

# Making Formal Property Verification Mainstream: An Intel® Graphics Experience

M Achutha KiranKumar V

Erik Seligman Aarti Gupta Bindumadhava S S Abhijith A Bharadwaj









- Formal needs no introduction:
  - Established 100% state space coverage
  - NO/Minimal Bug Escapes
  - High Verification Confidence
  - Testbench Free Verification
  - Various dedicated Apps for Specific problems:
    - RTL Equivalence Sequential Check, Connectivity, X verification, Security, Low power, etc

Still...

Why are the FV adoption rates are still low? Why was the dedicated FV deployment drive in VPG needed?



- Common FV Roadblocks:
  - FEAR:
    - MYTH: FV is only meant for PhDs
    - MYTH: Will drain my time/resources



Is it supposed to have ZERO gradient?

- ROI:
  - TRUTH: What eventually matters?:
     ROI
  - How to compare with DV results?



- No Progress Metrics
  - No intermediate check-points







## DESIGN AND VERIFICATION VPG FV Deployment Drive CONFERENCE AND EXHIBITION UNITED STATES



Preparation phase 1. Survey 2. GUI Prep 3. Cookie cutter lib



Training phase 1. Champion id 2. Trainings 3. Dojos



Test planning

1. Formal TP 2. Continuous progress plan



Execution & Closure 1. Execution &Tracking 2. Bug hunting/clearance 3. Coverage metrics/closure



Communicate results 1. ROI roll up 2. Integrate w/DV

Check-in &Regress

1. Proof Repository 2. Regular Regressions



6





Preparation phase 1. Survey 2. GUI Prep 3. Cookie cutter lib



Training phase 1. Champion id 2. Trainings Dojos



Test planning

1. Formal TP 2. Continuous progress plan



Execution & Closure 1. Execution & Tracking 2. Bug hunting/clearance 3. Coverage metrics/closure



1. ROI roll up 2. Integrate w/DV



Check-in & Regress

1. Proof Repository 2. Regular Regressions





- Survey
  - Immediate needs for most of the users:
    - Automated setup/property\_coding
    - Assertion Libraries for common design components
    - Dedicated Trainings
    - Local Contact Points



# 2017 VPG FV Deployment Drive:

## **FED STATES Preparation Phase**

# GUI creation

- Provided a quick setup interface to new users
- One stop shop for all formal flows
- Easy way of generating all collaterals for formal

	WELCOME TO FPV GUI				
Basic Setup	Automated Properties	Arbiter Propert	ties Manual Propert	ies Help	
Top file path for RTL Enter Top Module Name					Browse
Fast Clks (separated by a space) Slow Clks (separated by a space) Resets (separated by a space) BlackBox_Modules (names separated by a space) BlackBox_Instances (names separated by space)					Factor 2
Add Non_Re	esettable_regs to resets				
	Value	No	<u>±</u>		
Con	figuration Switch				
	Create Setup files				
Create FPV Golden TCL File		•			Gfx FV COE Bringing Formal close to you
	Check FSM props				
	Quit				

## VPG FV Deployment Drive: VPG FV Deployment Drive: UNITED STATES

- Cookie-cutter Property
   Templates
  - Automated properties

		WELCOME TO FPV GUI	(1) (1) (1) (1) (1) (1) (1) (1) (1) (1)		
asic Setup Automat	ed Propert	Arbiter Properties Manual Properties Help			
		Add output toggle covers			
		Add Mutex properties			
		FSM Arithmetic Overflow Signals toggling			
		Add the selected properties			
		Add the assert top module			
		Create assertion file for submodules			
		Assertion type	1		
		Add this SVA_LIB property			
		Add Standard Internace Property Set to the tot me			
		4			
Create	Setup files				
Create Create FPV	Setup files Golden TC	L File	Gfx FV COE		
Create Create FPV	Setup files Golden TC FSM props		Gfx FV COE Bringing Formal close to yo		

# VCON VPG FV Deployment Drive:

**TED STATES Preparation Phase** 

- Cookie-cutter Property
   Templates
  - Automated properties
  - Standard Interfaces

		WELCOME TO FPV GUI			
Basic Setup	Automated Properties	Arbiter Prope	erties Manual Proper	ties Help	
			Add output toggle cov	ers	]
			Add Mutex properties		
	FS Ar Się	M ithmetic Overfl gnals toggling	ow		
		Add	the selected prope	erties	
			Add the assert top mo	dule	1
		Create	assertion file for subm	nodules	
	As	sertion type	10100	<u>±</u>	(
			Line Ora_Line prope		
		Add Standa	rd Interface Property S	Set to the tcl file	
gt_convert gt_convert gt_convert gt_credit_c gt_credit_c gt_credit_c gt_credit_ gt_credit_ gt_credit_ gt_credit_ gt_credit_ gt_credit_ gt_credit_ gt_credit_ gt_credit_ gt_credit_ gt_credit_ gt_credit_ gt_credit_ gt_credit_ gt_credit_	inf_1x_2x inf_2x_1x inf_2x_1x jounter_sounter_repeater younter_sender younter_sender_1x2x younter_target younter_target_2x1x idle_inf_dvhold_sender inf_bcast_p2p_sender inf_bcast_sender inf_dvhold_sender inf_dvhold_sender inf_put_sender inf_sender inf_sender inf_sender inf_sinf is_counter_sender is_inf is_inf_bt_sender	gt_inf_ gt_inf gt_inf gt_inf gt_inf gt_put_ gt_put_ gt_put_ gt_vc_i gt_vc_i gt_vc_i gt_vc_i gt_vc_i gt_vc_i	creditrun_repeate creditrun_target sender_core target_core inf_sender inf_target ater_vc_inf er_vc_inf er_vc_inf er_vc_inf er_vc_inf sender nf_arb2 nf_arb3 nf_arb3 nf_arb4 nf_sender nf_target_demux2 nf_target_demux4 nf_target_demux4 nf_tracker		
	Check FSM props				Gfx FV COE Bringing Formal close to yo
	Quit				

