laser; (2) its ability to be flown from high altitudes while meeting eyesafety requirements at FAA flight separation distances, thus allowing for continued operations above other aircraft; (3) its ability to resolve fine structural detail, and (4) penetrate shallow water.

## 2.1 MMLA Science Data

The MMLA has undergone several test flights that have demonstrated its ability to detect surface features as illustrated in Figure 2.1. In this data, collected during a test flight over the Maryland Eastern Shore, MMLA clearly detects the outline of man-made structures and vegetation canopy.

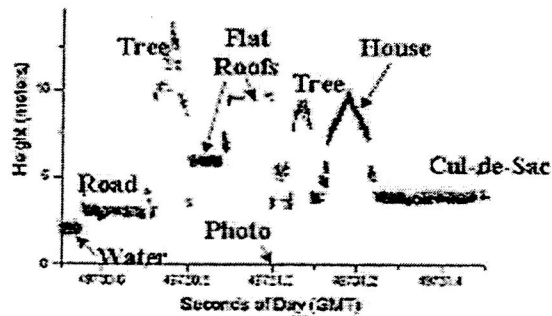
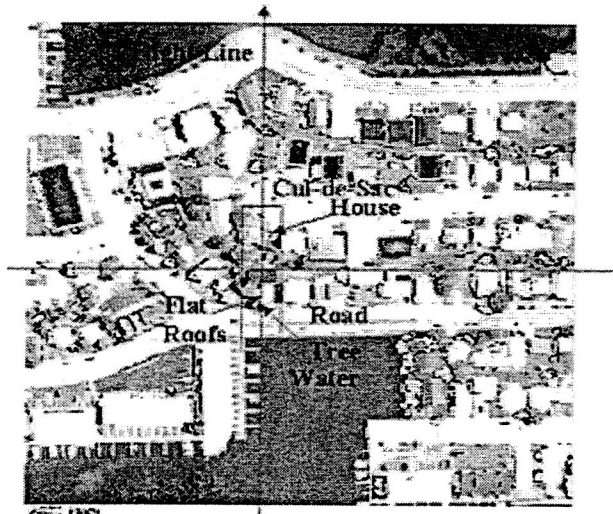


Figure 2.1 – MMLA Test Data

## 2.2 MMLA Data Processing

For each fire of its passively Q-switched laser, the MMLA detects and computes the time-of-flight for up to 16 returned photons as shown in Figure 2.2. A histogram is performed on these returns based on the laser fire time and the measured range of each return.

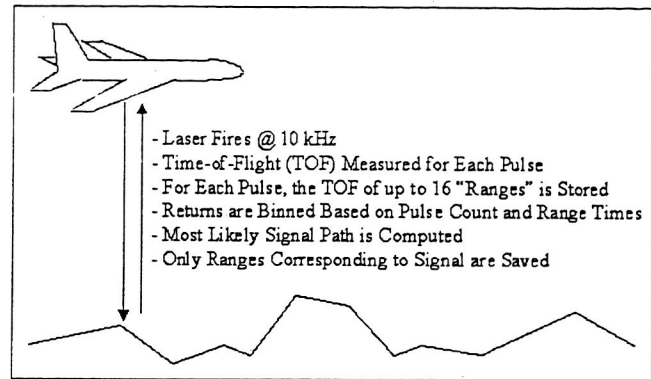


Figure 2.2 – MMLA Instrument Concept

*Plotdata Array Construction*—Illustrated in Figure 2.3, this results in the construction of a two dimensional "plotdata" array. The plotdata array is subdivided in the x dimension. This dimension corresponds to the laser fire time, and hence, the horizontal ground track of the aircraft) into frames. The array is subdivided in the y dimension into range bins. The entire span of the plotdata array in the x dimension is called a "superframe", which consists of an integral number of frames. The span in the y dimension is referred to as the range window.

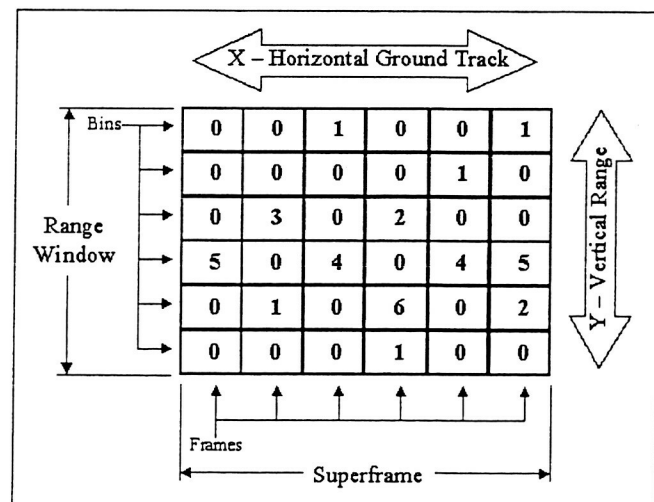


Figure 2.3 – Plotdata Array

Once the plotdata array is constructed, the MMLA selects the most likely signal path through the plotdata array. All returns falling along the selected signal path are then archived for future ground processing. This signal path through successive plotdata arrays is also displayed, providing a real-time indication of signal quality and ground terrain.

