

DSP Hardware in AMSAT P4 Geosynch Ride-share

Rincon Research Team

Dr. Michael Parker, KT7D

John Wallace KE7QVC

John Forinash, N8NJD

John Jensen

Presentation at AMSAT General Meeting, October 2015

1. Background

This spring Mike Parker was approached by Bob McGwier about the possibility of Rincon Research Corp. donating a DSP-based radio as part of ride-share opportunity on an Air Force satellite destined for geosynchronous orbit. Rincon was building hardware known as LPFE (Low Power Front End) for terrestrial use and planning to extend this hardware to space applications.

Rincon's management quickly approved donation to get started of at least one LPFE and a \$25K IRAD budget for preliminary work with AMSAT.

The purpose of this presentation is to describe:

- 1) How the LPFE fits into the rest of the package being developed by AMSAT
- 2) What the LPFE is (and isn't) capable of
- 3) How the LPFE operates in a terrestrial environment and changes necessary for space
- 4) Our concept of the development and test issues
- 5) The development/test facility currently interfaced to the internet at Rincon in Tucson and how it might be used

Figure 1 pictures the host satellite .



Figure 1: First meeting at Millennium Space Systems (MSS) showing a mockup of Air Force satellite which will carry a small AMSAT ride-share package. Rincon's Low-Power Front End (LPFE) is shown at left (terrestrial version, not to scale)

2. Payload

Figure 2 is a simplified concept of the AMSAT payload. Once the satellite is in geosynchronous orbit one side of the satellite will face earth. On this side will be mounted receive and transmit antennas. For purposes of this paper some assumptions have been made about the components that Rincon is not building. We have also left out the host spacecraft (which supplies power), the power supply of the AMSAT payload, etc. Frequencies illustrated are our current understanding.

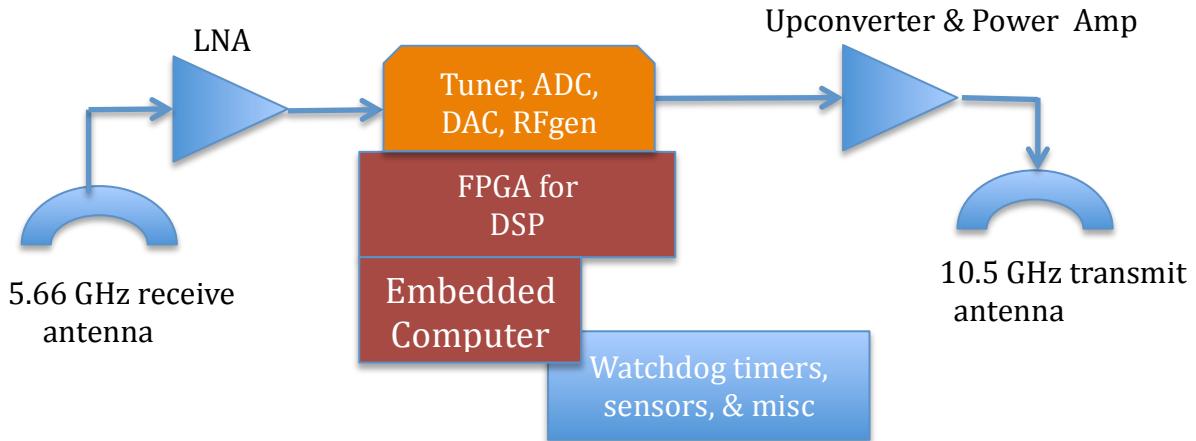


Figure 2: Simplified payload block diagram. Blue items are external to the LPFE DSP/Computer board. The LPFE board in brown and orange is Rincon supplied.

On the left of Figure 2 is the uplink antenna with a LNA setting the uplink noise figure that in turns feeds an amplified signal to the LPFE. The LPFE may be viewed performing three main functions.

