WR Switch and Node Architecture

Training material

Maciej Lipinski
WR Collaboration / CERN

WR Devices

WR switch

www.ohwr.org/project/white-rabbit/wikis/Switch

WR node

- Simple PCIe FMC carrier (SPEC)

- Simple VME FMC carrier (SVEC)

- PCI eXtensions for Instr. (PXI) module

- FPGA Mezzanine Card (FMC) module

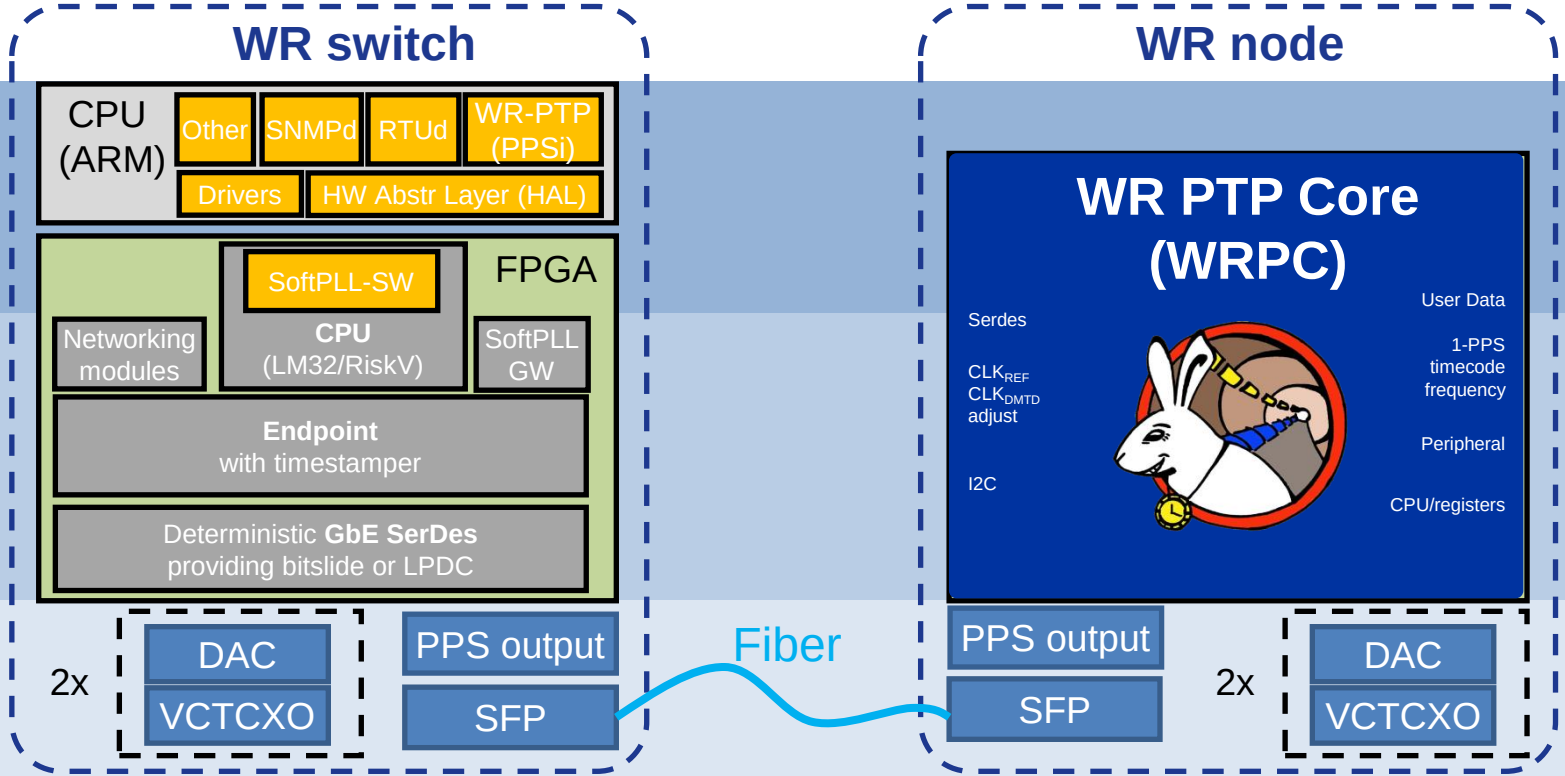
- CompactRIO White Rabbit

- Mini-WR to WR-enable carrier

- AMC FMC Carrier (AFC) compliant with MTCA.4

www.ohwr.org/project/white-rabbit/wikis/Node

