STRS Waveform Porting for NASA’s CoNNeCT Project

December 2, 2011
Dale Mortensen
Outline

- CoNNeCT project overview
- The Ported Waveform – TDRSS application
- "What is all this STRS stuff, anyhow?"
- Development approach
- Porting metrics & results

*Does STRS really make a difference?*
Communications, Navigation, and Networking reConfigurable Testbed

♦ a.k.a. “Space Communications and Networking (SCAN) Testbed”
♦ International Space Station (ISS) Exterior Payload, scheduled to launch in 2012
♦ Investigating the application of SDRs to NASA Missions
♦ SDR technology development
♦ Validating future mission operational capabilities
♦ First flight for STRS
# JPL Baseline Waveform Description

<table>
<thead>
<tr>
<th>Description</th>
<th>Transmit</th>
<th>Receive</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(return link)</td>
<td>(forward link)</td>
</tr>
<tr>
<td>Modulation</td>
<td>BPSK</td>
<td></td>
</tr>
<tr>
<td>Spreading</td>
<td>Direct Sequence Spread Spectrum (PN Short code)</td>
<td>Data Group 1, Mode 2</td>
</tr>
<tr>
<td></td>
<td>(with bypass option for DG2)</td>
<td>Data Group 2, non-coherent</td>
</tr>
<tr>
<td>TDRSS functionality</td>
<td>Data Group 1, Mode 2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Data Group 2, non-coherent</td>
<td></td>
</tr>
<tr>
<td>Forward Error Correction</td>
<td>½ rate convolutional encoding</td>
<td>½ rate Viterbi decoding</td>
</tr>
<tr>
<td>User Data Rates</td>
<td>24 kbps (spread), 192, 769 kbps (non-spread)</td>
<td>18 kbps (spread), 155, 769 kbps (non-spread)</td>
</tr>
<tr>
<td>Scrambling</td>
<td>IESS-308, V.35</td>
<td></td>
</tr>
<tr>
<td>Data Formatting</td>
<td>NRZ-M</td>
<td></td>
</tr>
</tbody>
</table>
Space Telecommunications Radio System

Radio Platform

GPM (GPP)
- STRS OE
  - WF App. (from PIM)
    - WF Control: Modulator

SPM (FPGA)
- STRS API
  - HAL
  - HID
  - Hardware Abstraction Layer

User Data Interface

Common APIs


Hardware Interface Definition

Data Conversion/Sampling

Data Format Converter

Encoder

Modulator

CLK

Carrier Synthesizer

HID

HAL
Development Approach

Waveform Development on COTS SDR

Port to JPL SDR (prototype)

TDRSS Performance Testing

TDRSS Firmware Heritage

STRS Reference OE

STRS Reference WF

STRS Compliance Testing

Flight SDR

BPM Prototype

CoNNeCT SDR Development

STRS Compliant OE

Documentation: HID, Dev Guide, Test WF
Processor Code porting - SLOC

- Estimate (3500)
- Device Driver
- WF Class

Source Lines Of Code

<table>
<thead>
<tr>
<th></th>
<th>Estimate</th>
<th>Device Driver</th>
<th>WF Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initial</td>
<td>3690</td>
<td>850</td>
<td>2346</td>
</tr>
<tr>
<td>Ported</td>
<td>3192</td>
<td>846</td>
<td>2346</td>
</tr>
</tbody>
</table>
## FPGA Utilization

<table>
<thead>
<tr>
<th>FPGA Resource</th>
<th>Initial Utilization</th>
<th>Ported Utilization</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total Slice Registers</td>
<td>94.5 %</td>
<td>59.8 %</td>
</tr>
<tr>
<td>4 input LUTs</td>
<td>90.0 %</td>
<td>70.4 %</td>
</tr>
<tr>
<td>occupied Slices</td>
<td>176.7 %</td>
<td>99.9 %</td>
</tr>
<tr>
<td>Slices containing only related logic</td>
<td>176.7 %</td>
<td>94.1 %</td>
</tr>
<tr>
<td>Slices containing unrelated logic</td>
<td>0 %</td>
<td>5.9 %</td>
</tr>
<tr>
<td>4 input LUTs</td>
<td>98.2 %</td>
<td>72.4 %</td>
</tr>
<tr>
<td>MULT18X18s</td>
<td>109.4 %</td>
<td>85.4 %</td>
</tr>
</tbody>
</table>

*porting of the waveform involved reducing the functionality of the original GSFC waveform so as to fit into the smaller JPL SDR FPGAs. There was also a speed reduction constraint.*
Porting Effort Overview

- 374 working (8 hour) days total effort divided between 3 engineers
- total calendar time 2 years
- tools used/required: Matlab/Simulink, Synplicity HDL synthesis (now Synopsis), Xilinx ISE, RTEMS development tools, Prototype BPM
- Does not include CoNNeCT System integration, performance, and environmental testing (vibe, thermal vacuum, EMI)
- NOTE: Porting effort blurs with system integration and flight platform specific functions. The COTS platform did not have an RF front end.
Almost half of the porting effort was not related to waveform reuse.
How did the WF port benefit with STRS?

1. Software for control was recompiled for new target processor, because of standard APIs.

2. Commanding and configuring from OE was the same, because of standard APIs.

The OE integration & WF Control slice would have been significantly larger.
Conclusions

1. Porting from more capable platform can be difficult:
   - Waveform design may need to change (e.g. analog I/Q mod instead of digital)
   - Reduction in features/performance.

2. SDR Platform should compensate for all temperature effects with OE and/or dedicated HW. However, some effects are waveform dependent.

3. STRS Architecture was helpful for this development:
   - despite the COTS to space-based platform disparity the standard APIs reduced porting effort.
   - Allowed for some parallel development, (forced by schedule constraints)

4. Better metrics could be found in a comparison of COTS to JPL Prototype, or a port of the current waveform on the JPL Flight SDR to another STRS flight SDR.
Dale J. Mortensen
ASRC Aerospace Corp. @ NASA Glenn Research Center
216-433-6698
dale.mortensen@nasa.gov

http://spaceflightsystems.grc.nasa.gov/SpaceOps/CoNNeCT/