Off the Shelf Testbeds for the Robotic Lunar Lander Risk Reduction Effort

Robert Davis

The Johns Hopkins University
APPLIED PHYSICS LABORATORY

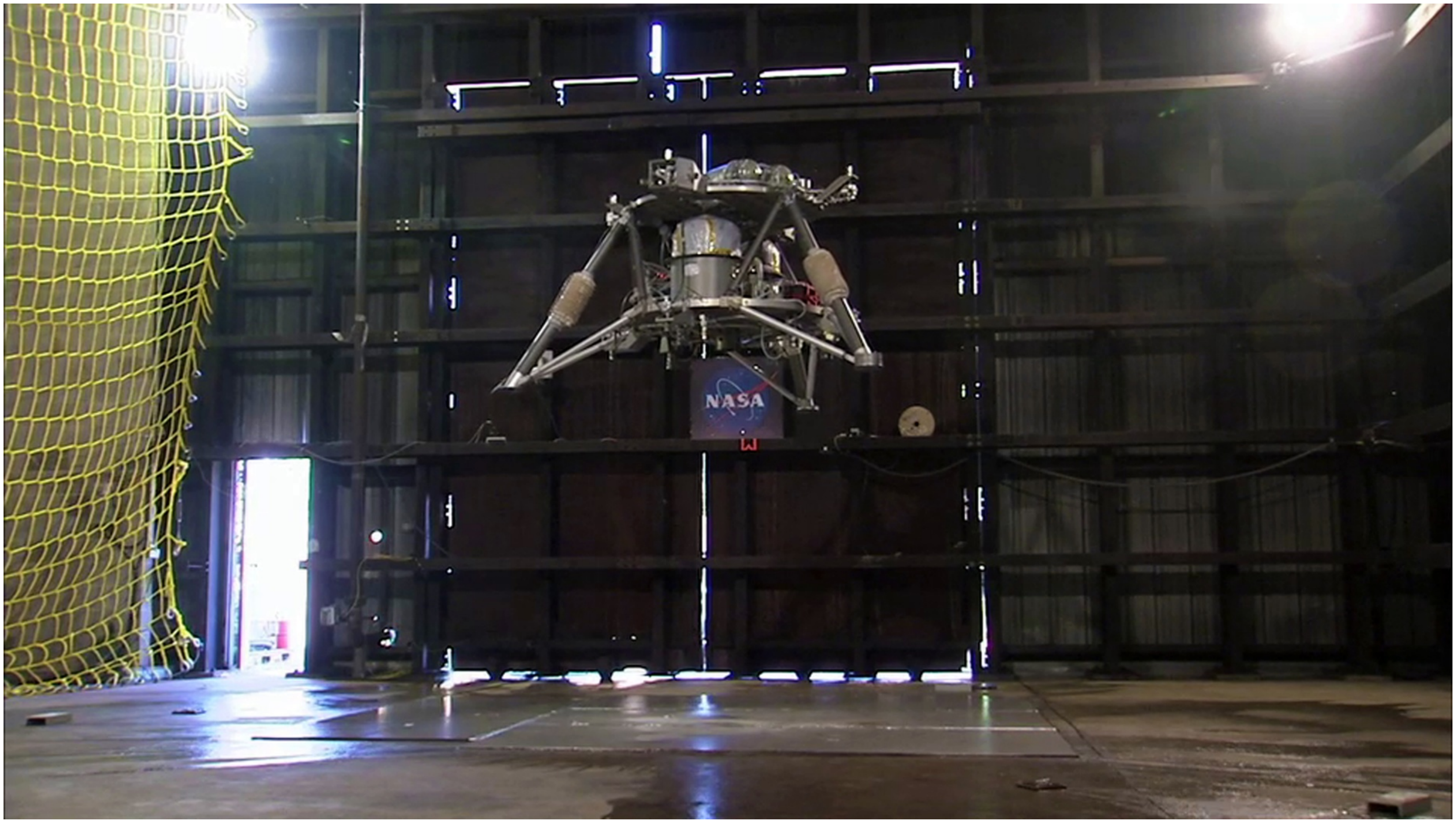
Introduction

Describe how off-the-shelf components were used to construct a testbed system for use on the Robotic Lunar Lander (RLL) project.