- 1) RF to digital and digital to RF interface module. This uses the same Analog Devices chip as in the B200/B210 boards that some of you may own. It directly tunes the uplink frequency of 5.66 GHz, but the output will be at an IF frequency which must be externally upconverted to 10.5 GHz and amplified for transmit.
- 2) An FPGA fabric for performing heavy-lifting DSP such as tuners, demodulators, signal sources, etc. occupies part of a Zynq 7030 or 7045 Xilinx(R) system on a chip.
- 3) The same Zynq chip contains two ARM computer cores for performing general purpose computing functions. In the terrestrial LPFE a version of Linux running on the ARM controls the signal processing and provides a digital control interface with the outside world. AMSAT plans to also run software that monitors and controls their payload on the ARM. This will require close coordination and testing.

A final important item in the diagram is an external watchdog circuit to provide reset commands in the event that something goes wrong. More about that later.

3. LPFE Overview

The LPFE is an existing Rincon Research design that is being used in terrestrial applications. It is an evolution of our Mountain Brik Product line that usually operates interfaced to a control computer either directly or via an Ethernet. A photograph of a terrestrial LPFE is in Figure 3.

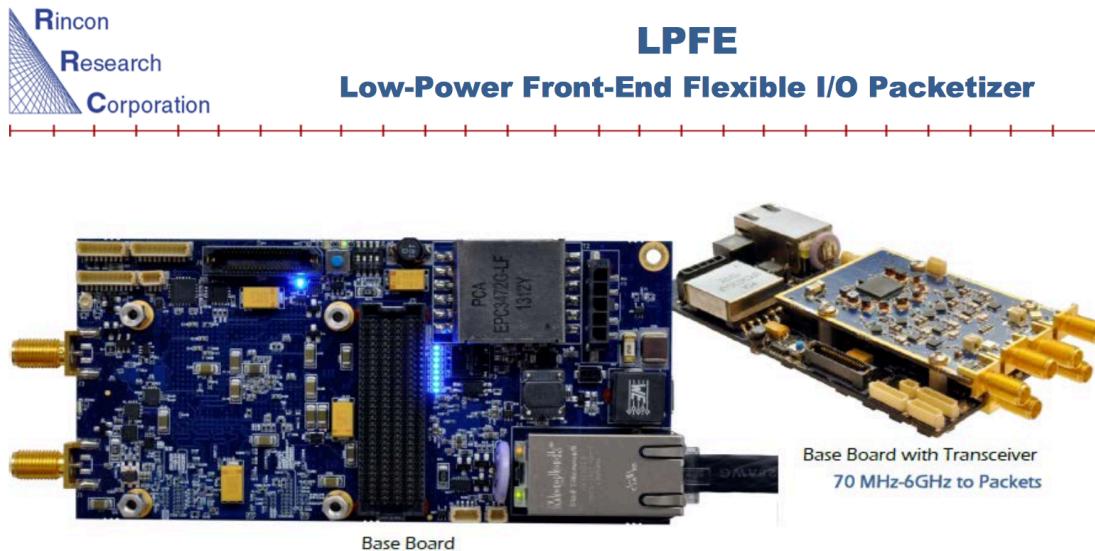


Figure 3: Photo of a terrestrial LPFE with mother board on left and combination with mounted daughterboard at right. Not shown on the reverse side is a Zynq 7030 (or 7045) system-on-a-chip. A conduction-cooled head sink removes heat from this and other hot chips.

Although schedules appear to be very tight, Rincon hopes to improve the board design for use by AMSAT. Planned modifications include:

- 1) Power supply more resistant to latch-up
- 2) Flash memory radiation resistant
- 3) Conduction cooling for attaching to spacecraft cold plate instead of air-cooled fins
- 4) The daughter card will be incorporated into the main card providing a single-card solution

Interfaces to AMSAT-provided watchdog circuits will probably be via existing connectors.

Major software changes are envisioned for the space version. These will be discussed later.

4. Schedule Drives Philosophy

Schedule is the thing that concerns Rincon the most. We understand that the AMSAT payload must be delivered to Millennium Space Systems for integration by July 2016. In order to meet this schedule, Rincon has adopted the philosophy of delivering only the absolute minimal functionality at integration with more to be supplied later. Remember the rule:

"If the minimum isn't good enough, it wouldn't be the minimum."