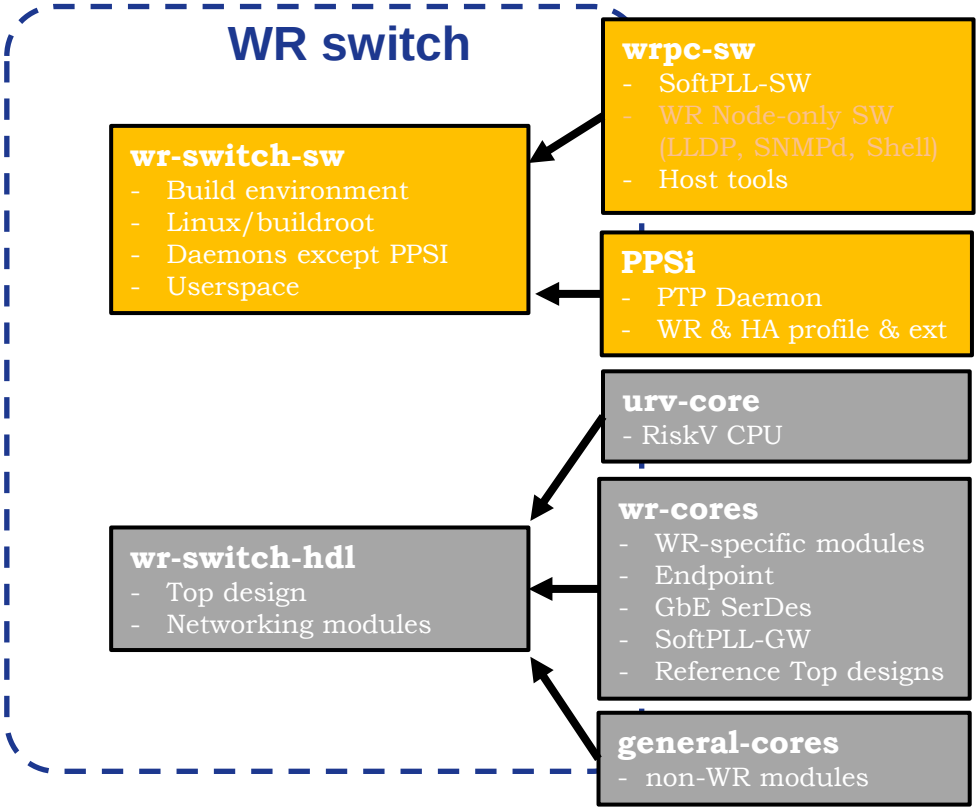
WR Devices – architecture



www.ohwr.org/project/white-rabbit/wikis/Switch

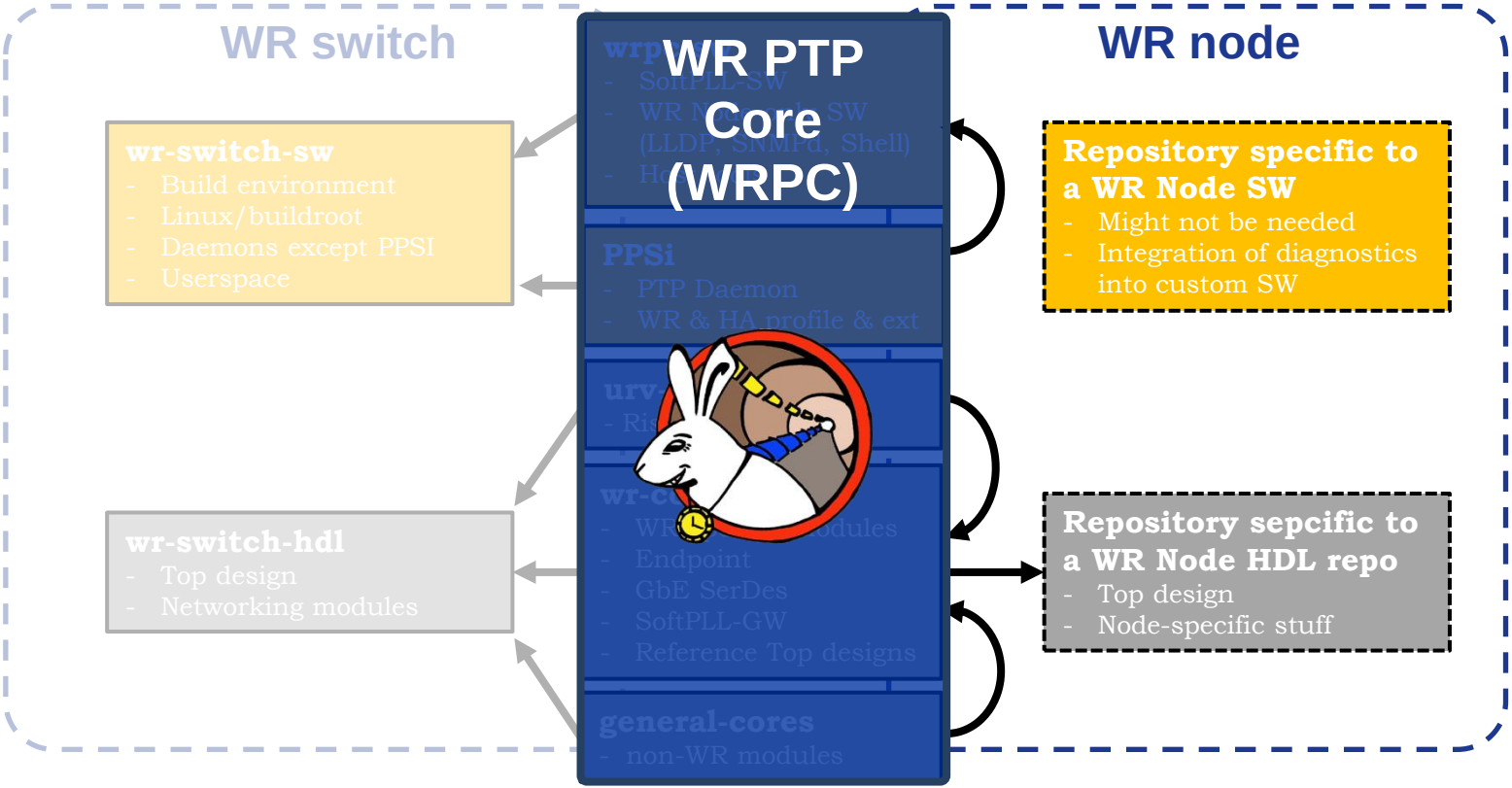
www.ohwr.org/project/white-rabbit/wikis/Node

Repositories: WR Switch



To get to the repo, add prefix https://ohwr.org/project/, e.g., https://ohwr.org/project/wr-cores

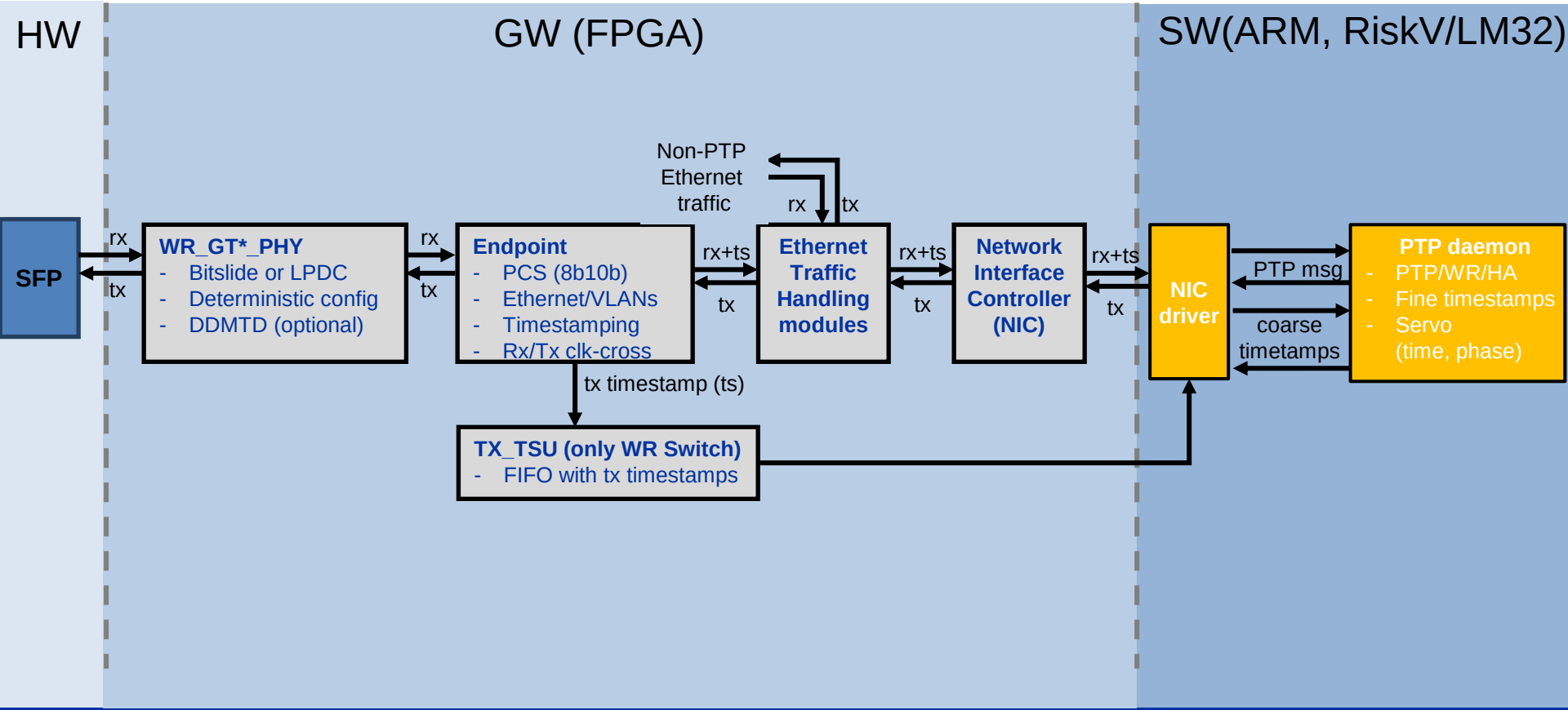
Repositories: WR Node



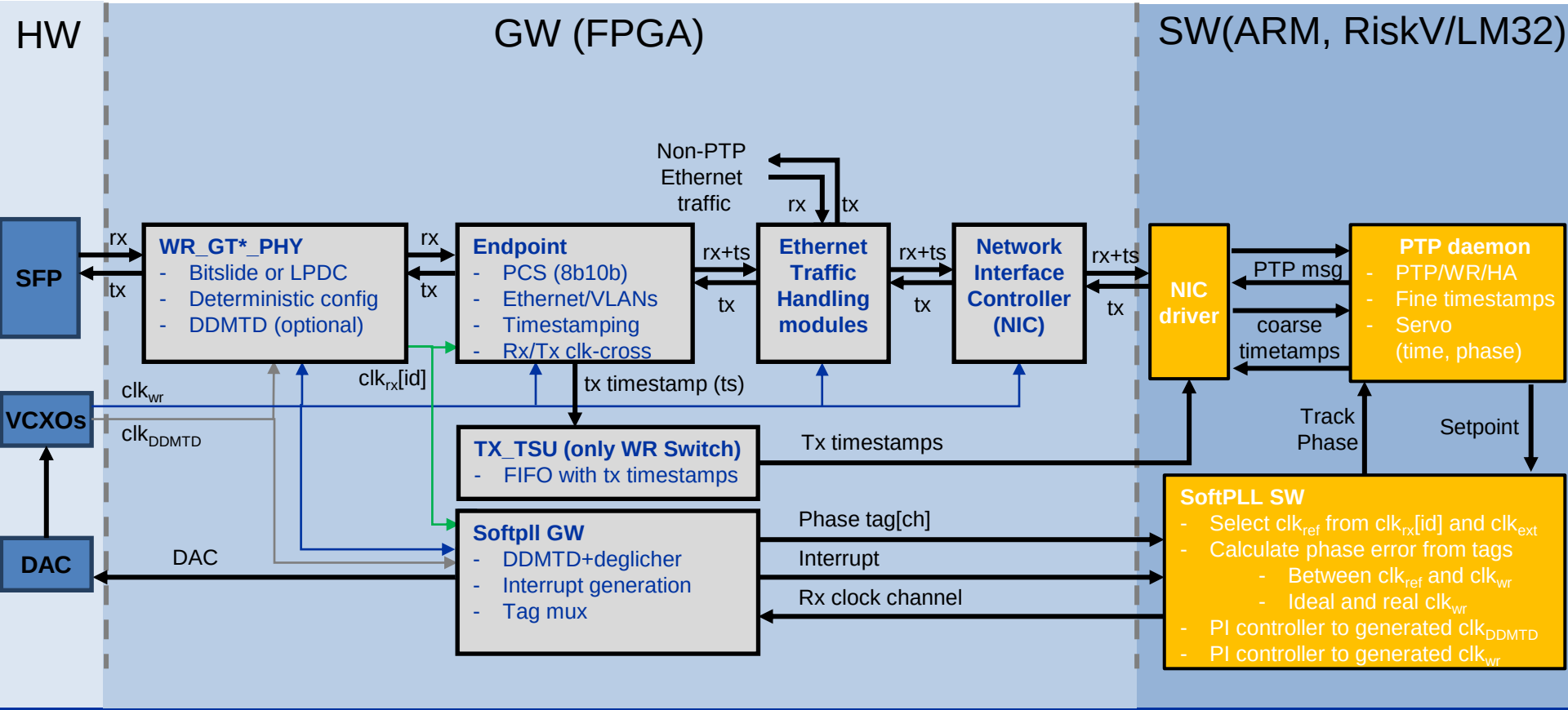
To get to the repo, add prefix https://ohwr.org/project/, e.g., https://ohwr.org/project/wr-cores

