

Partial Reconfiguration for Space Applications*

XRTC Annual Conference 2010

Jonathon W. Donaldson, Jeff L. Kalb
Sandia National Laboratories

UNCLASSIFIED

Approved for Unlimited Release
SAND 2010-XXXX



***Various slides/images courtesy Xilinx, Inc. and
Brigham Young University with modifications
by Jonathon W. Donaldson**



Agenda

- ◆ Introduction
- ◆ Approach
- ◆ BYU Bitstream Compression
- ◆ Hardware Support for PR
- ◆ Current Applications





Agenda

◆ Introduction

◆ Approach

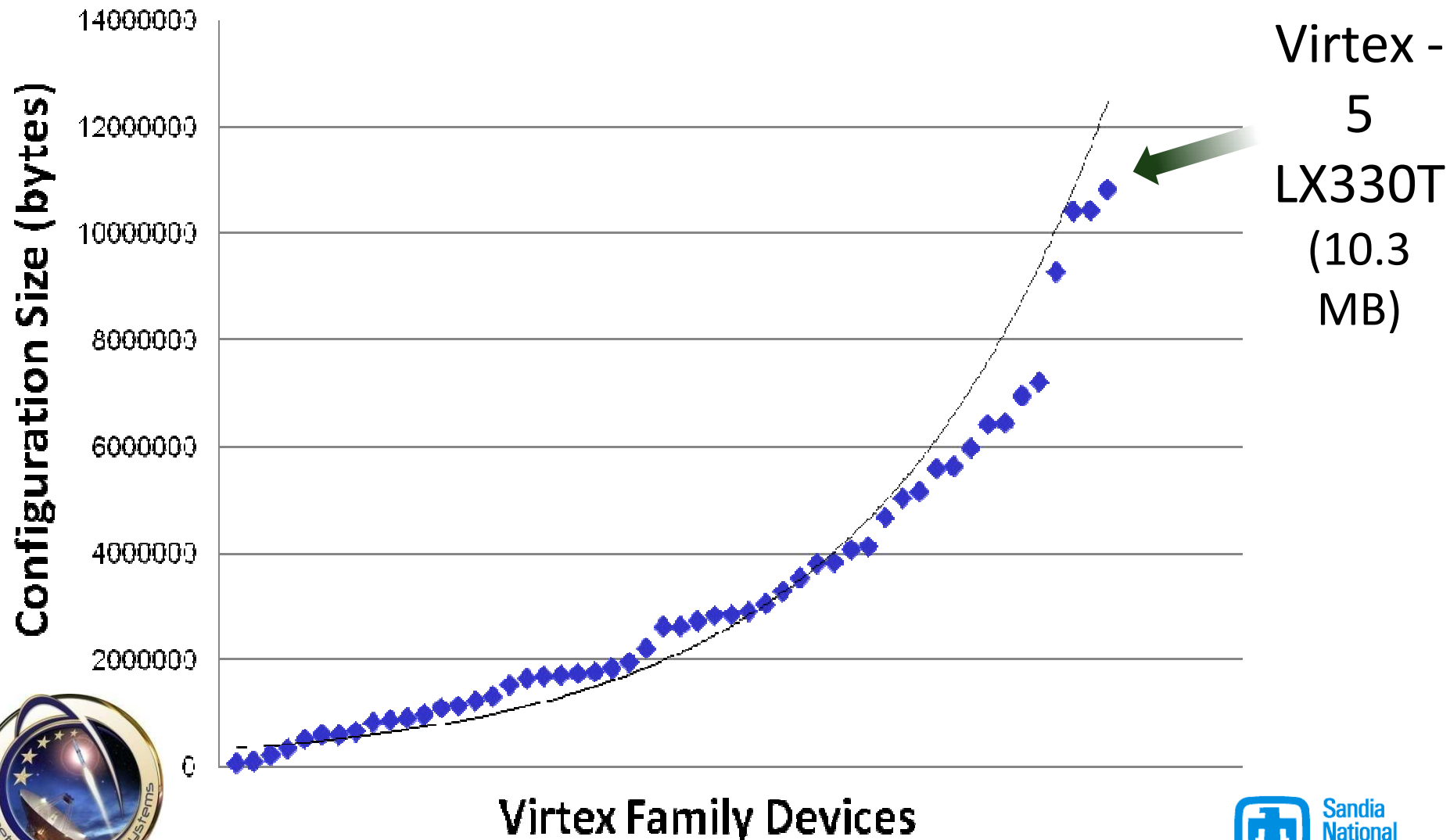
◆ BYU Bitstream Compression

◆ Hardware Support for PR

◆ Current Applications



Increasing FPGA Configuration Size



Virtex -
5
LX330T
(10.3
MB)