The goal is to have a rock solid and tested set of firmware/software ready for launch that allows us to load additional firmware/software once on orbit. This minimum capability must include features that allow us to test and control the payload. It might also include linear transponder functionality since it is relatively straightforward, but complicated digital communication protocols may have to wait. The plan is to develop and test additional functionality on system simulators while waiting for launch. This additional functionality will be loaded later.

If we have access to the payload before launch, we may be able to enhance the "launch load of firmware/software". But our baseline plan must be to load on orbit.

5. The Devil is in the Details

5.1 The Boot Process

It is useful to consider the details of the boot process because:

- 1) The system is dependent on fairly complicated software including a Linux operating system as well as applications code. If the system doesn't boot, the entire payload has failed.
- 2) Various failure scenarios can be envisioned ranging from radiation-induced malfunctions to control operator error. Various levels of system re-boot (including a cold start from the launch program) represent recovery modes.

The 1st stage of the boot process will start with the tested set of code stored in ROM before launch and then use it to load additional code after launch. This is the normal boot starting with power off, or it might be triggered by a watchdog timer expiring or by a command station reboot request.

It is possible that this 1st stage code stored in pre-launch ROM will get corrupted. Since functioning of the payload is dependent on the boot software being correct, one must pay careful attention to booting from an uncorrupted copy of the code.

One proposed plan is to store multiple “golden” copies of the code in ROM and use the RSA check function of the Zynq chip to verify validity of the code. If it fails, the next copy in sequence will be checked, etc. until a valid copy is found.

As mentioned, one function of the 1st stage boot is to allow 2nd stage boot code to be loaded into non-volatile RAM. If 2nd stage code is already present, a command from a control station might then select a specific application to boot in the 2nd stage. The validity of the software in the 2nd stage boot would be checked as before.

5.2 Functions at Launch

Table 1 represents our 1st cut at listing the minimum functions that must be present at launch.

- Power-up boot and various reset and recovery modes
- Telemetry downlink & beacon
 - Including command acknowledgement
 - Goal is to downlink spectral analysis or pre-D snapshot in telemetry for test, checkout, and monitoring
- Narrowband command uplink
 - High link margin reserved for command stations
 - Commands to change operating mode
- Payload comms uplink/downlink (wideband)
 - Reserved initial checkout, software updates, anomaly resolution, special monitoring
- Initial communications load
 - Minimum is linear transponder
 - Likely is several linear transponders with different bandwidths
 - Stretch goals (unlikely unless can load after delivery or there is a slip)
 - Transmultiplexing Interference Canceler (TMIC) in transponders
 - On-board demodulators, switching, and multiplexing into single digital downlink

Table 1: First cut at functions which must be present at Launch (the minimum)

You will note that some of our (and your) favorite communications ideas didn’t make the cut, except as “stretch goals”. That’s not because they aren’t good ideas. It means that we will likely have time to do only part of what we ultimately want to do before the payload is frozen for launch.

That is why we are planning to load and demonstrate cool things once the spacecraft is on orbit. Suggestions and advice are solicited.

5.3 Spacecraft Software Development

Figure 4 illustrates the current test bed implemented at Rincon Research in Tucson to allow spacecraft software development to proceed. Immediate access to a LPFE is key to having any chance of meeting our schedule. For now Rincon IT staff is providing support to manage the Firewall and the Linux SSH server. We anticipate that AMSAT developers will want to control software and use of the Development Support Computer and LPFE. Suggestions and assistance are welcome.