WR Timing Architecture of WR Switch and WR Node

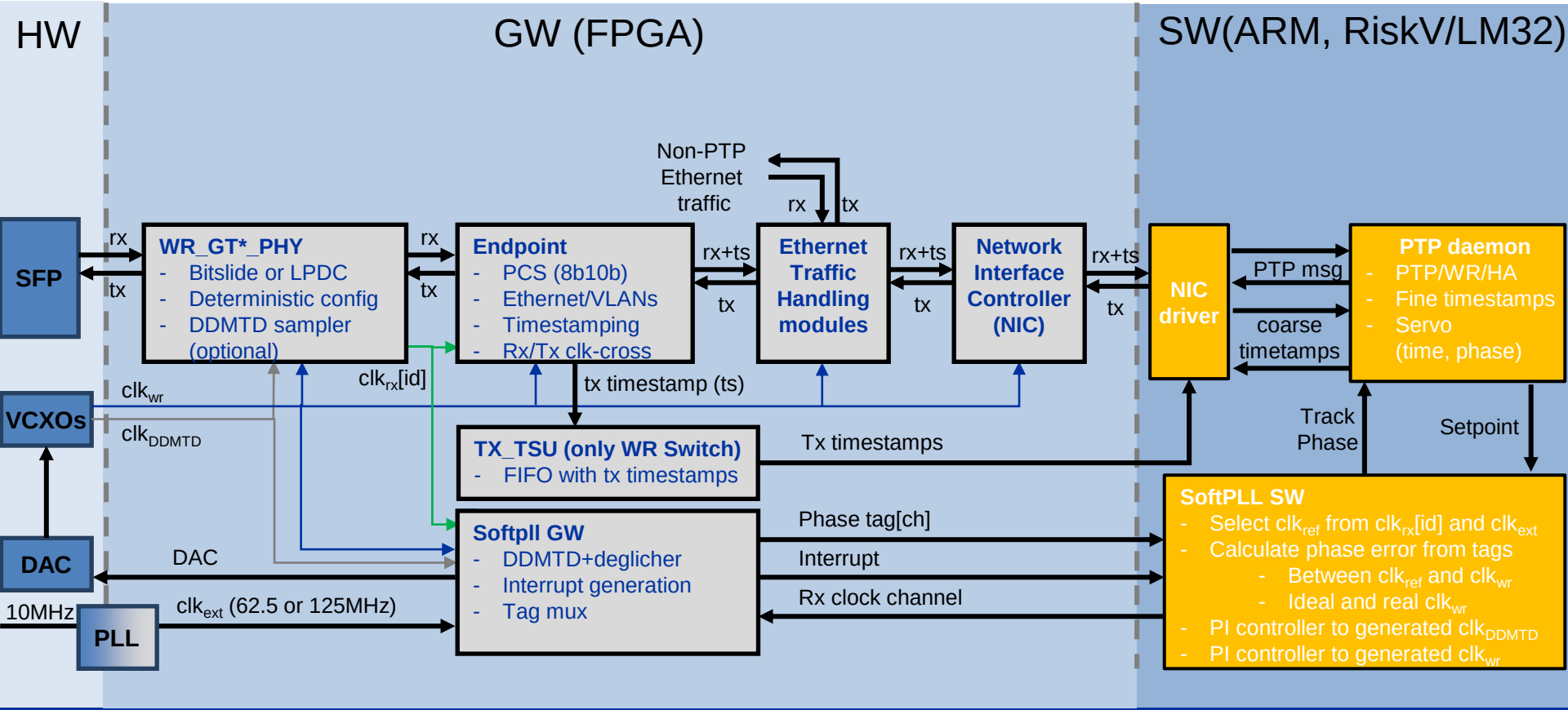
WR Timing Architecture



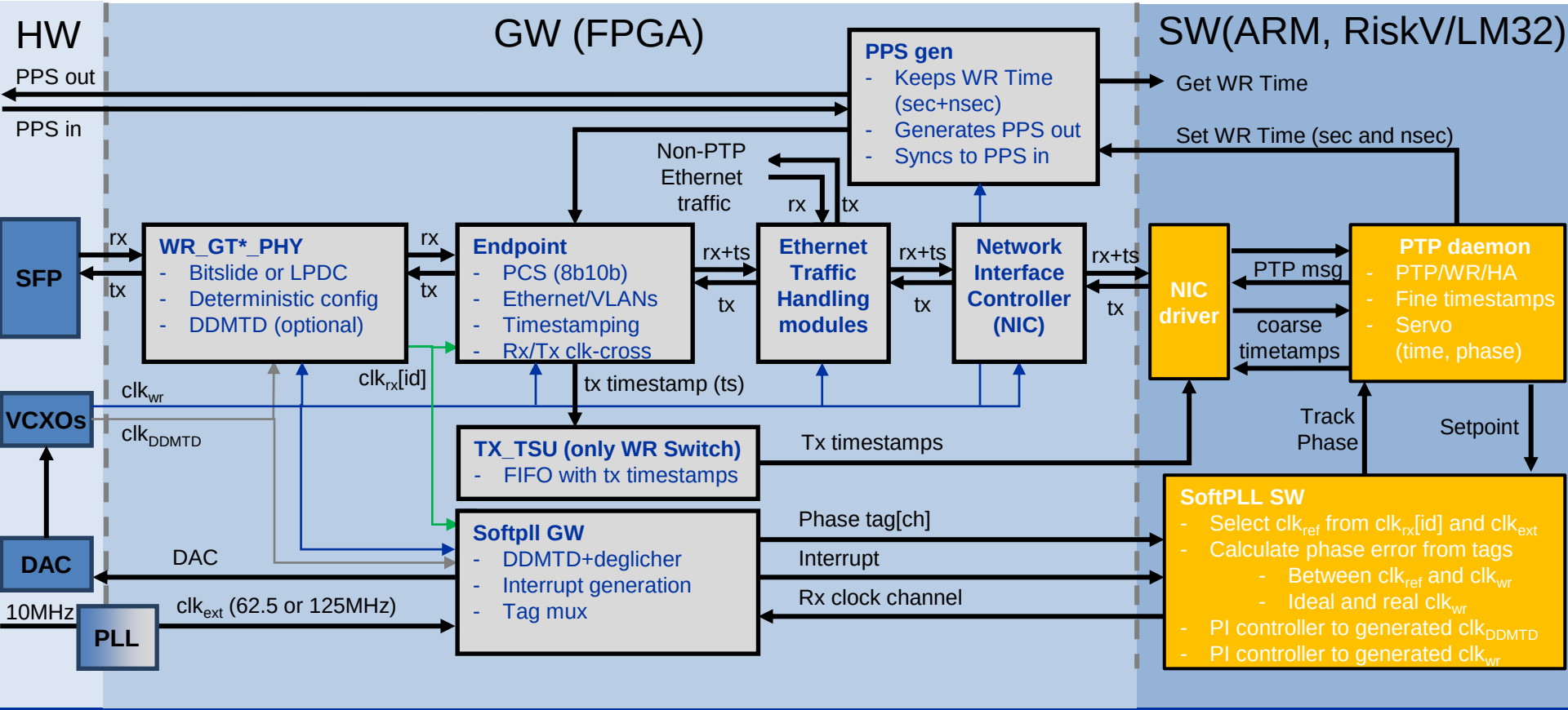
WR Timing Architecture



WR Timing Architecture

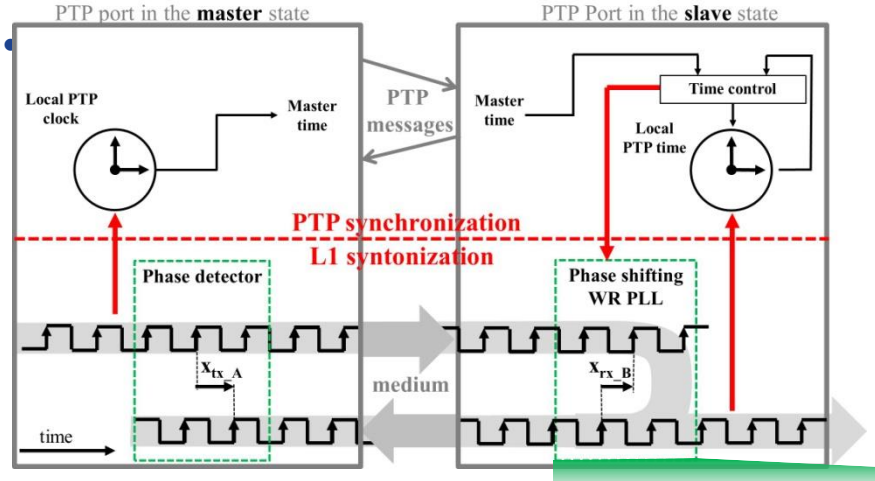


WR Timing Architecture



DDMTD and SoftPLL

DDMTD

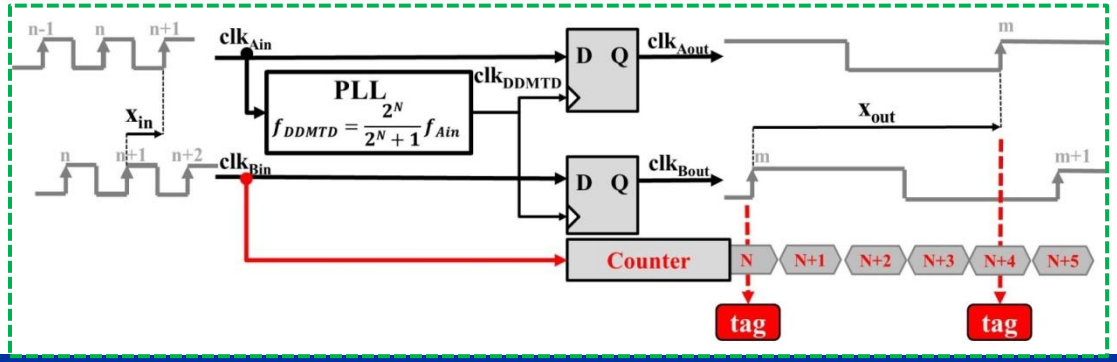


Digital Dual Mixer Time Difference (DDMTD)