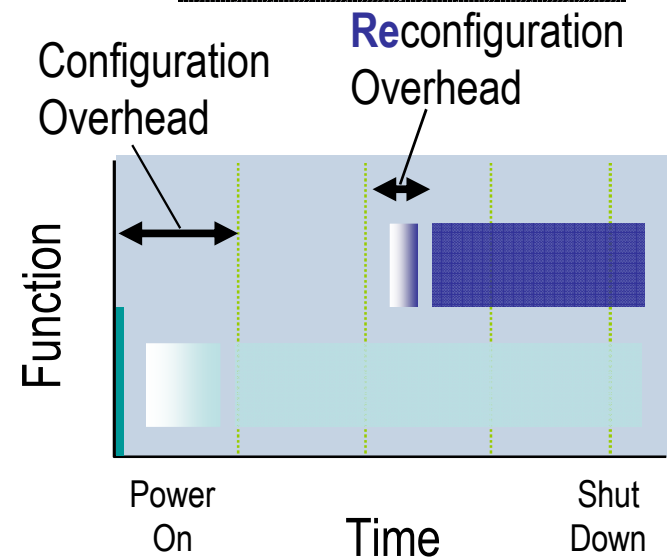
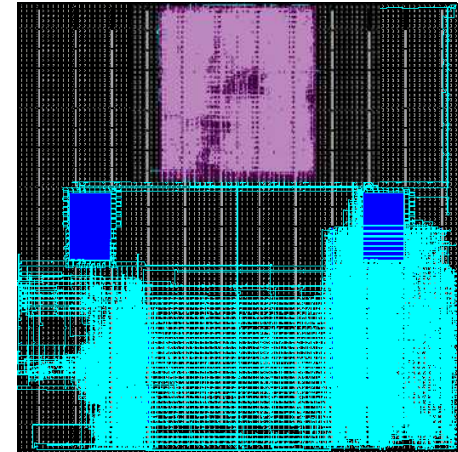
Why do we care about PR?

- Configuration mechanism of DOE/NNSA's Joint Architecture Standard (JAS) reconfigurable elements
 - Central NV memory allows reconfigurable flight system where applications can be loaded into nodes to meet system requirements and to recover from fault conditions
 - Boot FPGA off minimal bit file stored locally reduces need for local NV memory
 - Having a protocol endpoint in the static region allows reliable transport of the partial bit file over the network
 - Incorporating compression reduces network traffic and central NV memory requirements
- Management of individual applications
 - Allows operation of remaining functions during reconfiguration of a single application
 - Partial bitfiles reduce uplink bandwidth required for on-orbit hardware upgrades



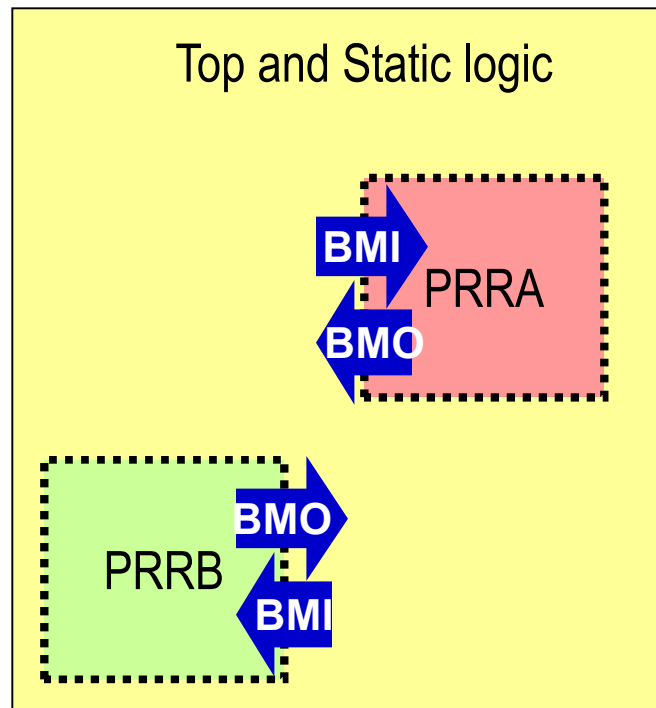
Dynamic Partial Reconfiguration

- A subset of the configuration data changes...
- But logic layer **continues operating** while configuration layer is modified...
- Configuration overhead limited to circuit that is changing...