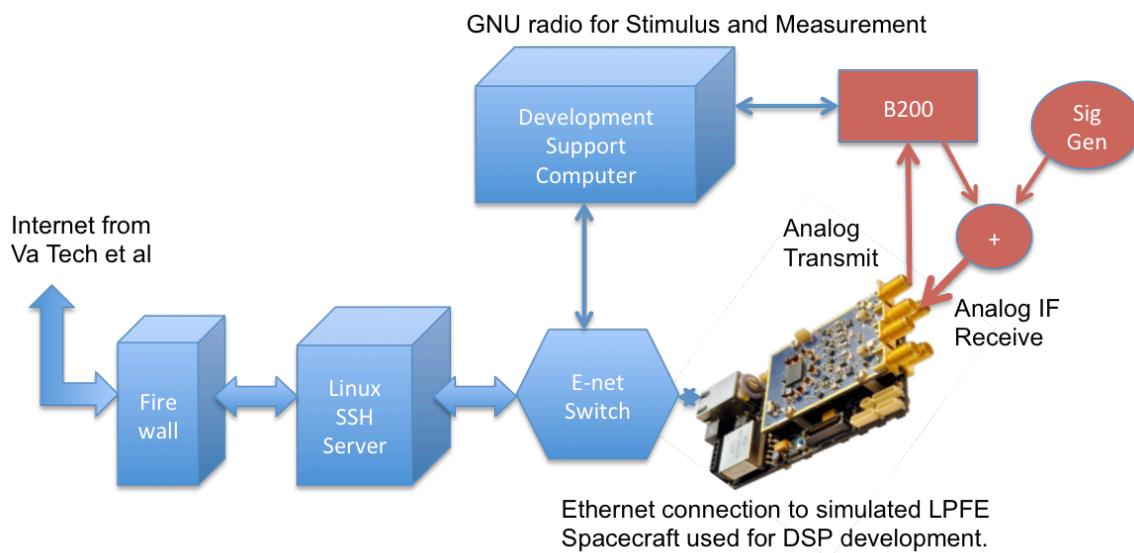


Figure 4: Block diagram of exiting Spacecraft Simulator test bed that allows remote login for spacecraft software development and testing. Rincon Research IT staff currently maintains the Firewall and the Linux SSH Server. Hard-core AMSAT developers can have immediate access.

This equipment is located in an area with controlled access. The area is air conditioned and has uninterruptable power. It is attached to the internet with 100 Mbps service shared with other Rincon traffic, but outside Rincon's company firewall. Placing other computers or devices on this AMSAT network is possible.

Rincon anticipates that AMSAT wishes to control who has access to the software that will be running on the payload, and that much development will be done at remote facilities. A Rincon-maintained firewall separates the internet from a SSH server dedicated to AMSAT. AMSAT personnel will be provided with access to the AMSAT local network by logging into the SSH server. AMSAT will be responsible for any additional computer security or procedures they deem advisable.

An additional computer is provided to assist with development. It is interfaced to a B200 board via USB. This computer will be controlled by AMSAT. It has two functions:

- 1) Assisting development of the LPFE spacecraft software by using the B200 board to generate stimulus and monitoring response. GNU radio may be used to do this.
- 2) Acting as a control station in concert with a B200 board.

The equipment and cabling marked in red carries RF signals for stimulating and monitoring the spacecraft simulated with the LPFE. Blue cables represent digital data paths (IP, Ethernet, and USB)

5.4 Possible Ground-Space Simulator

An extension of the Spacecraft Software Development described in the previous section is possible to form a complete Ground-Space Simulator. This would be implemented once we have working spacecraft code and is diagrammed in Figure 5. It assumes that B200 boards will be used for the ground ham stations. Four ground stations are represented by the B200 boards in the upper right of the slide. The 5.66 GHz output from these boards is combined in a summer with the output from a ground command station illustrated in the lower right. This is added to noise and perhaps interference from a signal generator and fed into the payload simulator.

