

FT-Unshades2: the User Friendly Framework as interface for designer support

Hipólito Guzmán-Miranda Javier Barrientos Luis Sanz Miguel A. Aguirre Patricio Molina

SEFUW, 3rd edition 15-17th March 2016



#### The story so far...

- Fault Injection as a concept is good overall, but can be improved
- The real value is in the INFORMATION we can get from the F.I. experiment
- Designers are typically busy with deadlines, campaigns and other issues
- Fault Injection tools should be usable, effective and with a reasonable difficulty curve
- In our experience, many gains on ease of use can be achieved through UI design



User Interface can help with:

- Design preparation
- Basic tool usage
- Basic post processing
- Easing difficulty curve

But it's more difficult to help with

- Many configuration options at the same time
- Advanced tool usage (custom campaigns)
- Custom post processing



### The experience with FT-Unshades2

- Design Preparation Flow
- FT-UNSHADES2 and Remote Access
- ASIC mode. Campaign settings.
- FPGA mode. Campaign setting
- Debugging. Detailed analysis
- Postprocessing
- Advanced issues



#### **Design Preparation Flow**

Implement a design using the standard design flow considering:

- Pinout fixed by the PCB
- SelectMap port has to remain open
- Bit allocation file (.II) is required
- Avoid DLLs and packing REGISTERS in PADs
- Generate a bitstream (.bit)

Stimuli set is obtained using a standard simulator

- A simulation of the circuit behavior is required
- Standard VCD format is used to obtain this file





### **FT-UNSHADES2** system





#### **Remote access**

Remote access (web based)

- User can use preferred Operating System (only a web browser is needed)
- User no longer needs to have all the hardware with him
- For maintenance purposes, Software and firmware can be upgraded without any friction to the user





#### ASIC mode

Fault Injection is performed only targeting the USER REGISTERS.

- After an injection the result is recorded into a Fault Dictionary.
- The procedure is repeated a significant number of RUNS. This is called CAMPAIGN.





### **Settings for ASIC mode**

#### Number of RUNS

Target registers and clk cycles

#### Type of campaign

- Random
- Exhaustive
- File
- Custom (tcl)
- Dummy

# Number of injections per run

grow is	CONSTRUCTION OF STRUCTURE			н	
reged as	Cepore -	101			Cosa 🔐 🤅
	att 📮Tas 🍸	tierarchy 🦻 🏞	1 Debre		
15.94	rpet FPGA-configuration	memora.bt		-	
3. 5	rget FPGA vectors.	memoria.dut			Your TWT assasion has expired or crashed
10.00	nih site:	100000		_	
6.0	heck residual darhape	140		1.1	
10.00	enege per nun	1			
6.0	tip in Ganage	Yes		-	
10.00	NOK teature wor	160		14	
10	illip allal run	No		-	
1.18	configure at error	No			
10 m	configure at time	0			
1 14	inder	Random – ran	dom cycle/hegeter.com	bratom	
	R	in Campaign			



#### **FPGA mode**

Fault Injection is performed targeting the CONFIGURATION BITS. Reuse the FT-UNSHADES2 hardware platform.

A fault on one configuration bit will not propagate to any other configuration bit (Is it totally true? Not really!)

Of course some faults on configuration bit can propagate to the circuit logic

□ Make a DYNAMIC injection (not only injections at time zero!)

□ Technique: inject and repair (fast scrubbing)





## Debugging

View the Detailed analysis of Fault Propagation. Registers and primary IOs are selected a priori

The system creates a waveform for each FPGA (golden & faulty)

Click on the register icon allows user to modify faulty's value

If there is a discrepancy between the two waveforms, it is highlighted in red.

rylling an allow a .	
American State	and the
0 4 · H · + · A ·	P - 12
	***************************************
Ci 🔒 Roberti	較終約較前於後國於總會佔總於給於於10.11.12.13.13.19.19.19.19.19.19.19.19.19.19.19.19.19.
	自然机构的外的局部的新闻机构新品牌的新品牌发展并且自己分布并且并且不是有
C 14 184	NONE INST ENGINEERED CONCERNMENT AND A CONCERN
# intellect	
C S Annala	827
	50%
0 * H	
🛛 🛔 Auto	
	ACCESSION OF COMPANY AND A COM
D 🖈 1440	
	£
	Ready



#### Postprocessing

# Multiple modes of operation force the system to log fault information in different files:

- injections.csv
- damages.csv
- reg\_name.csv
- stats.txt

Log	iged as hi	polito <del>–</del>					
C	reate new	file <mark>t</mark> x	📙 Files		•: Hierar	chy	1
		6	÷	Ē	ð		
	Plain te Directo User Co FPGA Ve Logic Le Analysis	xt file ry onstraint Fil ector File (. ocation File s of campa	le (.ucf) dat) from \ e (.ll) for co ign results	/CD dun nfigurat	np (.vcd) ion bits		