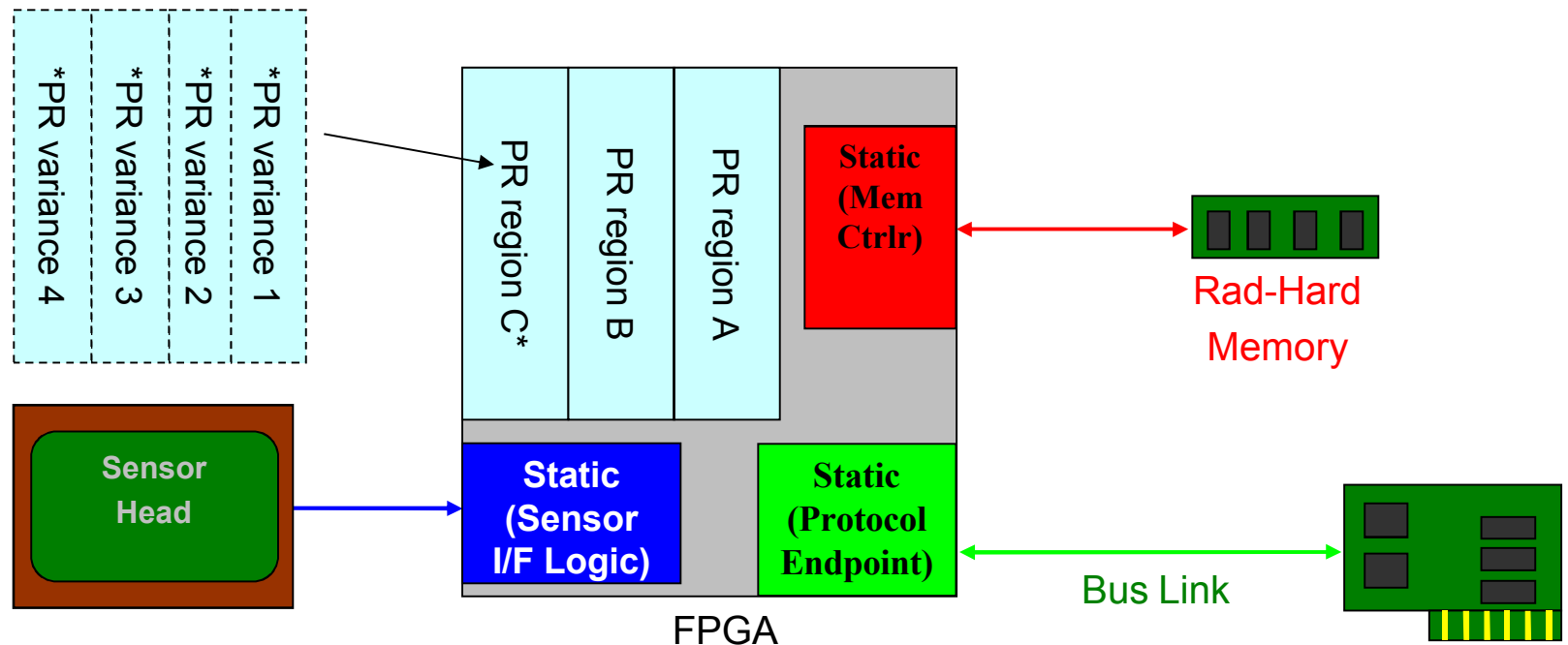
No More Bus Macros!

- All inputs and outputs for the PRRs used to require bus macro(s)
 - Used as fixed routing points
 - Think of it like a plug and socket for each PRR
 - All PRMs must have the exact same port list
- **Latest PR tools take care of this “auto-magically”**



PR Applications

- Maintain real-time link with other hardware during configuration change





Space-based FPGA Systems

- Storing FPGA bitstreams in radiation environments is difficult
- Radiation Hardened (RadHard) memory is very expensive
- RadHard memories have very low density
- Decompression circuitry required
- **Goal: Reduce configuration data stored in non-volatile memory**
 - How can we achieve this?





Agenda

◆ Introduction

◆ Approach

◆ BYU Bitstream Compression

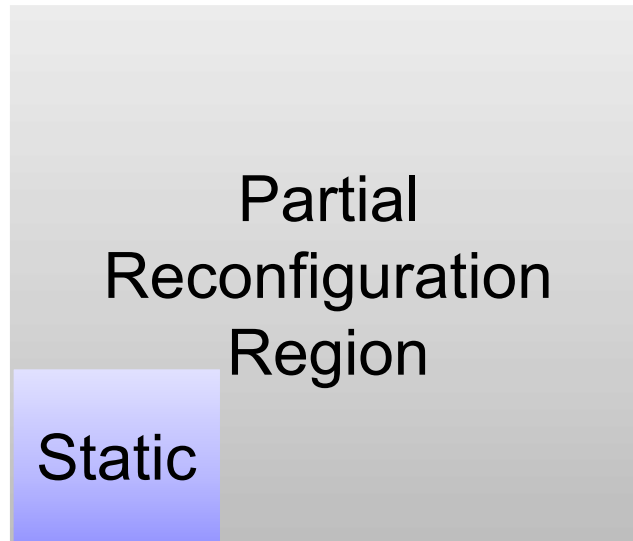
◆ Hardware Support for PR

◆ Current Applications





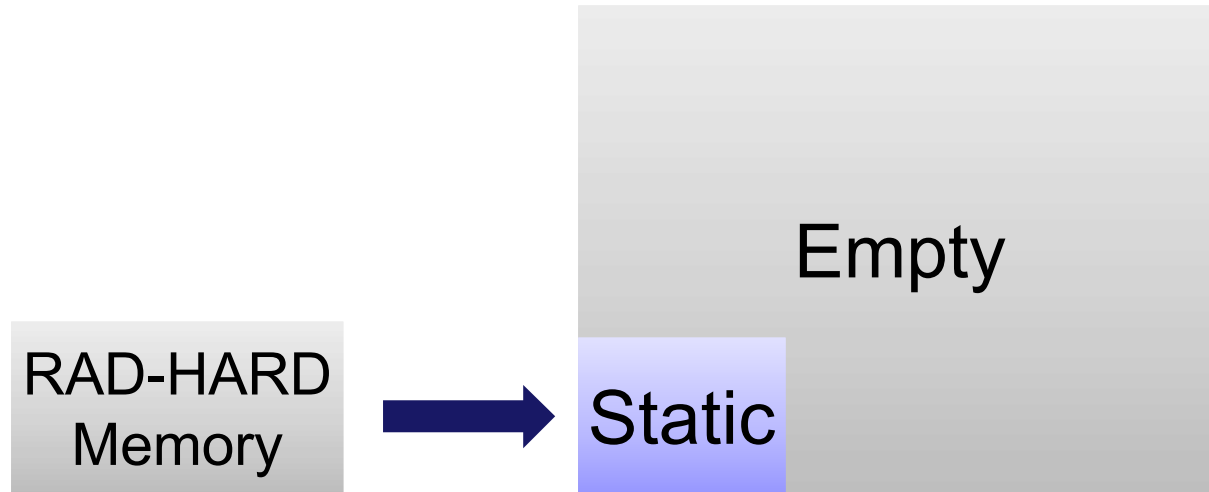
Organize Design into Two Components



- Static Region configured at initialization
 - Small – just enough for comm link
- Dynamic region configured after initialization through Partial reconfiguration