Compare RLL testbeds to the “usual” testbed systems.

Experiments with Symmetric Multi-Processing on a multi-core PC.

Warm Gas Text Article



Indoor Test Flight

Flight Test


06/13/2011 16:16:29 PTZ4
Marshall Space Flight Center
Robotic Lander
Test Facility
www.nasa.gov/roboticlander

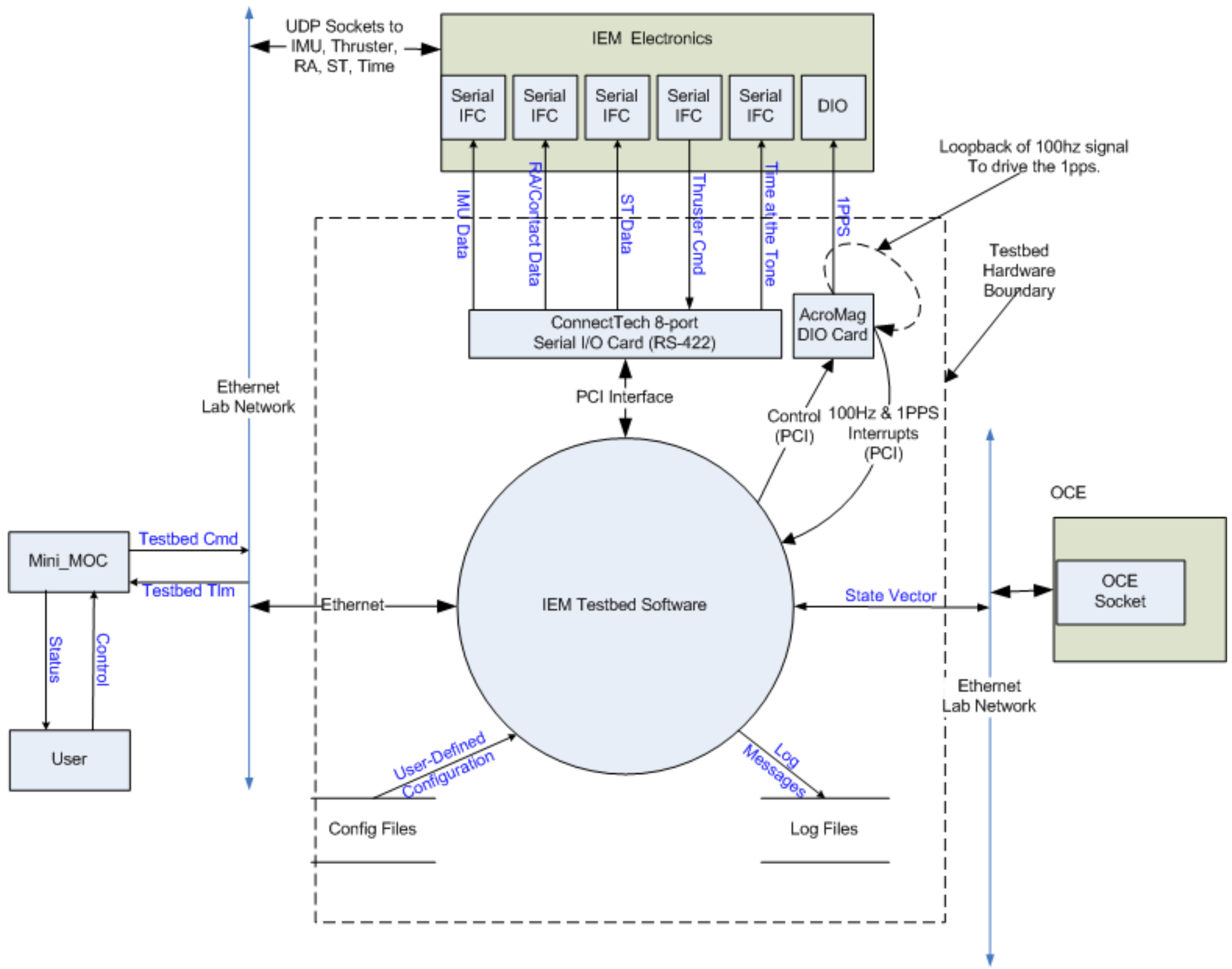
Outdoor Test Facility

Considerations

For the RLL it was decided to go with all off the shelf hardware for the testbeds.

- **Faster procurement**
- **No custom hardware to design, build, or debug**

RLL Software/Hardware Context



Hardware Components – Base System

- Processor: Dell R5400 rack mount PC.
  - Quad core
  - 4 GB of ram.
  - 320 GB hard drive
  - 256MB PCIe x16 nVidia Quadro FX570, Dual Monitor DVI Capable (standard)

- Digital I/O Card (for 1pps) - APC424 - 40 I/O channels: 24 differential and 16 TTL & Four 16-bit or two 32-bit counter/ timers. Also purchased SCSI 3 cables and breakout boxes.

- Serial I/O Card – ConnectTech Bluestorm/SP. A standard profile, 8 channel, RS 232/422/485 PCI card. Comes with a “fan-out” cable: has one 78-pin D-connector that “fans out” to eight 9-pin serial connectors.

Additional Equipment

- **Custom built serial I/O cable to the IEM.**
- **Custom cable to distribute the 1PPS signal from the testbed to other components.**
- **Dell 4220 42U Rack**

Multiple Interface Configurations (UDP, RS-422)

- Given the short time span in which to develop the system and taking into account the time to acquire IEM RS-422 communications hardware, we developed software for a UDP interface.
- Interfaces: IMU, thruster, time at the tone, Radar Altimeter data, and Optical Camera Emulator(OCE).
