NASA Operational Simulator for Small Satellites (NOS$^3$)

NASA IV&V Independent Test Capability (ITC) Team

Mr. Scott Zemerick
Scott.Zemerick@TMCTechnologies.com
Agenda

**STF-1 Intro**
- ITC Intro
- NASA IV&V CubeSat
- C&DH FSW Architecture

**NOS³**
- V1.0
- Architecture
- Simulators

**Conclusion**
- Next Steps
- Questions
Independent Test Capability (ITC) Introduction
NASA IV&V Independent Test Capability (ITC)

Charter

Acquire, develop, and manage adaptable test environments that enable the **dynamic** analysis of software behaviors for multiple NASA missions

Dynamic Analysis is performed on flight software to verify software behavior
NASA IV&V Independent Test Capability (ITC)

JIST

JWST Integrated Simulation & Test

S3

SLS Software-Only-Simulator

Automation and virtual deployment

JSTAR
Jon McBride Software Testing & Research Lab

NOS3

Small Sats

QEMU RAD750 Model

Wind River Simics Modeling

NASA IV&V
5000 NASA Blvd. Fairmont, WV 26554
www.nasa.gov/centers/ivv
Simulation-to-Flight (STF-1)
Introduction
Simulation-to-Flight (STF-1)

NASA IV&V ITC & West Virginia University (WVU) 3U Cubesat

- NASA Cubesat Launch Initiative (CLI) proposal submitted and accepted – NASA will pay manifest for future launch
- First WV Cubesat
- ITC is responsible for C&DH hardware/software, integration (hw/sw), and all testing
- WVU is responsible for payload hardware and software
- STF-1 is a “GSFC Cubesat” – partnering with GSFC/WFF and Dellingr Cubesat Team
- Current Launch Ready Date is August 2016 – not yet manifested – prefer polar orbit

Simulation-to-Flight (STF-1)

- **Primary Objective** – Showcase simulation technologies developed at IV&V
- **Secondary Objectives** – WVU Research into space weather, rad-hard materials, navigation instruments (GPS and IMUs), and camera

Sponsored by:

- JSTAR
- TMC²
- Orbital ATK
- WVHTC
- NASA
## Simulation-to-Flight (STF-1)

### STF-1 Flight Software / Hardware Design

- Working closely with the GSFC Dellingr 6U cubesat team
- FSW is Core Flight System (cFS)
  - Dellingr reuse, specifically on the radio cFS application
- ITC designed solar panel PCBs (Dellingr-based)
- Most hardware same as other GSFC cubesats

<table>
<thead>
<tr>
<th>Hardware</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Onboard Computer</td>
<td>Received</td>
</tr>
<tr>
<td>Solar Cells</td>
<td>Received</td>
</tr>
<tr>
<td>Power System</td>
<td>Ordered – 10 Week Lead Time</td>
</tr>
<tr>
<td>Chassis</td>
<td>Ordered – Unknown Lead Time</td>
</tr>
<tr>
<td>ITC Designed Solar Panel PCBs</td>
<td>Designed – Out for Quote</td>
</tr>
<tr>
<td>Radio</td>
<td>Ordered – 6 Month Lead Time</td>
</tr>
<tr>
<td>Clean Room</td>
<td>Procured and Setup for Ribbon Cutting</td>
</tr>
<tr>
<td>Deployable Antenna</td>
<td>Ordered – Unknown Lead Time</td>
</tr>
<tr>
<td>Camera</td>
<td>Received</td>
</tr>
</tbody>
</table>
Anatomy of STF-1

**Camera**
- Mounted to a PC104 protoboard
- Optional filters to provide earth science data

**CADET Radio**
- Half duplex UHF
- Low power design
- Store and Forward architecture
- 4GB memory buffer
- Up to 22 Mbps data rate

**ISISpace UHF/VHF Antennas**
- Deployable antenna system
- Four alloy tape antennas
- Up to 55cm in length
- Includes 30mm diameter center through-hole for pass-through

**ISISpace Chassis**
- Modular structure
- Each unit can be assembled independently
- COTS component
- Compatible with P-POD Cal-Poly specifications

**LCSEE**
- Two PC104 Boards
- Fits directly into stack without modification
- Three different LED carriers with different shielding levels

**Inertial Measurement Unit (IMU)**
- Micro Electro-Mechanical Systems
- Accounts for errors through calibration
- High quality inertial sensing with a MEMs IMU cluster

**Physics Payload**
- Particle detector
- VLF receiver
- Plasma Probe

**GomSpace Nanomind A3200**
- High-performance AVR32
- 512KB build-in flash
- 125Mb NOR flash
- 32MB SDRAM
- I²C, UART, CAN-Bus

**Novatel OEM625 GPS**
- On-orbit reprogrammable
- Precise orbit determination
- Open loop tracking
- Science data products: 100-Hz phase, TEC, S4

**2 x ClydeSpace Batteries**
- Lithium Polymer
- 80 Watt Hours Total
- Two independent boards for redundancy
- Internal heaters

**ClydeSpace Electrical Power System (EPS)**
- 10 command-able power switches
- Provides 3.3V, 5V, and 12V
- Optimized for Low Earth Orbit (LEO)
- Three independent battery charge regulators
Simulation-to-Flight (STF-1)
Simulation-to-Flight (STF-1)
Simulation-to-Flight (STF-1)

FSW Architecture

OS Abstraction Layer (OSAL)

STF-1

cFS

FreeRTOS

Flight Hardware
NASA Operational Simulator for Small Satellites (NOS³)

Introduction
What is NOS$^3$?