Compress and Store Configuration Bit Stream for Static Design

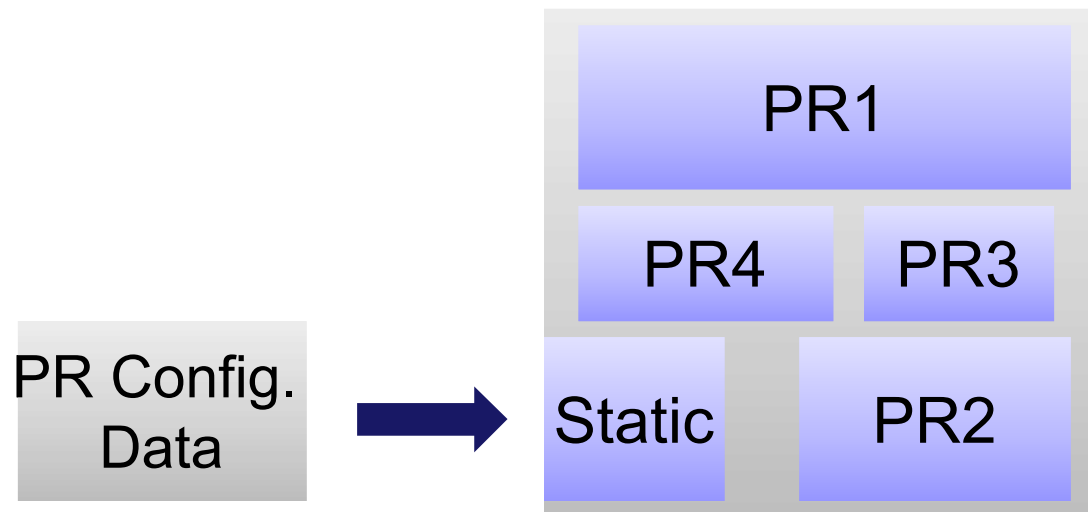


- Design only contains static logic (without PR)
- High compression ratio since static design is small
- Static design stored **locally** on node





Complete Functionality by PR



- Configure over network through static design using ICAP or direct access to SMAP/JTAG port
- Other mechanisms can be used to reliably store PR bit streams (ECC, disk, network, etc.)