- UDP sockets were implemented first. Simple and reliable but could be susceptible to congestion delays although we never encountered this. (~18Kbytes/second)
- Once the RS-422 serial interface hardware for the IEM was acquired the serial interfaces were developed and are the default communications method.
- This approach allows us to mix and match what interface types for specific communications channels if necessary.

Costs

Testbed Hardware

Dell R5400 Rack mount PC (LINUX installed)		$3,124	
QNX Operating System		$110	
BlueStorm Serial I/O Card, Driver, & Cable		$340	
Acromag Digital I/O Card		$700	
Breakout box & Cable		$285 *	
QNX Driver Software		$795 *	
	Total	$5,354	$4,274 (Minus * items)

Optional Items

Dell 42U Rack & Accessories	$1,000
QNX Momentics SDK	$12,000

* Future testbed systems using the same i/o cards will not need to purchase these items.

Comparative Costs

RLL

Dell R5400 Rack mount PC (LINUX)	$3,124	
QNX Operating System	$110	
BlueStorm Serial I/O Card & Cable	$340	
Acromag Digital I/O Card	$700	
Breakout box & Cable	$285*	
Driver Software	$795*	
	$5,354	**$4,274(Minus * items)**

Optional Items

Dell 42U Rack & Accessories	$1,000	*This houses 3 testbeds, 3 oces, 2 IEMs*
QNX SDK	$12,000	

RBSP

rack enclosure (38U) and accessories	$2,000
15C Series, Card cage, power supplies (cPCI)	$2,521
CT11T6C17KL1, single board computer	$4,270
CTM17A0001, I/O transition module for CT11	$375
CR11-SDK-LINUX, system development kit	$1,500
5C Series, cPCI card cage, backplane for BB IEM	$1,721
CP3-Quad100TX, cPCI Ethernet adapter	$760
Monitor/Keyboard for SBC	$0
bc635PMC, IRIG time code processor card	$2,245
CPCI-200B_BP, 4 slot IP carrier	$737
IP Modules and related hardware/software	$3,000
CD-RW 52X/DVD-ROM, CD read/write, DVD Reader	$90
Total	**$19,219**