**Signal Path Selection**—The selection of the most likely signal path is by far the most compute intensive step in the MMLA signal processing. This is performed “brute force” through an iterative process where a score is computed for every possible path through the plotdata array (within a set of constraints). The score of each path is computed by accumulating the total number of returns in each cell (array element) along the path. However, a path is considered to be a potential signal path only if at least NN of the MM cells of the path exceed a pre-defined “cell threshold” parameter. The signal path (if one exists) with the highest score is then selected as being the most likely signal path.

A path through the plotdata array is comprised of a sequence of cells traversing the array from the first column to the last column. The paths are constrained by a “cell spread” parameter that determines the set of valid cells in column x that could be transitioned to from a cell in column x-1. Hence, cells in adjacent columns cannot vary in vertical distance by more than (cell spread/2) cells. For example, with the typical cell spread of 3, a cell at location [x, y] can transition to cell [x+1, y-1], [x+1, y], or [x+1, y+1]. Figure 2.4 illustrates the path from a single starting cell with a cell spread of 3 and a length of 3.

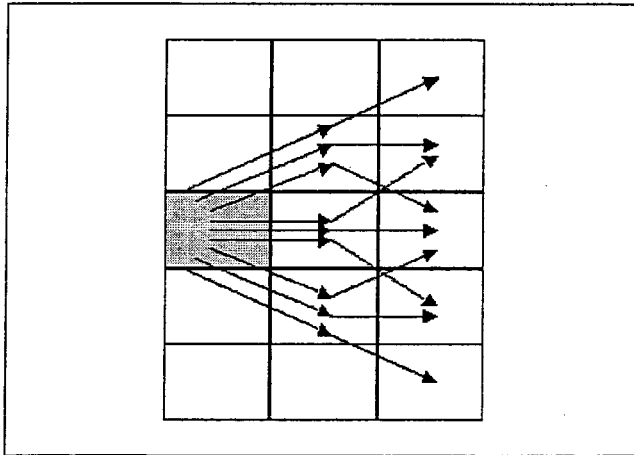


Figure 2.4 – Paths from Single Cell

The following equation gives the total number of paths through a plotdata array with dimensions [MM x Num\_Bins], where MM is the superframe size and Num\_Bins is the number of bins in the range window.

$$\text{Number of Paths} = \text{Num\_Bins} \times \text{Cell\_Spread}^{(\text{MM}-1)} \quad (1)$$

For a Cell\_Spread = 3 and a plotdata array with where MM = 8, Num\_Bins = 50, there are a total of 109,350 paths that must be computed.

### 2.3 MMLA Configuration

The MMLA instrument consists of two standard equipment racks containing COTS (commercial off-the-shelf) components and a transmitter/receiver assembly containing the laser, telescope, and detector [2]. The MMLA configuration is illustrated in Figure 2.5.

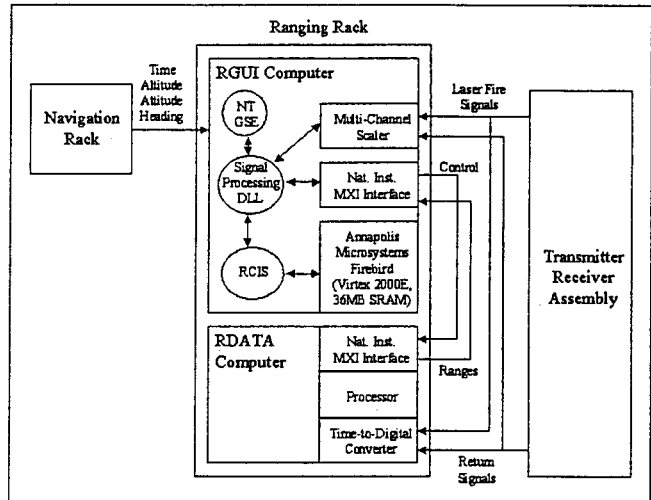


Figure 2.5 – MMLA Configuration

The Ranging Rack performs all of the ranging measurements, instrument control, and signal extraction processing. This rack houses two computers, the Ranging System Data Collector (RDATA) Computer and the Ranging System Graphical User Interface (RGUI) Computer. The RDATA is a VXI based computer that collects all ranging data and forwards it to the RGUI. The RGUI performs the instrument control, signal extraction processing, data archiving, and provides the user interface to the instrument. This PC is configured with dual Pentium 850MHz processors and uses the Windows 2000 operating system. The RGUI uses the NTGSE software which provides a generic platform for instrument control and monitoring. The MMLA signal extraction software is implemented as a separate DLL, containing functions called by the NTGSE.

The RGUI also includes an Annapolis Microsystems Firebird board to perform reconfigurable computing. This board is a commercially available PCI based FPGA computing board, which includes a Xilinx XCV2000E FPGA and 36 MB of SRAM arranged in 5 separately addressable banks.

The Navigation Rack houses the Navigation and Camera Control (NAV) Computer, which provides a GPS system time reference and determines the aircraft heading, altitude, and attitude. This data is sent via a shared file interface to

the RGUI where it is included with the range data for ground processing.

The Transmitter/Receiver Assembly houses the instrument laser, telescope, and detector. This assembly is mounted to the floor of the aircraft cabin where it transmits the laser pulses and receives the returns through a window in the bottom of the aircraft.

### 3. MMLA RECONFIGURABLE COMPUTING

#### APPLICATION

The MMLA onboard signal extraction is performed using a reconfigurable computing (RC) application [3]. This RC application consists of an FPGA image to perform the signal processing and a software module to integrate the hardware processing with the existing signal extraction software. The FPGA image consists of a set of core logic along with several generic modules from Annapolis Microsystems.