Agenda

◆ Introduction

◆ Approach

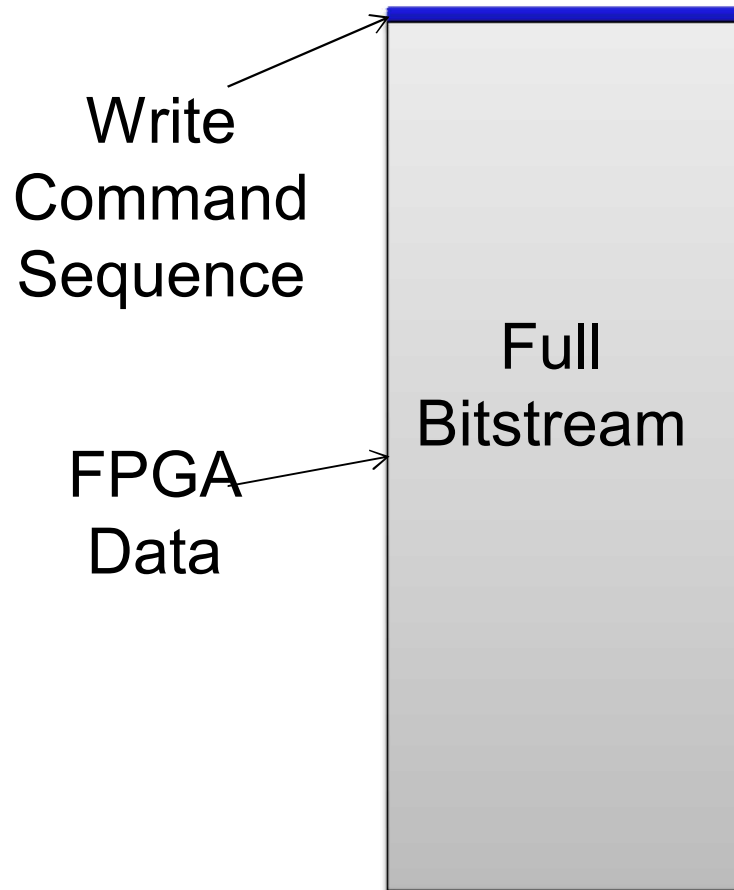
◆ **BYU Bitstream Compression**

◆ Hardware Support for PR

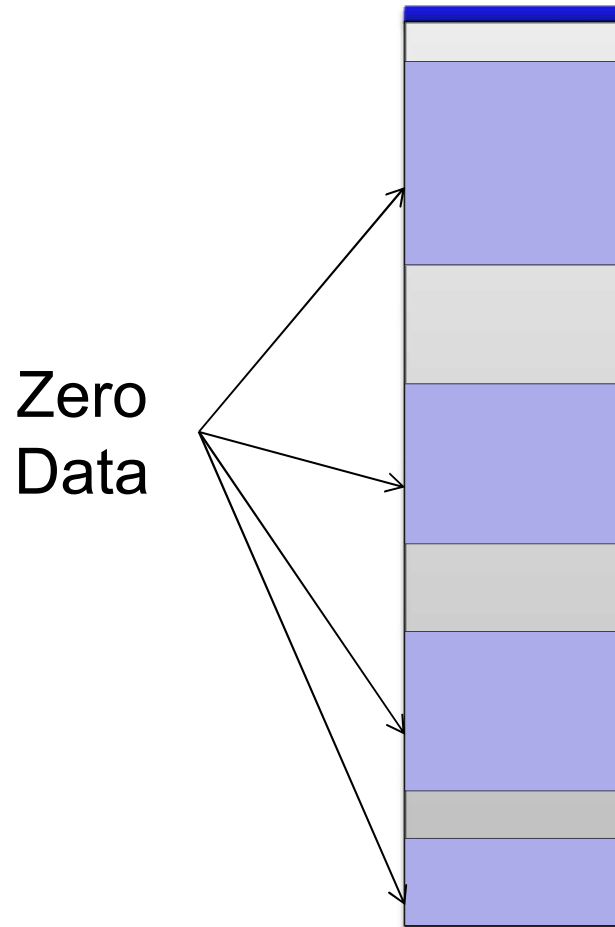
◆ Current Applications



Empty Frame Removal - Create Full Bit stream

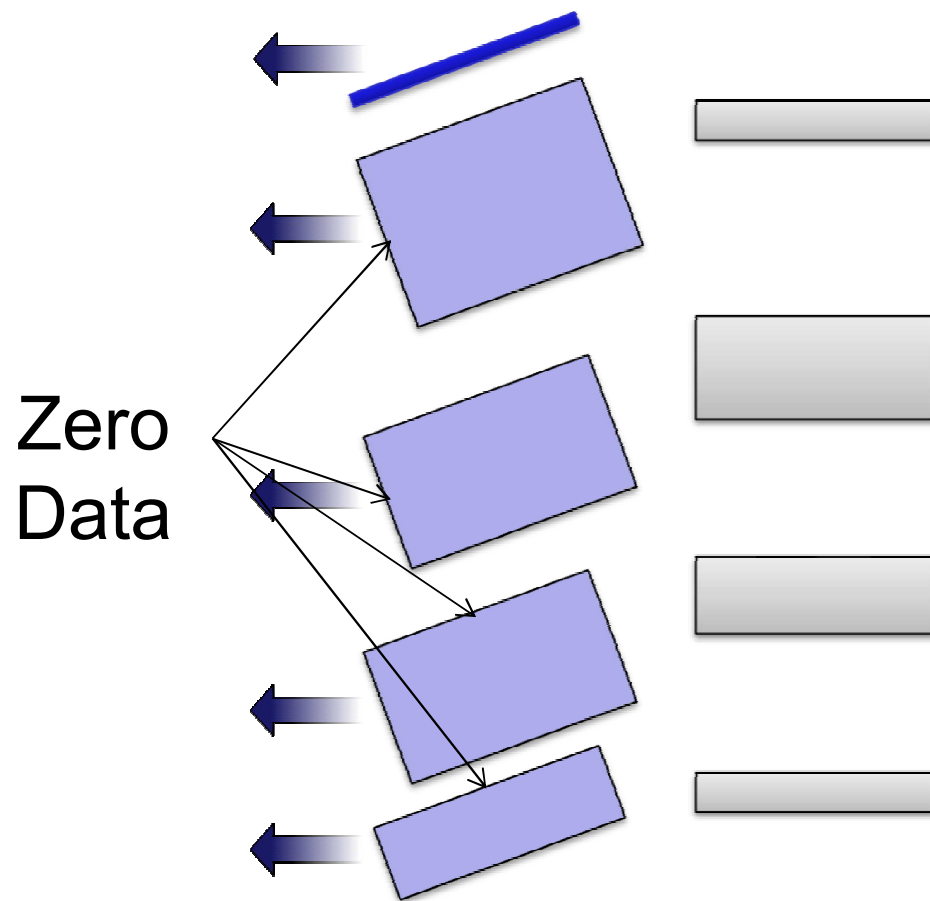


Empty Frame Removal – Identify Unused Logic

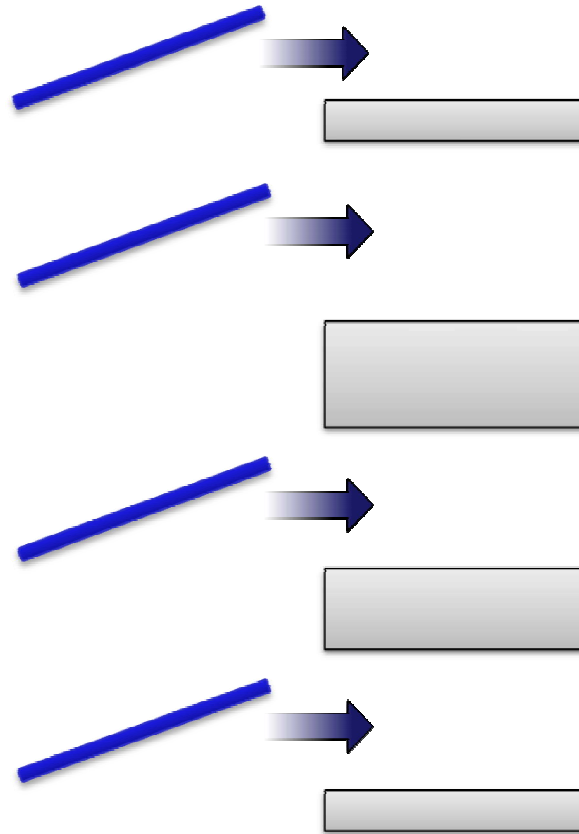




