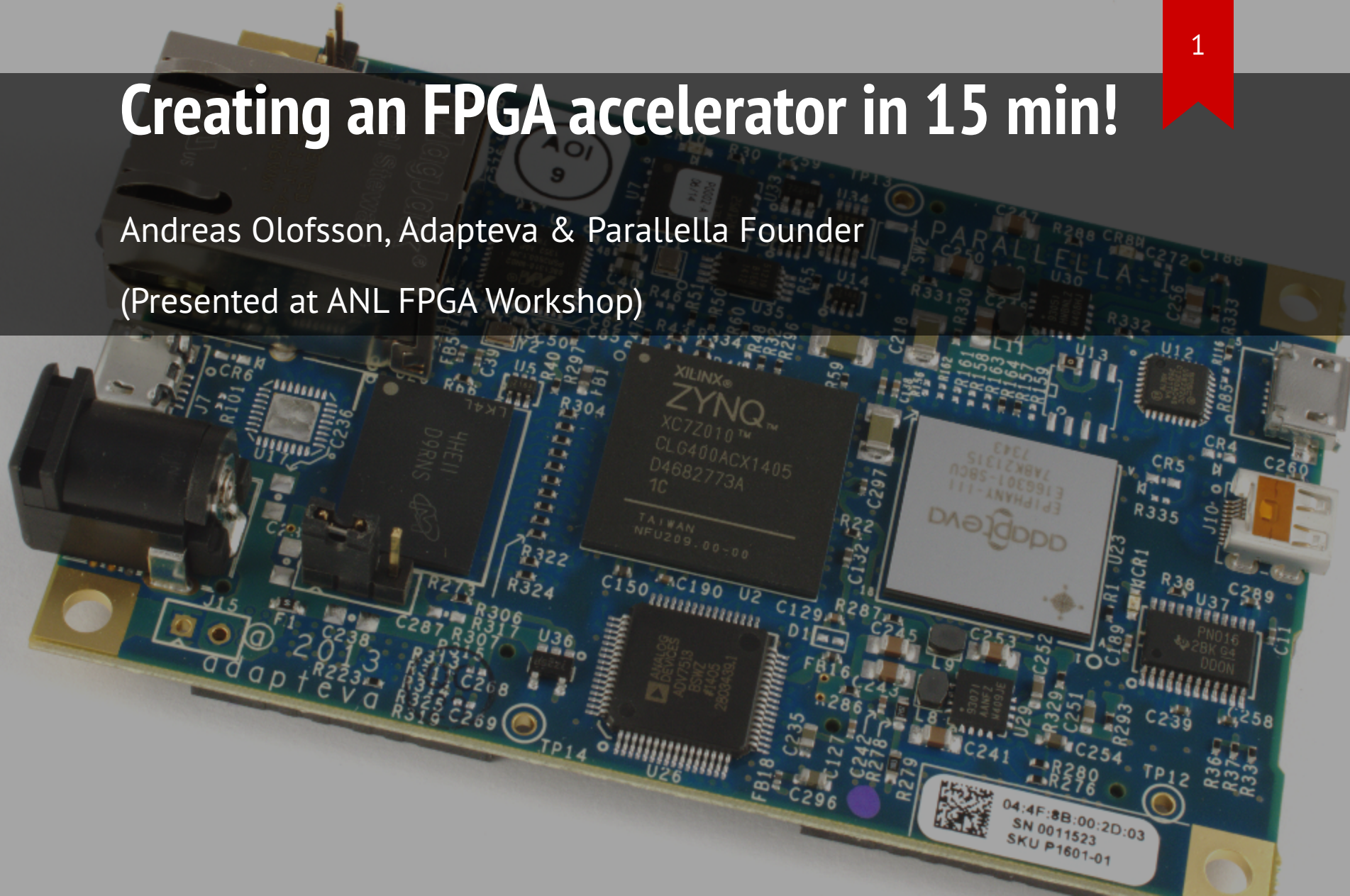


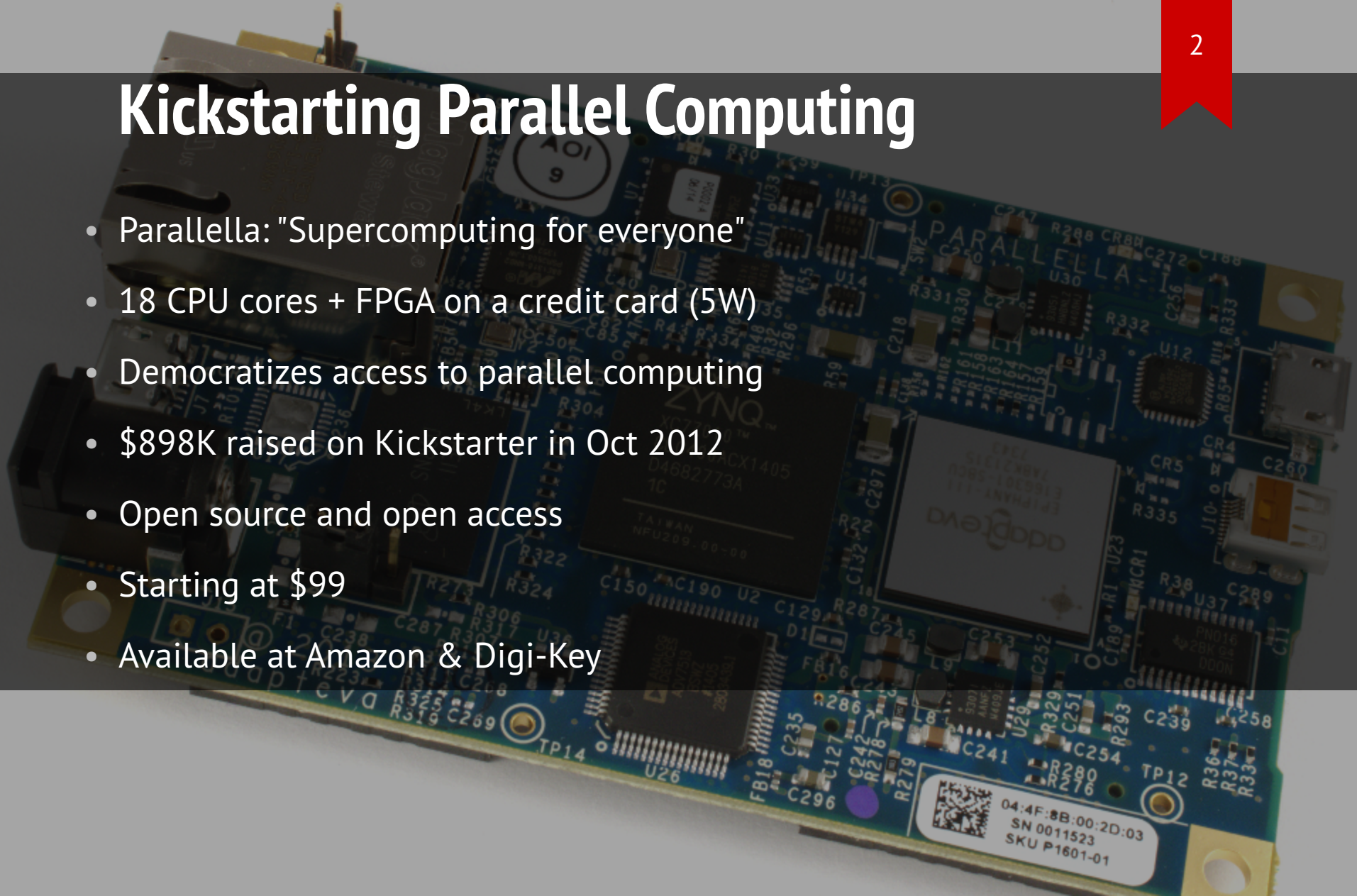
# Creating an FPGA accelerator in 15 min!

