Lynx: A scalable solution utilising FPGAs for high performance data processing

Part B: Firmware & Software Development December 2009

CSIR DPSS Presentation by Richard Focke Experimental Radar Systems Team Firmware / Software Developer





Introduction: Goals of my Presentation

- Background on firmware
- Demonstrate our design
- Showcase novel concepts
- Not only applicable to radar
- Possible issues



© CSIR 2009



Part B: Firmware & Software Development Topics

- **1. FPGA Firmware Background**
- 2. System Design Goals
- 3. Board Support Package
- 4. BSP Enabling Systems
- 5. Data Processing Firmware
- 6. Development Issues





© CSIR 2009

www.csir.co.za

1. What is FPGA Firmware?







1. HDL Uses

- HDL for hardware **simulation** only
 - Model 3rd party hardware
 - Test bench for functional verification
 - Read and write files
 - Dynamic structures
- HDL for FPGA firmware (or ASIC)
 - Not all language features
 - Register transfer level (RTL)
 - Model basic FPGA blocks
 - Instantiate advanced FPGA blocks



© CSIR 2009



Part B: Firmware & Software Development Topics

- 1. FPGA Firmware Background
- **2.** System Design Goals



© CSIR 2009



2. System Design Goals

- Adaptability & Scalability
- All FPGAs one processing space
- Migrate-able functions
- Decoupled hardware interfaces
- Functional independence
- Software: generic adaptable
- Consistent software interface
- Independent of processing flow



© CSIR 2009



Part B: Firmware & Software Development Topics

- 1. FPGA Firmware Background
- 2. System Design Goals
- **3.** Board Support Package



© CSIR 2009



4. Board Support Package (BSP)

- Make in-house boards usable
- Appropriate BSP crucial
 - Memory
 - Communications
 - Control
 - Monitoring
 - Real-time data viewing
 - Real-time data capture
- BSP abstract FPGA
 - Constrains function
 - Not dictate design



© CSIR 2009



3. BSP Components

- Components of the BSP:
 - Firmware wrapper (Processing FPGAs)
 - System controller (SCON)
 - RSP controller (RSPCON)
 - CCM FPGA
- Extra systems:
 - Display processor (DIP)
 - Data Acquisition Processor (DAP)





3. BSP Firmware Wrapper





3. BSP Control Interface





© CSIR 2009



3. BSP Dynamic Memory Map

- Command Interpretation
 - Address translation on each layer
 - Group size and implementation
- System Usability
 - Able to determine the meaning of each group
 - Can apply a template to define registers
 - Search for and use any register in any group
 - Rescan can automatically pick-up changes



© CSIR 2009



3. BSP High-speed data interface

- Based on Xilinx RocketIO
 - Utilising Xilinx Aurora (free)
 - SerialRapidIO, PCIe, 10GBE & FibreChannel
- Combine two RocketIO channels 6.4 Gbps
 - Carry 128-bit data at 40 MHz (640 MBps)
 - Potentially can combine up to 8 links!
 - Data conveyed as bursts with headers.





3. BSP External Interfaces





Part B: Firmware & Software Development Topics

- 1. FPGA Firmware Background
- 2. System Design Goals
- 3. Board Support Package
- **4.** BSP Enabling Systems



© CSIR 2009



4. System Controller (SCON) Overview

- Control Software Application for signal processor
- Architecture
 - Layered OO design using C++ Builder (Windows)



Role

- User interface for the signal processor
- RSP Setup, Control, Monitoring and BIT
- Centralised Facility Mode and Status Indication
- Developed in-house





© CSIR 2009



4. SCON Features

- High level of re-configurability using XML
 - Add extra processing functions
 - No recompile necessary
- Extensible event-driven OO framework
 - Add new software functionality
- Highly multi-threaded design allows multiple concurrent background and foreground tasks



© CSIR 2009



4. SCON Reconfigure-ability

• VME Boards

© CSIR 2009

- Memory Map in XML format (Gateway to Hardware)