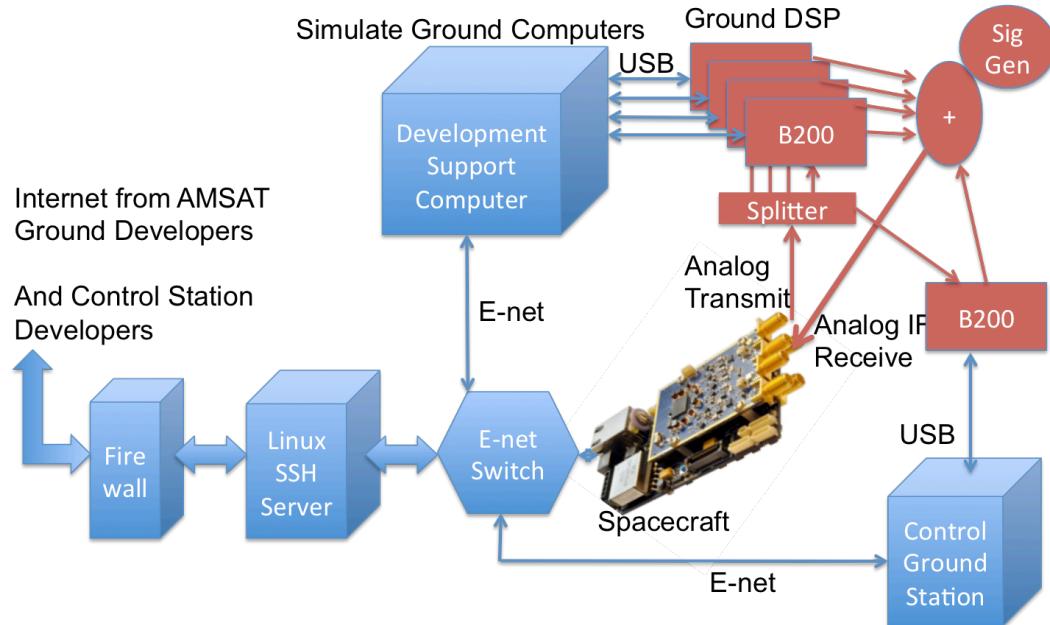


Figure 5: Extension of the Spacecraft Simulator to allow simulation and testing of the complete Ground-Space system including Ground Stations. This would occur once a working spacecraft simulator was complete.

The IF output from the satellite simulator is split and fed to the ground stations and to the command station. The simulator would operate at IF frequency without need for additional RF components to upconvert / downconvert to X-band. The diagram omits some components such as attenuators, noise sources, etc. that would be necessary for an accurate simulation.

We have also assumed that multiple copies of ground station software can be hosted in one computer. Otherwise, we would need a separate computer for each B200 board. You may also note that a Control Ground Station computer and its B200 board has also crept into the simulation. This is to somewhat separate Control Station functions from those of normal users.

This doesn't test RF paths and hardware like a "Ground Sat", but it allows AMSAT members across the country to participate in development of communication protocols while only tying up one spacecraft simulator.

5.5 More LPFE Details

This section will go into more details of the LPFE design.

5.5.1 RF and Analog-Digital Interface

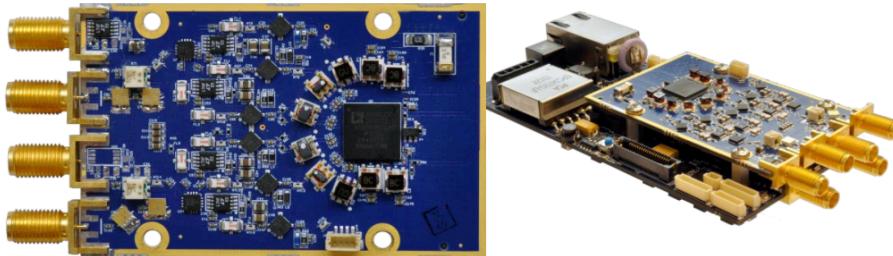
A block diagram of the daughterboard that interfaces between the RF and digital worlds is in Figure 6. At the left of the diagram are two input and two output connectors. Between these connectors and the Analog Devices chip are switches, buffer amplifiers, and filters. The Analog Devices' Catalina- AD9361 chip performs direct RF to digital and digital to RF conversion. The many-pin connector at the right of the diagram communicates digital data, clock, and control to/from the motherboard.

It is worth noting that the inputs are paired and can only operate at the same tuned frequency. The outputs may be tuned to a different frequency than the inputs, but are also paired.