Written in C and implemented as a Windows 2000 DLL, the RCIS (Reconfigurable Interface Software) integrates the FPGA processing with the existing signal extraction software. This software (1) initializes the Firebird board and the FPGA image, (2) receives and formats instrument return data for hardware processing, and (3) displays and archives the processed data. The RCIS receives instrument data as a "log record", containing one second of instrument returns, from the existing signal extraction DLL via a shared memory interface.

#### 3.1 Processing Flow

Upon receiving a log record, the RCIS constructs an array of  $[t, r]$  pairs corresponding to discrete returns, where "t" is the laser fire time and "r" is the measured range of that return. This array contains the consecutive  $[t, r]$  pairs that make up a single superframe. The RCIS then transfers this array to the "firedata" memory on-board the Firebird for hardware processing.

In hardware, three main processing steps occur as shown in Figure 3.1. In the Build Plotdata step, the  $[t, r]$  pairs in the firedata memory are binned to construct a plotdata array. As the plotdata array is constructed, each firedata memory location is annotated with its corresponding frame and bin number. In the Path Score step, all possible paths through the plotdata array are scored and the most likely signal path (if any exist) is stored. In the final Path Trace step, the frame and bin numbers of all  $[t, r]$  pairs in the firedata memory are compared against the frame and bin numbers of the cells along the selected path. Those  $[t, r]$  pairs that were binned into the cells of the selected path are then annotated with a "z" bit that indicates that it is a signal return.

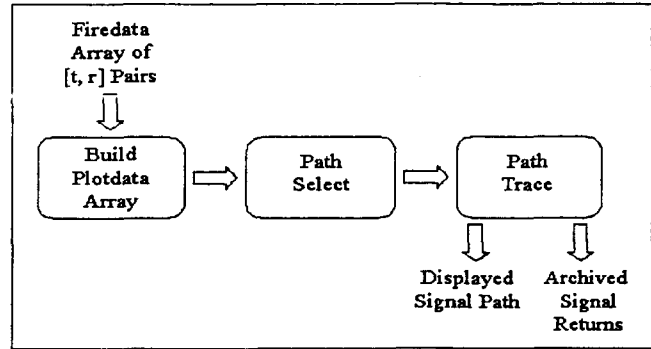


Figure 3.1 – RC Application Processing Flow

The RCIS then reads the plotdata array from the Firebird and displays it in a window as a scatter plot. This provides the user with a real time display of the detected surface. The RCIS also reads the firedata array from the Firebird and archives to disk the  $[t, r]$  pairs that are marked as signal via the z bit.

#### 3.2 FPGA Image Design

The FPGA image, implemented with roughly 9,000 lines of VHDL code, consists of a set of core logic that performs the signal processing. This core, illustrated in Figure 3.2, is divided into three basic modules, the Build Plotdata Module, the Path Select Module, and the Path Trace Module, each corresponding to a specific processing step. The Plotdata Memory is also implemented in the FPGA image as a 4K x 16 dual port BlockRAM.

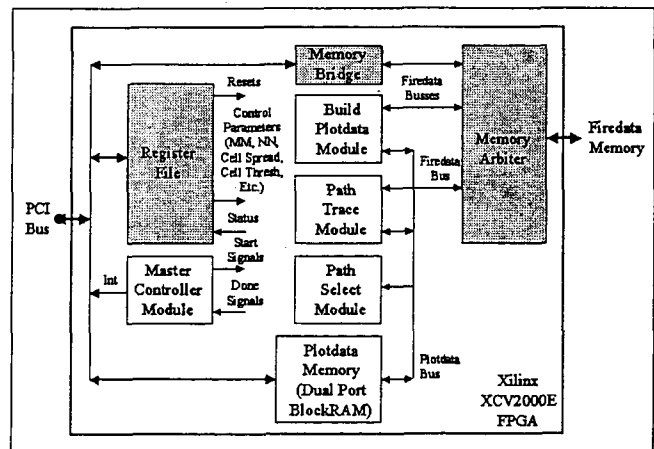


Figure 3.2 - FPGA Image Design

The FPGA image also uses several generic VHDL modules provided by Annapolis Microsystems (shown as shaded blocks in Figure 3.2). These modules perform such functions as clock management, off-chip memory arbitration, and register interfaces to the PCI bus.

**Build Plotdata Module**—This module constructs the plotdata array by binning the  $[t, r]$  pairs in the firedata array. To perform this function, this module is further divided into four lower level modules that (a) initially reset the firedata and plotdata arrays to zeros, (b) determine the minimum and maximum  $r$  values, (c) compute the (vertical) bin and (horizontal) frame that each  $[t, r]$  pair falls into, and (d) annotates the firedata array locations with bin and frame numbers and increments appropriate locations in the plotdata array.

**Path Select Module**—The Path Select Module, iteratively scores each possible path through the plotdata array, and selects the path with the highest score. This module consists of three lower level modules that (a) generate “path vectors” corresponding to each possible path through the plotdata array from a single cell in the first column to the last column, (b) iterates these paths for each cell in the first plotdata array column, and (c) computes the scores for each path. This module outputs the best start bin, the best path vector, the best score, and a flag indicating whether or not a valid path was found.