#### For advanced issues

- We strive to make our tool teach itself to the user: basic usage should prepare the designed to begin with advanced issues
  - i.e: Button clicks produce commands which can be reused
- User manuals available online
- Partial reconfiguration (scrubbing) through the user interface in interactive mode & scriptable
- Wanted: FTU2 application notes



#### Wishlist

- Integrate Implementation tools for FT-Unshades server-based FPGA implementation:
  - Not possible now because legal issues (would need custom software license)
  - Would simplify a lot design preparation process
  - Guarantee designs and bitstreams have compatible
    FTU2 configurations 100% of the time
- Provide script templates for some typical postprocessing operations
- FTU2 as a tool for radiation tests



#### Conclusions

- A tool for Fault Injection is available, it uses Xilinx SRAM-FPGAs
- The software that interconnects with the hardware is original and is determined to be an aid to the designer
- The application is linked to the platform and provides information in ASIC, FPGA and analysis mode
- Let's see an example



#### **Scrubbing emulation**

#### Perform a partial (or even complete!) reconfiguration **maintaining** the current contents of user FFs

This way scrubbing schemas can be tested

Let's see a brief example



Logged as hipolito -

ᡖ counter_8bit_partialreconf 🛛 🔓 Files	•: Hierarchy	📌 Run	🕷 Debug	
	2	·*		cl
	0 1 2 3 4 5 6	7 8 9 10		cl
🗆 🔜 /data_out	00 01 02	03 04 05 06		
	00 01/02	03 04 05 06 🖒		
🗆 📌 /enable				
🗆 🔜 /reg	00 01/02	03 04 05 06		
	00 01/02	03 04 05 06		
🗆 📌 /rst_high				
				-
	3			

Ready



javascript:Ftu.shell.run('obj\_bind \$FPGA\_CONFIG 2\nxc\_configure all');



Logged as hipolito -

ᡖ counter_8bit_pa	Step cycles	•: Hierarchy	📌 Run	🛞 Debug	
	+ a				cl
□ 📙 /data_out □ 📌 /enable	1 cycle 5 cycles 10 cycles other until next event	0 1 2 3 4 5 6 00 01 02 00 01 02	7 8 9 10 11 12 03 04 05 06 05 04 03 04 05 06 05 04	13 14 15 03 02 01 03 02 01	ob xc cll
🗆 🔜 /reg		00 01 02 00 01 02	03 04 05 06 05 04 03 04 05 06 05 04	03 02 01	
□ 📌 /rst_high					
		«.			

0	Mozilla Firefox	
<u>File Edit View History Bookmarks Too</u>	ols <u>H</u> elp	
∫ 💡 http://ftu.us.es/uff/ 🛛 🛪 🕀		
🍮 🖂 🏈 ftu.us.es/uff/		
Reconfigure target FPGAs with one of		
the bitstreams on board tialreconf	🕂 Hierarchy 🥓 Run 🛞 Debug	
	<i>p</i> - <b>n</b>	clł
counter_8bit_up.bit	0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15	obj
up2down.bit	00 01 02 03 04 05 06 05 04 03 02 01	
down2up.bit	00 01 02 03 04 05 06 05 04 03 02 01	CCF
🗆 📌 /enable		
🗆 🔜 /reg	00 01 02 03 04 05 06 05 04 03 02 01	
	00 01 02 03 04 05 06 05 04 03 02 01	
🗆 📌 /rst_high		
	2	
1		

javascript:Ftu.shell.run('obj\_bind \$FPGA\_CONFIG 3\nxc\_configure all');



Logged as hipolito -

ᡖ counter_8bit_pa	Step cycles	•: Hierarchy	📌 Run	🕷 Debug	
	+ @ -	2 - E			cl
	1 cycle	0 1 2 3 4 5 6	7 8 9 10 11 12	13 14 15 16 17 18 19 20	cl
🗆 🔜 /data_out	5 cycles	00 01 02	03 04 05 06 05 04	03 02 01 02 03 04 05 06	obj xc
	other	00 /01/02	03 04 05 06 05 04	03 02 01 02 03 04 05 06	cl
🗌 💉 /enable	until next event				obj xc
🗆 🔜 /reg		00 01 02	03 04 05 06 05 04	03 02 01 02 03 04 05 06	cl
		00 01 02	03 04 05 06 05 04	03 02 01 02 03 04 05 06	
🗆 📌 /rst_high					
		97			
		0			

javascript:Ftu.shell.run('clk\_step 5');

Ready



#### Conclusions

A tool for Fault Injection is available, it uses Xilinx SRAM-FPGAs

A Brave FPGA (NanoXplore NXT) version is intended for the future

The application is linked to the platform and provides information in ASIC, FPGA and analysis mode

We need your feedback to make our tool better ;)