Miscellaneous Items Purchased

Things we had lying around and did not need to buy:

- **KVM Switches and cables**
- **Monitors/keyboard/mice (although we did buy 1 testbed monitor)**
- **Surge protectors**
- **Network cables**
- **Oscilloscope – used in digital I/O card testing.**

Issues (1 of 2)

PCI Slots (or lack of)

- R5400 comes with 1 PCIX 64-bit and PCIe x16 (one of these used by video card)
- Added a riser with a PCI 5v and a PCI-X for the serial and digital I/O cards

Video Card (256MB PCIe x16 nVidia Quadro FX570)

- Not supported by QNX drivers
- Stuck at 640x480 resolution
- Other cards can be substituted but are limited by the number and type of PCI slots in the PC.
- Using SSH via office PC is the preferred access method.

Issues (2 of 2)

Did not build in a way to toggle a hardware output from software and observe it via an oscilloscope. Would have helped with verifying 1pps timing between *machines* as well as internal software timing. Probably could have used one of the digital card's outputs.

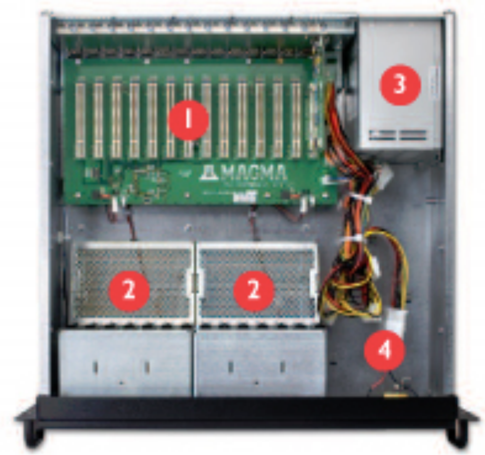
No way to do low level hardware communications testing: inducing bit and parity errors.

More PCI Slots – One Solution

Magma 13 Slot PCI Expansion System (cost: ~$2100)

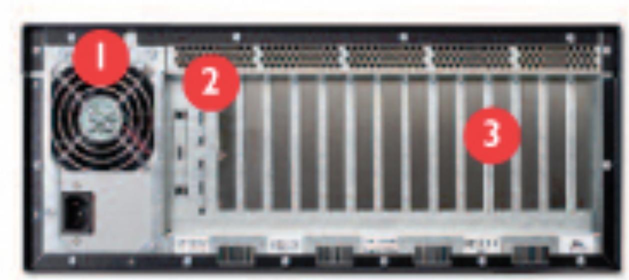
13 Slot - Top View

1. Thirteen slot PCI backplane
2. Cooling fan assembly
3. Power supply
4. Space for optional drive bays
5. Expansion cable
6. Host interface cards



13 Slot - Rear View

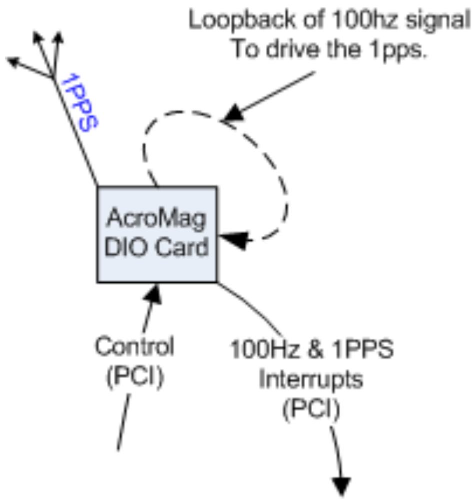
1. 400W power supply (Optional redundant power not shown)
2. Cable connections
3. PCI card slot openings



1 PPS

The 1 PPS signal for RLL originated from the testbed because:

- Did not have “normal” C&DH hardware to do so
- Only needed a common time reference
- The digital I/O card was capable of supplying it.



