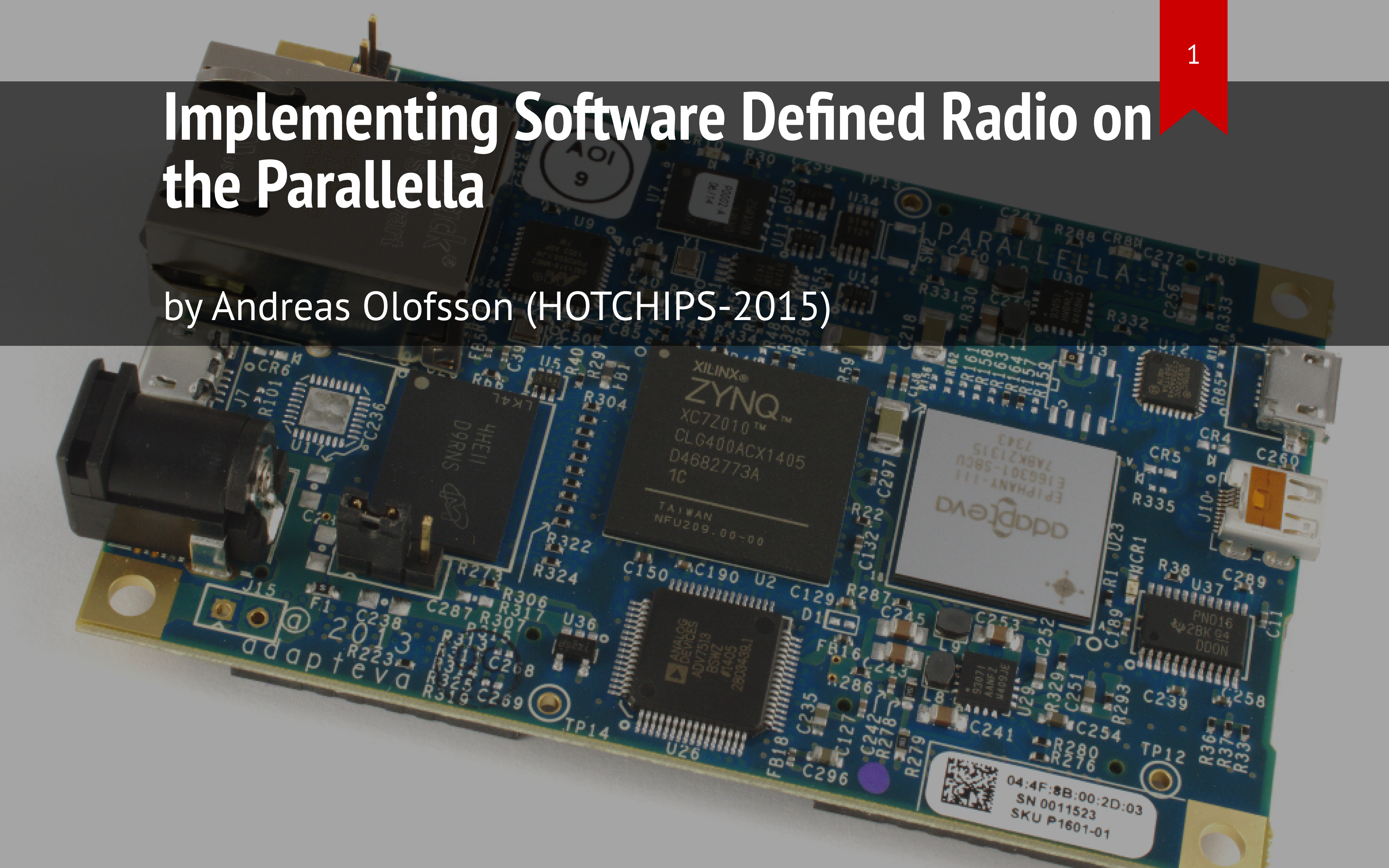
Implementing Software Defined Radio on the Parallella

