

L-3 Fuzing & Ordnance Systems

59th Annual Fuze Conference

May 5, 2016



L-3 FUZING & ORDNANCE SYSTEMS

PUBLIC DOMAIN. This document consists of general capabilities information that is not defined as controlled technical data under ITAR Part 120.10 or EAR Part 772.

Quality Driven Complex Logic Development Process

May 3rd - 5th, 2016

Open Session

Nick Adams



Objective and Exit Criteria

- Objective
 - Allow Groups an opportunity to understand the basic technical details of an FPGA, the overall development process and the specific design details that are undertaken by the L-3 FOS Complex Logic design team to provide a quality Complex Logic Device to our customer.
- Exit Criteria
 - Demonstrate an understanding of the FPGA complex logic component
 - Demonstrate an understanding of the development process
 - Demonstrate an understanding of the implementation methodology

Design Methodology

Areas involved in the development of a complex logic design

- Project planning and tracking
- Tool selection in the EDA (Engineering Design Automation) space
- Source control, version control and configuration management mechanisms
- Requirement traceability
- Design strategy and criteria
- Verification methods and coverage metrics
- Validation goals and a release and life cycle management plan

“Do it Once ... Do it Right”

Background Information

What is an FPGA?

An FPGA is **HARDWARE.**

- FPGA is a **Field Programmable Gate Array**
 - It **“IS”** an Integrated Circuit
 - It **“IS NOT”** Software
 - It **“IS”** Programmed
 - It **“IS NOT”** a program



FPGAs contain programmable components called "logic blocks", and a hierarchy of reconfigurable interconnects that allow the blocks to be "wired together".

Logic blocks can be configured to perform complex combinational functions, or merely simple logic gates like AND and XOR.

In FPGAs, the logic blocks also include memory elements or registers.

Hardware Descriptive Language (HDL)

In integrated circuit design, **Register Transfer Level (RTL)** description is a way of describing the operation of a synchronous digital circuit.

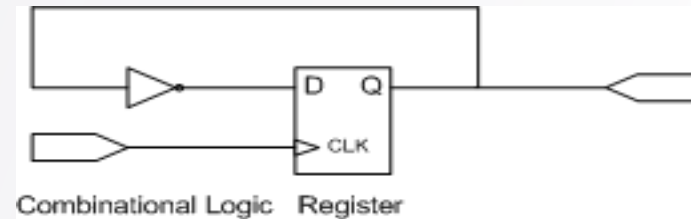
Behavior is defined in terms of:

- Flow of signals (transfer of data) between hardware registers.
- Logical operations performed on those signals.

RTL abstraction is used in **HDL's** or **Hardware Description Languages** like **Verilog** to create high-level representations of a circuit, from which lower-level representations and ultimately actual wiring is derived.