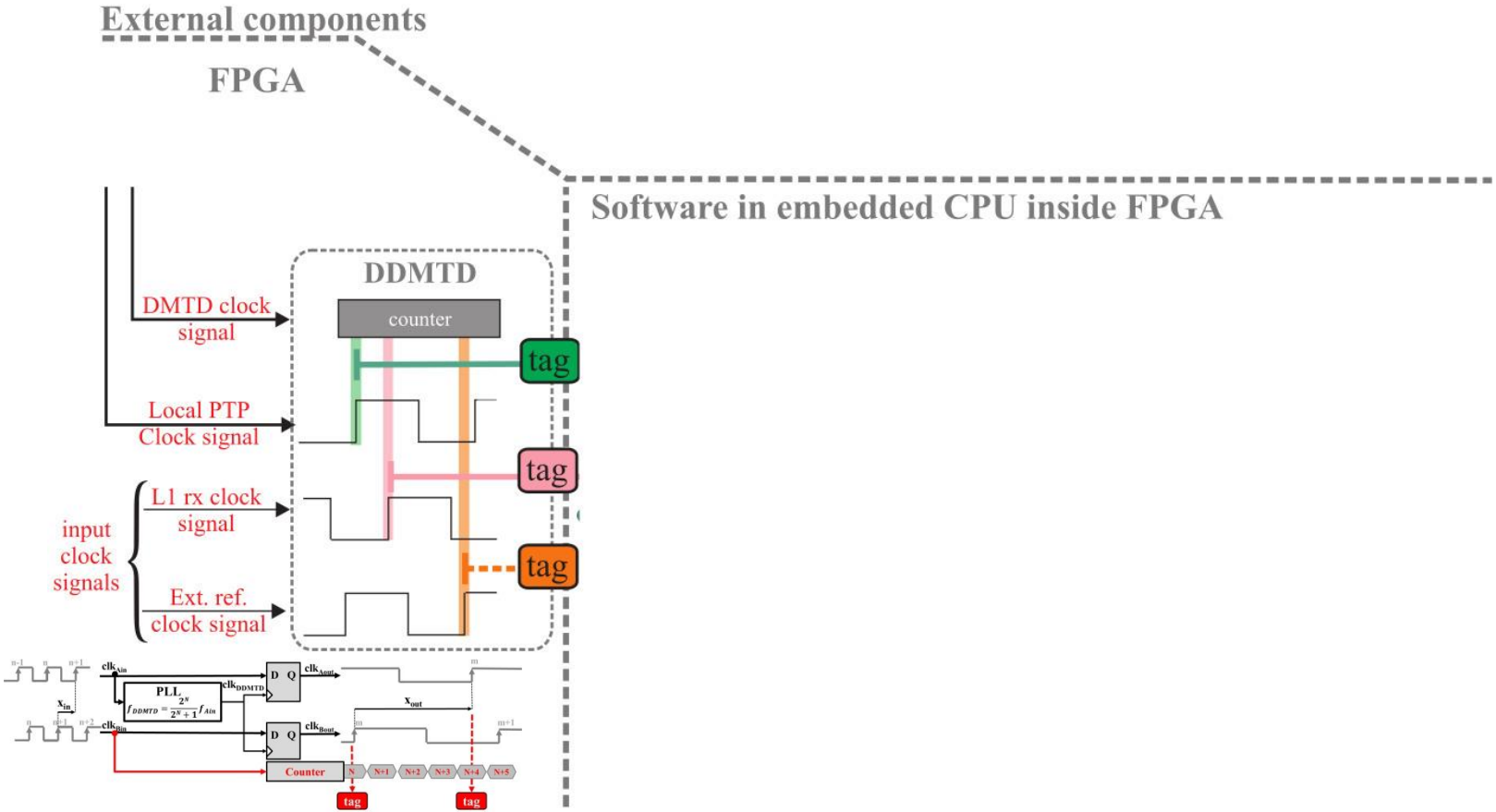
- Precise phase measurements in FPGA
- Used on Grandmaster to lock to external clock
- Used on GM and Master to enhance timestamps
- Used on Slave for phase-shifting PLL
- WR parameters:
  - clk in = 62.5 MHz
  - clk DDMTD = 62.496185 MHz (N=14)
  - clk out = 3.814 kHz
- Theoretical resolution of 0.977 ps