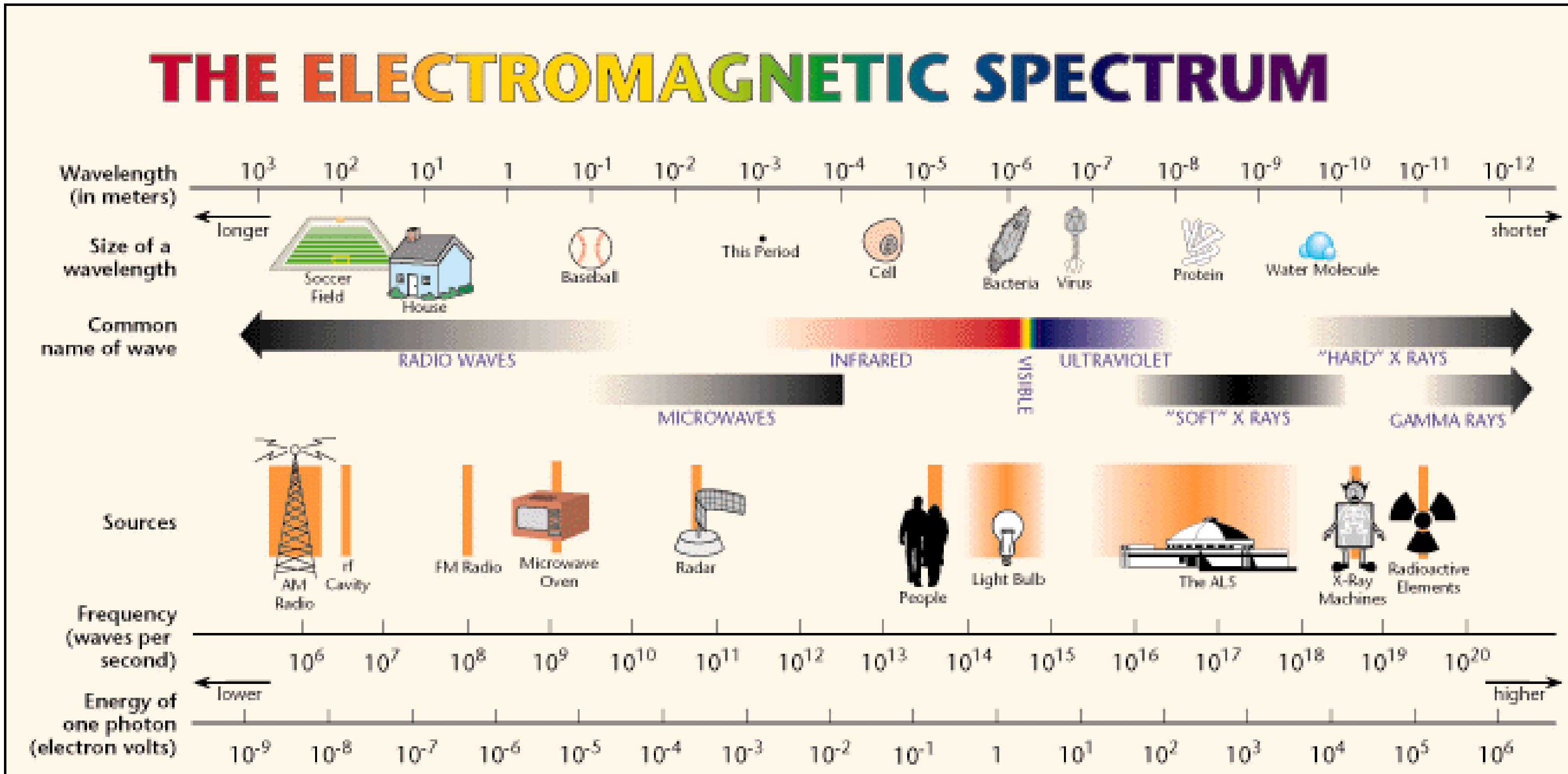
by Andreas Olofsson (HOTCHIPS-2015)



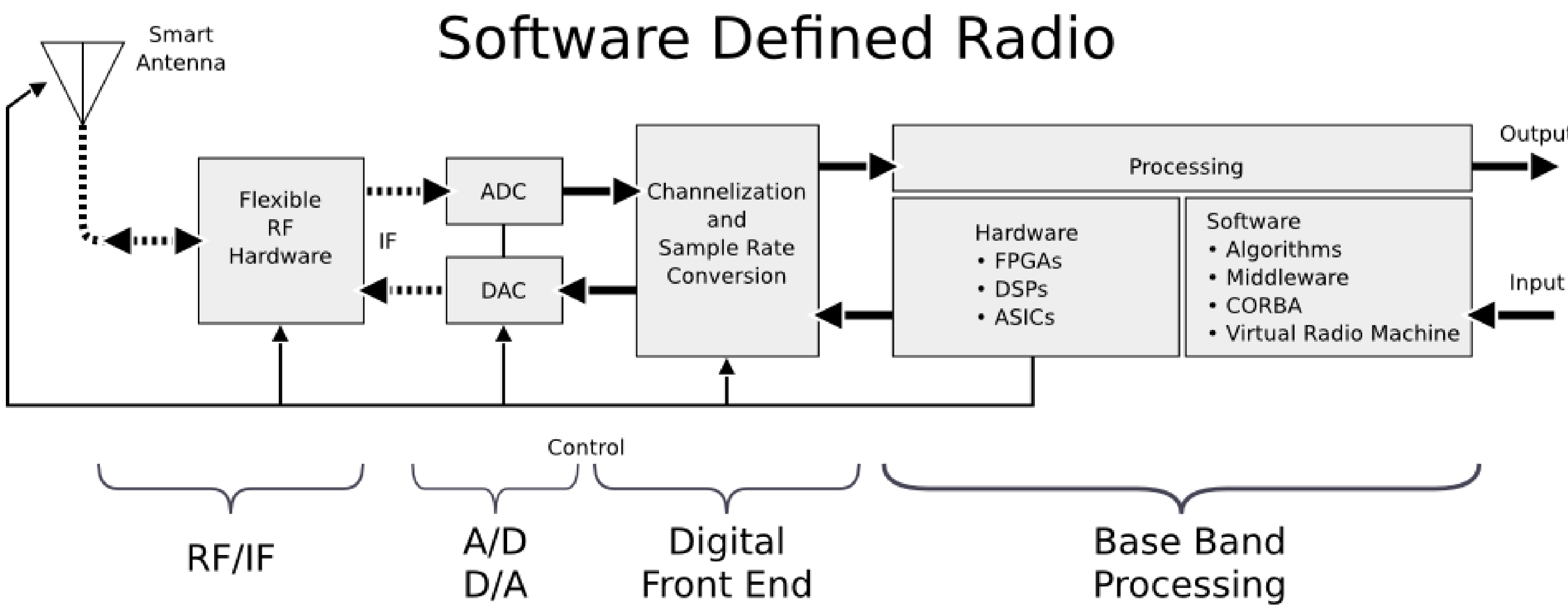
What is Software Defined Radio? (SDR)

"Radio in which some or all of the physical layer functions are software defined" --Wireless Innovation Forum

Not just "radio"!