 <Register Name="STATUS_REGISTER" Address="0x00000000"/>
- Static Memory Map
 - Board level registers
 - Parsed after VME discovery process
- Dynamic Memory Map (TLV representation)
 - Discovery process:
 - Read TLV-type registers of function
 - XMLs parsed based on structure and type value.
 - Rediscovery when:

- New Firmware Component added
- Firmware Component changes





4. SCON GUI Types

• Static design-time GUI



XML-Based Dynamic GUI

🗰 Functional Parameters						
Radar Signal Processor						
RSP Functions Burst Check Functions OAC Interface Oightal Pulse Generators Equalization Intermediate Frequency Sampler Timing Generation Function Control and Status	TGF Control Registers Mode Control TGF Mode Pattern Selection Pattern Source	Standby	Get Get Get Elevation Elevation	Get All Set All ↓	Auto Read Read on Load	.ast Update: ? I
- Static Parameters	DPG Control	0x0000000 Set	t Get TCF Target Range BIN	0x00000000	Set Get	
- Strobe Status - TSB 0 - TSB 1 - TSB 2 - TTG Control	TimeCode Current Time Current Time Fra	0xCEC79BB9 Set	t Get Pattern Control Pattern Number tionalGuiDescription_005_000_	0x00000000 000_000.xml	Set Get	
MUX Setun	STC Code Selec	dit <u>V</u> iew Project Team XML	Tools <u>W</u> indow Help			
	STC Code 1 🖳 Fund	ctionalGuiDescription_005_000_000_00	00.xml			
	Table Loading Sta WaveForm Table	1 ?xml version="1.0" enc 2 GuiDescription xmlns:x 3 <groupingcontrol td="" type<=""> 4 <groupingcontrol td="" ty<=""> 5 <regcontrol td="" uniqu<=""> 6 <item label="St</td></th><th>coding=" utf-8"?=""> <pre> xsi="http://www.w3.org/200 =="GroupBox" Label="TGF CC /pe="GroupBox" Label="Mode ueName="MODE_CTRL_REG" Loc tandby" Signal="0x00000000"</pre></item></regcontrol></groupingcontrol></groupingcontrol>	D1/XMLSohema_instance" ontrol Registers"> e Control" Top="22"> cation="D1:F5:I0:S0:i0: 0"/>	x <u>si:noNa</u> mes <u>p</u> :CONTROL_REG	aceSchemaLocation="Funct " Type="DropDown" Label*	
	Frequency Table	7 <item label="Op
8 </RegControl>
9 <RegControl Uniqu
0 <Item Label=" o"<="" th=""><th><pre>perate" Signal="0x0000000 neName="MODE_CTRL_REG" Loc '_Signal="0x00000000"/></pre></th><th>4"/> cation="D1:F5:I0:S0:i0:</th><th>:CONTROL_REG</th><th>" Type="DropDown" Label=</th></item>	<pre>perate" Signal="0x0000000 neName="MODE_CTRL_REG" Loc '_Signal="0x00000000"/></pre>	4"/> cation="D1:F5:I0:S0:i0:	:CONTROL_REG	" Type="DropDown" Label=



4. Ethernet-VME Bridge



- RSPCON COTS Single-board VME PC
 - Ethernet Messages ↔ VME Transactions
 - Decouples SCON from VME standard
 - Runs on VxWorks
- CCM Configuration Control and Monitoring FPGA
 - VME Transactions \leftrightarrow LVDS (or GBE \leftrightarrow LVDS)
 - Monitors hardware
 - Configures FPGAs
 - Determines placement of functions on FPGAs (Dynamic Memory Map)



© CSIR 2009



4. Display Processor

- Components:
 - Embedded PowerPC 440 running VxWorks
 - **Display GUI**
- Functions performed:
 - Display data from any processing point _
 - Intensity image plot, 3D & 2D Waterfall plot A-trace & R-trace
 - Implemented using T-chart -
- Enables:
 - Verification of processing functions -
 - Provides human comprehensible information -