Local WR Clock signal

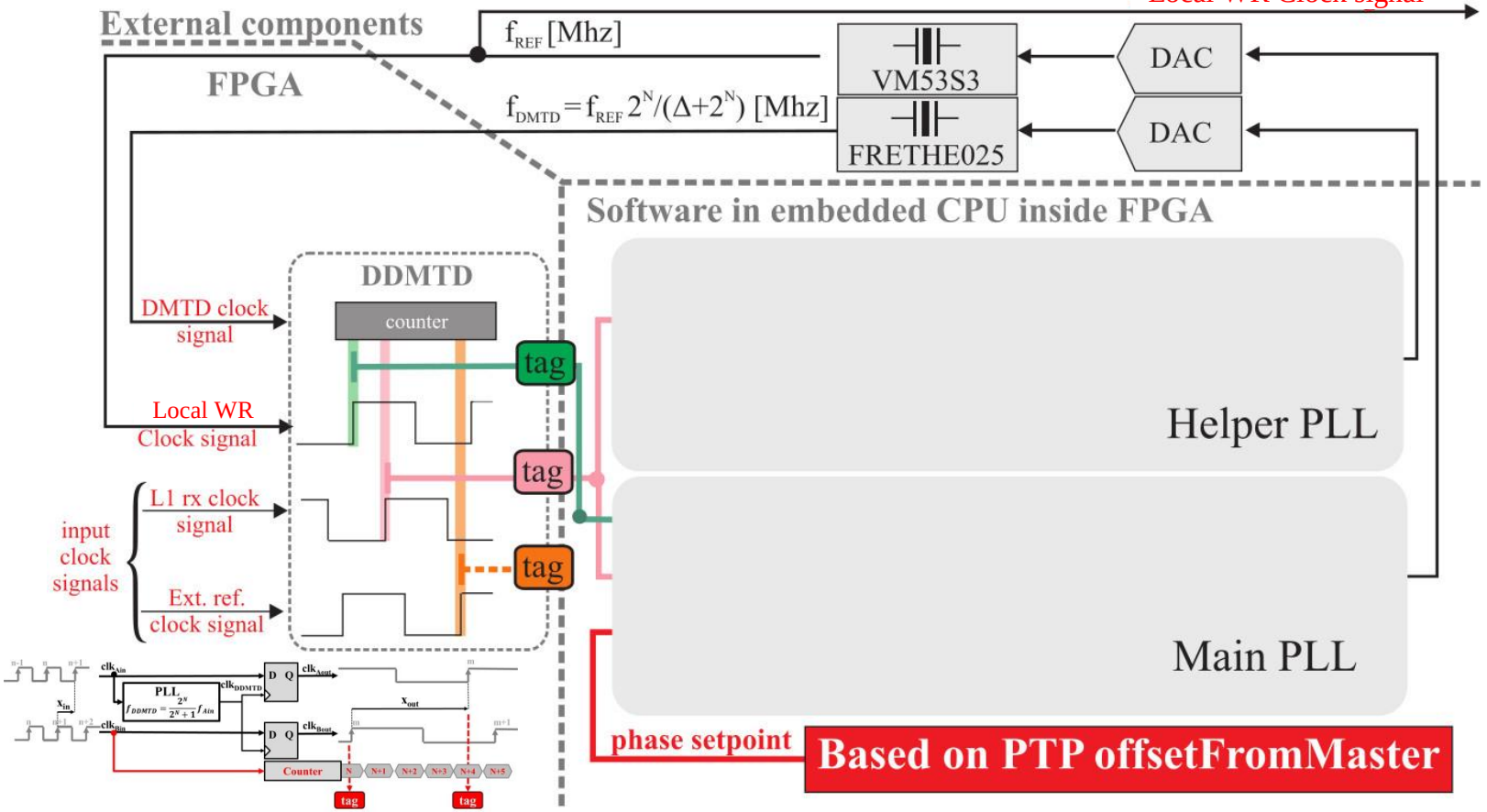
L1 rx clock signal



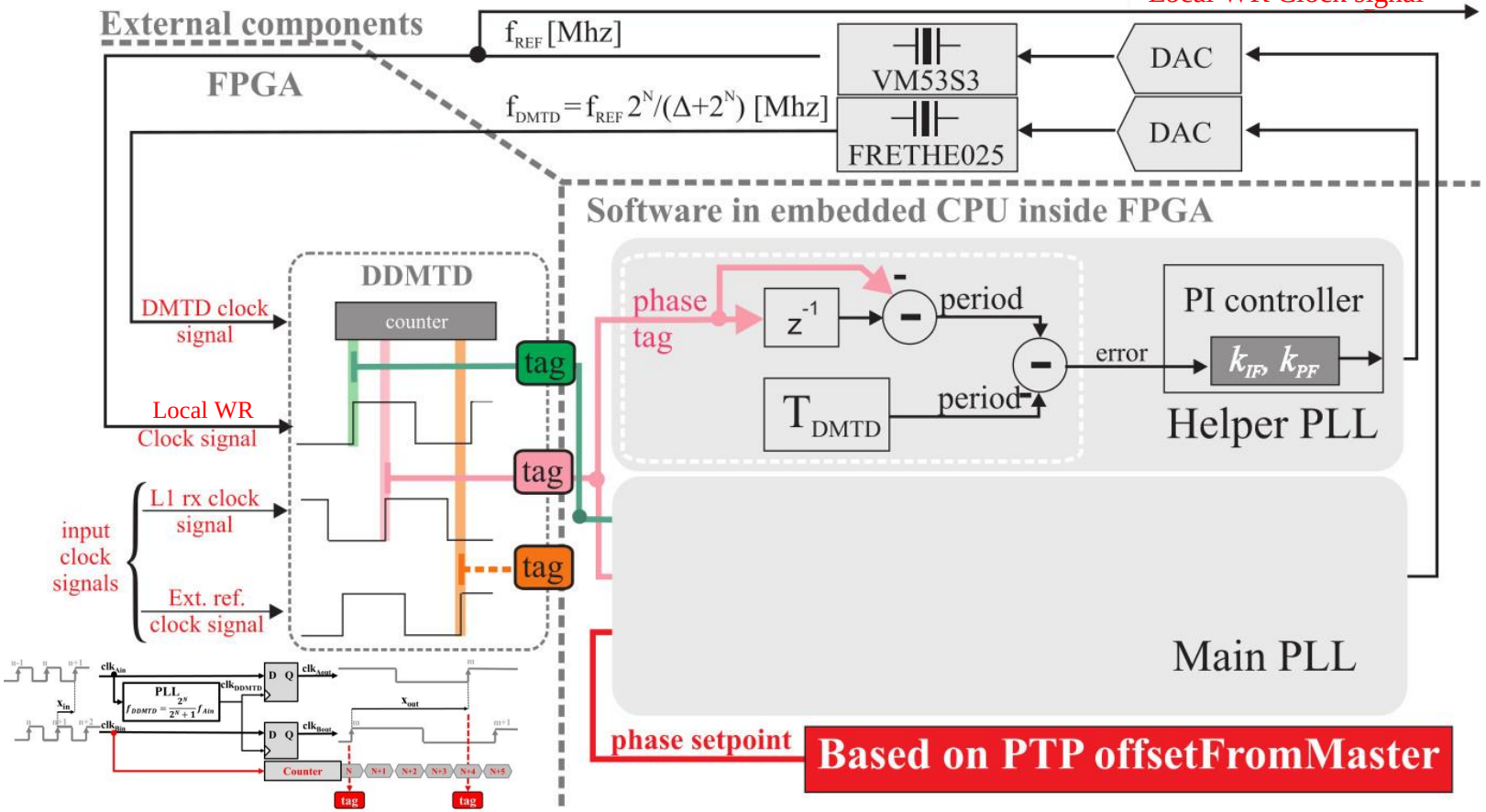
SoftPLL architecture



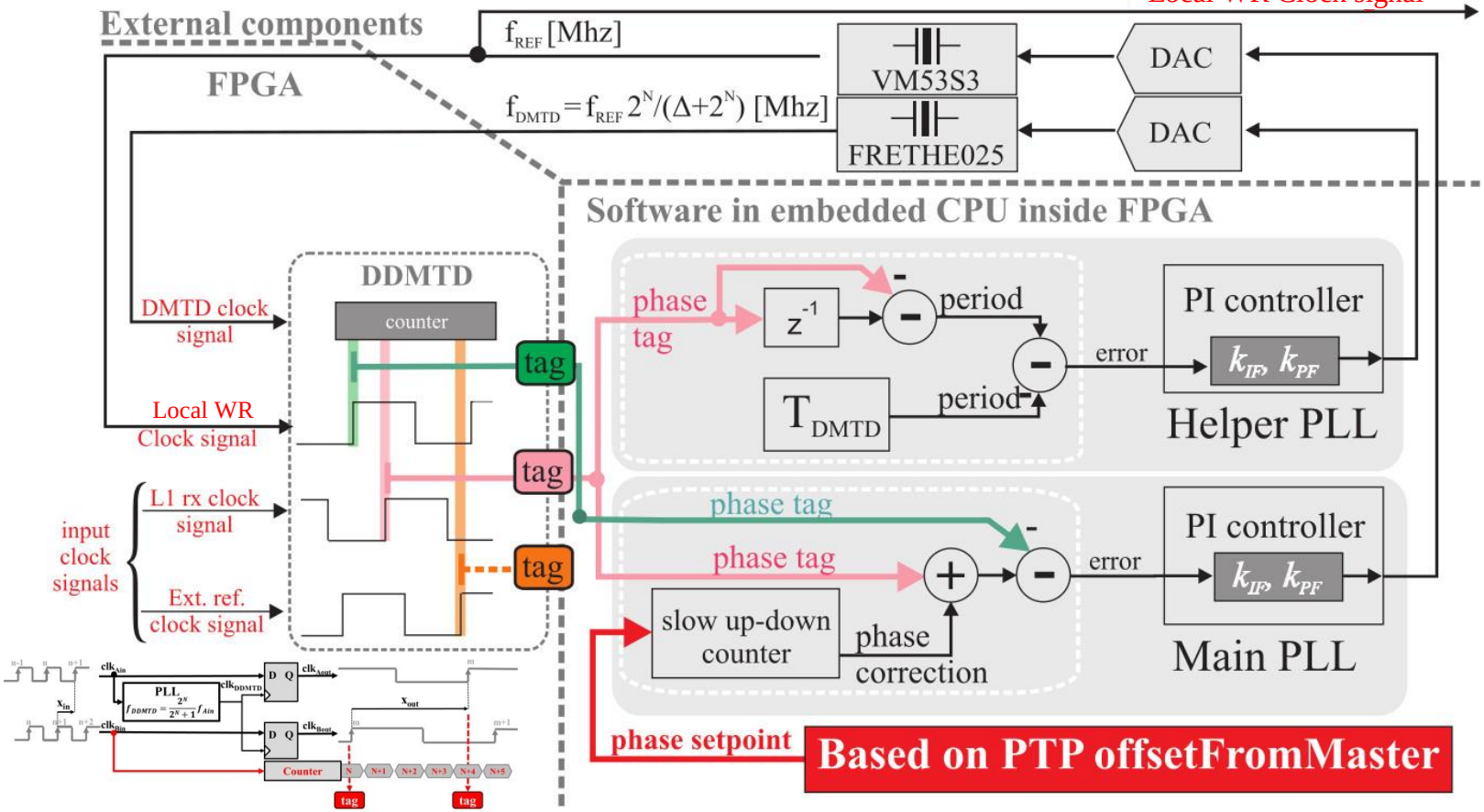
SoftPLL architecture