- FPGA's are programmed with HDL

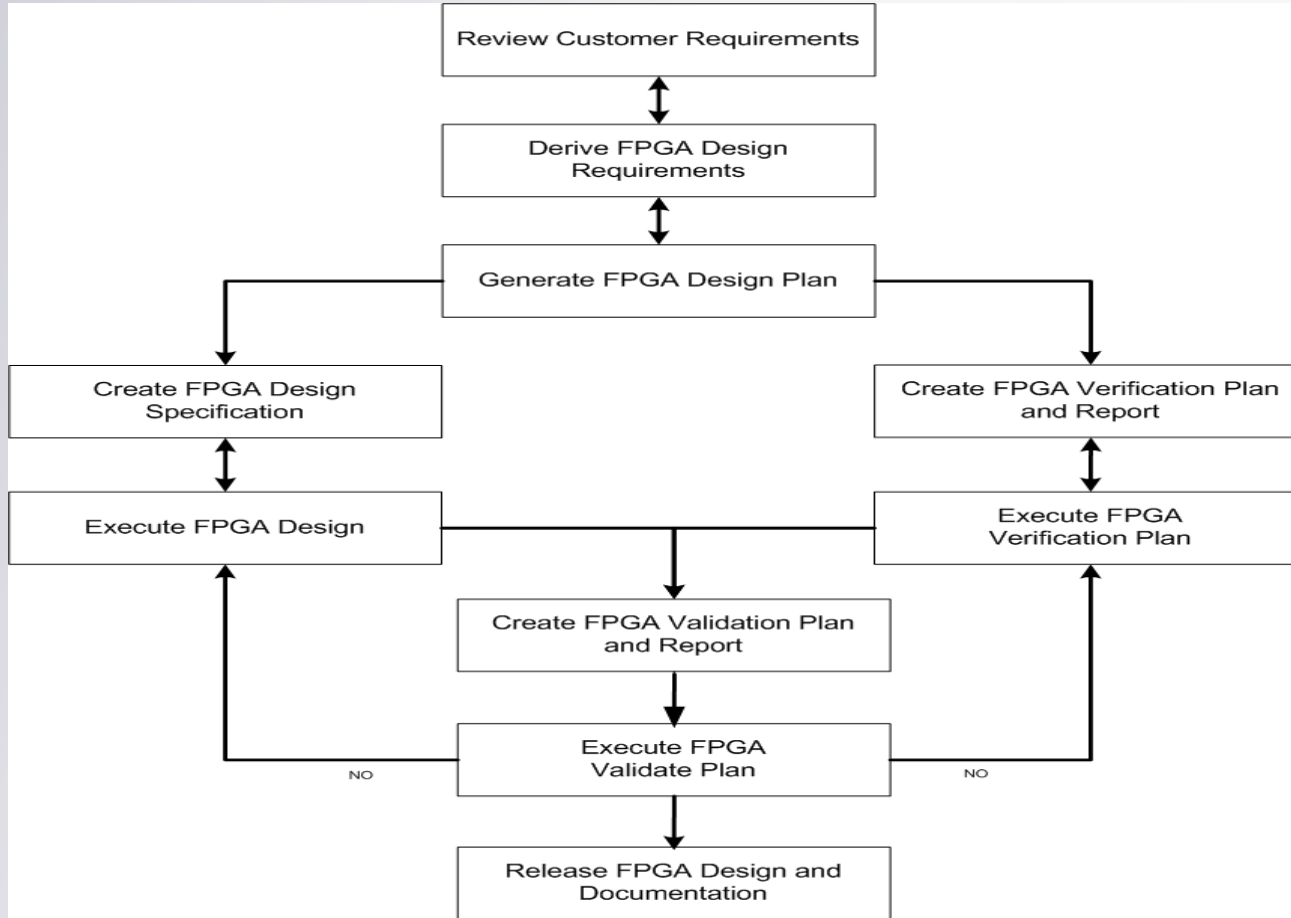
```
assign D = ~Q;
always @ (posedge clk)
begin
    Q <= D;
end
```



FPGA Design Methodology

- FPGA design process utilizes a number of supporting procedures and guidelines to facilitate in the development of safe and reliable quality driven complex logic devices.
- These include company as well as project specific procedures and guidelines:
 - FPGA Development Procedure
 - FPGA Design Plan
 - Development Checklist
 - L-3 FOS RTL Coding Guidelines
 - FPGA Directory Structure and Signal Naming Guidelines
 - Internal Design Review Procedure
 - Design Control Procedure

FPGA Design Process



FPGA Development Reviews

Action	Entrance Document	Exit Document	Responsible Engineer(s)	Review
Review Customer Requirements	Drafted Customer Requirements (SOW, ICD, EICD)	Baselined / Released Customer Requirements (SOW, ICD, EICD)	Lead Electrical Engineer	SRR
Derive FPGA Design Requirements	Baselined Customer Requirements	FPGA Design Requirements	Lead Electrical Engineer	IDR
Generate FPGA Design Plan (optional)	Baselined / Released Customer Requirements	FPGA Design Plan	Engineering Manager	FPGA Team Review
Create FPGA Design Description	FPGA Design Requirements	FPGA Design Description	FPGA Design Engineer	IDR
Create FPGA Verification Plan and Report	FPGA Design Requirements	FPGA Verification Plan and Report	FPGA Verification Engineer	IDR
Create FPGA Validation Plan and Report	FPGA Design Requirements FPGA Verification Plan and Report	FPGA Validation Plan and Report	Lead Electrical Engineer	IDR
Release FPGA Documentation Package	FPGA Validation Plan and Report	FPGA Design File	Lead Electrical	IDR

FPGA Project Tracking

Project: <####>	Charge Number: <####>	Project Type: Choose item		
Complex Logic Development Review Progress Tracking – Sign Off when completed				
Drawing Title:	Number and Revision:	Author:	Review Sign-off:	Status
FPGA Requirements	<####>			Completed
FPGA Design Description	<####>			Choose item.
FPGA Verification Plan/Report	<####>			Choose item.
FPGA Programmed Part	<####>			Choose item.
FPGA Label	<####>			Choose item.
FPGA Programming File	<####>			Choose item.
FPGA Design Files	<####>			Choose item.
FPGA Validation Plan	<####>			Choose item.
FPGA Validation Report	<####>			Choose item.

