ZIO - The Ultimate I/O Framework

Federico Vaga (federico.vaga@gmail.com),
Alessandro Rubini (rubini@gnudd.com)

2013-02-02
What is ZIO?

ZIO is an I/O framework for Linux. Its main targets are:

- big laboratories
- industrial applications
- high performance boards
- multi-function boards (Input/Output Analog/Digital)

Federico Vaga (federico.vaga@gmail.com), Alessandro Rubini (rubini@gnudd.com)
Example Board: [CERN] FMC ADC

http://www.ohwr.org/projects/fmc-adc-100m14b4cha

- 100 MSPS 14bit 4 channels

Federico Vaga (federico.vaga@gmail.com), Alessandro Rubini (rubini@gnudd.com)
Example Board: [CERN] FMC FINE DELAY

http://www.ohwr.org/projects/fmc-delay-1ns-8cha

- TDC Resolution: 28ps Precision (std. dev): 55ps
- Delay Resolution: 10ps Precision: 100ps

Federico Vaga (federico.vaga@gmail.com), Alessandro Rubini (rubini@gnudd.com)
Features

- Both input and output
- Sharp timestamps (better than 1ns)
- Large number of peripherals (bus)
- Small overhead (currently 0.35us)
- Large data items per trigger shot
- Fully described data (allows off-line work)
- Easy device configuration (sysfs)
- Mmap support for applications
- Hot-swapping for both trigger and buffer
- Supporting kernel v2.6.24 onwards
- Well documented

Federico Vaga (federico.vaga@gmail.com), Alessandro Rubini (rubini@gnudd.com)
All data transfers are block-based

- the block fully describes an acquisition
- it contains both data and metadata (control)
The Control

This area hosts attributes for the device and for the currently active trigger.

Device and trigger are each characterized by 16 "standard" attrs and 32 "extended" attrs. A bit-mask states which attrs are active.

Each attribute is a 32-bit word

TLV record for optional extra information

Federico Vaga (federico.vaga@gmail.com), Alessandro Rubini (rubini@gnudd.com)
Modules

- **Peripheral**: it handles physical data transfers
- **Buffer**: it stores blocks
- **Trigger**: it starts/stops acquisition
The Char Device Pipeline

- write
- store_block
- push_block
- raw_io
- data_doneretr_block
- retr_block
- free_block
- data_done
- alloc_block
- free_block

Federico Vaga (federico.vaga@gmail.com), Alessandro Rubini (rubini@gnudd.com)
Demo Time

DEMO

Federico Vaga (federico.vaga@gmail.com), Alessandro Rubini (rubini@gnudd.com)
Next Features

- **Double Buffering**
  - streaming
  - no driver changes

- **Monitoring**
  - sniffing support for external application
  - no interference with actual I/O

- **Socket Interface**
  - a single fd controls several channels
  - remote devices, local drivers

Federico Vaga (federico.vaga@gmail.com), Alessandro Rubini (rubini@gnudd.com)
Links

http://www.ohwr.org/projects/zio

git://ohwr.org/misc/zio.git