Canonical SDR architecture

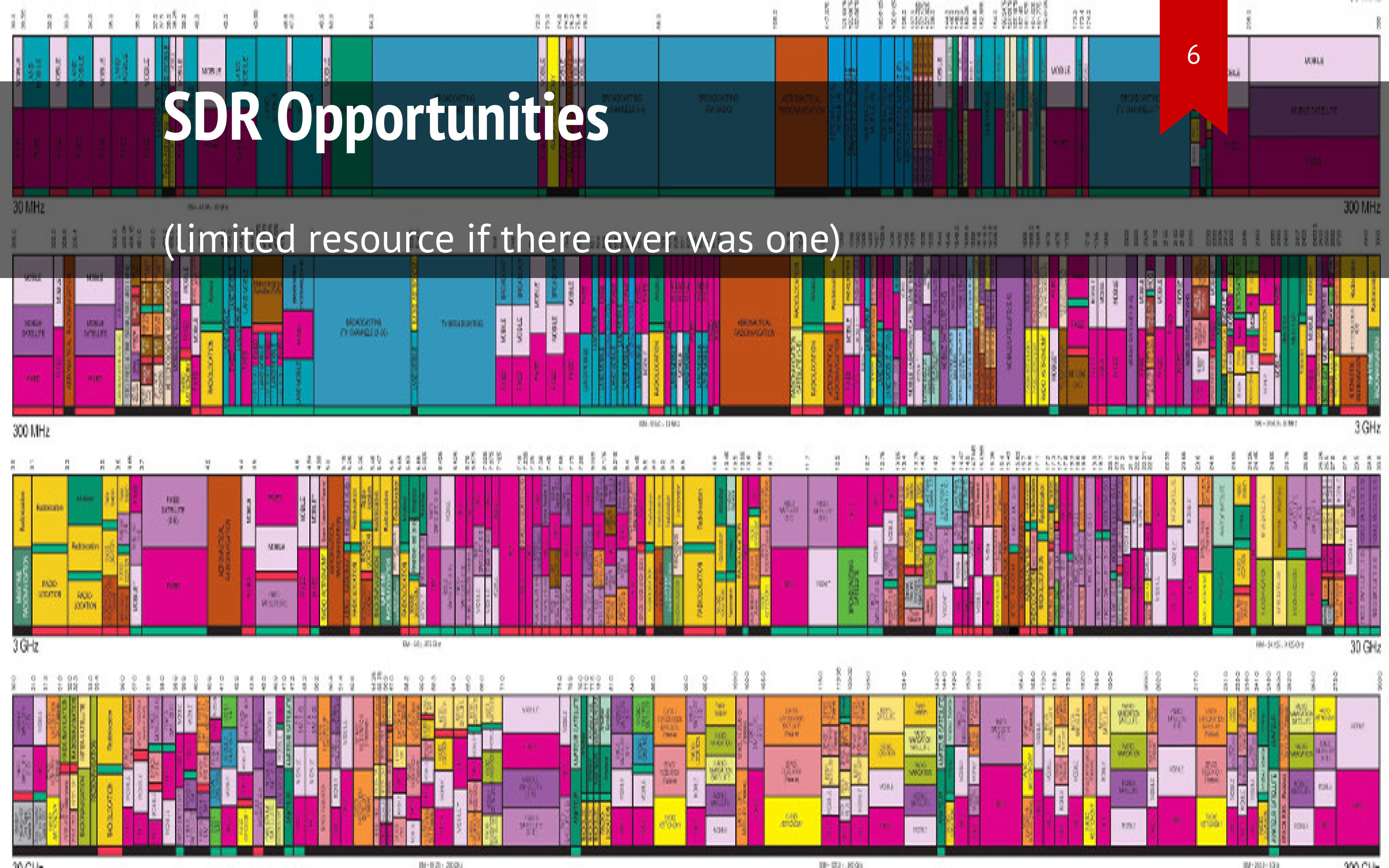


Why SDR is so cool (&hot)!!