Implemented via the AcroMag digital I/O card

- Configured to produce 100Hz interrupts driving the testbed’s main run loop. Accurate 100hz timing was possible for reading thruster commands, stepping the model, and sending model outputs.
- 100Hz counter fed the 1 PPS counter using the 32-bit counters.
- For future consideration this card can also receive signals on the digital inputs (feed in an external 1PPS or something faster?).

Serial I/O

- The ConnectTech serial card is jumper configurable between RS-232/422/485.
- Device driver startup controls buffer sizes, baud rates, and other configurable items. Runs as its own process.
- An RS-232 null modem cable provided for wrap-around testing of the hardware/software prior to integration with IEM hardware.
- Once IEM hardware was ready the card was jumpered to RS-422 operation.
- I/O with the hardware is done by open(), read(), write(), close(),and select() calls. This is the design pattern for QNX device drivers.

Multi-Core Processing (1 of 2)

- Tested the Symmetric Multiprocessing capabilities of the R5400 and QNX.
- On RLL, main testbed operation (@ 100hz) is all single-threaded which maintains the synchronization of thruster input, model stepping, and outputs to the flight side.
- QNX automatically assigns threads to unused cores in a multi-core processor.
- In order to test the possibility of running at faster rates (400hz), experiments were performed where the Dynamics Model processing was assigned to its own thread and locked to a specific core.
- Given the inherent speed of this PC the single threaded processing my be sufficient.

Multi-Core Processing (2 o 2)

- Testing has shown that running the dynamics model in its own thread and locked to a specific processor saves about 10-20% of the overall execution time vs. single-threaded operation.
- More tests are needed to investigate any synchronization issues when the model runs in its own thread.
- Original test: an event is sent to the model thread every 10ms and the model then steps as many times as programmed to do. Our first tests ran at 400hz (100x4 steps). Have not made it fly yet because the model needs updating to do so.
- Later tests indicate that even faster rates (1000 hz) may be attainable. Further testing still needs to be done to confirm that this is actually possible.
- An easy thing SMP thing to do is to run device drivers in their own threads and processor cores. Tested and works.

Conclusions

- An off the shelf testbed might not totally replace normal testbed configuration used by Space Department missions. More research needs to be done. It currently lacks the capacity for large numbers of communications interfaces and any custom interfaces that may be necessary.