FPGA Requirements Review		Date:	Click here to enter a date.
Checklist Item	Response		Comments
Is the Customer Specification referenced?	YES		
Is this a single or multiple FPGA Design?	Choose item		
Has a maximum FPGA resource utilization been specified?	Choose item		
Is the target FPGA package/device/technology specified?	Choose item		
Is the target FPGA Grade (Commercial/Industrial/Military) or temperature range specified?	Choose item		
Is the design input method specified (schematic or HDL)?	Choose item		

FPGA Design Requirements

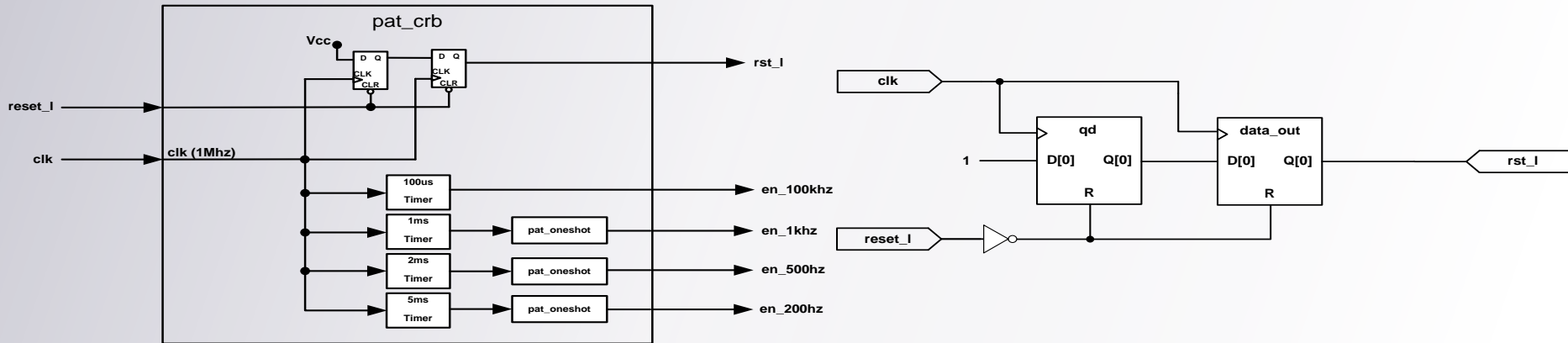
- Project begins with **Customer Requirements**
 - Externally supplied document for consumption by design team
 - Reviewed by design team; First release point once accepted
 - FPGA design requirements come directly from customer requirements
- **FPGA Design Requirements**
 - Gated by Customer Requirements
 - High level requirements document incorporating [REQ_TAG_ID]
 - “What” does the FPGA need to accomplish in the system
 - “What” are the constraints
 - “What” are the inputs and **how do** they behave
 - “What” are the outputs and **how should** they behave
 - Reviewed by design team; First release point once accepted
 - Risks understood and assessed for all TBD’s