© CSIR 2009



4. Data Acquisition Processor (DAP)

- Components:
 - DAP Console: GUI for setup and recording
 - Curtiss-Wright Phoenix M6000 VXS Card
 - Fibre-optic Storage Area Network
- High Speed Data Capture
 - From any processing point on any card:
 - data filtering
 - interleaving multiple channels
 - Supports recording data rates up to 640 MB/s (SI Mega)
 - 3.6 TB of storage capacity equates to 1 hour recording of LYNX RSP data at full data rate



© CSIR 2009



Part B: Firmware & Software Development Topics

- 1. FPGA Firmware Background
- 2. System Design Goals
- 3. Board Support Package
- **4.** BSP Enabling Systems
- **5.** Data Processing Firmware



© CSIR 2009



5. Data Processing Firmware

- Types of functions:
 - Generation of timing & signals
 - Data conditioning functions
 - Mathematical processing
- Required elements:
 - Memories
 - Adders, subtractions, multipliers & dividers
 - State-machines
- Design constraints:
 - FPGA configurable logic
 - FPGA DSP building blocks
 - Memory capacities



© CSIR 2009



5. How to Create a Data Processing Function

- Control / Status Interface:
 - Compile list of configurable parameters
 - Determine the necessity for tables of data
 - Implement the required DMM interface
- Data Processing:
 - Partition available FPGA resources per function
 - Using actual numbers determine which algorithms will fit
 - Make use of over-clocking and SIMD to expand the available resources
 - Implement the chosen algorithm using suitable tools



© CSIR 2009



Part B: Firmware & Software Development Topics

- 1. FPGA Firmware Background
- 2. System Design Goals
- 3. Board Support Package
- 4. BSP Enabling Systems
- 5. Data Processing Firmware
- **6.** Development Issues





6. Development Issues

- Mapping firmware into clock regions
- Full FPGAs (> 75%) cause routing problems
- High speed designs > 400 MHz placement sensitive
- Automatic BRAM and DSP mapping: poor timing & no control
- IP cores require a lot of FPGA resources: sometimes have bugs
- Tools over-promise & under-deliver



© CSIR 2009





Conclusion

- Adaptable & scalable system
- Easy use through BSP
- Generic data processing
- Covered development issues



© CSIR 2009





Part B: Firmware & Software Development Topics

A. Development Tools



© CSIR 2009



A. Mentor Graphics HDL Author & Modelsim



Lynx LVDS Design



© CSIR 2009



A. Mentor Graphics HDL Author & Modelsim

- Hierarchical Graphical Design for HDL
- Pros:
 - FPGA Vendor Independent
 - Graphical view intuitive & easy to manipulate
 - Hierarchical design: maintainable & reusable
 - Only leaf nodes of hierarchy coded in HDL
- Cons:
 - Not an integrated development environment
 - Expensive license
 - Poor integration with some design tools