**Path Trace Module**—Based on these outputs, the Path Trace Module then annotates the plotdata array to indicate which cells contain signal, and annotates the firedata array to indicate which  $[t, r]$  pairs are signal returns. This module contains lower level modules that (a) generate the bin and frame indices of each cell along the best path, (b) flag the plotdata array locations corresponding to these indices, (c) compare these indices to each firedata array element, and (d) flag the firedata array locations corresponding to the indices.

### 3.3 Results

The RC application has been integrated with the existing MMLA data system has been successfully flight-tested, during which the RC application operated in parallel with the software processing. Comparison of the output data verified the correct operation of the RC application.

The FPGA image currently requires roughly 15% of the resources of the XCV2000E FPGA and operates at a speed of 74 MHz. With this performance, the RC application can process 1,000,000 paths per second. For the current airborne MMLA data system, the RC application shows a performance gain over the software processing. With the laser firing at 8.5 kHz, the RC application processes data using an MM of 6, where the software can only process with an MM of 4. This indicates that the RC application is capable of processing 9 times more paths than the software processing.

Figure 3.3 shows the test flight results of the MMLA RC application. The upper trace is the signal extracted using software operating with an MM of 1. The lower trace is the signal extracted by the RC application operating with an MM of 6.

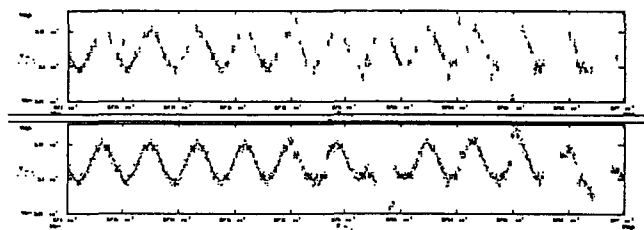


Figure 3.3 – Test Flight Results

Using test software, the performance of the RC application was compared the performance of software processing on a 1.2 GHz Pentium workstation for varying superframe sizes. Results indicate that the RC application does not match the performance of the software processing for values of MM less than 6. This is due to the fact that at low values of MM, a higher proportion of time is required to transfer data to the Firebird and to build the plotdata array. However, for larger values of MM, the RC application shows performance gain of 1.5 to 2.3 over the software processing.

As a key purpose of this IR&D project is to develop reconfigurable computing for space applications, it is more pertinent to compare the performance against currently available spaceflight qualified processors such as the RAD6000 and the RAD750. Although the performance of the MMLA signal extraction on these processors has not been measured, a very rough comparison can be inferred by comparing their standard Dhrystone MIPS benchmark results. With the 1.2 GHz Pentium, RAD6000, and RAD750 processors rated at 2230MIPS, 35 MIPS, and 300 MIPS respectively, it can be inferred that the RC application would have a performance gain of roughly an order of magnitude over the RAD750 and two orders of magnitude over the RAD6000. This figure also assumes that the RAD6000 or RAD750 processor is dedicated solely to MMLA data processing.

## 4. SPACEFLIGHT INSTRUMENT APPLICATION

There are many spaceflight applications for single-photon-counting laser altimetry. These applications include low earth orbit (LEO) missions measuring ground topography, vegetation canopy height, ice sheet thickness, and shallow water depth. Planetary applications include surface mapping missions for the moon, asteroids, Mars, and Jupiter's moons. Given its capabilities for producing surface maps with exceptional resolution, this laser

altimetry technique can also be used to map potential landing sites for other spacecraft.

#### 4.1 Spaceflight Data System Constraints

While the MMLA instrument successfully demonstrated the feasibility and benefits of this laser altimetry technique on an aircraft platform, a spaceflight instrument must have an entirely different implementation. MMLA was implemented with commercial-off-the-shelf (COTS) components to operate on a large aircraft. It consisted of two standard equipment racks, operated from 110 VAC, and was designed for manual operation.

A spaceflight single-photon-counting laser altimeter must be implemented with a spaceflight-qualifiable architecture that (a) reduces size, mass, and power, (b) interfaces to a spacecraft bus, (c) operates autonomously, (d) survives the mechanical stresses of launch, and (e) operates in a thermal vacuum environment. Additionally, the instrument electronics must operate in a radiation harsh environment that can vary widely from mission to mission. Specifically, the electronics must tolerate the expected total ionizing dose (TID), mitigate any single event upsets (SEUs), and be immune to single event latchup (SEL).

SEU mitigation is a key issue for the RC application, as it operates on SEU-soft SRAM-based FPGAs. There are two ways in which an SEU can effect the operation of a reconfigurable computing application [4]. First, an SEU can upset the configuration memory of the FPGA, which holds the logic design programmed into the device. This can cause a change in the FPGA logic or internal routing, which may result in a functional error. An SEU can also upset storage elements that are part of the FPGA logic design will result in a functional error. To operate reliably in a space environment, the RC application must mitigate both of these types of errors.

#### 4.2 Spaceflight Data System Architecture

One data system architecture for a spaceflight single-photon-counting laser altimeter, illustrated in Figure 4.1, is based on the industry standard 6U Compact-PCI bus. This architecture uses COTS Processor and RCC boards along with custom Housekeeping/Scan-Motor Drive and Timing Electronics boards. In this architecture, the reconfigurable computing application performing signal extraction is comprised of logic running on FPGAs on the RCC board along with control software running on the Processor board. This data system architecture is currently being prototyped at GSFC as part of the AMPL (Advanced Mapping Photon-counting Lidar) instrument development.