FPGA Design Plan

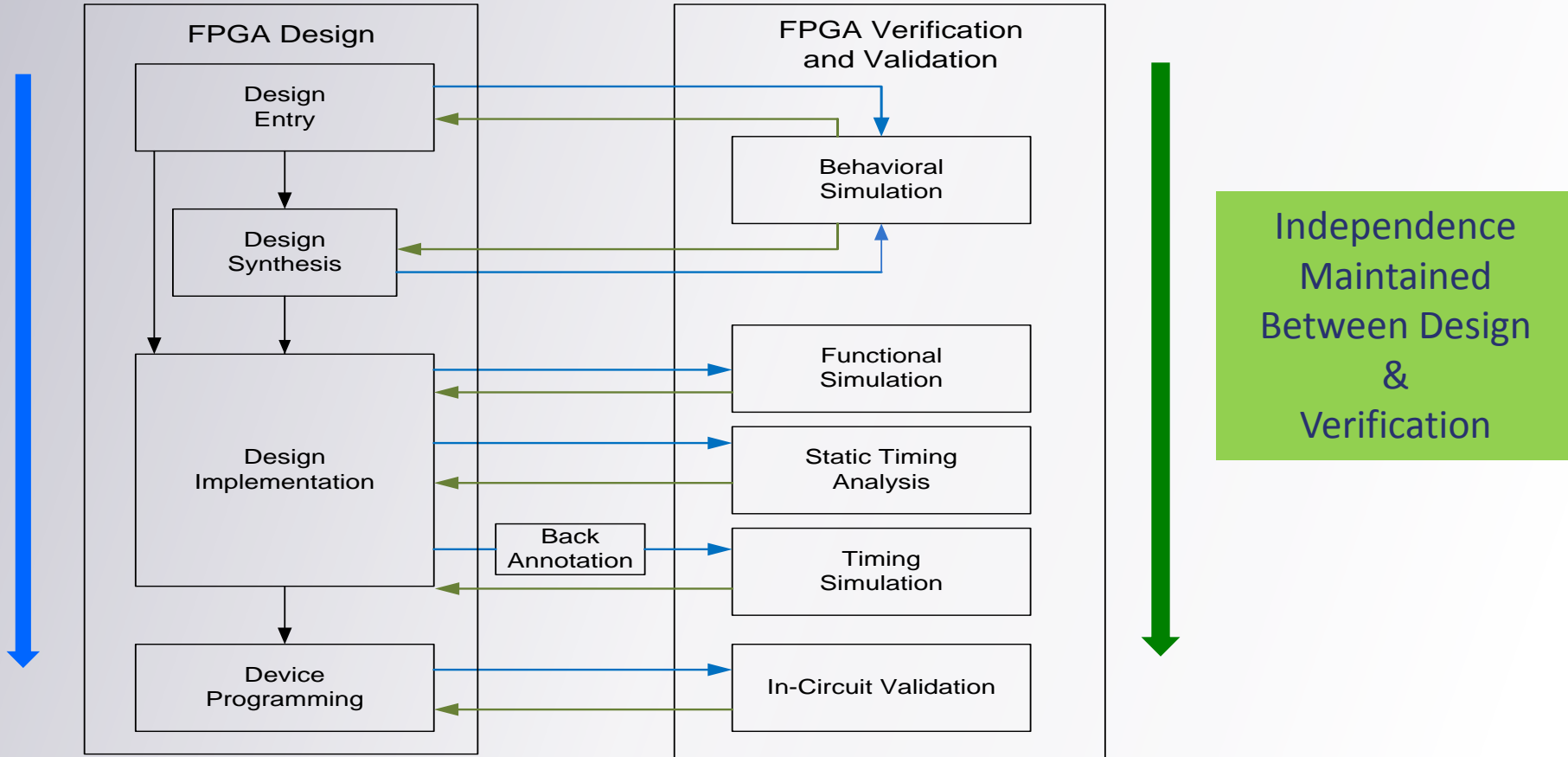
- Project management resource
- Defines people, methods and strategies to be implemented
 - Scope and Schedule
 - Team members and responsibilities
 - Entrance and exit criteria
 - Meeting and review procedures
 - Status and progress reporting practices
 - Issue / Bug reporting and tracking
 - Methods, tools and conventions utilized

FPGA Design Description

- Gated by FPGA Design Requirements
- Details “How” the design requirements will be met in hardware
- Describes device architecture
- Defines structural blocks functions and interfaces
- Reviewed by design team; First release point once accepted



FPGA Design Strategy



FPGA Design Execution

- Design execution and verification typically run in parallel
- Design Execution includes:
 - Logic partitioning
 - Design Entry
 - Design Synthesis
 - Block level verification
 - Design implementation
 - Device programming
- Design ensures hardware implements defined requirements
- Verification ensures hardware meets derived requirements
- Verification and Validation
 - Verification is 100% functional coverage done in EDA environment
 - Validation is a subset of Verification done in hardware
 - Not all verification points can be realized in hardware



