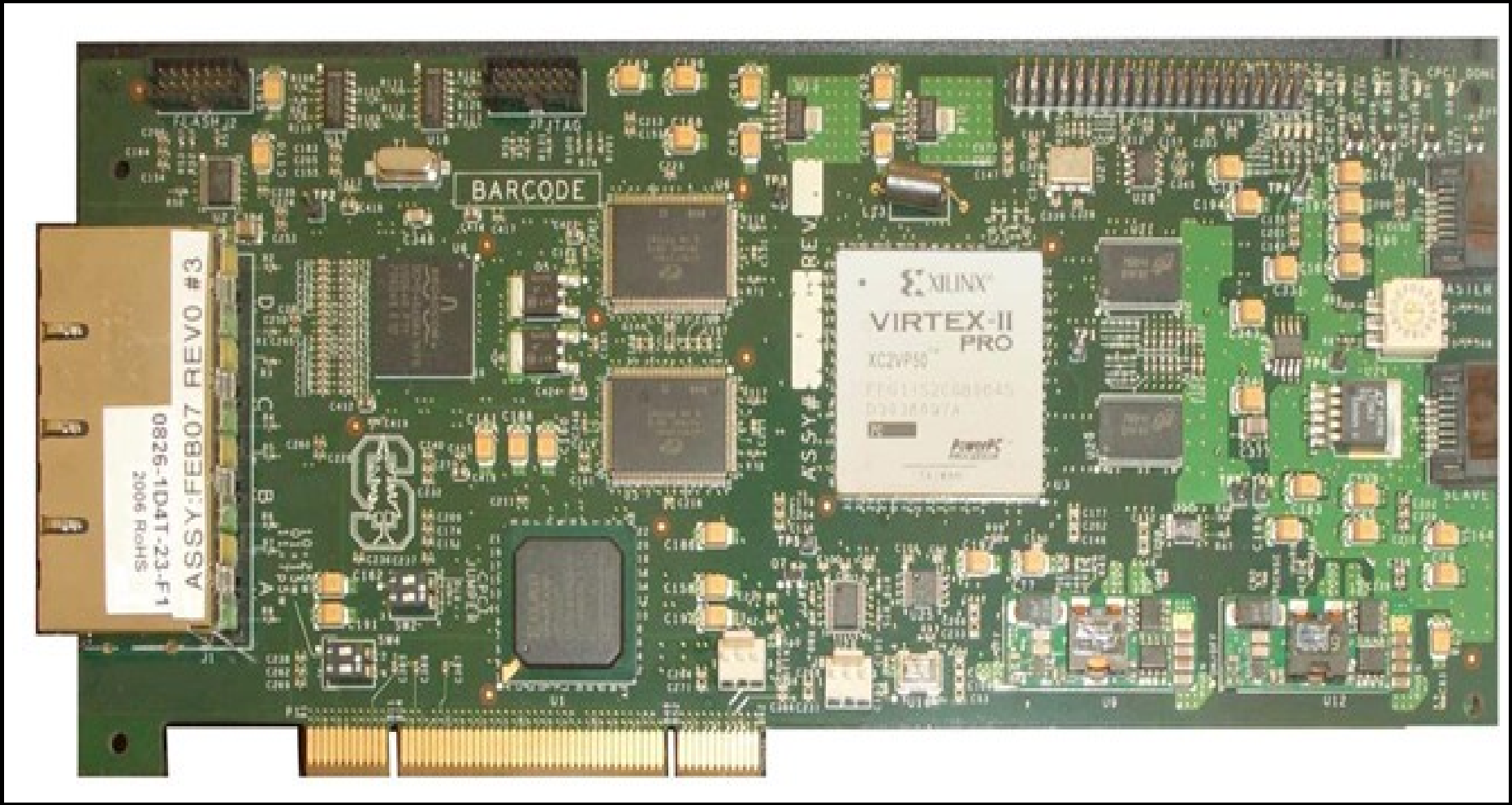
FreeBSD support for Stanford NetFPGA

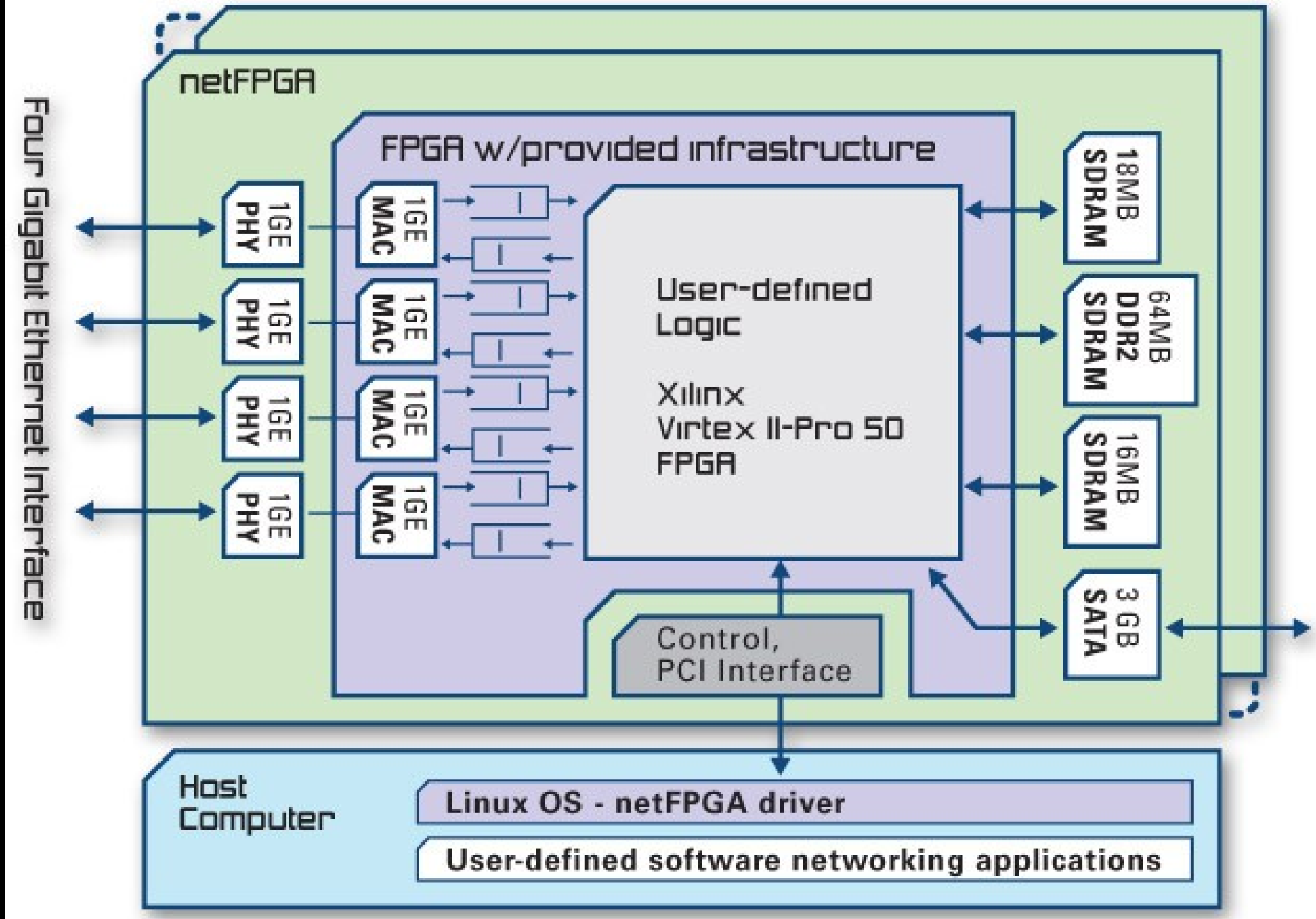
Wojciech A. Koszek
wkoszek@FreeBSD.org
2009.08.25

NetFPGA card



NetFPGA card

- 4 ports of Gigabit Ethernet
- 2 high speed, serial I/O connectors
- 64MB of DDR2 DRAM, 4.5MB of SRAM
- Fully reprogrammable without additional hardware
- PCI interface


Four Gigabit Ethernet Interface
netFPGA
FPGA w/provided infrastructure
1GE PHY
1GE PHY
1GE PHY
1GE PHY
1GE MAC
1GE MAC
1GE MAC
1GE MAC
User-defined Logic
Xilinx Virtex II-Pro 50 FPGA
18MB SDRAM
64MB DDR2 SDRAM
16MB SDRAM
3 GB SATA
Control, PCI Interface
Host Computer
Linux OS - netFPGA driver
User-defined software networking applications