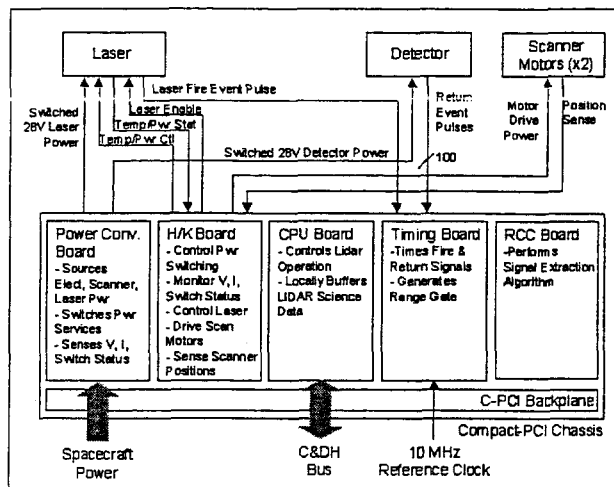


Figure 4.1 – Compact-PCI Data System Architecture

The above architecture has the benefits of using an industry standard architecture, which include the availability of COTS space qualified Compact-PCI hardware as well as IP cores for use within FPGAs in custom designed hardware.

While this architecture is an attractive option for instruments on LEO missions, its size, mass, and power make it unattractive for use on planetary missions, where the launch cost per kilogram is far higher. For these missions, a higher degree of integration is needed where more functions are combined on fewer boards. With these constraints, it may be beneficial to integrate the reconfigurable computing hardware (the FPGAs, SRAM, and control circuitry) with the timing electronics on a single board. Alternatively, the reconfigurable computing hardware can be integrated into the Processor board. Yet another option is to base the data system on a "Platform FPGA", such as the Xilinx Virtex-II Pro, which combine FPGA logic resources, processors, and memory on a single device. While they are yet untested for radiation susceptibility, these devices have great potential in this application, provided that they are found to be radiation tolerant.

### 5. SEAKR RE-CONFIGURABLE COMPUTER (RCC)

SEAKR Engineering has developed a high performance Re-Configurable Computer (RCC) for space applications using an array of space qualified re-programmable FPGAs. Unlike conventional One Time Programmable (OTP) FPGAs typically used in space applications, re-programmable FPGAs support an unlimited number of re-programming cycles. Design updates, multi-platform support, or even changing mission objectives post-launch

are possible. When these devices are applied in a flexible architecture, a single board design can be used for multiple platforms, applications, and missions by changing the VHDL code and application program. Reduced cycle time, risk, and cost are all by-products of this technology. RAM based FPGAs, however, present new design challenges with respect to fault tolerance in a radiation environment.

### 5.1 Xilinx XQVR FPGA

The Xilinx XQVR product line is a radiation tolerant version of the Xilinx Virtex series FPGAs. The Virtex series is an SRAM based device that supports up to 1 million configurable gates and system performance of greater than 100 MHz. The radiation tolerant version utilizes an epitaxial process and the Virtex commercial mask set. This creates a product that is latch-up immune to a LET of  $>125\text{MeV}\cdot\text{cm}^2/\text{mg}$ , radiation tolerant to 100krad TID, and still supports the commercial tools and libraries. Though latch-up immune, these devices are susceptible to Single Event Upsets (SEU). These upsets manifest themselves not only in the traditional sense of logical state changes in registers and memory elements but also in device configuration errors.

Xilinx has extensively characterized the Virtex product line with respect to Single Event Effects (SEE). There are several excellent papers that detail the result of their testing [4,5]. SEE testing demonstrates that the devices are susceptible to traditional SEUs that alter logic states of flip-flops, latches, and RAM elements but also to upsets that alter the internal configuration of the device. Unlike ASICs, which have all their internal routing hardwired, the Virtex uses logic stored in SRAM to define the device routing and configuration. These SRAM latches are susceptible to SEU's and therefore a hit to one of these latches by a high-energy particle can change the configuration of the device. Because of these configuration latches, SEU rate calculations for re-configurable FPGAs are different from traditional ASIC SEU analysis.

In traditional ASIC designs, an upset to a memory element such as a register or SRAM, will essentially cause a functional error in the design. However, an upset in an FPGA configuration latch does not necessarily cause a functional error. A 300k FPGA gate Virtex device has approximately 1.75M total bits that are susceptible to SEUs. Of these 1.75M total bits, approximately 85% of them control routing pips in the device. Table 5.1 shows the number of bits per device element for an XQVR300 FPGA [5].

Table 5.1 Latch types in a XQVR300 FPGA

Latch Type	Function	# Bits
CLB	Configuration Logic Blocks	6,144
IOB	Programmable IO Blocks	948
LUT	Look Up Tables	98,304
BRAM	Block RAM	65,536
	Routing and Other Bits	1,579,860
	Total	1,750,792

A typical design (80%-90% device utilization) uses less than 10% of the routing pips. This means that when a high-energy particle hits a cell, there is a high percentage chance that the particle will hit a cell that is not actually used in the design and will not cause a functional error. This implies that the functional error rate is much lower than the raw SEU rate for the device. Work done by Xilinx, BYU, and LANL have verified this theory [7,8]. The actual functional upset rate is design dependant but their work has shown typical functional upset rates as low as 1 functional error per 6 configuration latch upsets for the Virtex I devices and perhaps 1 in 10 for the Virtex II devices. Device element saturation cross-sections can be obtained from Xilinx data sheets.