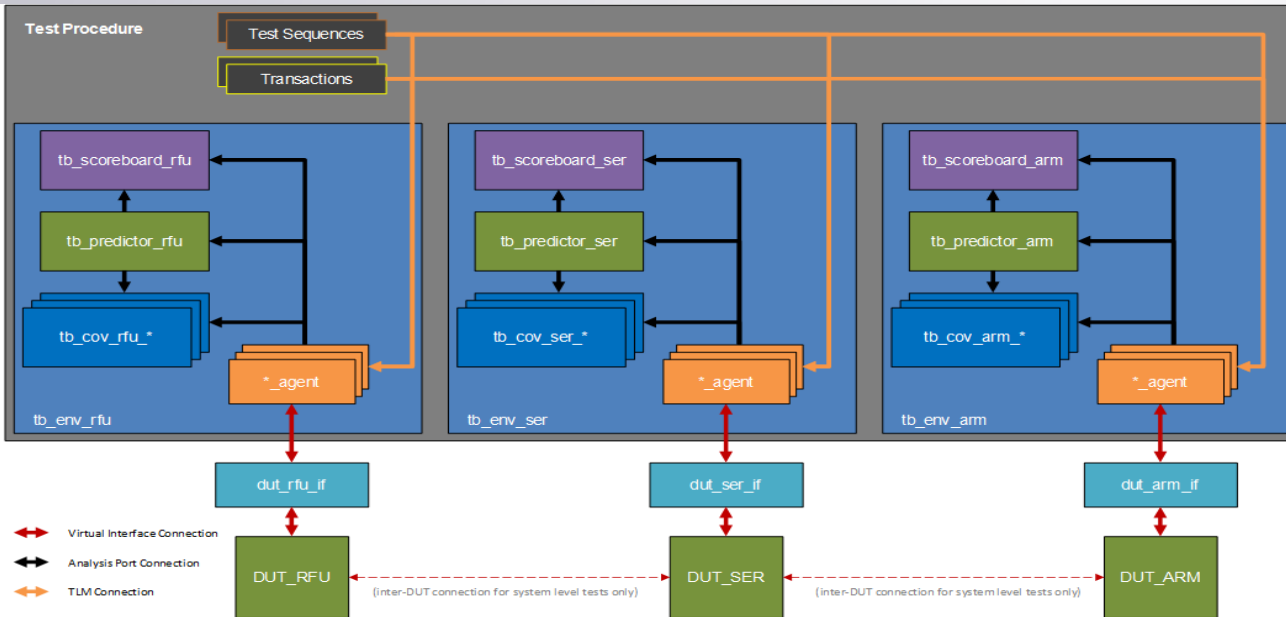
Verification Plan

- Design verification analyzes a design for proper performance and function as defined in the Architecture Specification.
 - Performed repeatedly during design development.
 - Ensures that the building blocks exhibit required behaviors and then that the entire design exhibits the required behaviors.
 - Done in an EDA environment not in hardware.
(when we hit hardware we have entered Validation).

FPGA Verification: Simulation Environment

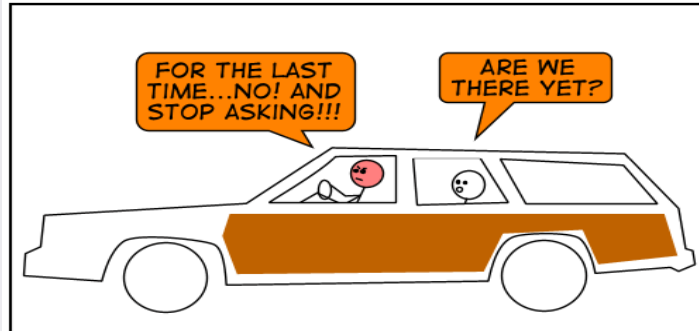


- Universal Verification Methodology (UVM) version 1.2
- Questa 10.4a simulator
- Separate simulation environments for each FPGA, and a combined simulation environment with all three FPGAs connected together
- Full code coverage and functional coverage collection with links to FPGA requirements document



FPGA Verification

- FPGA Verification Test Plan and Report document contains inspection and simulation test cases identify the requirements that are being verified, the stimulus that is being performed, and the expected behavior of the FPGAs
- Simulation test environment is also documented in the verification test plan and report
- Verilog designs are listed as part of the verification
- Functional Covergroups are used to ensure all simulation test case stimulus and conditions are simulated
- Verilog Code Coverage metrics are captured to ensure the design is tested
- Synthesis and Layout configuration and log files are inspected to verify the FPGAs meet timing, have the correct pin-out, device selection, etc.



Verilog Linting with Aldec's ALINT

- ALINT is a design rule checking software that is used to identify coding style, functional, and structural problems that are difficult to debug in simulators and in hardware
- FPGA designs are “linted” early in the development as part of the design verification
- Target Goal:100% Design Quality (Info, Warning, Critical Warning and Error Free).

Design Quality: 87%.

This report was generated based on the following settings:

- Base classification: Rules
- Include unused/disabled rules: No

Violated Rules:

Rule ID	Severity Level	Title
ALDEC_VLOG.1003	Warning	Avoid unconnected ports

...

Not Violated Rules:

Rule ID	Severity Level	Title
ALDEC_VLOG.1001	Warning	Constant conditional expressions should be avoided
ALDEC_VLOG.1004	Warning	Decimal constants should be used for repeat statement expressions
ALDEC_VLOG.1005	Warning	Do not assign input signals
ALDEC_VLOG.1007	Warning	At least one loop iteration is required

...