NetFPGA: 3 puzzles

Bitstream: functionality provider

Userspace tools: for bitstream upload

Kernel driver: low-level glue

NetFPGA naming

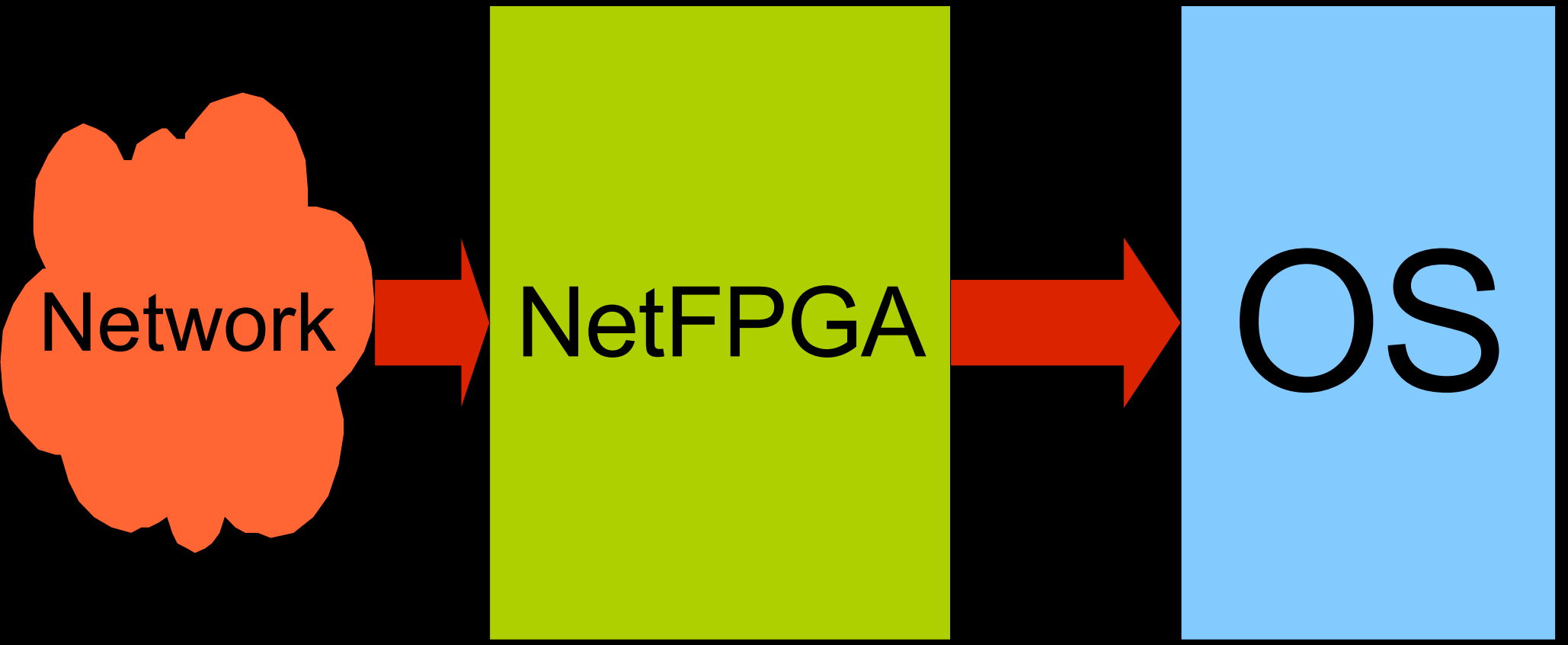
CPCI: small FPGA (Spartan2) responsible for PCI interface

CNET: BIG FPGA for Ethernet control

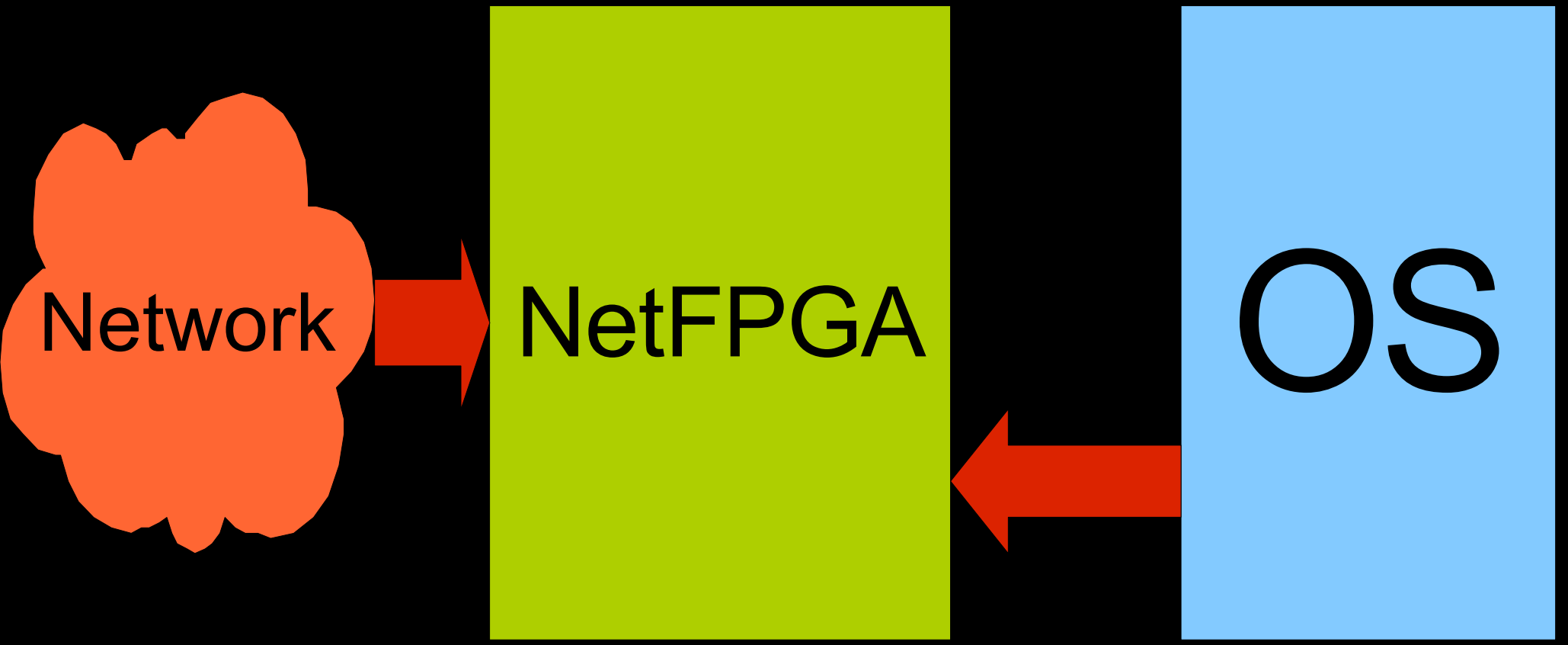
Packet transmission in the NetFPGA world