The results of radiation testing of the Xilinx XQVR devices show that the devices are radiation tolerant to 100k rad TID and latch-up immune. However, these devices are susceptible to Single Event Upsets in both configuration RAM and logic units. Single Event Functional Interrupts (SEFI) in the POR and selectMap circuits has also been identified, however the mean time between occurrences is in the 10's of years. And the recovery from the SEFI only requires a full reconfiguration and does not require any power cycling. Unmitigated, the devices exhibit multiple SEUs per day. In order to use these devices reliably in space applications, mitigation methods must be employed.

### 5.2 Single Event Effects and Mitigation

To develop a reliable system using these devices in space applications, we must be able to detect and mitigate the many different SEE characteristics of the device. In general, the mitigation techniques required for these devices fall into two categories: system level and device level. System level SEE mitigation techniques typically do not effect the processing performance of the system. Table 5.2 shows the system level SEE mitigation.

Table 5.2 System Level Mitigation

Name	Description
Configuration readback	Read back configuration bitstream and detect upsets in bitstream
Partial Reconfiguration	Reprogram FPGA. Single or multi-frame
POR SEFI Detection	Detection and handling of the POR SEFI
SelectMap SEFI	Detection and handling of the selectmap SEFI. This SEFI prevents proper SelectMap interface operation

Many of the device level mitigation techniques have a dramatic impact on system performance. Table 5.3 details device level SEE mitigation

Table 5.3 Device Level Mitigation

Name	Description
Triple Modular Redundancy (TMR)	Triplification of design with majority voting
Half latch removal	Performed at the design level
Elimination of dynamic LUT usage	The usage of LUTs as shift registers prevents the use of configuration readback
Block RAM EDAC	Error detection and correction on block RAM

System and device level mitigation must be employed for a reliable system. System level techniques are fairly complicated, however, since they do not impact the processing performance, they can be implemented once at the system level and used for all applications. This allows a designer to ignore all the complexities related to system level mitigation and focus on FPGA design and device level mitigation. This was the approach taken with the SEAKR RCC boards. The board architecture provides all the system level mitigation, and it is up to the FPGA designer to implement an acceptable amount of device level mitigation for the target application.

The RCC provides all required system level mitigation. An on-board configuration controller, implemented in a radiation tolerant Actel FPGA, provides configuration scrubbing for all four FPGAs. The device accesses the on-board non-volatile memory from the selected program location and programs the devices. The board contains 256 Mbits of triplicated non-volatile memory. This is enough to store more than 16 different configuration files. The scrub routine then performs a configuration readback on each FPGA at a programmable interval. CRC codes are calculated during the readback cycle and compared with known good CRC codes for that frame of configuration

data. If a mismatch occurs the frame location is logged and at the end of the device readback, the bad frame locations are retrieved from non-volatile memory and reprogrammed into the FPGAs. The device also detects POR SEFI and SelectMap SEFI modes. Upon SEFI detection the device is programmable to generate either an automatic full reconfiguration of the FPGA in SEFI or generate an interrupt and wait for processor interaction.

Device level mitigation is normally implemented by using TMR with majority voting [6]. Unfortunately, TMR is costly in terms of area and speed and can dramatically effect the performance of the design. However, speed, density, and power consumption is dramatically being improved for each generation of FPGAs. Making the increase area, power and speed penalties acceptable for many designs. For example present Virtex-II Pro devices have more than 8M FPGA gates, system speeds of 300 MHz, and 8Mbits of internal memory. Though this device is not radiation tested yet, it provides insight into the possible roadmap of capability. Today, typical designs will typically TMR critical FSM controls and elements with feedback where an upset could persist even after the error was removed. Data path or functions where upsets will quickly "flush-out" of the system are not normally TMR'd. The concept of this approach is to allow the occasional transient error but prevent the persistent error. Other methods of SEU mitigation have also been explored. Data buffering with the use of test vectors can provide a high level of SEU mitigation without the high cost of TMR. This approach is more complex from a system design standpoint but provides exceptional performance when TMR is not feasible.

### 5.3 RCC Design

The SEAKR RCC system consists of four 1 million gate Xilinx FPGA. Each FPGA has an interface to an internal PCI bus, shared FPGA bus, high speed I/O port, and a 256 Mbyte local memory array. Figure 5.1 shows a block diagram of the SEAKR RCC board. The prototype board also provides a PMC slot to allow prototyping of COTS PMC technologies. The Compact-PCI backplane interface allows rapid integration into Compact-PCI designs. The system is a 6U form factor with conduction cooling. Figure 5.2 shows a picture of the prototype RCC board.