© CSIR 2009



A. Xilinx Integrated Synthesis Environment (ISE)

Surces for: Implementation Implemen	rogramming File Generated o Errors S372 Warnings I Signals Completely Routed Constraints Mat (Setup: 0, Hold: 0, Component Switching L (Timing Report)	mplementation State: Programming File Generated • Errors: • Routing Results: • Routing Results: • Routing Results: • Routing Constraints: All Sinals Completely Routed • Timing Constraints: All Constraints Met • Final Timing Score: • Getup: 0, Hold: 0, Component Switching Limit 0) Chining Resort) XPS Reports Errors: Errors: • Used: • Used: • Variable: • Used: • Sinalbe: • Used: • Sinalbe: • Sinalbe:		X		^	I_FPGA_TOP	LYNX1 Project Stat	us (12/0	3/2009 - 12	:11:40)		
Image: Second	o Errors 5372 Warnings 15 grads Completely Routed 1 Constraints Met (Setup: 0, Hold: 0, Component Switching L (Timing Report)		Sources for: Implementation	× .	IOB Properties	Project File:	I FPGA TOP LYNX1.ise	Implementation S	itate: Programming File Generated				_
Image Constraints Timing Constraints Image Constraints Timing Constraints Image Constraints Phote Report Image Constraints Phote Report Image Constraints Static Timing Image Constraints Al Signals Completaby Image Constraints Image Constraints Image Constraints Signals Completaby Image Constraints Signal	S372 Warnings Signals Completely Routed Lonstraints Met (Sectup: 0, Holds 0, Component Switching L (Timing Report) mings Infos		Hierarchy	A G	Module Level Utilization	Module Name:	I_FPGA_TOP_LYNX1	• Errors:		No Errors			
Image: Spring and Spring	I Signals Completely Routed Constraints Met (Setup: 0, Hold: 0, Component Switching L mings Infos	Routing Results: Al Signals Completely Routed Timing Constraints: Al Constraints Met Final Timing Score: U Setup: 0, Hold: 0, Component Switching Limi 0 (Setup: 0, Hold: 0, Component Switching Limi 0 (Setup: 0, Hold: 0, Component Switching Limi VSF Reports Varings Vari	I_FPGA_TOP_LYNX1		Timing Constraints	Target Device:	xc5vfx200t-2ff1738	• Warnings:		25372 Warni	ings		
Border Lipsen Versienser (DC) Constraints Design Summary/Reports Dors Arts, Bufferen, 0 - LVOS, RX, Warrings Design Summary/Reports Design Constraints	I Constraints Met (Setup: 0, Hold: 0, Component Switching I (Timing Report)		(D: \Projects\Lvnx\RSP\Firmware\T	c 🕻 🕻	Clock Report	Product Version:	ISE 11.2	• Routing Re	sults:	All Signals Co	mpletely Route	ed	
Besign Strategy: Xiinx Default (unlocked) Final Timing Score: O (Setup: 0, Hold: 0, C O) (Timing Recort) Besign Strategy: Xiinx Default (unlocked) Final Timing Score: O (Setup: 0, Hold: 0, C O) (Timing Recort) Besign Strategy: Xiinx Default (unlocked) Final Timing Score: O (Setup: 0, Hold: 0, C O) (Timing Recort) Besign Strategy: Xiinx Default (unlocked) Final Timing Score: O (Setup: 0, Hold: 0, C O) (Timing Recort) Besign Strategy: Xiinx Default (unlocked) Final Timing Score: O (Setup: 0, Hold: 0, C O) (Timing Recort) Besign Strategy: Xiinx Default (unlocked) Final Timing Score: O (Setup: 0, Hold: 0, C O) (Timing Recort) Besign Strategy: Xiinx Default (unlocked) Final Timing Score: O (Setup: 0, Hold: 0, C O) (Timing Recort) Besign Strategy: Xiinx Default (unlocked) Final Timing Score: O (Setup: 0, Hold: 0, C O) (Timing Recort) Besign Strategy: Xiinx Default (unlocked) Final Timing Score: O (Setup: 0, Hold: 0, C O) (Timing Recort) Besign Strategy: Xiinx Default (unlocked) Final Timing Score: O (Setup: 0, Hold: 0, C O) (Timing Recort) Besign Strategy: Xiinx Default (unlocked) Final Timing Score: O (Setup: 0, Hold: 0, C O) (Timing Recort) Besign Strategy: Xiinx Default (unlocked) Final Timing Score: O (Setup: 0, Hold: 0, C O) (Timing Recort) Besign Strategy: Xiinx Default (unlocked) Final Timing Score: O (Setup: 0, Hold: 0, C O) (Timing Recort) Besign Strategy: Xiinx Default (unlocked) Final Timing Score: O (Setup: 0, Hold: 0, C O) (Timing Recort) Besign Strategy: Xiinx Default (unlocked) Final Timing Score: O (Setup: 0, Hold: 0, C O) (Timing Recort) Besign Strategy: Xiinx Default (unlocked) Final Timing Score: O (Setup: 0, Hold: 0, C O) (Timing Recort) Besign Strategy: Strategy: Strategy Bitint Hessages Bitint Hessages Besintror and Warnings	(Setup: 0, Hold: 0, Component Switching L (Timing Report) mings Infos		(D: \Projects \Lynx \RSP \Firmware \T	c	Static Timing	Design Goal:	Balanced	• Timing Con	straints:	All Constrain	ts Met		
Def: Diff. Singen Messages Diff. Distant Messages Distant Messages Distant Messages Distant Messages Distant Messages Distant Messages Distant Messages Distant Messages Distant Messages Distant Log File Encore and Warnings Distant Log File Libgen Log File Design Utilities Synthesis Messages Distant Monte Messages Map Messages Distant Monte Messages	mings Infos	XPS Reports Infos Errors Warnings Infos I - - - I - - - I - - - I - - - I - - - I - - - I - - - I - - - I - - - I - - - I - - - I - - - I - - - I - - - I - - - I - - - I - - - I - - - I - - - I -	I_FPGA_TOP_LYNX1 - struct (D: DMM_PACKAGE (D:\Projects\L) DFE CLOCKS TOP I - DFE CL	3	PS Errors and Warnings Platgen Messages Libgen Messages	Design Strategy:	Xilinx Default (unlocked)	• Final Timing	Score:	0 (Setup: 0, 0) <u>(Timing Re</u>	Hold: 0, Compo	onent Switching	g Lim