Data is being sent to the card

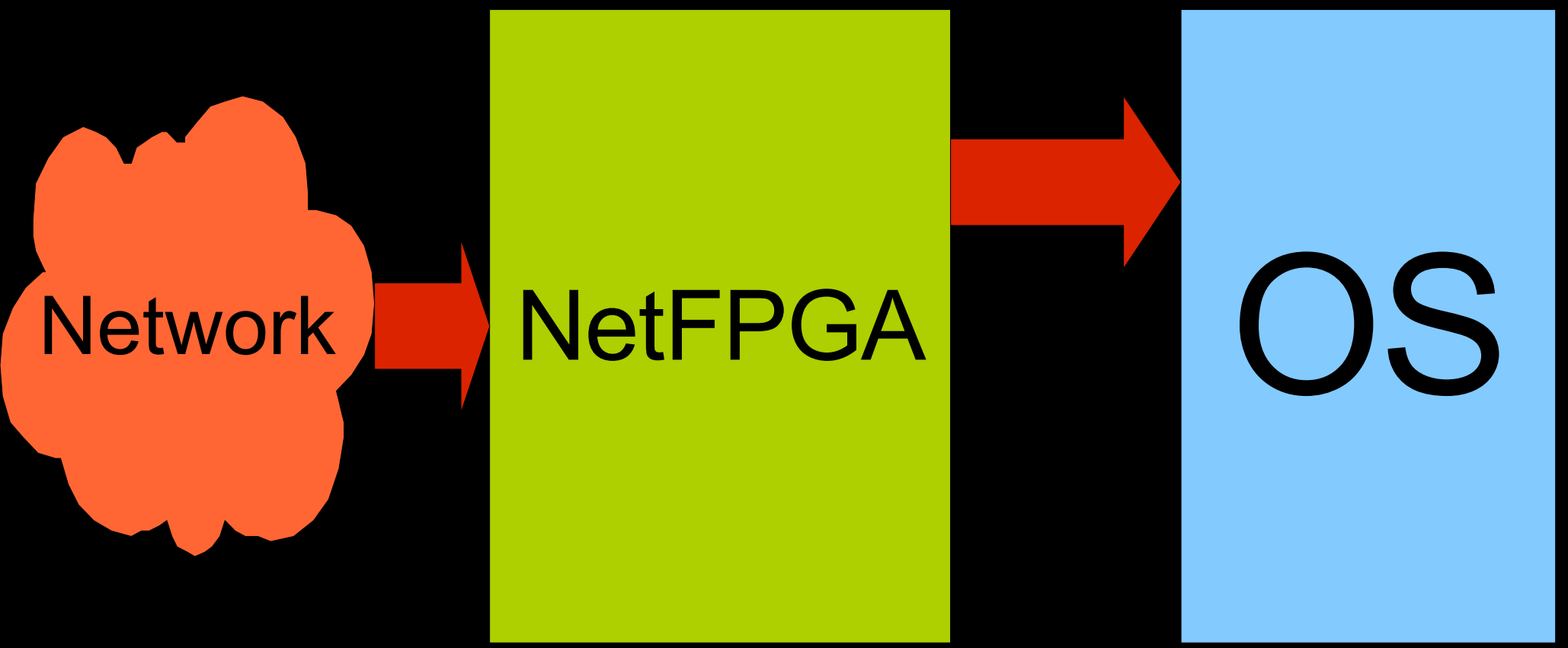
Interrupt is delivered



In order to „see”, which port has a data available, you read a register

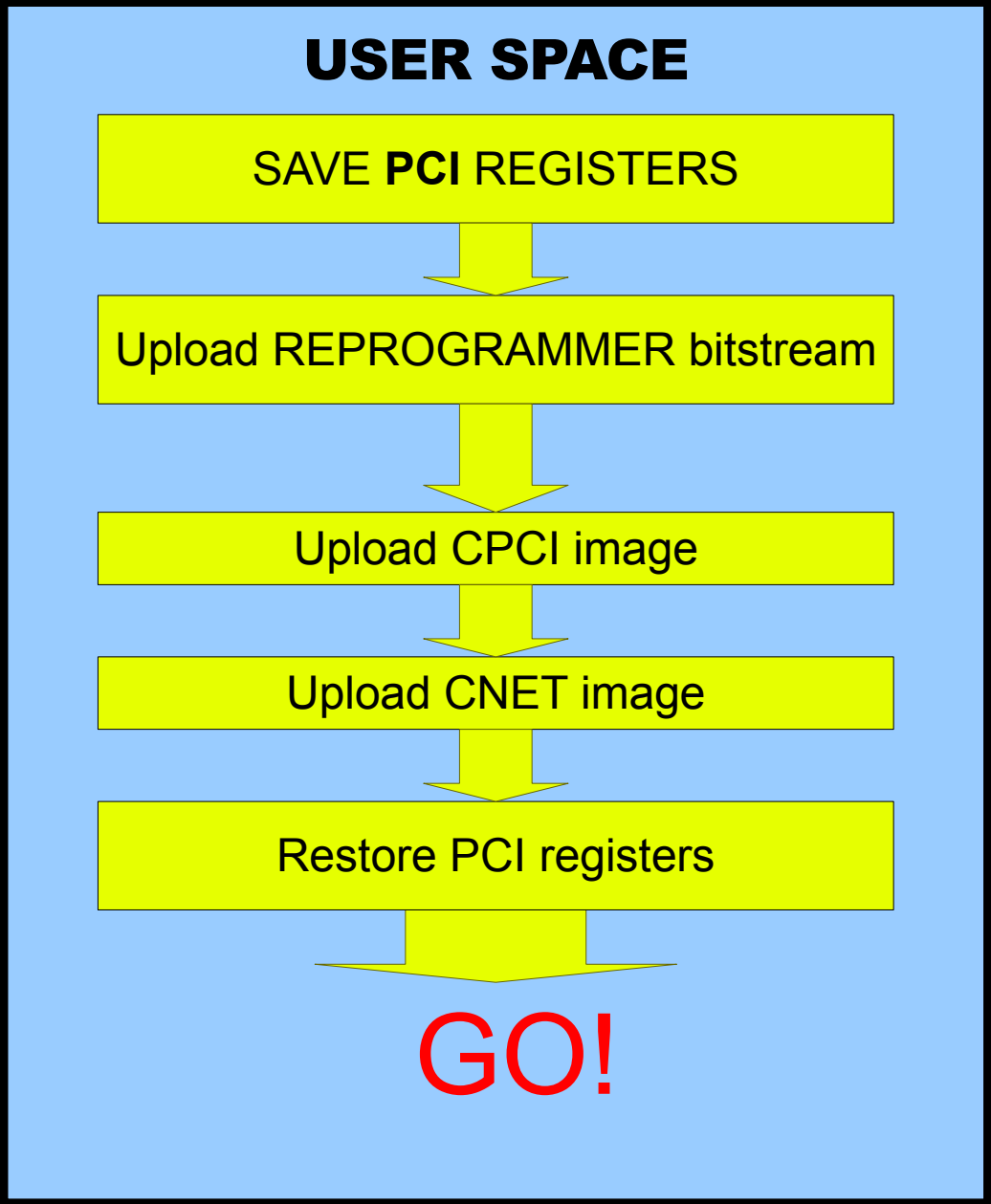


DMA transfer is started



NetFPGA programming

NetFPGA programming (Linux)



PCI registers saved from userspace :-(

Additional dependency on user-space tools

NetFPGA driver in the (Linux) kernel is monolithic..

