Intro	Why FPGA (and what the FPGA is all about)	Software approach – simulation first	Going to the hardware	What's next	Contact info
00	0	000	000	0	0

On-hardware debugging of IP cores with free tools

Anton Kuzmin

2020-02-02



Software approach - simulation first Going to the hardware What's next Contact info

Who am I

- not really a software developer ... but write code sometimes
- developing embedded systems for 25 years
- VME, CompactPCI, AdvancedTCA, SoM
- FPGA and SoC-FPGA (Altera/Intel, Microsemi/Microchip)
- VHDL (RTL-code, no, it is not a software)

My usual problem with the software is how to make it run on a hardware which is known not to be working yet and how to bring-up and test this hardware. With a soft-core CPU it is getting even worse.



Intro	Why FPGA (and what the FPGA is all about)	Software approach – simulation first	Going to the hardware	What's next	Contact info
00	0	000	000	0	0

Intro

Why FPGA (and what the FPGA is all about)

Software approach – simulation first

Going to the hardware

What's next

Contact info



Software approach – simulation first Going to the hardware Contact info

Why software develoers should care about FPGA

- conventional hardware architectures are stuck
- the only two mainstream HDLs represent software technology level of a stone age (well, last century)

Therefore...

- you need it to make next generation software run on heterogeneous and malleable hardware
- industry needs your help to move forward



Free software for HDL developers

Simulate it!!

- Verilog HDL
 - Verilator (https://www.veripool.org/wiki/verilator)
 - Icarus Verilog (http://iverilog.icarus.com/)
 - Yosys (http://www.clifford.at/yosys/)
- VHDL: ghdl (http://ghdl.free.fr/) GHDL is an open-source simulator for the VHDL language.
- GTKWave (http://gtkwave.sourceforge.net/) GTKWave is a fully featured cross-platform wave viewer

Unit and regression testing for hardware blocks (Python cocotb, VPI, etc.)



000

Software approach - simulation first Going to the hardware What's next Contact info

```
VHDL code snippet
```

```
process (d. mode)
begin
  if mode = '0' then
     d_next <= std_logic_vector(unsigned(d) + 1);</pre>
  else
     d_{next}(7 \text{ downto } 1) \leq d(6 \text{ downto } 0);
     d_{next}(0) \le d(7) \text{ xor } d(5) \text{ xor } d(4) \text{ xor } d(3);
  end if:
end process;
```



GTKWave with GHDL simulation results

			GTKWave -	data_gen_tb.ghw				
File Edit Search Time	Markers View Help							
🙀 🖷 💼 🖂 👄 😽 I 🗞 II 🤞	⇒ From: 0 sec. To: 2	Marker: 260 ns C	ursor: 8284 ns					
SST	Signals	Waves						
≅top	Time	6 us 7 us	8 us 9 us	s 10 us	11 us 12 us	13 us	14 us	15 us 16 u
adata_gen_tb	clk=1							
¢≑dut	cnt=7							
-■d_next	reset=0							
-■d ■data	reset=0							
-=data	enable=1							
-data	mode=0							
ignals	data[7:0]=01			((()))))((()))())(((())))((((()))))(((((
	data[7]=0							
	data[6]=0							
	data[5]=0							
1	data[0]=1							
lter:								
Append Insert Repla	ace							

... and live demo

Software approach – simulation first Going to the hardware

Going to the hardware What's next Contact info

On-chip instrumentation

Proprietary and vendor specific tools (and their problems)

- Altera/Intel SignalTAP
- Xilinx ChipScope
- Synopsys Identify RTL Debugger
- Microsemi/Microchip SmartDebug

What is in common: standard inteface to the hardware IEEE 1149.1 (JTAG)



Software approach – simulation first

Going to the hardware What's next Contact info

Assembling a puzzle

- Free software to speak with hardware on IEEE-1149.1
 - UrJTAG (http://urjtag.org/)
 - OpenOCD (http://openocd.org/)
- libsigrok (https://sigrok.org/)
- PulseView (sigrok Logic Analyzer GUI)
- SpiderBoard with SpiderSoM (http://www.spiderboard.org/) or MX10 (https://www.aries-embedded.com/system-on-module/fpga/) Altera/Intel MAX10 FPGA module with built-in USB-to-JTAG interface



Software approach – simulation first Going to the hardware

Going to the hardware What's next Contact info

Logic Analyzer trace

			Session	1 - PulseView			×
Run * Session	10						
Session 1	🕮 🔤 fosdem'20 vjtag	1-la-mx10 · 🛪 🍠 25	5 MHz ·				
-54.0 μs	-53.0 µs	-52.0 µs	-51.0 µs	-50.0 µs	-49.0 μs	-48.0 µs	-47.0 μs
DO							
D3							
D4							
D5							_
D6							-

... and live demo



Intro Why FPGA (and what the FPGA is all about) 00 0 Software approach – simulation first Going to the hardware

Going to the hardware What's next Contact info

Challenges ahead

• integration

- support for different hardware interfaces, FPGA vendors, device families
- automated design instrumentation (kudos to vendor tools)
- IEEE-1149.7, IEEE-1687 (2014), etc.
- integration with software debugging tools



Intro Why FPGA (and what the FPGA is all about) 00 0

Software approach – simulation first

Going to the hardware What's next

ct Contact info

Thank you!

Anton Kuzmin anton.kuzmin@cs.fau.de https://github.com/ak-fau/

Questions?..



