

Simulation Environment Based on the Universal Verification Methodology (UVM)





uASIC



Adrian Fiergolski¹⁾

on behalf of the CLIC detector and physics (CLICdp) collaboration ¹⁾The European Organization for Nuclear Research

CLICpix2

Geneva, Switzerland

The Universal Verification Methodology (UVM) is a standardized approach of verifying integrated circuit designs, targeting a Coverage-Driven Verification (CDV). It combines automatic test generation, self-checking testbenches, and coverage metrics to indicate progress in the design verification. The flow of the CDV differs from the traditional directed-testing approach. With the CDV, a testbench developer, by setting the verification goals, starts with a structured plan. Those goals are targeted further by a developed testbench, which generates valid stimuli and sends them to a device under test (DUT). The progress is measured by coverage monitors added to the simulation environment. This way, the non-exercised functionality can be identified. Moreover, the additional scoreboards indicate undesired DUT behaviour.

Such verification environments were developed for three recent ASIC and FPGA projects which have successfully implemented the new work-flow:

- the CLICpix2 65 nm CMOS hybrid pixel readout ASIC design;
- the C3PD 180 nm HV-CMOS active sensor ASIC design;
- the FPGA-based DAQ (µASIC) system of the CLICpix chip.

Different interfaces (Ethernet, trigger and timing interface, I2C, SPI) which stimulate the DUTs are handled by complex and versatile testbenches enabling

an exhaustive system verification and identification of difficult-to-track design flaws.



Coverage-Driven simulation (CDV) random generation of test vectors constrained by a user

Step 1: Verification plan

Section	Title	Description	Link	Туре	Weight	Goal
2	SPI interface				1	100
2.1	Single or back to back	This selects whether the transaction is single transfer or a back to back tr	▶spi_cvg,*:single_or_continuous	CoverPoint	0	100
2.2	SPI_read_corresponds_to_register_model	Read data corresponds the behavioral model.	SPI_read_corresponds_to_register_model	Assertion	1	100
			SPI_access_cg:second_readout	Coverpoint		
2.3	write_read_all_addresses	Write and read all valid addresses.	SPI_read_corresponds_to_register_model	Assertion	1	100
			SPI_access_cg:first_readout	Coverpoint		
2.4	check_default_values	Check default values of the registers after reset.	SPI_read_corresponds_to_register_model	Assertion	1	100
			SPI_matrix_configuration_model_vs_registers_mask			
			SPI_matrix_configuration_model_vs_registers_th_adj			
			SPI_matrix_configuration_model_vs_registers_count_mode			
2 5	SPI matrix configuration model vs. registers	Checks whether values of the pixel matrix model correspond with the DTL	SPI_matrix_configuration_model_vs_registers_IP_en	Accortion	1	100
2.5	SPI_mainx_connguration_model_vs_registers	Deed back metric configuration	Deedeut_egyreedeut_kind	Assertion	1	100
2.0	readback_matrix_conliguration	Read back mains configuration.	Readout_cg.readout_kind	Coverpoint	1	100
27	different matrix configuration sconario	Check different matrix configurations	MatrixConfiguration_cg:matrix_configuration_scenario	Coverpoint	1	100
2.1				CoverPoint	1	100
2.0	Depet signal	Access duminy addresses	SPI_access_cg.uummy_auuress	Coverpoint	1	100
3	Reset signal				0	100
4	Readout Interface				1	100
4.1	UNDEFINED_X_STATE_ON_THE_LINK	No X or Z state on the bus.	UNDEFINED_X_STATE_ON_THE_LINK	Assertion	1	100
4.2	Number_of_columns	Readout with different number of parallel columns being read	Readout_cg:parallel_collumns	Coverpoint	1	100
			PACKET_HEADER_SHOULD_BE_A_REGULAR_DATA_WORD			
			RCR_IN_PACKET_HEADER_DIFFERRS_FROM_REGISTER			
12	booder and corrier extend check	Check beader and carrier extend		Accortion	1	100
4.5	neauer_anu_carner_extenu_cneck	Check header and carrier extend	FOUND_UNEXPECTED_CONTROL_WORD_IN_PACKET	Assertion	1	100

- extract verification points from the chip specification
- choose the most appropriate implementation of the coverage model for each point of the plan: : checking permutations of condition and state when a known Covergroup Modelling result is achieved (System Verilog Covergroups) **Cover Property Modelling:** checking that a set of state transitions has been observed

(System Verilog Assertions)

Tip: Questa provides add-on for MS Excel and Word and supports Open Office

Step 2: Verification platform

• build a generic environment eventually customized by the specific tests

- high level of abstraction (Transaction Level Modelling)
- EDA vendors provide libraries to verify popular interfaces (Ethernet, SPI, I2C, etc.)

System Verilog extremely scalable

- extension of Verilog
- Verification (and RTL)
- C++ like syntax
 - easy data modelling
 - object oriented
- interfaces
- assertion language (SVA)
- randomization and coverage support
- Direct Programming Interface (DPI)

-🗾 uvm root 🖃 🗾 uvm test top env 1 sar 🗏 🚮 spi env 🕒 🇾 sqr 🔄 🇾 slave_agent 🗾 scoreboard 🗄 🗾 monitor 🖃 🗾 master_agent -🗾 spi coverage 🗄 🧾 sequencer 🛃 monitor - 🗾 listener -🗾 driver serial_agent 🗾 monitor scoreboard 🗄 🗾 SPI_scoreboard

庄 🗾 Readout_scoreboard 🔄 🎵 MatrixConfiguration scoreboar

charge_agent



+- 🗾 sqr 🗾 monitor

对 driver



- use EDA libraries for standard interfaces (*Tip: Questa Verification IP*)
- create a library and reuse your agents for your custom interfaces
- take advantage of UVM configuration, logging and factory classes
- follow UVM patterns (environment, item execution, etc.)
- do not use hierarchical paths to DUT inside the verification environment
 - create dedicated interfaces
 - use assertions in a top module or interfaces
 - use System Verilog directives (`define)
- define basic transactions
 - recording makes a waveform more readable
- make extensive use of randomization (use constraints)
- integrate the simulation with software over **DPI**
 - facilitates debugging (eg. Ethernet packets view in Wireshark)
 - allows debugging of the DUT interfaced with its slow control/DAQ software
- develop tests
 - generic test
 - Inherit from the above specialized tests with dedicated sequences, eg. spiTest, chargeInjectionTest...