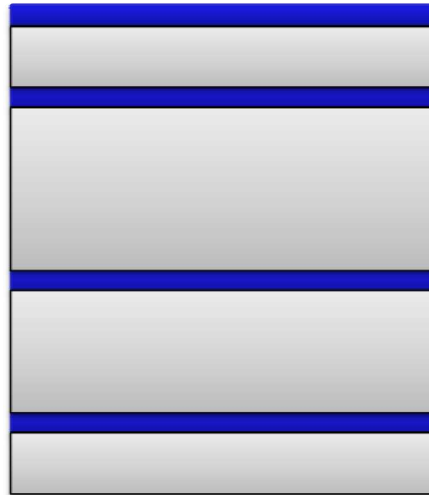
Empty Frame Removal – Remove Unused Logic and Old Command Sequence



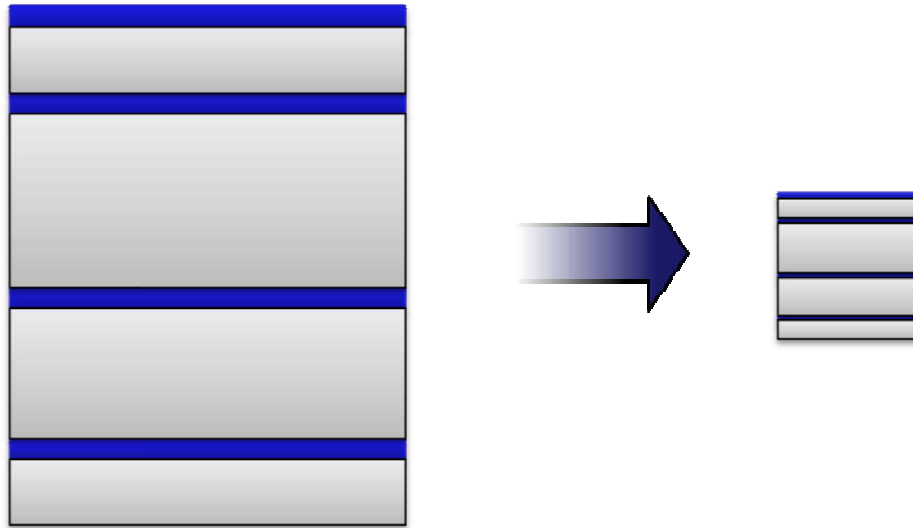
Empty Frame Removal – Add New Write Command Sequences



Empty Frame Removal – Final Compressed Bit Stream



Hardware-based Algorithmic Compression (LZRW3)



- **Further reduction using algorithmic compression**
 - Requires on-board decompression circuitry