CostC _ MAUSSR _ Pool LOC Cost > Report S Bit DoS _ MX_ESTR _ 0 - UDS _ RX, CostC _ MAUSSR _ 0 - UDS _ RX, CostC _ MAUSSR _ 0 - UDS _ RX, Dispert Log File CostC _ MAUSSR _ 0 - UDS _ RX, Dispert Log File CostC _ MAUSSR _ 0 - UDS _ RX, Dispert Log File Dispert Log File Dispert Log File Dispert Log File Dispert Log File Design Summary Reports Synthess Messages Bit CostC _ MAUSSR _ MAUS	rnings Infos	Linfox Unitation Errors Warnings Infos Image: Second	DFE_RIO_I - DFE_RIO_TOP - s	t (A)	Singen Messages BitInit Messages			VPC Describe					
NDS_RX_SUFFER_0 - I/DS_RX Liben Log File Report name Generated Crrois Warnings Processes: [_PPCA_TOP_LYNK1 - struct BitInit Log File Liben Log File Liben Log File Liben Log File Design Summary/Reports Translation Messages BitInit Log File Liben Log File Liben Log File Warnings Design Summary/Reports May Messages BitInit Log File Liben Log File Design Utilities May Messages BitInit Log File Liben Log File Liben Log File		Litors Workings IIIOs Important Important Important Important Impo	LOGIC_ANALYSER_POD_I - LO		Platgen Log File	Report Name	Concepted	APS Reports	V	Varminac	Info		
VIDS_RX_BUFFER_1 + LIDS_RX_M amgent cog me dopen Log File dopen Log File dopen Log File Singen Log File Singen Log File Singen Log File Singen Log File amgent cog me		Use Available Utilization Note(s) 65,234 65,234 53% 53%	LVDS_RX_BUFFER_0 - LVDS_RX		Libgen Log File	Plateen Log File	Generated	LITOIS	V	rannings	1110		_
Image: Second		Use Available Utilization Note(s) 65,249 122,880 53% 53%	LVDS_RX_BUFFER_1 - LVDS_RX	. 🞽	BitInit Log File	Libgen Log File							
rocesses: I_FPGA_TOP_I/NVIX - struct		Use Vallable Utilization Note(s) 65,249 122,880 53% 53% 65,234 65,234 53% 53%	***************************************	4	Errors and Warnings	Singen Log File							
Uesgn Julites Uesgn United States I Design Un		Use Utilization Summary Use Utilization Note(s) 65,249 122,880 53% 53%	ocesses: I_FPGA_TOP_LYNX1 - struct		Synthesis Messages	BitToit Log File							
User Constraints		Vertilization Summary Used Available Utilization Note(s) 65,249 122,880 53% 53% 65,234 65,244 65,248 53% 53%	 Design Summary/Reports Design Utilities 	115	Map Messages								
		Used Available Vlitization Note(s) 65,249 122,880 553%	Ser Constraints		Disco and Basts Massage								
Synthesize - XST Device Utilization Summary Device Utilization Summary	And the second s	Used Available Utilization Note(s) 65,249 122,880 533% 65,234	Synthesize - XST		Enable Enhanced Design Summary		Dev	rice Utilization Sum	mary				
A inperient Lesion Used Available Utilization Used Available Utilization Used Available Utilization	vailable Utilization Note(s)	65,249 122,880 53% 65,234	Translate		Display Incremental Messages	Slice Logic Utilizat	tion		Used	Available	Utilization	Note(s)	
Contract Part Contents Map Contents Map Contents Map	122,880 53%	65,234	в 🖓 🚺 Мар		Optional Design Summary Contents	Number of Slice Regi	sters		65,249	122,880	53%		
Place & Route Show Clock Report Number used as Flip Flops 65,234			Place & Route		Show Clock Report	Number used as Flip Flops		65,234					
Configure Tracet Device 15 Now Warnings Number used as Latch-thrus 15		15	Configure Target Device		Show Warnings	Number used as La	Number used as Latch-thrus						
Update Bitstream with Processor Data 🚽 🚽 Show Errors Number of Slice LUTs 68,066 122,880	122.880 55%	In the second	Update Bitstream with Processor Da	ta	- D Show Errors	Number of Slice LUTs	Number of Slice LUTs			122,880	55%		_
Analyze Design Using Chipscope Show Partition Data Number used as logic 65,010 122,880	122,000	68,066 122,880 55%	Analyze Design Using Chipscope		Show Partition Data	Number used as lo	Number used as logic			122,880	52%		_
	122,880 52%	68,066 122,880 55% 65,010 122,880 52%		>		Number using O	6 output only		49,130				_
Show Profs big/ubb 122,880	177.8801 55%-1	CO. 0.C. 100.000	ta	Show Errors		Number of Side LUIS	Number of Sice LUTs			122,880	55%		
Number used as logic 65,010 122,880	122,000 3376	68,066 122,880 55%	ng empscope	-		Number used as lo	Number used as logic			122,880	52%	-	
	122,880 52%	68,066 122,880 55% 65,010 122,880 52%				Number using O	6 output only		49,130				_
Number using O6 output only 49,130	122,880 52%	68,066 122,880 55% 65,010 122,880 52% 49,130	Files Libraries	ک	Design Summary (Programm	ing Hile Generated)							
Files Libraries Empirical Number using Ob output only 449,130	122,880 52%	68,066 122,880 55% 65,010 122,880 52% 49,130											++