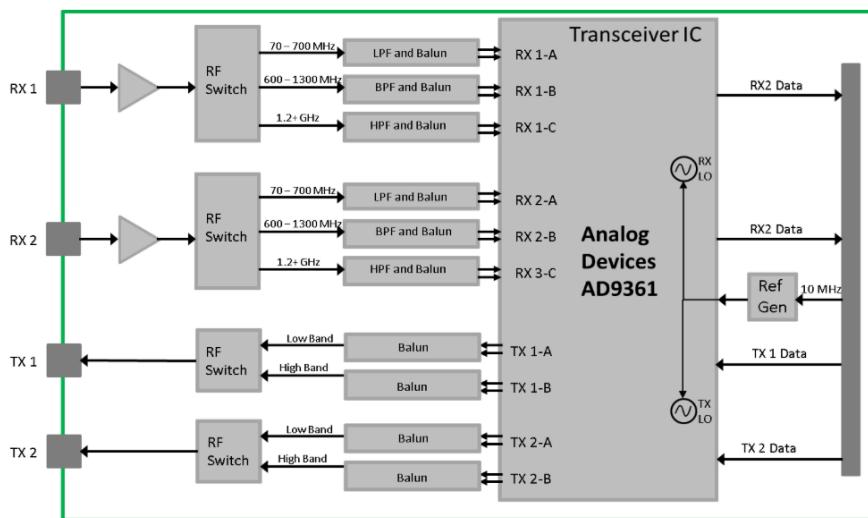
We note that Rincon also manufacturers a daughter board with multiple DACs and ADCs that offer advantages for certain applications involving IF frequencies up to about a GigaHertz. At the risk of adding complexity to the external RF hardware, this is an alternative should the Analog Devices chip not be sufficiently radiation tolerant

LPM-RFXR-0200

Low-Power RF Transceiver



The LPM-RFXR-0200 provides an RF-to-bits capability for the LPFE and IPPV7



70 MH to 6 GHz Tuning Range, Diversity Transmit/Receive (MIMO)

Figure 6: Picture and block diagram of RF Analog-Digital daughter board on terrestrial LPFE. This functionality will be integrated into the mother-board in the AMSAT design.

5.5.2 LPFE Mother Board Block Diagram

A diagram of the mother board shows the details of a LPFE motherboard. Input from the RF Analog-Digital daughter board arrives at the I/O Mezz. connector interfaces directly to the Zynq FPGA fabric.

You will note 8 Gb of DDR3 memory is available to the FPGA fabric and 4 Gb available to the Dual ARMs. An external interface is shown to an SD Card. An SD card is normally used for the power-up boot. This may be changed due to radiation considerations.

An Ethernet external connection is often used for terrestrial applications. In one application a user can remotely log into the LPFE via SSH through Linux running on

the ARMs. The same application provides a web server on the ARMs so that a remote application can control the LPFE and receive data and power spectra computed by the FPGA.