Tip: Look for help on https://verificationacademy.com/

Tip: Check https://gitlab.cern.ch/CLICdp/HDLVerificationLibrary

Step 3: Actual verification

• use issue ticketing system to share with other developers spotted bugs and ambiguity of the DUT specification (Tip: Jira)

 managemen 	t	tools	
-------------------------------	---	-------	--

- reduce maintenance
- improve maintenance
- instead of one long, launch few shorter simulations in **parallel** change randomization seed

Tip: Check Questa Verification Management

- control progress of the verification process
 - coverage metrics
 - corner cases can be reached by specialized sequences
- swap the RTL description of the DUT with a back annotated netlist and rerun the tests
 - more realistic DUT description including internal delays

•	No Results - Delault Configuration 1 (New) - The Sep 15	15.07.40 CEST 2010	- Delault						
Ru	unnable/Action (NoFilter)	Status	Total Coverage Elapse	d Time	Host Name	CPU Time	Seed	Test Status Reason	Test Status Time
4	<pre>// CLICpix2_test/compile/compile_init/execScri</pre>	Passed		18	lnxmicv21.cern.ch				
4	<pre>F CLICpix2_test/compile/optimize/execScript</pre>	Passed		53	lnxmicv21.cern.ch				
4	CLICpix2_test/tests/spiTests-1/spiTest/exec	Passed (Warning)	58.23	7890	lnxmicv21.cern.ch	7708.53	1	UVM_WARNING	490715625000.00
4	CLICpix2_test/tests/spiTests-1/spiTest/tpla	Passed	0.00	1	lnxmicv21.cern.ch				
4	CLICpix2_test/tests/spiTests-1/spiTest/merg	Passed	58.23	12	lnxmicv21.cern.ch				
4	CLICpix2_test/tests/spiTests-2/spiTest/exec	Running			lnxmicv21.cern.ch				
4	CLICpix2_test/tests/spiTests-3/spiTest/exec	Passed (Warning)	59.88	9201	lnxmicv21.cern.ch	9022.09	3	UVM_WARNING	36500000.00
4	CLICpix2_test/tests/spiTests-3/spiTest/merg	Passed	60.09	12	lnxmicv21.cern.ch				
4	CLICpix2_test/tests/chargeInjectionTests-1/	Failed (Error)	54.36	10774	lnxmicv21.cern.ch	10646.7.	. 1	Assertion error.	991137500000.00
4	CLICpix2_test/tests/chargeInjectionTests-2/	Passed (Warning)	55.15	9407	lnxmicv21.cern.ch	9318.64	2	UVM_WARNING	25311875000.00
4	CLICpix2_test/tests/chargeInjectionTests-2/	Passed	63.67	12	lnxmicv21.cern.ch				
4	CLICpix2_test/tests/chargeInjectionTests-3/	Failed (Error)	53.35	9403	lnxmicv21.cern.ch	9313.50	3	Assertion error.	540193125000.00

Pending: 0 Queued: 0 Running: 1 (8.3%) Suspended: 0 Passed: 9 (75.0%) Failed: 2 (16.7%) Timeout: 0 Killed: 0 Skipped: 0 Dropped: 0 Coverage: 63.67 Testplan: 90.13

Sec#	Testplan Section / Coverage Link	Type Coverage		Goal	% of Goal	Status
0	🖃 🔆 testplan	Testplan	90.12%	-	90.12%	
1	吏 🔆 CLICpix specs for testbench design	Testplan	100%	100%	100%	
2	🚍 🔆 SPI interface	Testplan	87.25%	100%	87.25%	
2.1	🗊 🔆 Single or back to back	Testplan	100%	100%	100%	
2.2	🕒 🔆 SPI_read_corresponds_to_register_model	Testplan	0%	100%	-	
2.3	🕀 🔆 write_read_all_addresses	Testplan	96.66%	100%	96.66%	
2.4	🕀 🔆 check_default_values	Testplan	96.66%	100%	96.66%	
2.5	🕀 🔆 SPI_matrix_configuration_model_vs_registers	Testplan	100%	100%	100%	
2.6	🛓 🔆 readback_matrix_configuration	Testplan	100%	100%	100%	
2.7	🛓 🔆 different_matrix_configuration_scenario	Testplan	33.33%	100%	33.33%	
2.8	🛓 🔆 dummy_address	Testplan	96.87%	100%	96.87%	
3	👾 🔆 Reset signal	Testplan	100%	0%	100%	
4	🕀 🔆 Readout interface	Testplan	83.12%	100%	83.12%	
5	🗓 🔆 Pulse generation	Testplan	100%	100%	100%	
6	🖃 🔆 Power pulsing	Testplan	100%	100%	100%	

Topical Workshop on Electronics for Particle Physics (TWEPP 2016), September 26th-30th 2016, Karlsruhe, Germany