SoftPLL architecture

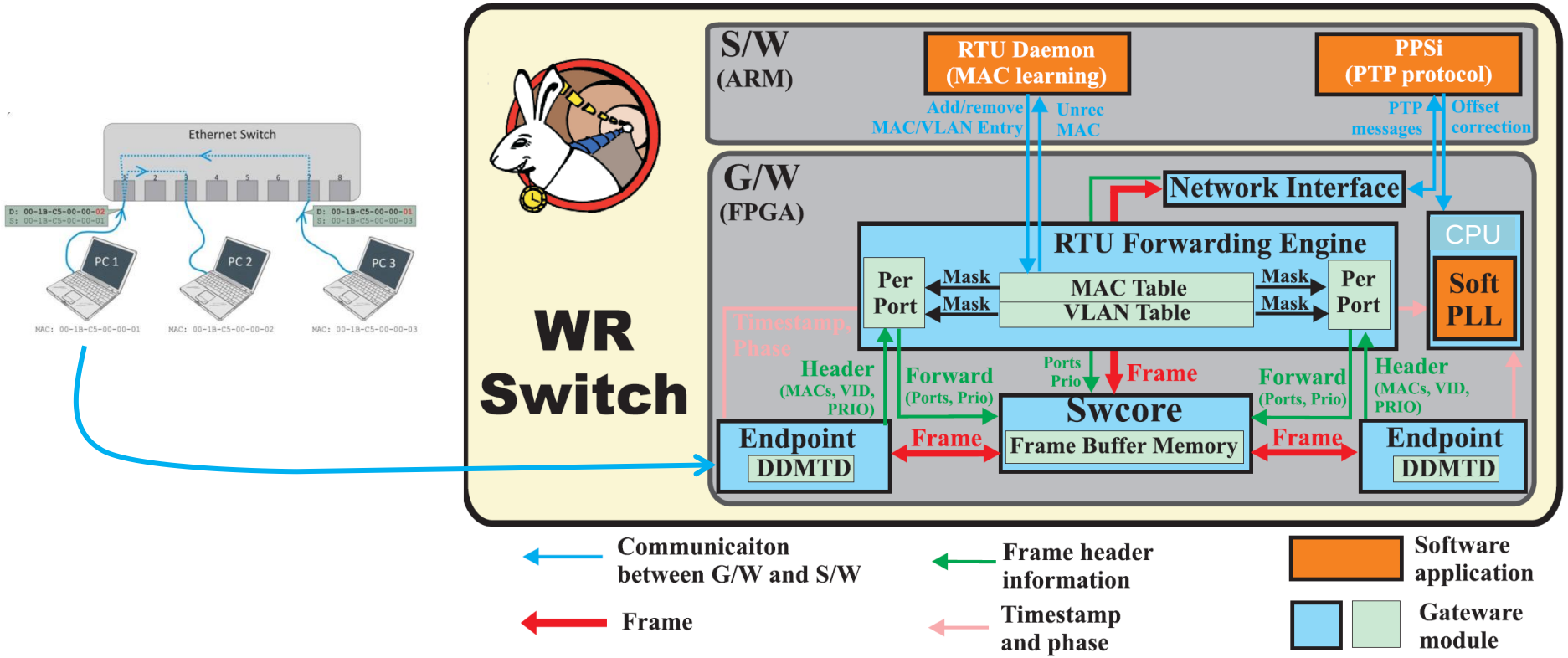


SoftPLL architecture



Data Forwarding Architecture of WR Switch

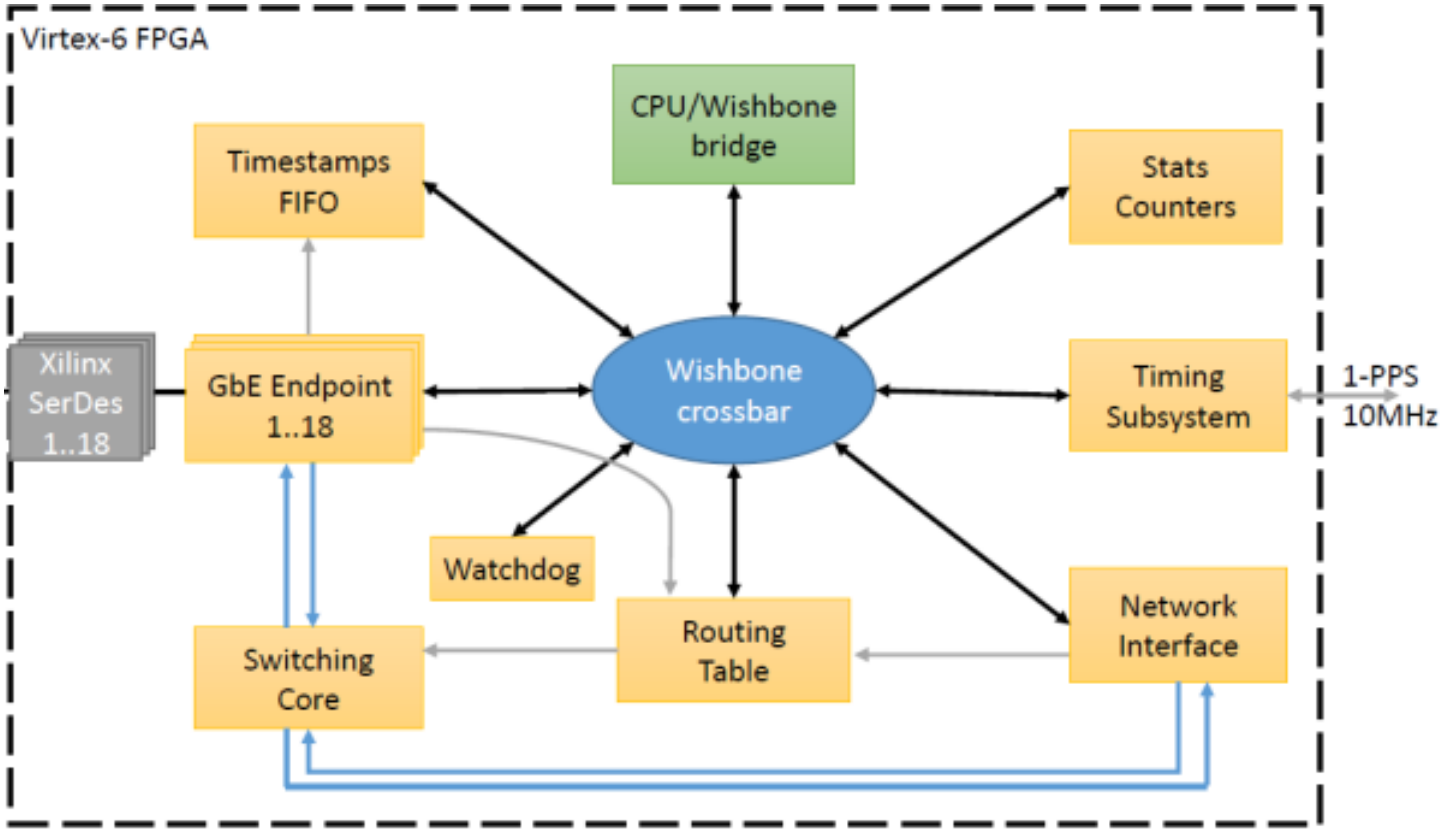
WR Switch - Data Forwarding Architecture