• Lynx TGF ISE Project



© CSIR 2009



A. Xilinx Integrated Synthesis Environment (ISE)

- Integrated Design Environment for Xilinx
- Pros:
 - Quick and easy access to all design flows
 - Integrates well with all Xilinx applications
 - Easy to use and configure
- Cons:
 - Specific to Xilinx
 - Poor file management
 - Expensive license



© CSIR 2009



A. System Generator for DSP



Lynx IF Sampler Example



© CSIR 2009



A. System Generator for DSP

- High Level DSP Design Environment
- Pros:
 - Easy implementation of DSP algorithms
 - Visualisation tools for multi-rate systems
 - Big ready-to-use block set
 - Hardware in the loop simulation
 - Simulate using existing Matlab Simulink blocks
- Cons:
 - High level: difficult to trace problems
 - Poor state-machine design control
 - Matlab based, expensive



© CSIR 2009



A. Xilinx Embedded Development Kit (EDK)



Lynx DIP Embedded Processor



© CSIR 2009



A. Xilinx Embedded Development Kit (EDK)

- Embedded Processor Implementation
- Pros:
 - XPS: build processor subsystem quickly
 - SDK: eclipse IDE with GNU debugger
 - IP block set for processor type functions
 - Support for Linux, VxWorks and Xilkernel
- Cons:
 - Integrating own IP not well managed
 - Black-box approach to IP
 - Specific to Xilinx



© CSIR 2009