STARC_VLOG.1.1.1.1	Warning	File names should be as follows: ".v".
STARC_VLOG.1.1.1.10	Warning	Do not use the same instance name or cell name as the ASIC library being used
STARC_VLOG.1.1.1.2	Warning	Only alphanumeric characters and the underscore '_' should be used, and the first character should be a letter of the alphabet

...

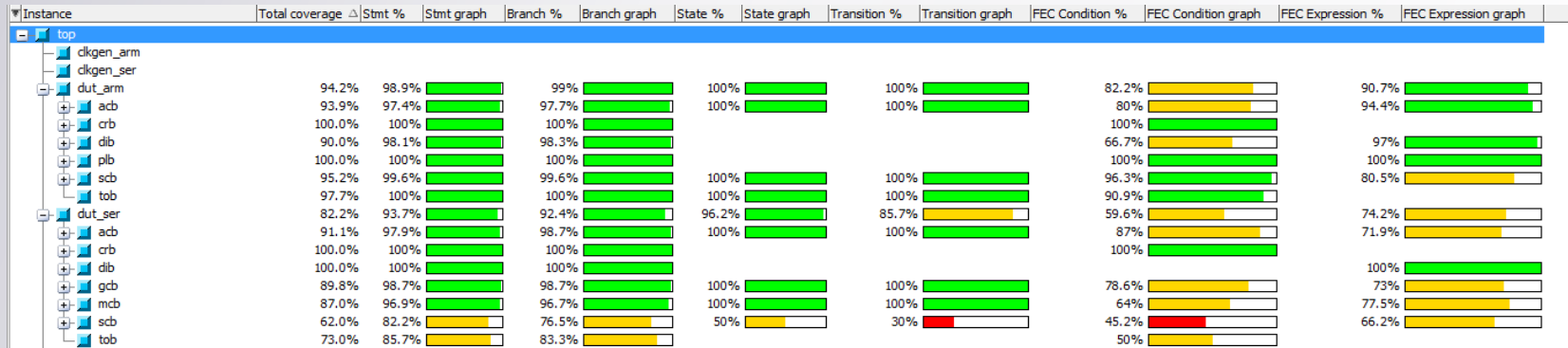


FPGA Code Coverage

Coverage Report Totals BY FILES: Number of Files 41

Enabled Coverage	Active	Hits	Misses	Weight	% Covered
-----	-----	-----	-----	-----	-----
Stmts	1726	1641	85	1	95.0
Branches	1540	1453	87	1	94.3
Conditions				1	84.1
UDP Condition Rows	0	0	0	1	100.0
FEC Condition Terms	500	341	159	1	68.2
Expressions				1	88.8
UDP Expression Rows	0	0	0	1	100.0
FEC Expression Terms	301	234	67	1	77.7
FSMs				1	94.2
States	116	113	3	1	97.4
Transitions	158	144	14	1	91.1

Total coverage (Code Coverage Only, filtered view): 85.9%




Issue Tracking and Revision Control

SharePoint Issue Opened

Reply Reply All Forward

Wed 04/06/16 9:10 AM

 FOSnet_Baymax@SP.KDI.L-3com.com

A task for Validity check of accel response "s" bits not working has been assigned to you

To Adams, Nick @ SSG - PE - FOS

A task was assigned to you by Desch, Noah @ SSG - PE - FOS on 4/6/2016.
Due by: 4/8/2016

Task: Resolution Task - 521 - Validity check of accel response "s" bits not working

Description: Please take a look at the bug report below:
<http://fosnet/depts/ESAD/FPGA/Bugs/Lists/Bugs/DisplayForm.aspx?ID=521>

After coming up with a resolution, please fill out additional info.

Please complete the task form located [here](#).

Click below to see the bug report:<http://fosnet/depts/ESAD/FPGA/Bugs/Lists/Bugs/DisplayForm.aspx?ID=521>

Issue is assigned a unique ID

Subversion Revision Control

Issue fixed, design updated and committed into subversion repository for verification

The screenshot displays a Subversion repository browser interface. At the top, a header bar shows the revision number '478', the author 'nadams', the date and time 'Wednesday, April 06, 2016 2:42:02 PM', and the bug ID '521'. The main content area is a table of revisions, with the following columns: Revision, Actions, Author, Date, Bug-ID, and Message. Revision 521 is highlighted, showing a message that reads: 'fix bug id 521 gate te accel good with not st accel good'. Below the table, a detailed view of revision 521 is shown, including the path 'fix bug id 521 gate te accel good with not st accel good' and the action 'Modified'. The interface also includes search filters, a 'Show 100' button, and a 'Refresh' button.