CPCI/CNET programming is tied to Ethernet port structures

FreeBSD driver design

Driver has 2 parts:

Programming interface

Ethernet interface

Card itself appears as ``NetFPGA controller''

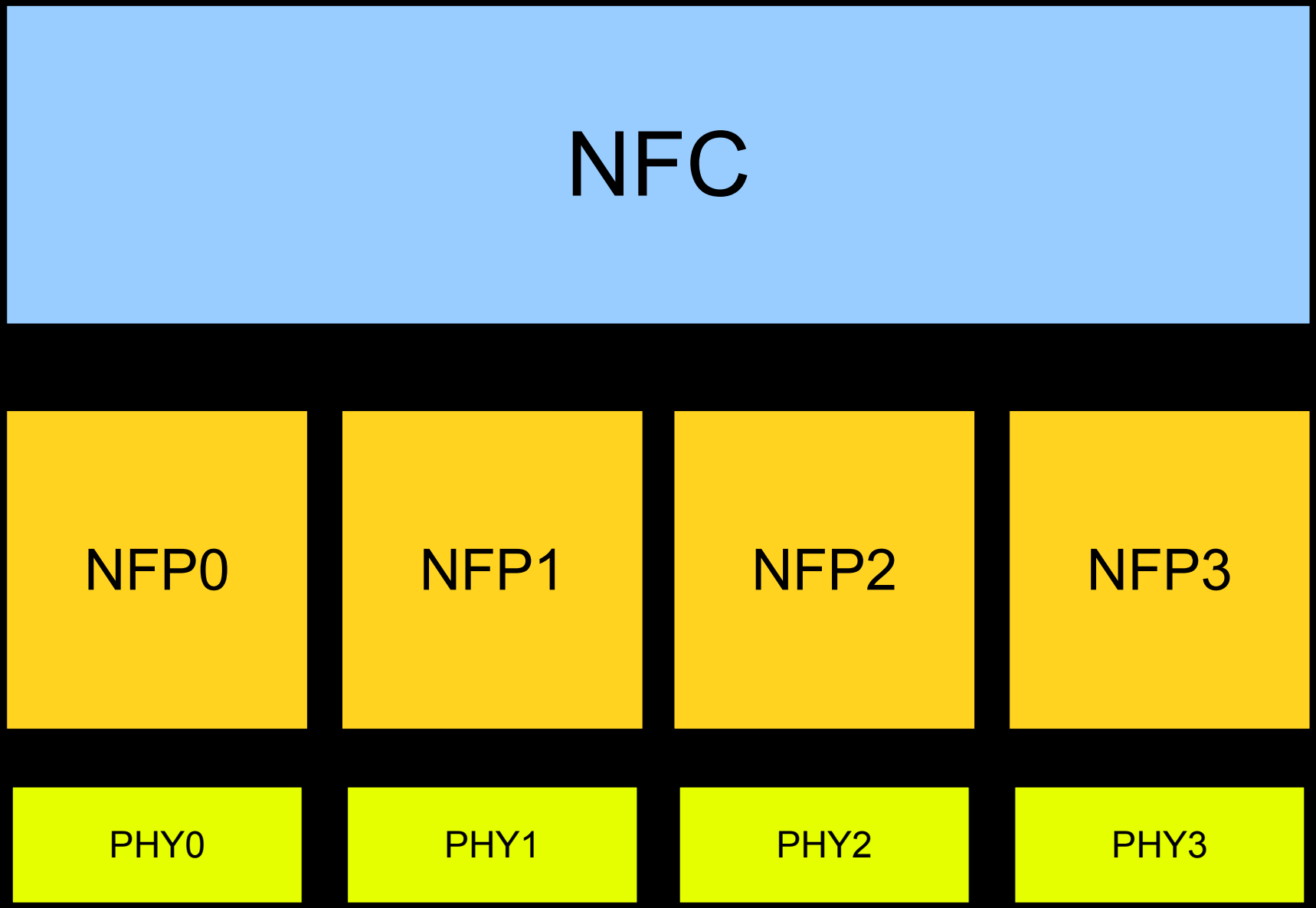
Later called (NFC)

It's up to the controller to export CPCI/CNET interface

Each NFC has 4 Ethernet gates named:

``NetFPGA ports''

Later called: (NFP)


NFC
NFP0
NFP1
NFP2
NFP3
PHY0
PHY1
PHY2
PHY3

Interface appears as separate device

/dev/netfpga

Userspace application works by opening /dev/netfpga and sending commands to it

Two level driver architecture?