Agenda

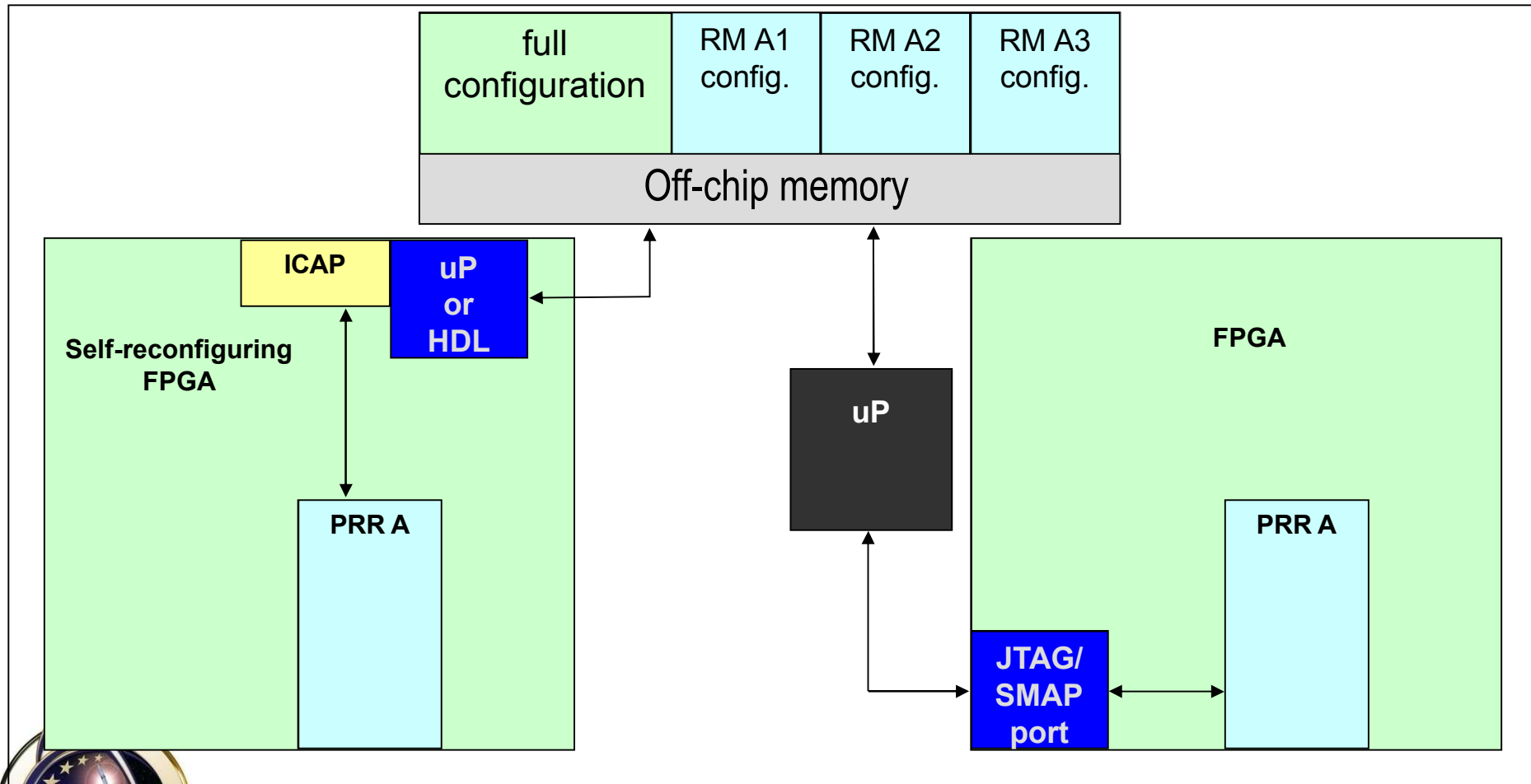
- ◆ Introduction
- ◆ Approach
- ◆ BYU Bitstream Compression
- ◆ **Hardware Support for PR**
- ◆ Current Applications



Methods for Loading a Partial Bit File

Internal Reconfiguration

External Reconfiguration via uP





Internal Configuration Access Port (ICAP)

- ICAP is the internal configuration access port for Virtex-II and later-generation Virtex devices
- It is a functional subset of SelectMap and is accessible internally via a user design
- It allows the user design to control device reconfiguration at run-time
- It becomes available **after** initial (externally controlled) configuration is complete