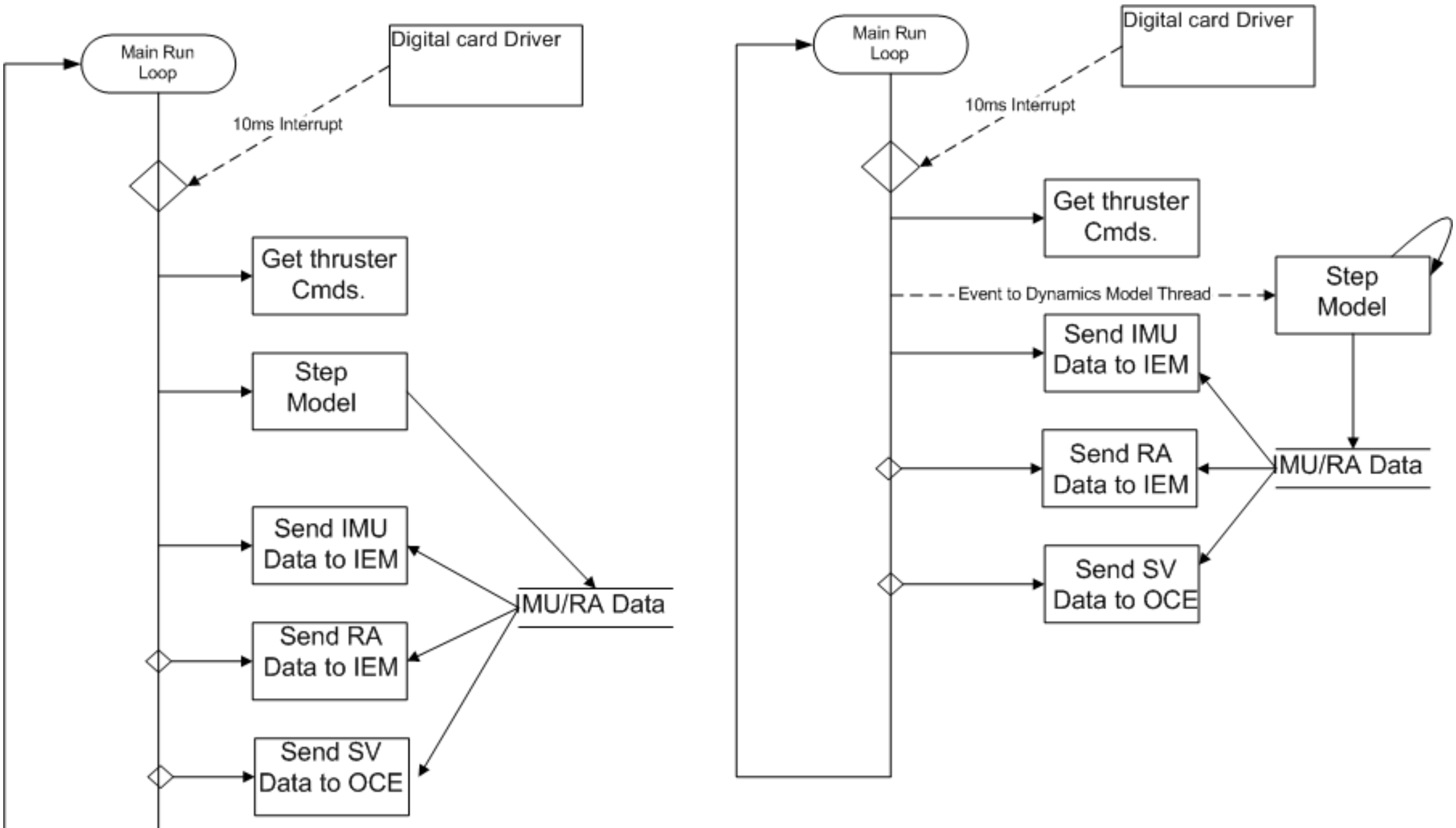
- For early testbed development and to endure delays in hardware delivery, the off the shelf concept can be a temporary work around.

Backup Slides

Additional multi-core processing information

Outdoor Hover Test

Multi-Core Processing (3 of 3)



SMP Code

```
/* This code is placed within the thread to be "core-locked" */
   unsigned num_elements = 0;
   int *rsizep, masksize_bytes, size;
   unsigned *rmaskp, *imaskp;
   void *my_data;
   int status;

   // Global: struct syspage_entry *_syspage_ptr
   num_elements = RMSK_SIZE(_syspage_ptr->num_cpu);
   masksize_bytes = num_elements * sizeof(unsigned);
   size = sizeof(int) + 2 * masksize_bytes;
   my_data = malloc(size);
   memset(my_data, 0, size);

   rsizep = (int *)my_data;
   rmaskp = (unsigned int *)(rsizep + 1);
   imaskp = rmaskp + num_elements;

   *rsizep = num_elements;
   /* CORE NUMBER 2 */
   RMSK_SET(2, rmaskp);
   RMSK_SET(2, imaskp);
   status = ThreadCtl(_NTO_TCTL_RUNMASK_GET_AND_SET_INHERIT, my_data);
```