It would be nice to be able to enable Ethernet interface layer only when there's Ethernet support in a bitstream

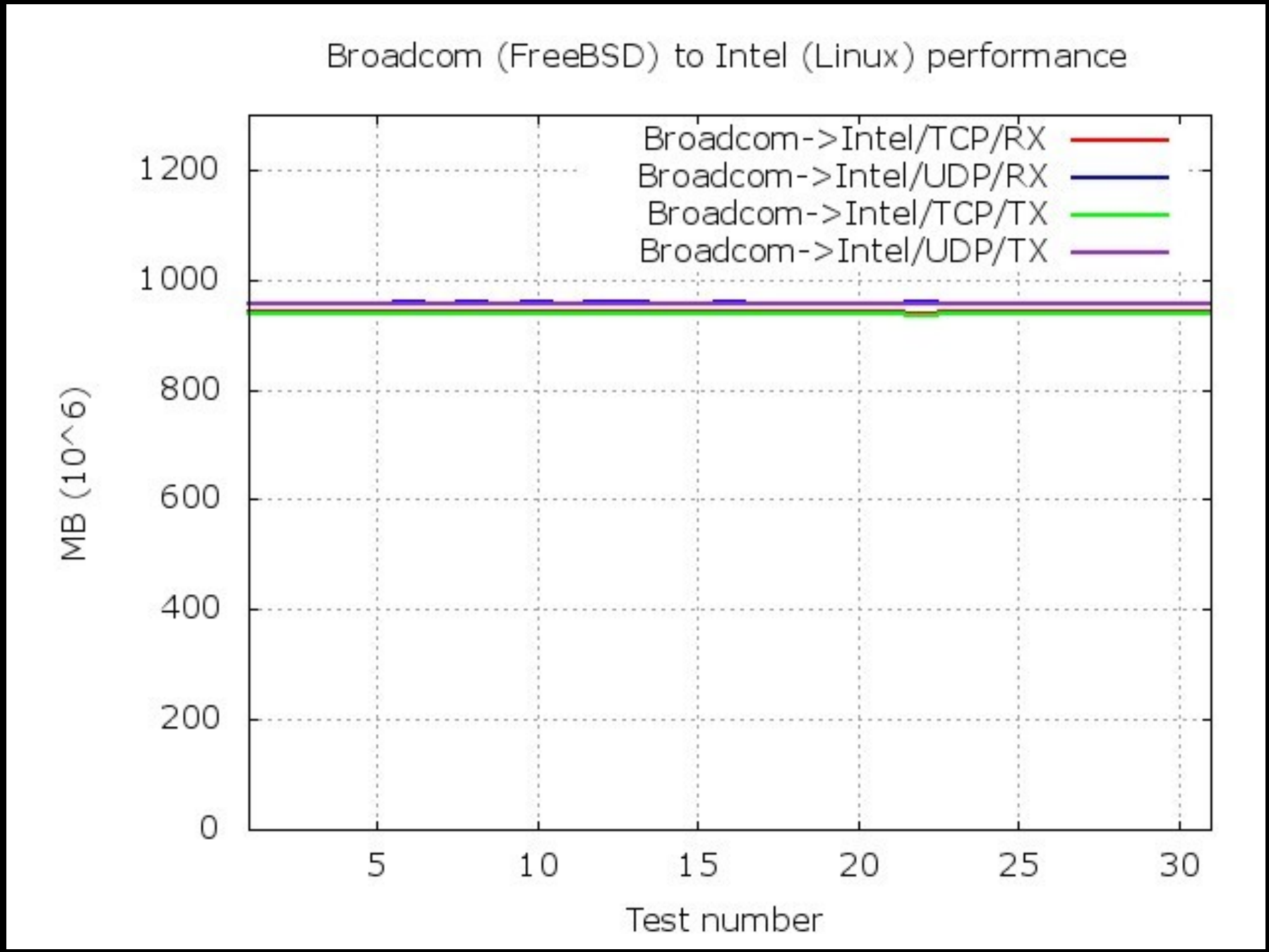
Every NFP is visible to the system as Ethernet interface

```
nf2c0: flags=8843<UP,BROADCAST,RUNNING,
       SIMPLEX,MULTICAST> metric 0 mtu 1500
       options=28<VLAN_MTU,JUMBO_MTU> ether
         00:6e:66:32:63:30
       inet 10.0.0.1 netmask 0xff000000
       broadcast  10.255.255.255 media:
       Ethernet autoselect (none)
```