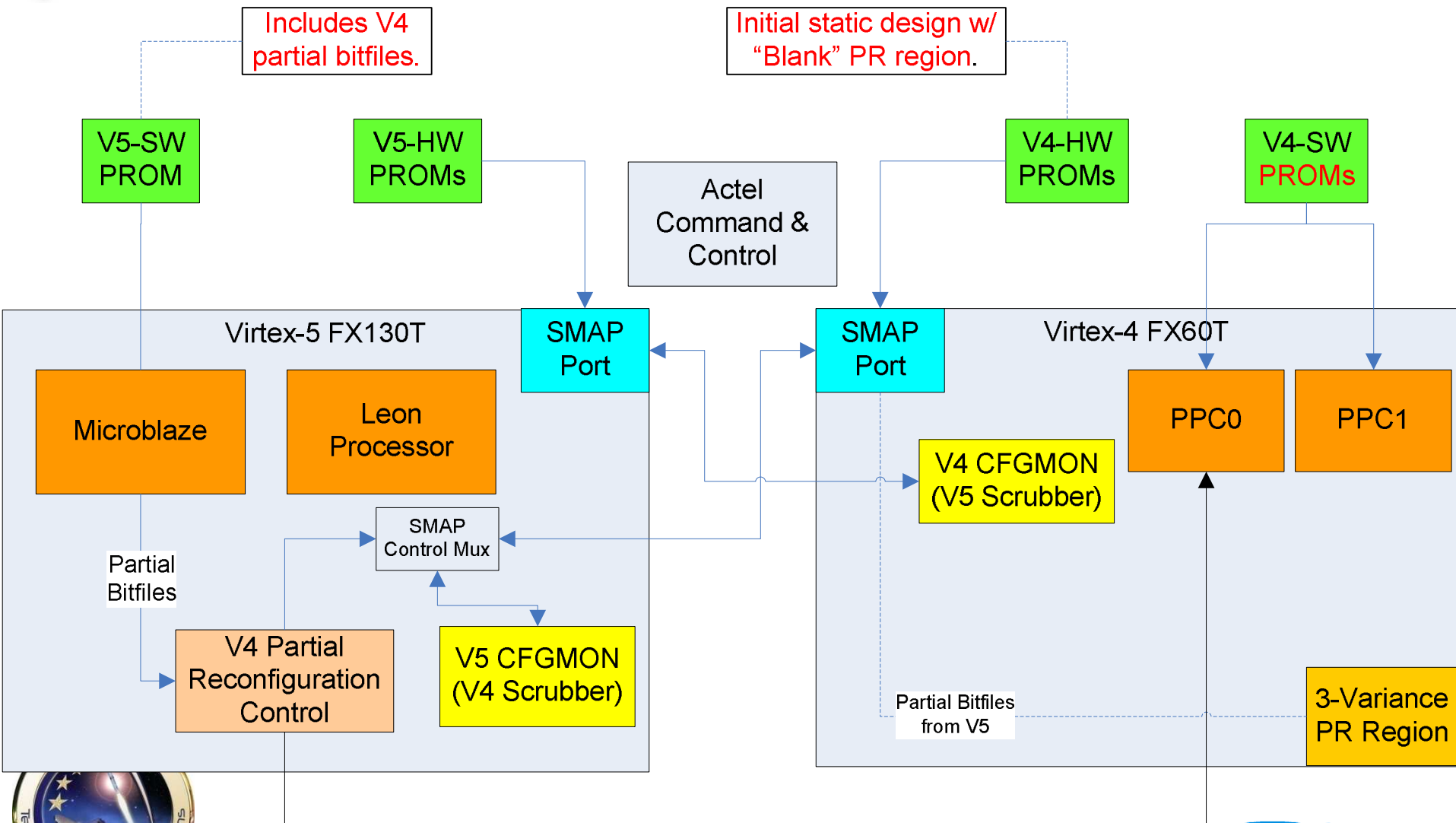


Agenda

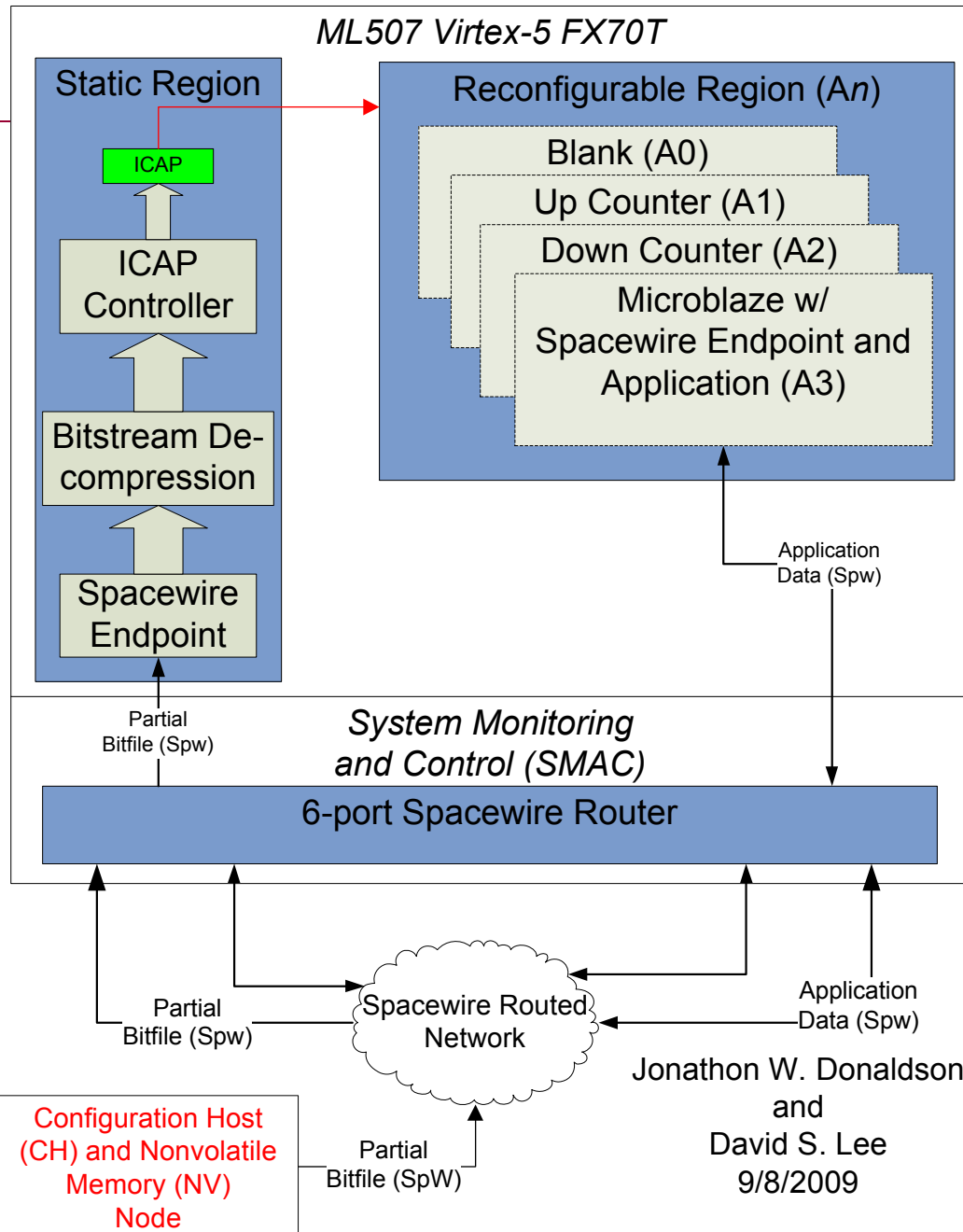
- ◆ Introduction
- ◆ Approach
- ◆ BYU Bitstream Compression
- ◆ Hardware Support for PR
- ◆ **Current Applications**



Loading a Partial Bit File on MISSE8



PR Over Spacewire for JAS



Jonathon W. Donaldson
and
David S. Lee
9/8/2009