	HW	SDR
Compilation	Months	Minutes
Cost	$50K	$500
Hurdle	RF, HW, SW	SW
Real time configurable	No	Yes
Future proof	No	yes

SDR Opportunities

(limited resource if there ever was one)

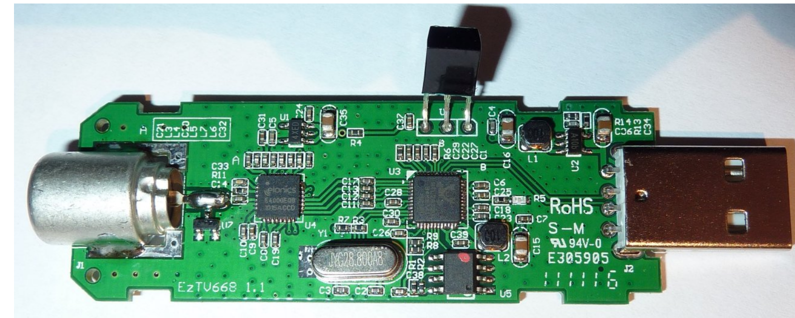


SDR Application Examples

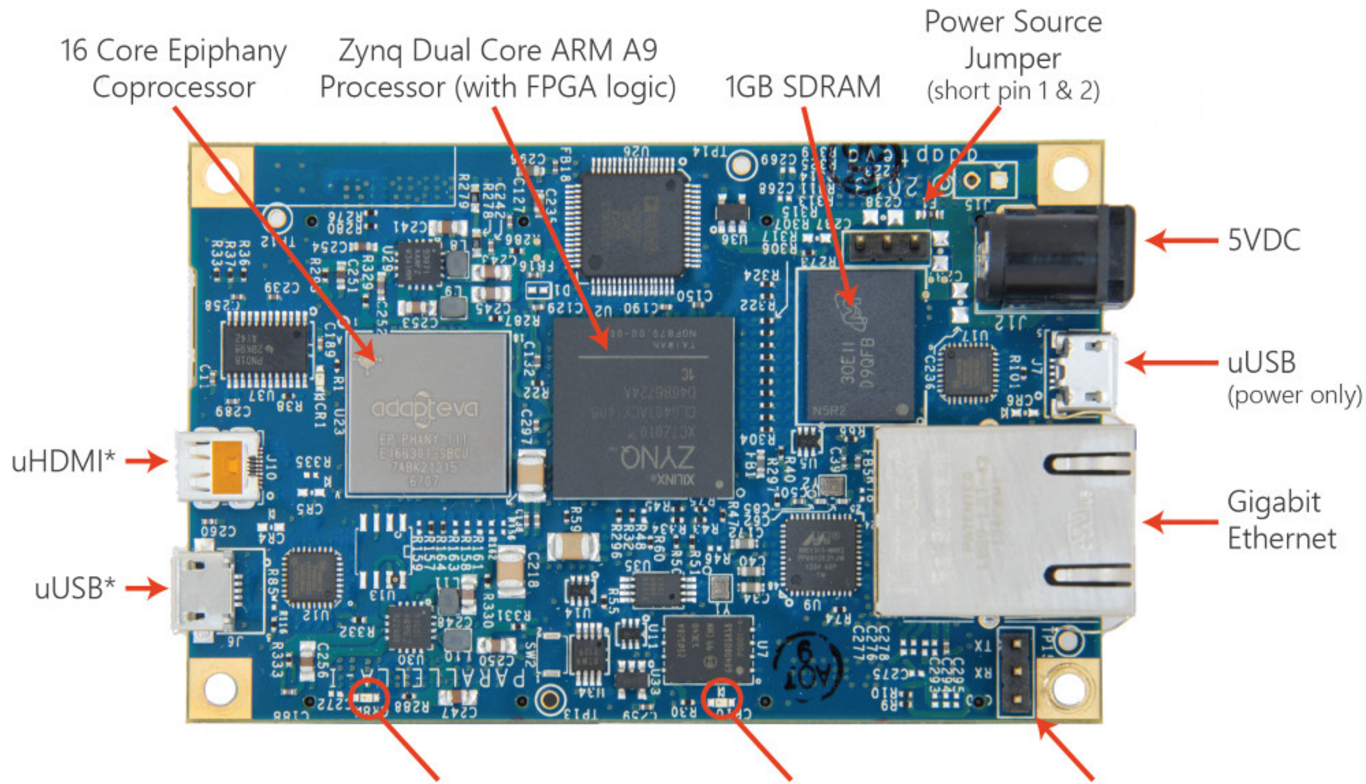
- Amateur radio (HAM, etc)
- Radio astronomy
- Legacy modem emulation
- Wireless comms (GSM, LTE)
- Wireless research (5G)
- Spectrum analysis
- Teaching DSP

SDR Challenges

- Latency (microsecond)
- Throughput (gigabits)
- Size, Weight, Power (SWAP)
- Cost ($20-->$30,000)



Parallella Introduction



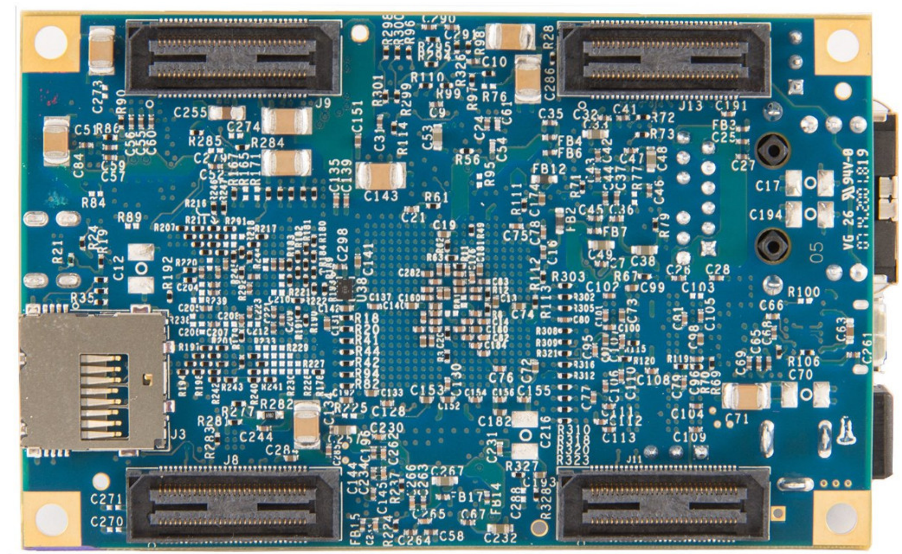
* Not on P1600-xxx models

Parallella Specs (parallella.org)