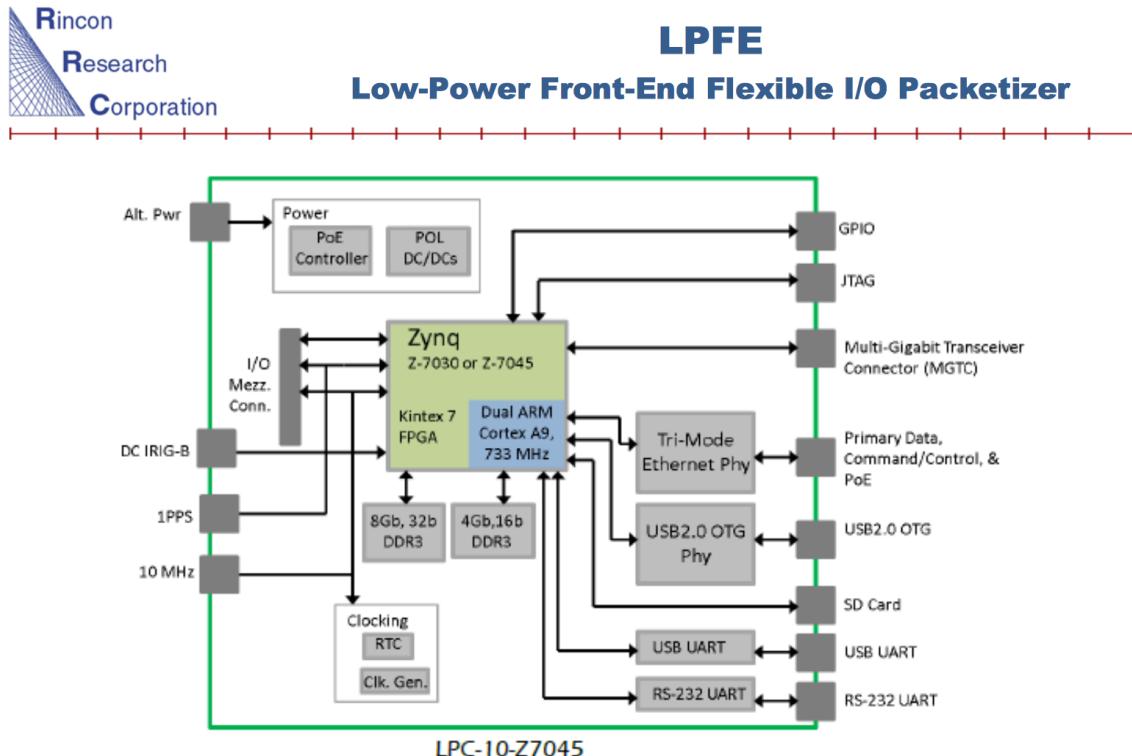


Figure 7: Block diagram of the LPFE motherboard

We will use the Ethernet connection during payload development, but once on orbit all commands, control, data, etc. must be transmitted over the RF ports. This represents a major software change inside the LPFE.

Additional external interfaces may be used during development, but presently have no function once on orbit.

5.5.3 What's a Zynq?

The LPFE incorporates several major functions.

- 1) An RF daughterboard that can transmit or receive between 70 MHz and 6,000 MHz. This is based on an Analog Devices Catalina chip on a Rincon-designed board. This converts analog bandwidths up to 56 MHz between analog and digital domains.
- 2) A processing board centered around a Zynq 7030 (or 7045) system on a chip. The heavy lifting of digital signal processing is done by the FPGA portion of the chip, and the ARM processor cores provide random logic and control. A Rincon-modified version of Arch Linux runs on the ARM processors.

Rincon typically develops the FPGA processing using VHDL. Wiring hardware functions is a foreign concept to software developers, and is very time consuming. We are monitoring alternative means of programming the FPGAs, but so far have not seen reason to disbelieve the old saying that one gets 10 times the performance with 10 times the programming effort.

We do believe that attempting to implement random logic in FPGA fabric, for example using a version of C, is a very bad idea. However, we continue to monitor concepts for stitching together pre-designed modules with higher-level code. Concepts like RFnoC or perhaps OpenCL on the FPGA are intriguing. If anyone thinks that is the best approach, we would be supportive. Perhaps we could learn something.

The Zynq family has undergone radiation tests and proved to be reasonably radiation tolerant when certain precautions are taken such as disabling cache memory.

5.5.4 Terrestrial Configuration of LPFE

There will be substantial differences between a space based and its terrestrial counterpart. I will start by describing operation of the terrestrial version and then describe changes that are contemplated for the Space LPFE.

The typical terrestrial LPFE is programmed to boot on power up and load a image from flash into RAM. Once the image, including Linux, is loaded, a Linux terminal interface is accessible from a local or remote terminal using the Linux SSH command (TCP/IP protocol) to log into the Linux running on the Zynq's ARM. Multiple applications have been designed to run on the LPFE. In one data capture application the ARM runs a web server which presents a graphical interface over the Ethernet to local or remote computers. In this application the ARM controls the RF tuner frequency, bandwidth, and gain through receipt of HTTP commands from the remote user's graphical interface. As an alternative, one can issue "curl" commands from the remote terminal to the web server.

Note that the above application has a mixture of data communication mechanisms ranging from one-way UDP packets whose delivery is not guaranteed to handshakes via TCP/IP. While this was a good mixture for the particular application, it needs to be re-thought for the AMSAT mission. For example, sending commands that require 2-way TCP/IP handshakes before action is taken may not always be a good idea. Likewise, it is well-known that the standard TCP/IP protocol stack causes problems transferring data when encountering $\frac{1}{4}$ second round-trip delays to geosynchronous satellites.

5.5.5 Changes for Space

Rincon Research has identified changes to the terrestrial design for space use listed in Table 2.