Revision	Actions	Author	Date	Bug-ID	Message
493		nadams	Friday, April 08, 2016 11:31:33 AM		added comments
489		nadams	Thursday, April 07, 2016 5:21:40 PM	517	fix bug id 517 needed to reset counter on entering safe
488		nadams	Thursday, April 07, 2016 5:03:55 PM	522	fix bug id 522 adt set too short for accepting based on 5ms en
490		nadams	Thursday, April 07, 2016 4:25:56 PM	523	fix bug id 523 again
481		nadams	Thursday, April 07, 2016 1:56:12 PM		fix delay req req done issue
484		nadams	Thursday, April 07, 2016 1:20:36 PM		add st_cmd arbitration
490		nadams	Wednesday, April 06, 2016 4:40:21 PM	525 526	fix bug id 526 525 gate sts on not in POST add parity error to state machine
478		nadams	Wednesday, April 06, 2016 2:42:02 PM	521	fix bug id 521 gate te accel good with not st accel good
471		nadams	Wednesday, April 06, 2016 8:46:36 AM	520	fix bug id 520 removed the en_200Hz from the counter clear so was clearing each count which is wrong
468		nadams	Tuesday, April 05, 2016 3:30:01 PM		fix not safing on las_en at adt timeout was waiting for sad timeout
463		nadams	Tuesday, April 05, 2016 1:21:03 PM	519	fix bug id 519 delay entering safe state to enable the arm_conv_tm to be active for the 2ms minimum
461		nadams	Monday, April 04, 2016 4:46:00 PM	518	fix bug id 518 add las_lim_arp to launch2safe condition
459		nadams	Monday, April 04, 2016 3:01:47 PM	516	fix bug id 516 needed to sequence falling edge of valid sep after receipt of charge cmd
456		nadams	Monday, April 04, 2016 8:50:39 AM	513	fix bug id 513 clearing the destruct timer exp when we safe
454		nadams	Friday, April 01, 2016 12:09:22 PM	514	fix bug id 514
450		nadams	Friday, April 01, 2016 8:55:25 AM	511 512	fix bug id 511 512 had check for valid instead of invalid
446		nadams	Thursday, March 31, 2016 1:37:47 PM	509	fix bug id 509 add destruct valid into 50ms counter logic for safing when armed
444		nadams	Wednesday, March 30, 2016 4:03:26 PM	508	fix bug id 508 parity errors from accel
442		nadams	Wednesday, March 30, 2016 1:35:18 PM		remove condition in qpl_fem
441		nadams	Wednesday, March 30, 2016 11:28:06 AM		add parity error to req checks during init
440		nadams	Wednesday, March 30, 2016 10:05:50 AM	306	fix bug id 506 was validating the deflection prior to offset adjustment
439		nadams	Tuesday, March 29, 2016 4:52:28 PM	505	fix bug id 505 add accel response error to confic and config_x errors
437		nadams	Tuesday, March 29, 2016 3:44:15 PM	504	fix bug id 504 failsafe out not working when not concurrent destruct and charge cmd
432		nadams	Thursday, March 24, 2016 4:30:29 PM		fix the accel response error
431		nadams	Thursday, March 24, 2016 3:20:31 PM	501	fix bug id 501 overwriting accel data with test setup command responses
425		nadams	Thursday, March 24, 2016 8:20:13 AM	500	fix bug id 500 cap mon out returning low if use safe after pre_launch
423		nadams	Wednesday, March 23, 2016 4:41:06 PM	499	fix bug id 499 cap mon was following destruct
422		nadams	Wednesday, March 23, 2016 4:31:04 PM	498	fix bug id 498 addsd window req 52
420		nadams	Wednesday, March 23, 2016 3:47:09 PM		fix arm_conv_tm
416		nadams	Wednesday, March 23, 2016 11:11:33 PM	407	fix bug id 407 arm_conv_tm 2ms dwell from launched to arm_enabled state
410		nadams	Tuesday, March 22, 2016 8:51:55 AM		PAS_CS_BE78 SX32A CS_7D87
408		nadams	Monday, March 21, 2016 2:48:17 PM		added 1 second delay for failsafe_just after safing when destruct is high and charge command is lan...
405		nadams	Friday, March 18, 2016 11:47:01 AM		fix arm_conv_tm
404		nadams	Friday, March 18, 2016 10:18:59 AM		fix accel_cfl_fem transition to safe

Issue ID's and Design Revision are linked together in Subversion



SharePoint Issue Tracking

Issue resolved, committed in subversion project repository and reassigned to be verified and closed

fixed in revision 478 gate te accel good with not st accel good

CONFIGURATION
FOSNet > ESAD Engineering > FPGA > Bug Database > Bugs > Validity check of accel response "s" bits not working