Performance	~30 GFLOPS
Memory	1GB DDR3
IO	~25 Gb/s (48 GPIO)
Size	credit-card
Weight	38g
Power	<5W
Cost	$99 -> $249

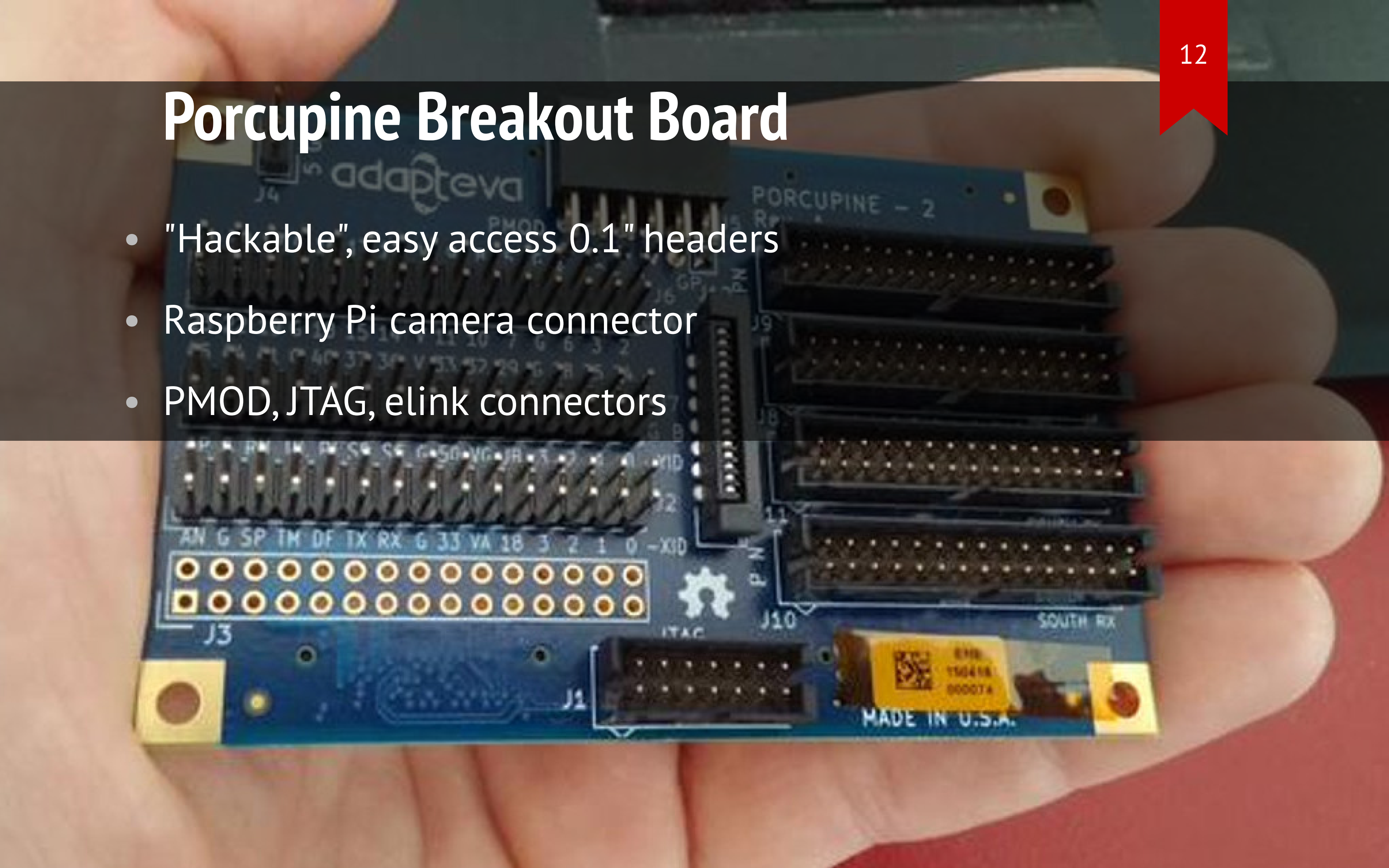
Parallella IO

- 0.5mm Samtec connectors
- 48-pin/24Gbps FPGA link
- 2 Epiphany links (20Gbps)
- JTAG, UART, I2C, SPDIF
- LVDS/CMOS
- Adjustable I/O voltage



Porcupine Breakout Board

- "Hackable", easy access 0.1" headers
- Raspberry Pi camera connector
- PMOD, JTAG, elink connectors



