wr-nic software: status and plans

Alessandro Rubini
Independent consultant
Working for CERN "hardware and timing" group
The WR NIC project

The hardware demo is made up of:

- Fosaelml aseadarat ds;
- Koea asras;t asiopaa
- Idasr asl as as
- Oas aasjkhlsa asasrjal
- Tasr slkmla taklat hkathak
- Bboairalm mtlkam iatathuh
- Ilklhtl h alj ajshtahaa;ljah
- Eakjrakl a tjklaj alk jal
Software is included in spec-sw package

To simplify the task for installers:

- Everything is included in a single sw package
- Full documentation for developers is included
- 7S supplies an extra manual for non-developers
The NIC can send and receive Ethernet frames.

It can timestamp frames with an 8ns resolution.

```
spusa.root# stamp-frame wr1
stamp-frame: Using interface wr1
timestamp  T1:  1476.381349032
timestamp  T2:  1476.381403352
timestamp  T3:  1476.391563248
timestamp  T4:  1476.391617560
round trip time:  0.000108632
forward time:     0.000054320
backward time:    0.000054312
```
Software Features: DIO

The hardware is made up of two PCBs

The NIC is spec-specific, not dio-specific

- Use of netdevice ioctl commands
  - PRIV_MEZZANINE_ID to identify the FMC
  - PRIV_MEZZANINE_CMD to actually run stuff
- As of today, no identification is there, yet

Old good command line interface

```bash
# Pulse channel 4 for 0.1 seconds now
wr-dio-cmd wr0 pulse 4 .1 now
# Pulse for 10 microseconds in the middle of the next second
wr-dio-cmd wr0 pulse 4 .00001 +1.5
# Pulse for 1ms at 17:00 today
   wr-dio-cmd wr0 pulse 4 .001 $(date +%s --date 17:00)
# Make a train of 5 pulses, 0.5ms wide, every ms at next second
wr-dio-cmd wr0 stamp 4 0.0005 +1 .001 5
```
We have simpler tools too (still CLI)

tornado.root# /tmp/wr-dio-agent wr0 &
spusa.root# wr-dio-ruler wr1 IN4 L3+.001 R4+.001 R2+.001
wr-dio-ruler: configured for local channel 3, delay 0.001000000
wr-dio-ruler: configured for remote channel 4, delay 0.001000000
wr-dio-ruler: configured for remote channel 2, delay 0.001000000

[... wait a few seconds ...]
spusa.root# wr-dio-cmd wr1 stamp 3
ch 3, 385.001000000
ch 3, 386.001000000
ch 3, 387.001000000
ch 3, 388.001000000

Tornado.root# wr-dio-cmd wr0 stamp 2
ch 2, 385.001000000
ch 2, 386.001000000
ch 2, 387.001000000
ch 2, 388.001000000

Tornado.root# wr-dio-cmd wr0 stamp 4
ch 4, 385.001000000
ch 4, 386.001000000
ch 4, 387.001000000
ch 4, 388.001000000
Nothing, the wr-nic is perfect

Actually, we have the over-engineering syndrome
- Have the NIC driver as a carrier-specific thing
- Use SDB to self-detect everything
- DMA for better speed (full gigabit?)
- Better time stamps
Thank you for your attention