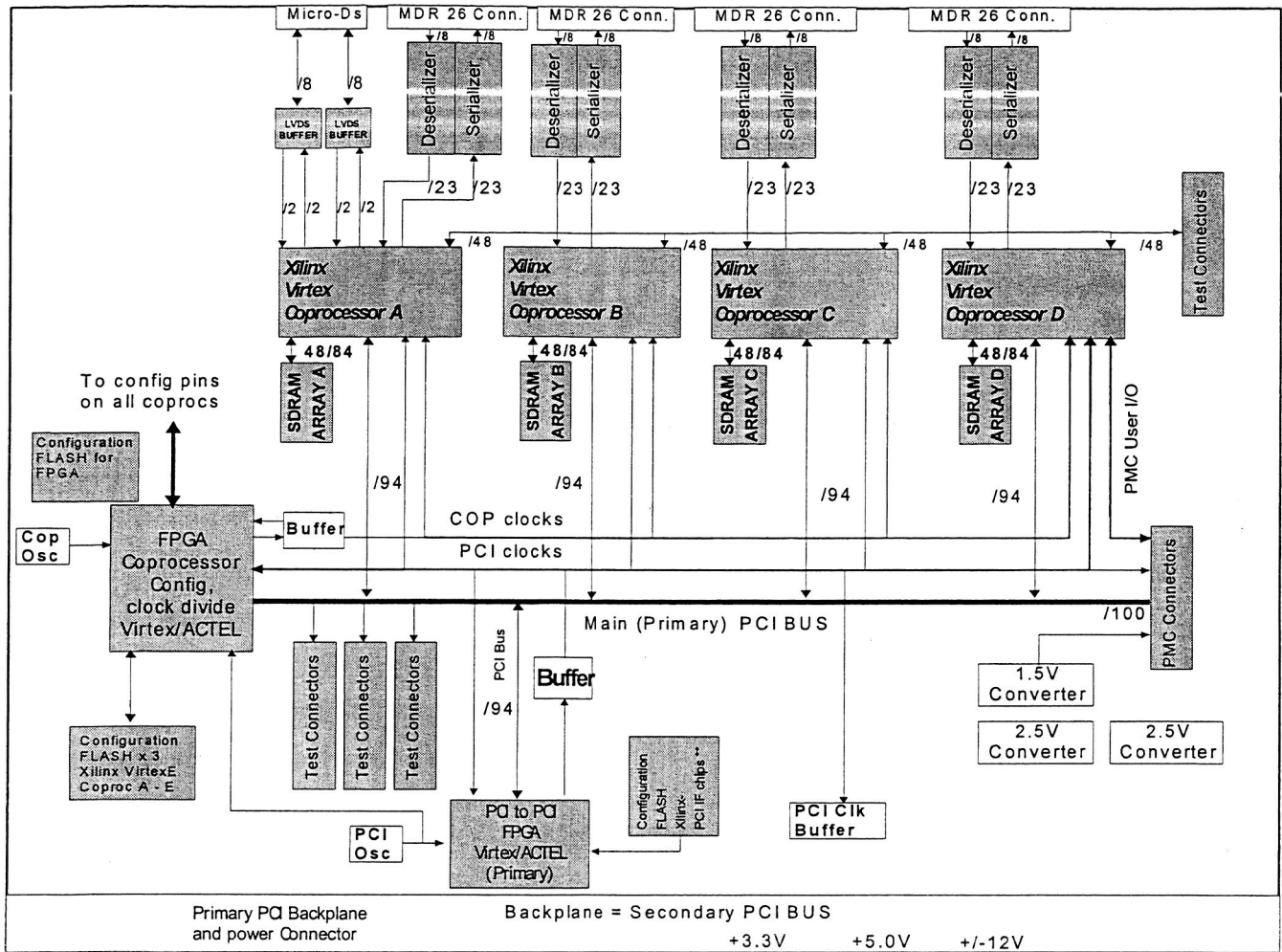


Figure 5.2 - SEAKR RCC Block Diagram

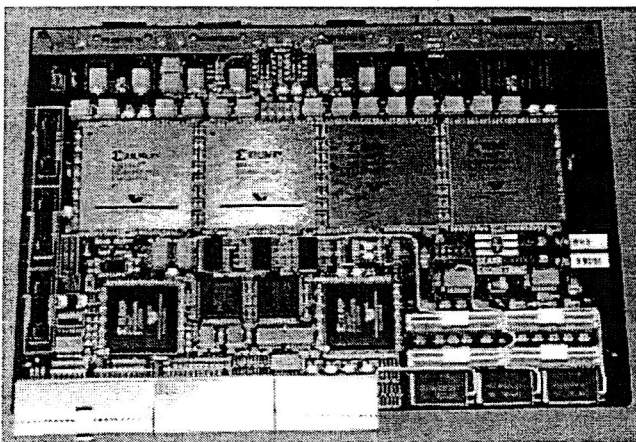


Figure 5.1 - Photograph Of Prototype SEAKR RCC

Re-configurable FPGAs for space applications are ushering in new advances in processing by enabling re-configurable computing. By using configuration RAM scrubbing techniques and Triple Module Redundancy, systems that

exhibit low upset rates can be realized. By implementing these devices in a flexible architecture with an industry standard interface, many applications can be implement on a common platform reducing program cost and risk.

## 6. SPACEFLIGHT RC APPLICATION

In the Compact-PCI data system concept for a spaceflight single-photon-counting laser altimeter, signal extraction can be performed via a reconfigurable application. This application consists of software running on the Processor board and logic programmed into the four FPGAs on the RCC board. Given the flexibility of the Virtex FPGA and the SEAKR RCC board, there are many possible ways to implement the signal extraction logic. Perhaps the simplest option is to program identical copies of the MMLA signal extraction logic, modified for the RCC board, into each FPGA. Illustrated in Figure 6.1, each set of logic would operate in parallel, with the sequencing controlled by a dedicated task in software. This implementation has the advantage that a single FPGA design can be designed, simulated, and standalone tested. After debugging, the design can then be used on all four FPGAs.