Parallella SDR Platform

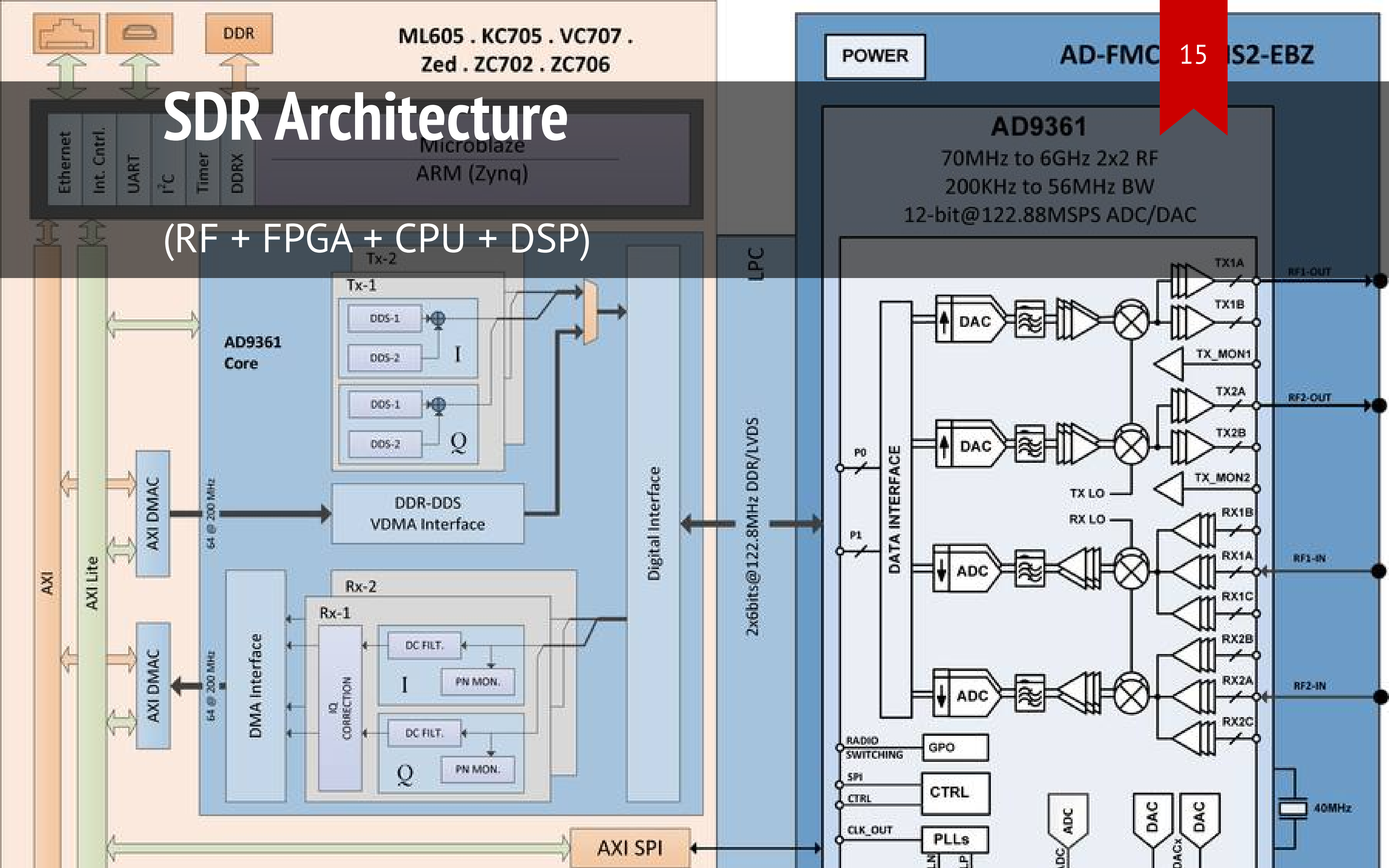
- 70MHz - 6GHz
- [ADI FCOMMS* board (FMC)]
- Parallella carrier
- FMC adapter board
- 100% Open source SW

AD9361 Overview

- RF 2 × 2 transceiver
- 12-bit DACs and ADCs
- 70 MHz to 6.0 GHz
- TDD/FDD support
- BW: <200 kHz to 56 MHz
- Noise figure < 2.5 dB
- Independent AGC

SDR Architecture

(RF + FPGA + CPU + DSP)



Xilinx Zynq SoC Architecture

- **ARM SOC:**

  -Dual A9 CPUs (up to 1GHz)