Source: M. Lipinski, Methods to Increase Reliability and Ensure Determinism in a White Rabbit Network, WUT PhD, Warsaw, Poland 2016

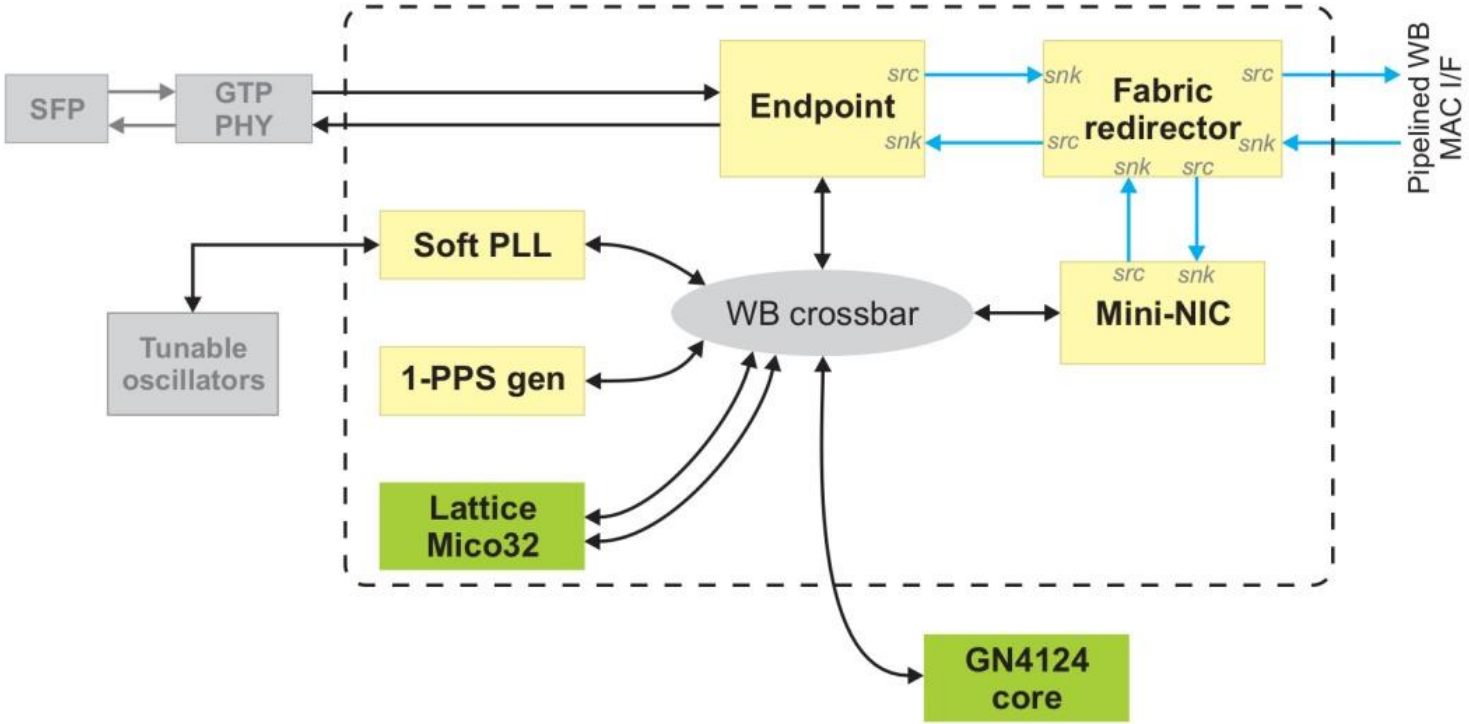
Interconnections of HDL modules in the WR Switch

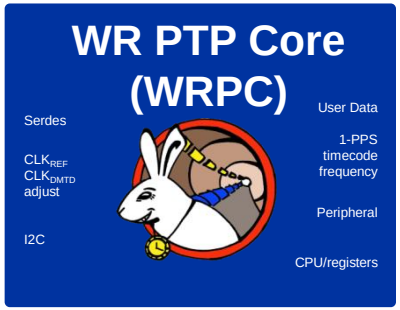
Interconnections of HDL modules



Interconnections of HDL modules in the WR Node

WR Node – HDL modules and their communication





Software environment of the WR Switch

Compilation of software

- Repository: https://ohwr.org/project/wr-switch-sw
- Developers Manual: link
- Compiling all software - execute **wrs_build-all,** ideally:

```
git clone git://ohwr.org/white-rabbit/wr-switch-sw.git
git submodule init
git submodule update
/path/to/wr-switch-sw/build/wrs_build-all 2>&1 | tee logfile \ | grep "^20..-..-.. ..:“
```

- Compiling your own SW to work on WRS

```
BRX=/home/a/wr/wrs/clean/build/build/
LINUX=$BRX/linux-3.16.38/
CROSS_COMPILE=$BRX/buildroot-2016.02/output/host/usr/bin/arm-buildroot-linux-gnueabi-
make
```

Summary

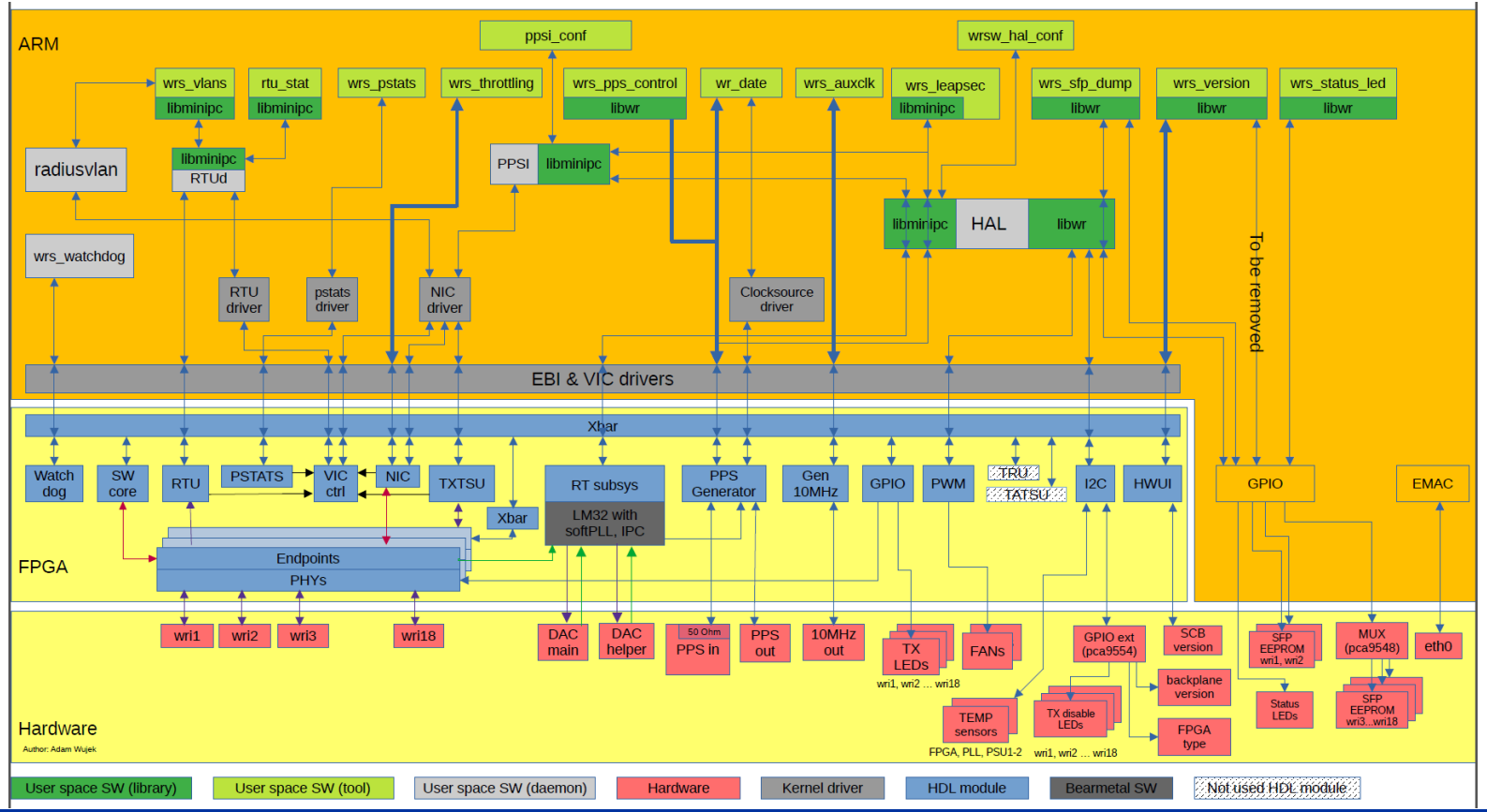
- WR Device Architecture and repositories
- WR Timing Architecture in the WR Switch and WR Node
- DDMTD and SoftPLL
- Data Forwarding Architecture on the WR Switch
- Compilation errors