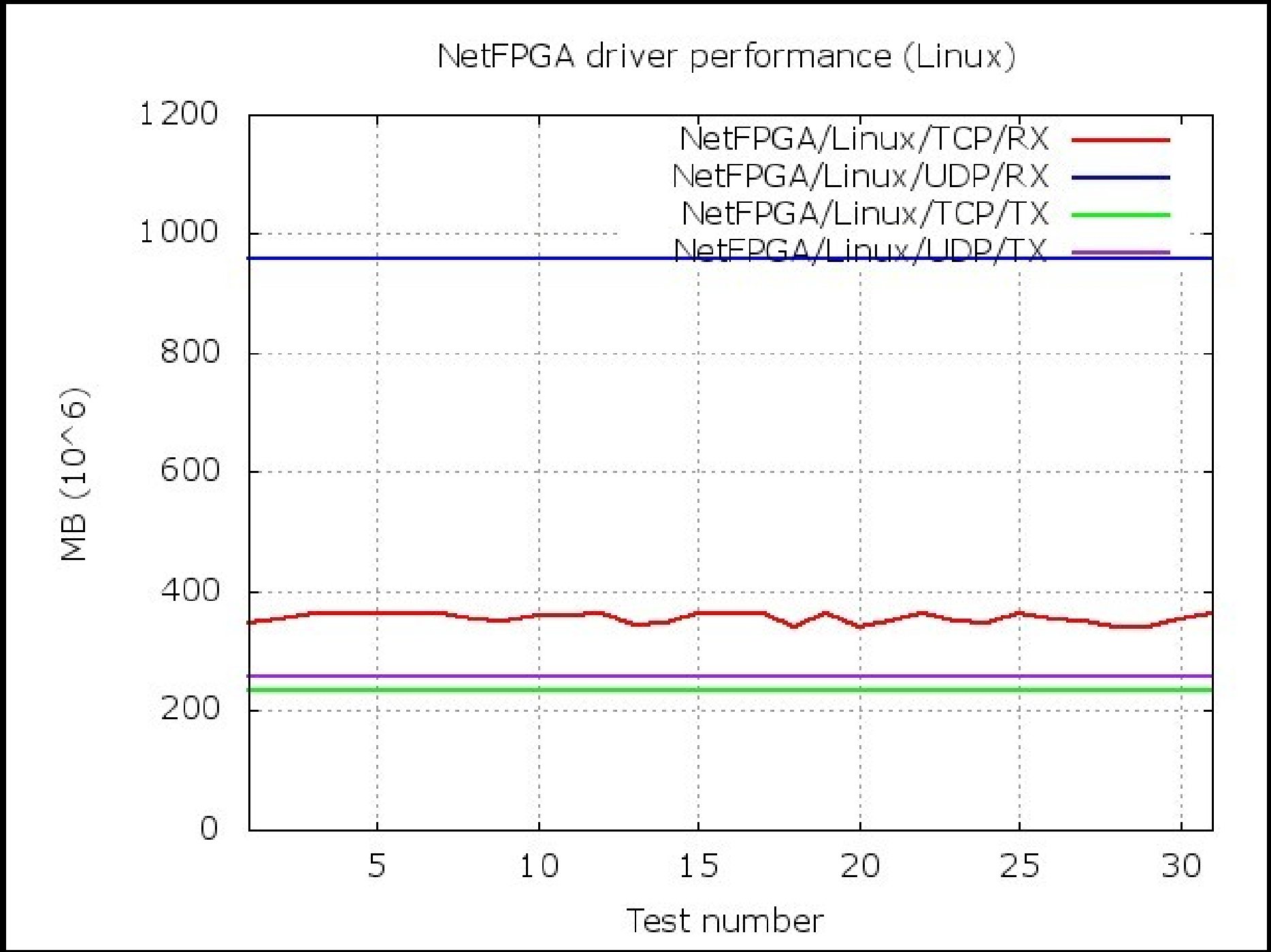
...but after driver is loaded, Ethernet interfaces always appear automatically

even if there's no bitstream

Non-NetFPGA performance



NetFPGA performance (Linux)



NetFPGA performance (FreeBSD)

NOT YET :-(

Sorry!

Performance problems..

Minimum DMA transfer is 60 bytes

I could get handling of this limitation wrong

Performance problems..

There's no access to hardware-assisted RX/TX of packets

I could get software mitigation of typical primitives wrong.

Problems (1)

When to start Ethernet ``ifnet'' layer in the kernel?

Right now you don't know if the „4 Gigabit Ethernet reference design” has been loaded or not

Problems (2)

Problems with new register system made it impossible to work with new NetFPGA release

We used 1.2.5 release

Problems (3)

Reset of the PHY chip seems to take some „undeterministic amount of time”.

Correct reset should be done each time Ethernet interface is brought up.

Plans

- Bring NetFPGA support to the FreeBSD source code base
  - Work on stability
  - Work on PERFORMANCE
    - Being better than Linux would be nice!

Project's repository

```
svn co
https://vcs.hiit.fi/svn/psirp/s
rc/netfpga/projects/freebsd_net
fpga
```

This is still Work-In-Progress!

Special THANKS...

- Pekka Nikander
- Jussi Kangasharju
- Bengt Sahlin

This presentation will be available on:

http://FreeBSD.czest.pl/~wkoszek/netfpga/

The End

Wojciech A. Koszek
wkoszek@FreeBSD.org
2009.08.25