  -GigE,USB,UART,I2C, …

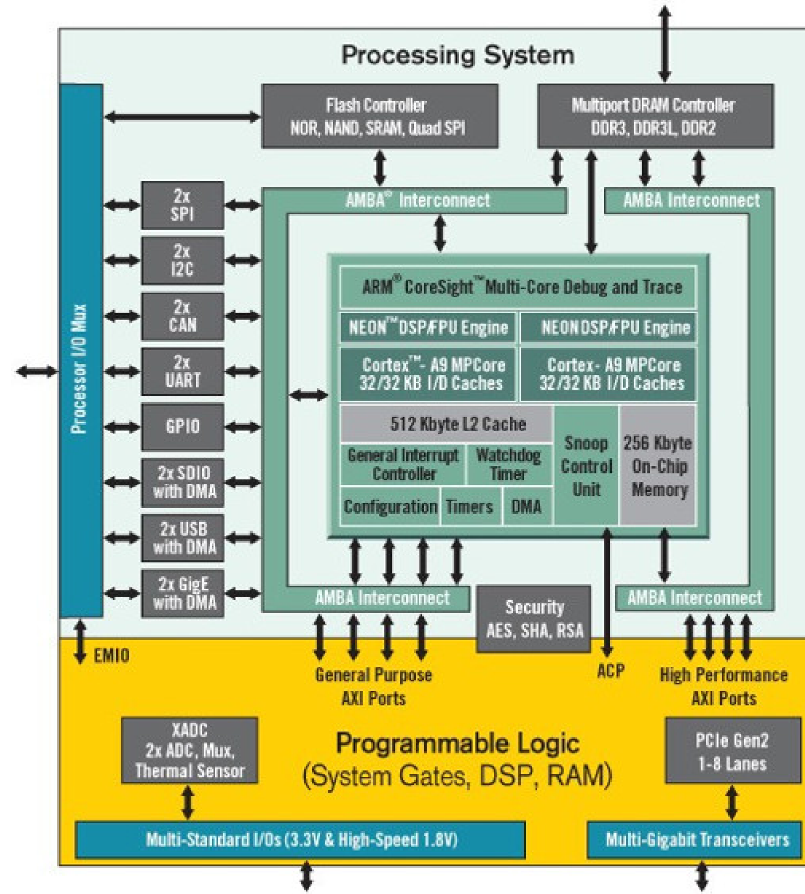
  -Flash & DDR3 controller

- **Programmable Logic:**

  -I/O: (86 --> 470)

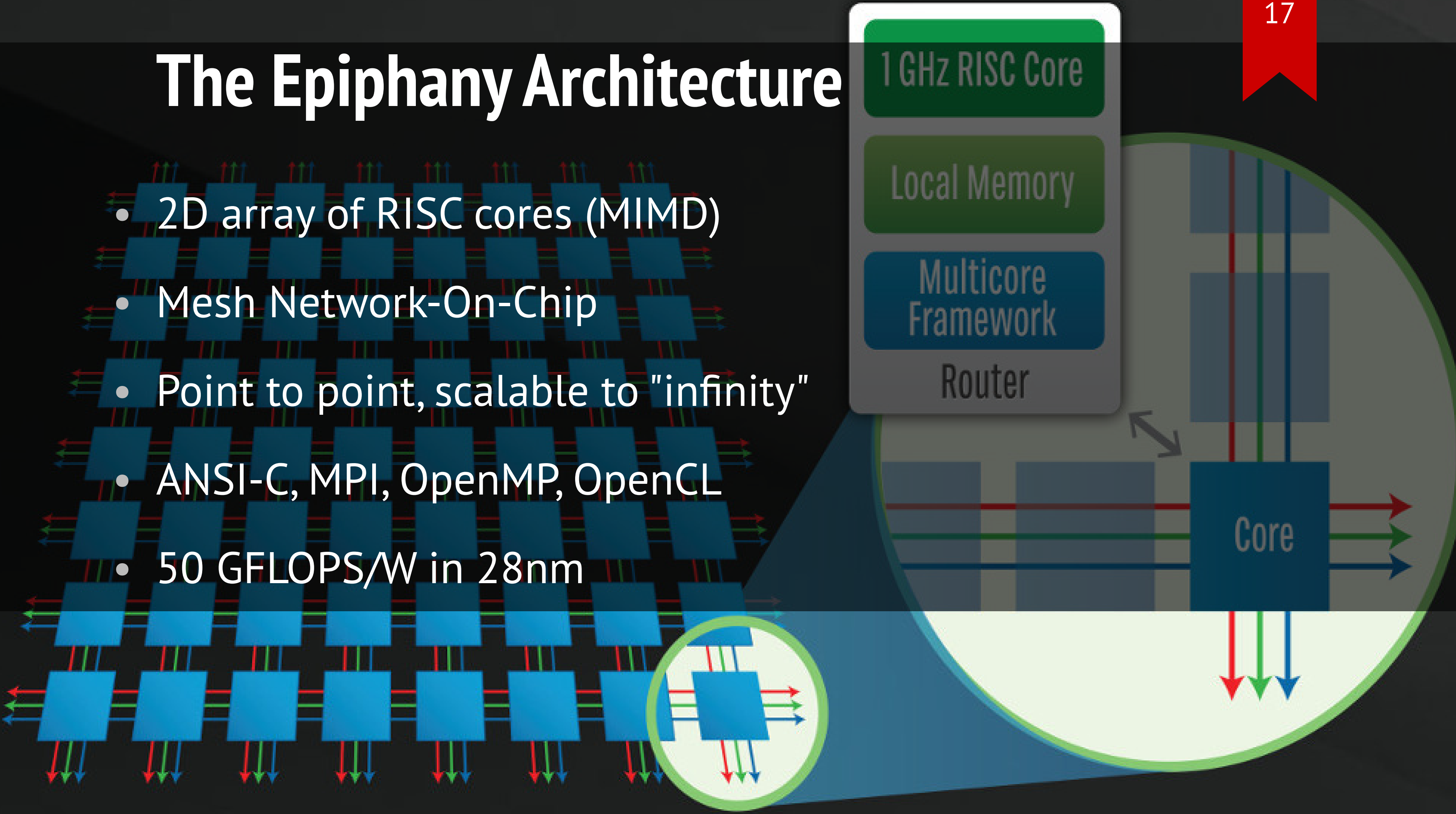
  -LUTS: (17 --> 277K)