NOTE: See backup slides for figures with detailed:

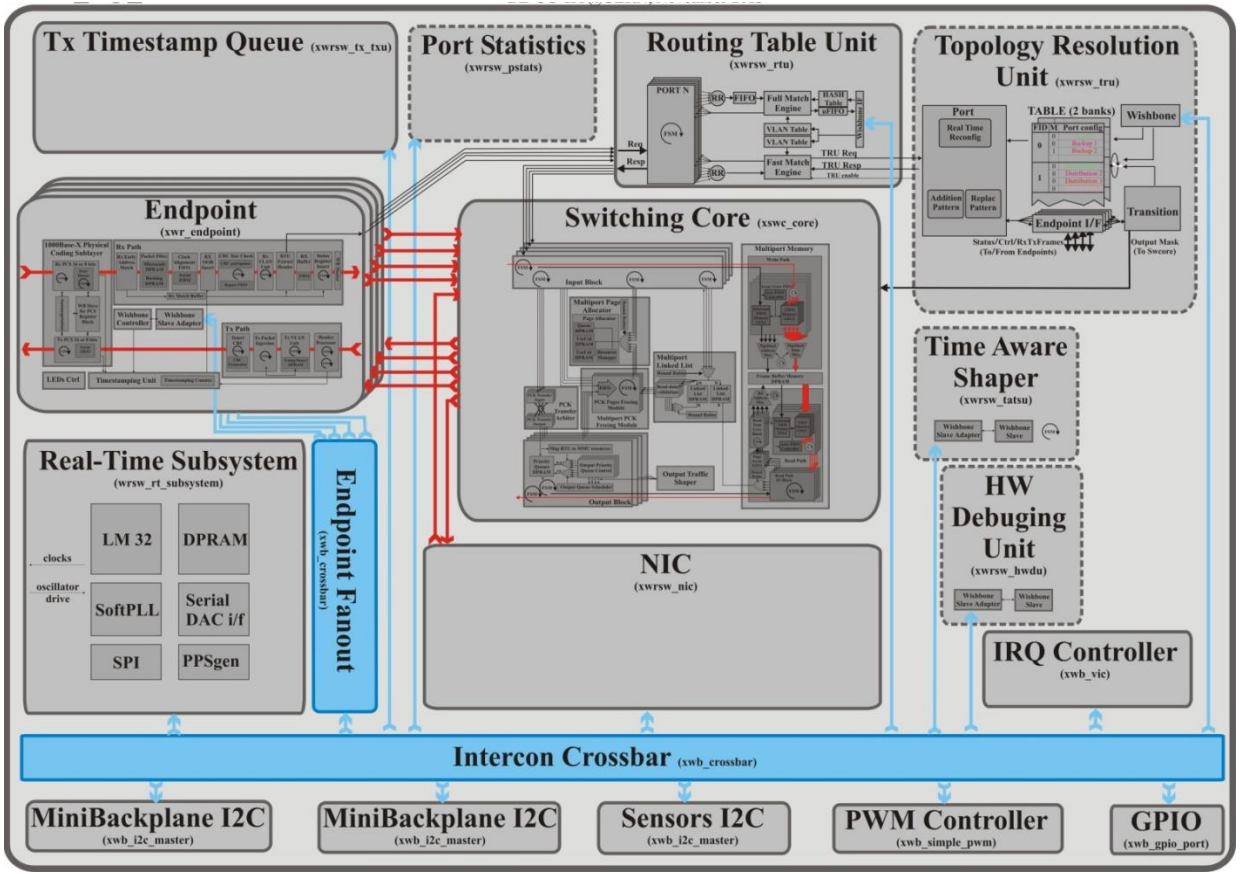
- WR Switch HW-GW-SW interactions
- WR Switch – detailed HDL modules design
- Endpoint Block Diagram

Backup slides

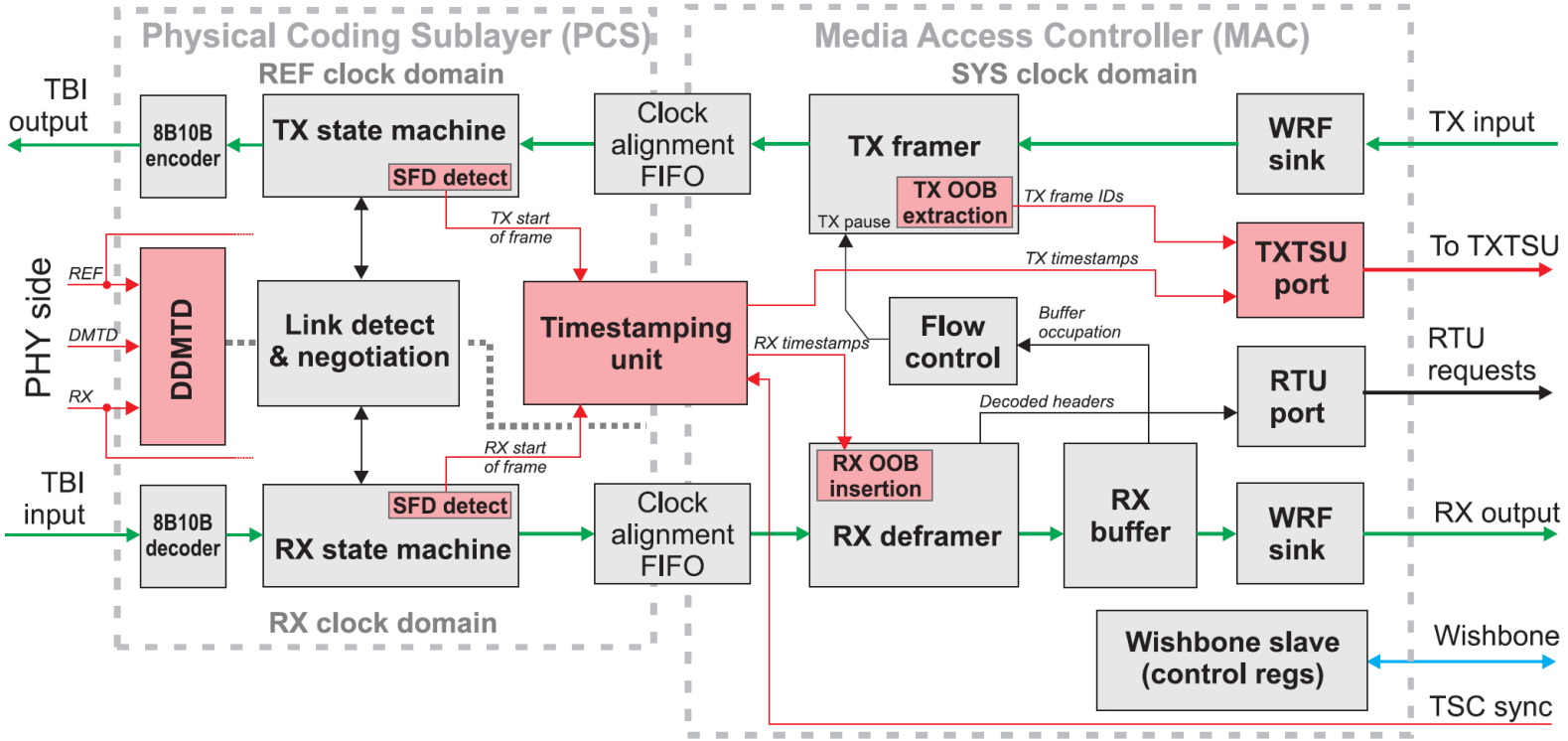
WR Switch HW-GW-SW interactions



WR Switch – detailed HDL modules design



Endpoint Block Diagram



Source: T. Włostowski. *Precise time and frequency transfer in a White Rabbit network.* WUT Master's thesis, Warsaw, Poland, May 2011.