# VPG FV Deployment Drive:

**TED STATES Preparation Phase** 

- Cookie-cutter Property
   Templates
  - Automated properties
  - Standard Interfaces
  - Arbiter Properties

F	🖓 💳 WELCOME TO FPV GUI 😽 🕄 🔺								
Basi	sic Setup Kutomated Propertix Arbiter Properties I lanual Properties Help								
Assumptions     General Arbiter Properties     Round Robin Properties     Priority Robin Arbiter Properties     Covers									
	Genera	Arbiter Properties	ASSERT_ONE_RE	Q_GNT	•	1	Ada	d this Arbiter property	
When	ASSERT_ONE_RE ASSERT_ONE_RED_GNT (name, num, ASSERT_ONE_RED_GNT (name, num, ASSERT_ONE_CAN ASSERT_ONE_CAN ASSERT_ONE_GN ASSERT_GNT_PR ASSERT_GNT_ON ASSERT_CONLY_CAN ASSERT_ON		ASSERT_ONE_RE ASSERT_ONE_GN ASSERT_ONE_GN ASSERT_ONE_GN ASSERT_GNT_PA ASSERT_LIVENES ASSERT_ONLY_CO ASSERT_GNT_ON	EQ_GNT ILY_ON_REQ IT_PER_REQ IT_AT_A_TIME EESERVED_TIL SS DNE_GNT_PER I_ALL_PORTS	L_DATA_ _CLIENT_	inal tc l fi	le		
			RBITER PRO	PERTIES		<b>(</b> =	2년 7		
			the property						۲
		Enter the Reque	st Signal					x FV COE	
		Enter the gran	tsignal					Formal close to y	ou
느		Enter the clock	(signal						
		Enter the Rese	tsignal	. —				_	
	Er	ter the Error message	e to be displayed	•					
		Enter	text						
		clear	Text						
		Exit this Arbiter_p	roperties windo	N					

# DESIGN AND VERIFICATION VPG FV Deployment Drive



Preparation phase 1. Survey 2. GUI Prep 3. Cookie cutter lib





Test planning

1. Formal TP 2. Continuous progress plan



Execution & Closure 1. Execution & Tracking 2. Bug hunting/clearance 3. Coverage metrics/closure



Communicate results 1. ROI roll up 2. Integrate w/DV



Check-in &Regress

1. Proof Repository 2. Regular Regressions



### VERIFICATION VERIFICATION VPG FV Deployment Drive: CONFERENCE AND EXHIBITION UNITED STATES

- Identifying Local Experts
  - FV champion from each cluster
  - Immediate support
- Setting-up Self Help Materials
  - Dojos
  - FAQs
- Training Dungeons
  - 2-week rigorous training











- Clear Goal Definitions
  - Assigning FV problem to correct category





#### CONFERENCE AND EXHIBITION CONFERENCE AND EXHIBI

# Progress Continuum

- Defining measurable subgoals
- Allows progress to be graded w.r.t. time
- Decided % completions to be achieved for each Val Milestone

	Over Constraint	Checker set 1	Checker set2	Checker set3	Checker set 4	Checker set 5
	Phase 1	1 week				
	Phase 2	1.5 week	2 weeks			
)	Phase 3	2 weeks	2.5 weeks	3 weeks		
	Valid Constraints	3 weeks	3.5 weeks	4 weeks	5 weeks	6 weeks

#### CONFERENCE AND EXHIBITION CONFERENCE AND EXHIBI

# Activity Ownership

Designer needs to monitor
 FV completion for his unit



# DESIGN AND VERIFICATION VPG FV Deployment Drive



Preparation phase 1. Survey 2. GUI Prep 3. Cookie cutter lib



Training phase 1. Champion id 2. Trainings 3. Dojos



Test planning

1. Formal TP 2. Continuous progress plan



3. Coverage metrics/closure



1. ROI roll up

2. Integrate w/DV



Check-in &Regress

1. Proof Repository 2. Regular Regressions



# SIGN AND VERIFICATION VPG FV Deployment Drive:

## **EXECUTION & Closure**

- Execution and Tracking
  - Round-the-clock support extended to FV owners after dungeons
  - Weekly FV progress tracked and published
- Deep Bug Exploration
  - Abstractions, different engine settings
  - Lots of Covers added
- Coverage and Closure
  - Line and branch coverage obtained
  - Properties added for holes