  -BRAM: (0.24MB --> 3MB)



The Epiphany Architecture

- 2D array of RISC cores (MIMD)
- Mesh Network-On-Chip
- Point to point, scalable to "infinity"
- ANSI-C, MPI, OpenMP, OpenCL
- 50 GFLOPS/W in 28nm



Epiphany Chip Features

- 16 x 1GHz RISC processors (65nm)
- 32 bit IEEE floating point
- 512KB on-chip SRAM
- North, East, West, South IO links
- 32 GFLOPS peak performance
- 512 GB/s local memory BW
- 128 GB/s NOC bandwidth
- 8 GB/s IO bandwidth

Epiphany RISC DSP

- 32 bit, dual issue in order, 5/8 stage pipeline
- 16/32 bit instruction set
- 64 general purpose registers
- IEEE754 floating point (FMADD,FMUL,..)
- Nested interrupts, 2-Channel DMA, debug unit
- ISA: B, BL, JR, JAL, LDR, STR, TESTSET, ADD, SUB, ASR, LSR, LSL, ORR, AND, EOR, BITR, FADD, FSUB, FMUL, FMADD, FMSUB, FABS, FIX, FLOAT, MOV, MOVT, MOVFS, NOP, IDLE, RTS,

Epiphany Memory System

- 32 bit addressing
- Shared flat address space, no HW caches!
- Upper 12 bits specify coordinates in 2D map (64 x 64 mesh)
- 32KB SRAM per core in E16G301. Accessible by all cores.
- 4 independent 64 bit local memory transactions per cycle
- Fetch, load, DMA, emesh supports 32 byte access per cycle
- Strict local memory ordering, VERY relaxed remote odering.

Epiphany Network-On-Chip

- 3 meshes: on-chip writes, read requests, off-chip writes
- 104 bit atomic single cycle packets
- Non-blocking round robin routing
- x/y static routing
- 8 bytes transfered per cycle (on chop write mesh)
- 1.5 clock cycle latency / hop
- Extends off chip to I/O (elinks)

Parallella SDR

Software

Free Software Resources

- GNURadio: Open source SDR platform
- Epiphany SDK: Epiphany compiler, debugger
- Vivado: FPGA synthesis tools
- COPRTHR: OpenCL, MPI, Threads
- OpenMP: OpenMP 4.0 device
- PAL: Optimized open source math/dsp library

Creating a Parallella SD card

- Download image
- Insert SD card in laptop

```
$ gunzip -d <releasename>.img.gz
$ df -h
$ umount <sd-partition-path>
$ sudo dd bs=4M if=<release-name>.img of=<sd-device-path>
$ sync
```

- Remove SD card and insert into Parallella

Install Vivado

- Download Vivado from Xilinx (Choose the web installer)

```
$ sudo unlink /bin/sh
$ ln -s /bin/bash /bin/sh
$ chmod u+x ./Xilinx_Vivado_SDK_2015.2_0626_1_Lin64.bin
$ ./Xilinx_Vivado_SDK_2015.2_0626_1_Lin64.bin
$ source 2015.2/settings64.csh
```

Install GNURadio Dependancies

```
$ sudo apt-get -y install git-core cmake g++ python-dev swig \
pkg-config libfftw3-dev libboost1.55-all-dev libcppunit-dev \
libgsl0-dev libusb-dev  python-wxgtk2.8 \
python-numpy python-cheetah python-lxml doxygen libxi-dev \
python-sip libqwt-dev libfontconfig1-dev \
libxrender-dev python-sip python-sip-dev
```

Download GNU Radio (from ADI)

```
$ sudo dd if=/dev/zero bs=1MiB of=/home/<user>/swap.img
$ sudo mkswap /home/<user>/swap.img
$ sudo swapon /home/<user>/swap.img
$ git clone https://github.com/analogdevicesinc/gnuradio.git
$ git clone https://github.com/analogdevicesinc/libiio.git
$ cd gnuradio
$ git checkout master
```

Building software

```
$ mkdir gnuradio/build; cd gnuradio/buid
$ cmake -DENABLE_DOXYGEN:bool=false ..
$ make -j2
$ sudo make install
$ sudo make -C gr-iio install
$ sudo ldconfig
$ cd ~/libiio
$ cmake ./
$ make all
$ sudo make install
```

Shortcut…b/c life is too short

[PARALLELLA SDR IMAGE](#)

REFERENCES

FMCOMMS3 User Guide (ADI)

GNURadio Installation (ADI)

IIO-scope User Guide (ADI)

SD-CARD WIKI (ADI)

ADI at FOSDEM

SDR demo

- FCOMMS2 + Adapter board + Parallella
- ARM + FPGA
- ADI Oscilloscope application
- ...

Epiphany demo

Single core FFT prepared by Sylvain Munaut (SDR guru)

Source code

Platform	Results
fftw (ARM A9)	221.977509 Mflops
fftw (A9-Neon)	409.619659 Mflops
epiphany C	170.642029 Mflops
epiphany ASM	668.507629 Mflops