- A software test bed for small satellites
- Based upon STF-1 hardware, but sufficiently generic
- Easily-interfaces to cFS FSW, but cFS not required
- Currently open-loop, closed loop planned
- Openly distributed solution Ready-to-Run (RTR) – Looking for Users!
- A collection of Linux executable and libraries

What is it used for?

- FSW early-development – NOS$^3$ provides real-world inputs to FSW
- FSW V&V – Testing FSW, invalid inputs, behavior, stress conditions
- FSW Integration – Used for early-app development and payload team integration
- Mission Planning – Example: power analysis
NOS³ Ready-to-Run (RTR)

Leverage ITC virtual deployment technologies

• ITC does NOT distribute virtual machines
• Virtual machines are built on-the-fly by the user
• Deployment Steps
  • Obtain files ITC
  • Install virtual machine provisioner such as Virtual Box
  • Run 1 Command – generates virtual machine
  • Login to virtual machine and build cFS with RTR script

Ready-to-Run (RTR) for...

• cFS development environment
• NOS³ environment
• Ground system software
• Software integration testing
NOS³

Linux Software Architecture

- cFS
- FSW
- NOS Engine
- Sim
- Process
- Process
- Process
## NOS\(^3\) v1 Included Simulators

<table>
<thead>
<tr>
<th>Simulator</th>
<th>Hardware Modeled</th>
<th>Sim Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Magnetometer</td>
<td>Honeywell HMC5843</td>
<td>FSW data source for development</td>
</tr>
<tr>
<td>Electrical Power System</td>
<td>Clydespace Gen III</td>
<td>Power analysis – software control of switches</td>
</tr>
<tr>
<td>(EPS)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GPS</td>
<td>Novatel</td>
<td>FSW data source for development and software commanding of GPS</td>
</tr>
<tr>
<td>Camera</td>
<td>ArduCam Mini OV2640 SPI/I2C</td>
<td>FSW data source for development and large data packet handling</td>
</tr>
</tbody>
</table>
# NOS$^3$ Components

<table>
<thead>
<tr>
<th>Component</th>
<th>What is it?</th>
<th>How is it used?</th>
</tr>
</thead>
<tbody>
<tr>
<td>NASA Operational Simulator (NOS) Engine</td>
<td><em>Engine</em> is ITC-developed message passing middleware designed specifically for use in simulation. Includes time synchronization, data manipulation, and fault injection.</td>
<td>Serves as the NOS$^3$ glue to tie all components together into a common interface to FSW</td>
</tr>
<tr>
<td>Hardware Model</td>
<td>A model for a specific piece of flight hardware, often focusing on the inputs/outputs of the device from the FSW perspective.</td>
<td>Serves as virtual hardware in order to provide FSW with an accurate representation of its data</td>
</tr>
<tr>
<td>42</td>
<td>42 is an open-source general purpose simulator developed at NASA Goddard Space Flight Center for spacecraft attitude and orbit dynamics.</td>
<td>Serves as an <em>Environment Data Provider</em> – chosen to provide magnetic field data and positional data as inputs to the magnetometer and GPS simulators</td>
</tr>
</tbody>
</table>

**Diagram:**
- **NOS Engine Interface:** time, input bytes
- **Hardware Model:** output bytes
- **Environment Data Provider:** environmental data, time
NOS$^3$ Components

FSW

NOS Engine Interface -> NOS Engine Interface -> NOS Engine Interface

time, input bytes -> time, input bytes -> time, input bytes

output bytes -> output bytes -> output bytes

Hardware Model -> Hardware Model -> Hardware Model

environmental data -> environmental data -> environmental data

time -> time
STF-1 FSW + NOS$^3$

No FSW code changes

OS Abstraction Layer (OSAL)

- Linux
- NOS$^3$
- FreeRTOS
- Flight Hardware

STF-1
STF-1 FSW + NOS$^3$

No FSW code changes

OS Abstraction Layer (OSAL)

Linux (x86, ARM/PI)

NOS Engine

Hardware Models

Environment Data Provider

FreeRTOS (AVR32)

Flight Hardware

CMake Cross-platform Make

NOS$^3$
STF-1 FSW + NOS³

No FSW code changes

OS Abstraction Layer (OSAL)

STF-1

cFS

Hardware Lib

Linux
(x86, ARM/PI)

NOS Engine

Hardware Models

42

FreeRTOS
(AVR32)