# DESIGN AND VERIFICATION VPG FV Deployment Drive



Preparation phase 1. Survey 2. GUI Prep 3. Cookie cutter lib



Training phase 1. Champion id 2. Trainings 3. Dojos



Test planning

 Formal TP
 Continuous progress plan



Execution & Closure 1. Execution &Tracking 2. Bug hunting/clearance 3. Coverage metrics/closure





Check-in &Regress

1. Proof Repository 2. Regular Regressions

21

#### VPG FV Deployment Drive: CONFERENCE AND EXHIBITION CONFERENCE AND EXHIBITION CONFERENCE AND EXHIBITION COMMUNICATE Results

- ROI calculation standard was established
- ROI numbers calculated for activity and compared against DV

ROI Parameter	Formal	Dynamic	Relative
Bugs% found by FV	NB_F/(NB_F+NB_D)	NB_D/(NB_F+NB_D)	NB_F/NB_D
Total coverage achieved	Covtot_F	Covtot_D	
Bug - engineering cost ROI	BEC_F = NB_F/CE_F	BEC_D = NB_D/CE_D	BEC_F/BEC_D
Bug - machine cost ROI	BMC_F = NB_F/Cm_F	BMC_D = NB_D/Cm_D	BMC_F/BMC_D
Coverage - engineering cost ROI	CEC_F =	CEC_D =	CEC_F/CEC_D
	Covtot_F/CE_F	Covtot_D/CE_D	
Coverage - machine cost ROI	CmC_F =	CmC_D =	CmC_F/CmC_D
	Covtot_F/Cm_F	Covtot_D/Cm_D	

Parameter	Depiction		
Number of engineers worked on	NE_F		
the problem			
Average bandwidth/engineer	Bw_f		
Number of weeks spent on effort	Ww_F		
Total time spent in hours	Ts_F = Bw_F <sup>a</sup> Ww_Fa <sup>a</sup> 40 <sup>b</sup>		
Project schedule in weeks	Pw_F		
Scaled engineering costs	CE_F = Ts_F/(Pw_F <sup>a</sup> 40)		
Machines used for computations	Mc_F		
Runs/week	Rw_F		
Number of days of run	Dw_F		
Total machine-days/week	Tm_F = Mc_F <sup>a</sup> Rw_F <sup>a</sup> Dw_F		
Project machine-days/week	Pm_F		
Scaled machine costs	Cm_F = Tm_F/Pm_F		
Coverage (% cover points hit)	Covtot_F		
Bugs found by the method	NB_F		

<sup>a</sup>Assuming eight hours/day for a five-day work week = 40 hours/week. <sup>b</sup>Machine runs are not limited to a working day but for the whole week

# DESIGN AND VERIFICATION VPG FV Deployment Drive



Preparation phase 1. Survey 2. GUI Prep 3. Cookie cutter lib



Training phase 1. Champion id 2. Trainings 3. Dojos



Test planning

1. Formal TP 2. Continuous progress plan



Execution & Closure 1. Execution & Tracking 2. Bug hunting/clearance 3. Coverage metrics/closure



Communicate results 1. ROI roll up 2. Integrate w/DV



1. Proof Repository 2. Regular Regressions

23



- FV cannot be done once & forgotten
  - RTL in constant churn
  - Regressions to be enabled to find bugs due to changes







## DVCONFERENCE AND EXHIBITION CONFERENCE AND EXHIBITION UNITED STATES

## • Quality of reception:

- 80+ engineers became FV experts
- 15 bugs caught within 2 weeks of activity

FV is awesome; I write code, run FV before even committing it to my local repo, and get it clean.
If it wasn't for this tool, I'd be building micro-testbenches before the ULT is ready.

It uncovered many scenarios in the beginning for which the Fulsim tests came very late. it has given very high confidence that new RTL has achieved the original health.

FV is the prime mode of validation for one of my unit. While DV gave me a bill of 6-8 weeks, we verified all the DCN changes in less than 2 weeks using FV

Found couple of dozen bugs even before DV was up and helped clean the RTL On a FV Quality bug hit: Hitting this bug in functional random env would have taken a long time and the bug is fatal enough to cripple the feature.

### CONFERENCE AND EXHIBITION UNITED STATES Results of VPG FV Deployment

# • Quality of Bugs:

- 85+ bugs caught in 12 weeks (activity was started after half way through DV drive)
- Most of the bugs caught fell in corner-case category



### CONFERENCE AND EXHIBITION UNITED STATES Results of VPG FV Deployment

- ROI:
  - Weighted ROI of 220x compared to DV
  - Some activities gave as much as 138x w.r.t. engg costs; 3692x w.r.t. machine costs





# FORMAL VERIFICATION IS NOW FORMALLY IN THE PROJECT !!





# **Q & A**



Achutha KiranKumar V M, Intel Corp.