- Hardware
 - External keep-alive reboots on failure or missing heartbeat signal
 - Radiation-tolerant power supplies
 - Radiation-tolerant ROM
 - Re-design conduction cooling
 - ECC memory (HW or SW ?)
 - Integrate two boards into one
- Software/firmware
 - Disable cache on microprocessors
 - Use Xilinx secure image function and sequence through multiple “gold” copies of boot if parity fails.
 - Perhaps enhance error detection and correction?
 - Possibly run a configuration memory scrubber
 - Reboot on kernel panics or other Linux error messages
- RF instead of Ethernet interface for command/control/data
- Add spacecraft control and monitoring function
 - RF power control
 - Communications mode control
 - LPFE power control
 - Full function, telemetry only (rest of FPGA off), sleep (microprocessor only), ALL OFF

Table 2: Desirable Changes to the Terrestrial LPFE for Space Use

A key hardware item that we expect AMSAT to provide is an external watchdog circuit that includes power recycle in the event of a heartbeat not being present from the LPFE or in the event that a watchdog timer is not reset periodically by a ground command station. Rincon plans an LPFE hardware rework to implement the items listed in table 2.

In the software arena, there are several things that could be done to improve radiation tolerance. Experience of other designers suggests that the cache on the microprocessors is subject to single event upsets and should be disabled. They also suggest using the Xilinx secure image function to detect whether a bit image is correct. Multiple “gold copies” of the bootable image would be stored, and sequenced through until a correct image was found.

It is also suggested that the a configuration memory scrubber could be used to detect and repair single event upsets in the main FPGA configuration memory.

Finally, it is suggested that Linux kernel panics and other error messages should trigger certain reboot actions including possibly a power reset.

A major conceptual change for space use of the LPFE is routing all of the command, control, and data via RF links rather than the Ethernet or other wired interfaces as is normally done in terrestrial scenarios.

Finally, AMSAT is planning to implement spacecraft control functions in the LPFE's ARM processors. This has a number of advantages, but it does complicate development and testing of software as two disparate functions share the operating system and compute resources of the ARMs.

Fault tolerance is very important. We note that many of the fault tolerant schemes have a significant software component, so they are subject to software design choices, programmer availability, and schedule limitations. We have already talked about some of these, but we felt it important to re-emphasize them.

- External watchdog to monitor LPFE heartbeat and keep alive timer and reset if necessary
- RSA-authenticated bootstrap from NAND flash with multiple “golden” copies
- ECC memory controller for DDR3 within Zynq
- Internal watchdogs within Zynq
- Potential FPGA configuration scrubber
- Redundant processing streams with “voting”

Table 3: Some (largely software) items that could improve fault tolerance. Our ability to implement some of these may be limited by the schedule and the size of the programming team.

Items not yet mentioned include the possibility of repeat “scrubbing” of the FPGA bit mask to discover and correct for any single-event-upsets in the FPGA fabric. This is an item unique to FPGAs and may not be familiar to programmers. Another new item, redundant processing streams with “voting” is well-known in concept, but of course there are details that can achieve most of the beneficial effects of voting while using less compute resources.

6. Summary

Rincon Research is very excited by the prospect of incorporating Rincon Hardware into an AMSAT payload. We do this in part to pay back all the people and organizations that have helped us over the years. We also do it because many of us are hams and it is a cool project.

Time is extremely short. This project would be impossible if we were not starting with an existing design that allows us to begin software development immediately. It would also be impossible to have a payload ready for integration in less than a year if we didn't adopt the approach of having a minimal set of spacecraft code ready at spacecraft integration and loading additional code on orbit.

In this paper we have outlined a success-oriented plan for the LPFE part of the payload. As this is being written in early September 2015 we have no visibility into progress on the rest of the payload or how this will all come together.

Rincon Research has a sense of urgency in two areas:

- 1) Completing a rework of the LPFE board optimized for space
- 2) Giving AMSAT developers access to LPFEs for software development and testing

Rincon views spacecraft software development as a significant schedule risk, and Rincon has put a spacecraft software development facility on the internet for use by AMSAT developers.

The authors and Rincon Research: All authors are employees of Rincon Research Corporation headquartered in Tucson, AZ. Rincon is a company of about 200 employees mainly supporting the US government. It also has offices in Colorado and Virginia, and employees are scattered around the globe. Rincon sponsored and provided technical support for RinconSat, an early CubeSat built by the University of Arizona. Unfortunately its Russian launch vehicle failed.