ACTIONS
Log a Bug
Add a Project

BY ASSIGNMENT
Assigned to Me
Opened by Me
Unassigned Bugs
All Bugs

Title	Validity check of accel response "s" bits not working
Project	D3749 PATRIOT
Category	Design
Status	Active
Comments	Per FPGA_REQ-95, 96, and 97, the FPGA should not accept accel responses with wrong s bits. do run_test esad_normal_test iter=1 seed=2433867173 FPGA is in test-mode-enabled, and accel is in self-test-enabled, so proper s-bits are 2'b10. Accel sample at 10062ms has wrong S-bits (2'b00), but integrator runs. No
Repro Steps	
Safety Critical	
Severity	(2) Normal
Priority	(2) Normal
Assigned To	Desch, Noah @ SSG - PE - FOS
Resolution	Fixed
Resolved By	Adams, Nick @ SSG - PE - FOS
Resolution Due Date	04/08/2016, 04:00 AM
Resolution Comments	fixed in revision 478 gate te accel good with not st accel good
Duplicate Bug	
Related Bugs	
Verified By	
Verification Due Date	2016-04-13T04:00:00Z
Verification Comments	

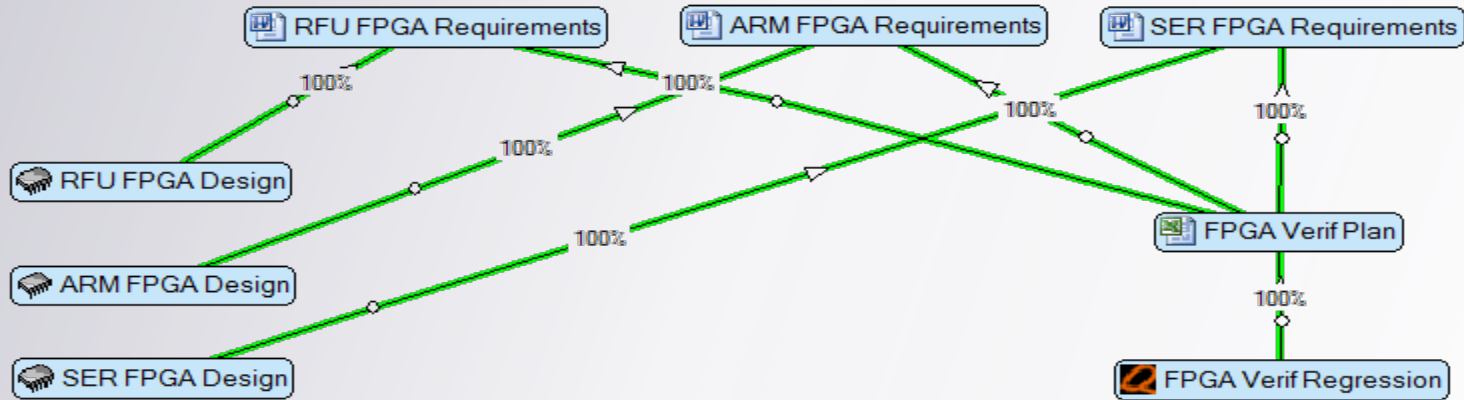
Version: 4.0

Design Revision is captured in Issue Resolution Comment

Requirement Traceability

ReqTracer from Mentor Graphics

Requirements are tagged



RTL is tagged

Verification tests are tagged

Tags in Requirements, Verification and Design are all linked together

FPGA Documentation

FPGA Package Includes:

Description
FPGA Programmed Part
FPGA Label
FPGA Fuse File
FPGA Electronic Design & Verification File
FPGA Design Requirements
FPGA Design Description
FPGA Verification Test Plan & Report
FPGA Validation Plan & Report

Conclusion

The end result is a complex logic design process that approaches or is equivalent to a CMMI (Capability Maturity Model Integration) Maturity level 4, Quantitatively Managed, for product development.

CMMI is a process appraisal program and service required by many DoD and U.S. Government contracts applying to firmware and hardware development for complex logic devices.



FOS

Mission Statement

Our mission is to support the warfighter by providing highly reliable fuzes, safety and arming devices, proximity sensors and related products. We will continue to innovate and develop unique solutions by leveraging our valued workforce.



Contact Information

L-3 Fuzing & Ordnance Systems
Business Development Department
3975 McMann Road
Cincinnati, Ohio 45245
513-943-2000