Flight Hardware

CMake
Cross-platform Make

NOS³

NOS³
## Future Plans – NOS\(^3\) v2

<table>
<thead>
<tr>
<th>Capability</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 Axis Gyroscope Simulator (In Progress)</td>
<td>InvenSense MPU-3300</td>
</tr>
<tr>
<td>Temperature Sensors (In Progress)</td>
<td>I2C Temperature Sensors</td>
</tr>
<tr>
<td>Electrical Power System (EPS) Sim Maturity (In Progress)</td>
<td>Add battery sim to Clydespace Simulator</td>
</tr>
<tr>
<td>UHF Radio Simulator (under consideration)</td>
<td>L3 Cadet Radio</td>
</tr>
<tr>
<td>Visualization / User Interface</td>
<td>Provide the user with a generic NOS(^3) user-interface.</td>
</tr>
<tr>
<td>Integrate with Ground System Software</td>
<td>Currently looking into COSMOS and ITOS.</td>
</tr>
<tr>
<td>Tighter 42 Integration</td>
<td>Programmatically sync FSW time to 42 time so that NOS(^3) hardware models and FSW are in sync</td>
</tr>
</tbody>
</table>
NOS³ Work in Progress

FSW (Linux)

- TimeSync App
- EPS App
- Magnetometer App

- STF-1 HWLIB
- LIBA3200NOS
- NOS Engine

FSW (Nanomind)

- EPS App
- Magnetometer App

- STF-1 HWLIB
- LIBA3200
- I2C
- SPI
- UART

Other Sim 1
- NOS Engine

Other Sim 2
- NOS Engine

Ethernet

- I2C Adapter
- SPI Adapter
- UART Adapter

Camera
- GPS

Camera Sim
- GPS Sim
- NOS Engine
- NOS Engine
NOS$^3$ Visualization
NOS$^3$ Ground System Integration
Questions?
Backup Slides
NOS$^3$ FSW Architecture
NOS$^3$ FSW Architecture
NOS$^3$ 42 Integration

42 Input Files
- Inp_Sim
  - Configuration Parameters of Simulation
- Inp_Cmd
  - Commands read by 42, currently the starting quaternion
- Orb_LEO
  - Orbit Parameters
- SC_STF1
  - Spacecraft parameters for STF-1

42 Outputs Files
- FOTON.42
- MagField.42
- MATLAB.42

42 Executable

NOS3 Simulators
- Magnetometer Sim
- FOTON GPS Sim
## EPS Simulator

### Board
- **Firmware**: 1
- **Revision**: 2
- **Cmd Invalid**: Cmd
- **Data**: Data
- **POR**: POR
- **Last Reset**: Last Reset
- **BOR**: BOR
- **WDT**: WDT

### Battery Charge Regulator

<table>
<thead>
<tr>
<th></th>
<th>BCR1</th>
<th>BCR2</th>
<th>BCR3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Va</td>
<td>20.25</td>
<td>19.5</td>
<td>6</td>
</tr>
<tr>
<td>Ia</td>
<td>2.25</td>
<td>1.75</td>
<td>0.25</td>
</tr>
<tr>
<td>Ib</td>
<td>2.75</td>
<td>2</td>
<td>0.75</td>
</tr>
<tr>
<td>Ta</td>
<td>5</td>
<td>7.5</td>
<td>20.5</td>
</tr>
<tr>
<td>Tb</td>
<td>10</td>
<td>15</td>
<td>21.75</td>
</tr>
<tr>
<td>SDA</td>
<td>100</td>
<td>600</td>
<td>511</td>
</tr>
<tr>
<td>SDB</td>
<td>775</td>
<td>300</td>
<td>512</td>
</tr>
</tbody>
</table>

### Power Conditioning Module

<table>
<thead>
<tr>
<th></th>
<th>BAT</th>
<th>12V</th>
<th>5V</th>
<th>3.3V</th>
</tr>
</thead>
<tbody>
<tr>
<td>V</td>
<td>7.7</td>
<td>12.3</td>
<td>5.1</td>
<td>3.3</td>
</tr>
<tr>
<td>I</td>
<td>4.2</td>
<td>1.2</td>
<td>4.1</td>
<td>3.9</td>
</tr>
</tbody>
</table>

### Power Distribution Module

<table>
<thead>
<tr>
<th></th>
<th>SW1</th>
<th>SW2</th>
<th>SW3</th>
<th>SW4</th>
<th>SW5</th>
<th>SW6</th>
<th>SW7</th>
<th>SW8</th>
<th>SW9</th>
<th>SW10</th>
</tr>
</thead>
<tbody>
<tr>
<td>V</td>
<td>11.9</td>
<td>0</td>
<td>7.7</td>
<td>0</td>
<td>4.9</td>
<td>0</td>
<td>5.1</td>
<td>0</td>
<td>3.2</td>
<td>0</td>
</tr>
<tr>
<td>I</td>
<td>1.25</td>
<td>0</td>
<td>1.75</td>
<td>0</td>
<td>2.25</td>
<td>0</td>
<td>2.75</td>
<td>0</td>
<td>3.25</td>
<td>0</td>
</tr>
</tbody>
</table>

[Reset | Update]

[Reset | Update]