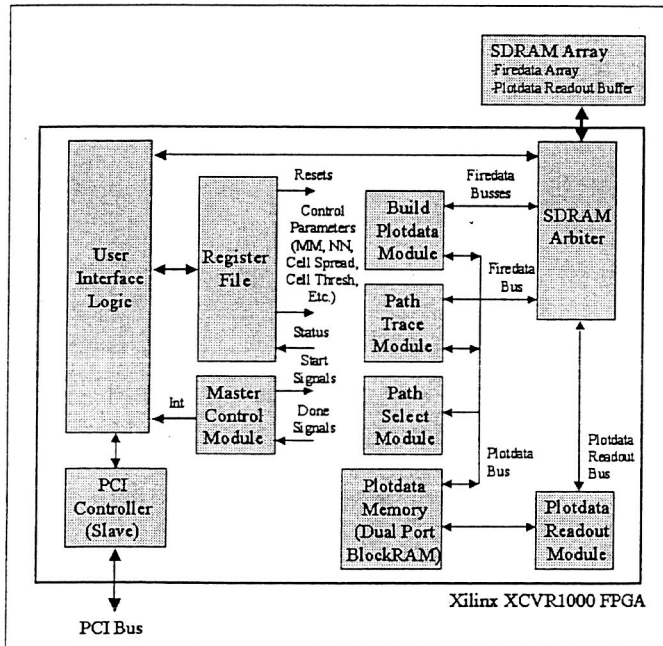


Figure 6.1 – Spaceflight RC Application FPGA Design

As previously stated, the RC application must perform device level SEU mitigation. One option for this mitigation is to implement TMR on all circuitry. However, given the device utilization for the MMLA FPGA logic, it is unlikely that the XQVR1000 would have the capacity to fully accommodate the triplicated logic.

A more efficient approach is to check for functional errors after processing each superframe, and then reprocess the data when an error is detected. With this approach, some of the circuitry, specifically control logic, would still require TMR. However, much of the circuitry would rely on a set

of relatively simple checks to detect functional errors. These checks include:

- Summing the contents of all plotdata array elements and verifying that the total equals the number of firedata array elements.
- Computing a checksum on the firedata array when written to the FPGA and then verifying it upon readback.
- Verifying that sum of the plotdata array elements along the signal path equals (a) the total number of firedata array elements flagged as signal and (b) the computed best path score.
- Performing limit checks at intermediate processing steps.
- Maintain a software time out for superframe processing.

While requiring only minimal FPGA logic resources and software overhead, these checks along with selective use of TMR can allow the signal extraction algorithm to be reliably performed using this spaceflight RC application.

The overall performance of the spaceflight RC application can be inferred from the measured MMLA performance results. For the MMLA, a single set of signal extraction operating at 74 MHz processed 1,000,000 paths per second. Assuming that each of the four FPGAs on the RCC board will operate in parallel at the 33 MHz Compact-PCI bus speed, the spaceflight RC application should be capable of processing roughly 1,750,000 paths per second.

## 7. CONCLUSIONS

Single-photon-counting laser altimetry is a new laser altimetry technique that offers significant advantages in resolution, size, mass, power, and complexity. As demonstrated with the MMLA instrument, reconfigurable computing is an essential technology for performing the onboard signal extraction needed by single-photon-counting laser altimeters.

An instrument data system, based on the commercially available, spaceflight qualified SEAKR RCC can be achieved that will enable single-photon-counting laser altimeters for LEO missions. Operating on this Compact-PCI data system, a spaceflight RC application, based on the MMLA design, can be implemented that can exceed the MMLA performance while mitigating device level SEUs.

## REFERENCES

- [1] J. Degnan, J. McGarry, T. Zagwodzki, P. Dabney, J. Geiger, R. Chabot, C. Steggerda, J. Marzouk, A. Chu "Design and Performance of an Airborne Multikilohertz

Photon-Counting, Microlaser Altimeter", ISPRS Workshop on Land Surface Mapping and Characterization Using Laser Altimetry. October. 2001b. Annapolis. MD.

[2] J. Degnan, J. "Photon-Counting Multikilohertz microlaser altimeters for airborne and spaceborne topographic measurements", J. Geodynamics, 34, pp. 503-549, 2002a.

[3] W. Powell, P. Dabney, E. Hicks, J. McGarry, M. Pinchinat "Application of Reconfigurable Computing Technology to Multi-KiloHertz Micro-Laser Altimeter (MMLA) Data Processing", Military and Aerospace Applications of Programmable Logic Devices (MAPLD), September 2002, Laurel, MD.

[4] C. Carmichael, "Correcting Single-Event Upsets through Virtex Partial Reconfiguration," Xilinx Application Note XAPP216, June, 2000

[5] E. Fuller, M. Caffrey, P. Blain, C. Carmichael, N. Khalsa, A. Sasazar, "Radiation Test Results of the Virtex FPGA and ZBT SRAM for Space Based Reconfigurable Computing", MAPLD 1999 Proceedings, C\_2, September 1999

[6] C. Carmichael, "Triple Module Redundancy Design Techniques for Virtex FPGAs", Xilinx Application Note XAPP197, June, 2001

[7] P. Graham, et al, "Consequences and categories of SRAM FPGA configuration SEU's", 2003 Military and Aerospace Programmable Logic Devices International Conference, September 9-11, 2003.

[8] J. Fabula, et al, "The NSEU Response of Static Latch Based FPGAs", 2003 Military and Aerospace Programmable Logic Devices International Conference, September 9-11, 2003.