Andreas Olofsson, Adapteva & Parallella Founder  
(Presented at ANL FPGA Workshop)



# Kickstarting Parallel Computing

- Parallella: "Supercomputing for everyone"
- 18 CPU cores + FPGA on a credit card (5W)
- Democratizes access to parallel computing
- \$898K raised on Kickstarter in Oct 2012
- Open source and open access
- Starting at \$99
- Available at Amazon & Digi-Key





# Parallella Specs (<http://parallella.org>)

|              |                       |
|--------------|-----------------------|
| Performance  | ~30 GFLOPS            |
| Architecture | ARM + FPGA + MANYCORE |
| Memory       | 1GB DDR3              |
| IO           | ~25 Gb/s (48 GPIO)    |
| Size         | credit-card           |
| Power        | <5W                   |
| Cost         | \$99 -> \$249         |

# "Hello World" in Software

1. **CODE:** `main() { printf("Hello World\n"); }`
2. **COMPILE:** `gcc hello.c`
3. **TEST:** `./a.out`
4. **DEBUG:** `printf`, `gdb`

# "Hello World" in Hardware

1. **CODE:** Verilog/VHDL source
2. **CODE MORE:** Verilog/SystemC testbench
3. **TEST:** VCS/NC/Icarus/Verilator
4. **DEBUG:** Waveform debugging
5. **SYNTHESIZE:** HDL-->NETLIST-->POLYGONS
6. **BURN:** FPGA/ASIC
7. **TEST MORE:** Pray that it works...

# Hardwave vs Software

|                 | SW      | HW             |
|-----------------|---------|----------------|
| Compile Time    | seconds | minutes/months |
| Libraries       | lots    | little         |
| Debugging       | "easy"  | an art         |
| Cost of mistake | low     | VERY HIGH!!!!  |

# Let's start..."hello world"

```
assign result[31:0]=input0[31:0]+input1[31:0];
```



*Now what??*

# What's missing

1. Control code
2. Host/Accelerator Hardware interfaces
3. Test environment
4. Synthesis scripts (non trivial)
5. Drivers (software)

“ *How many man-years is that?* ”



# OH! (Open Hardware Library)

- Verilog
- MIT license
- ~15K lines of code so far
- Best practices based on 20 years of chip design
- Silicon proven building blocks
- **Small:** FIFOs, synchronizers, muxes, arbiters, etc
- **Big:** chip to chip link, mailboxes, memory translators
- <http://github.com/parallela/oh>
- Yes, we do accept pull requests!

# DEMO

04:4F:8B:00:2D:03  
SN 0011523  
SKU P1601-01

# Accelerator Case Study

1. **Coding:** 2hrs
2. **Simulate/Debug:** 2hrs
3. **Synthesize:** 2hrs
4. **Debug 1st "Bus Error":** 1hr
5. **Debug 2nd "Bus Error":** 2hrs



*9hrs to put together something that takes 30 seconds in C!*

# Files Used

REPO: [github.com/parallella/oh/accelerator](https://github.com/parallella/oh/accelerator)

1. **Code:** hdl/{accelerator.v,axi\_accelerator.v}
2. **Testbench:** dv/{dut\_axi\_accelerator.v,build.sh,run.sh}
3. **Synthesis:** fpga/{package.tcl, run.tcl}
4. **Drivers:** sw/{driver.c,test.c}

# How to Run

## VERIFY:

```
$ cd accelerator/dv
$ ./build.sh # build
$ ./run.sh tests/hello.emf # load data
$ gtkwave waveform.vcd # view waveform
$ cd ../fpga
$ ./build.sh # build bitstream
```

# Conclusions

1. Yes, today you CAN build an FPGA accelerator in 15 min
2. Anything new is still 100x more expensive to develop than SW
3. Develop for FPGAs, but keep ASIC option open

“ *...to make FPGA universally viable we need to catch up with >>\$trillion investment in software infrastructure*

**Email:** andreas@adapteva.com

**Twitter:** @adapteva