

ZIO - The Ultimate I/O Framework

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

2013-02-02



What is ZIO?

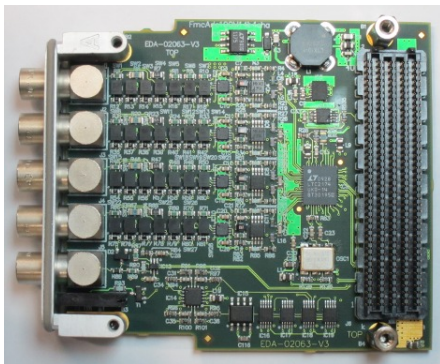
ZIO is an I/O framework for Linux. Its main targets are:

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

Example Board: [CERN] FMC ADC

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

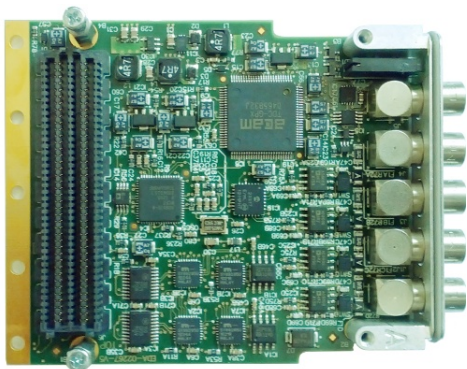
- 100 MSPS 14bit 4 channels



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



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

The Block

All data transfers are block-based

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



The block is overall blue



Control Cyan



Data Darker

The Control

0x00	V	v	A	a	sequence	nsamples	ssize	nbits
0x10	fam		type		host-identification		device-id	
0x20	cset		chan		device name			
0x30	tstamp: secs				tstamp: ticks			
0x40	tstamp: bins				mem-addr		reserved	
0x50	flags			trigger name				
0x60	<p>This area hosts attributes for the device and for the currently active trigger.</p> <p>Device and trigger are each characterized by 16 "standard" attrs and 32 "extended" attrs. A bit-mask states which attrs are active.</p> <p>Each attribute is a 32-bit word</p>							
0x1F0	TLV record for optional extra information							

All Characters in ZIO



Luser Lemon



Fops Forest



Socket Salmon



Buffer Brown



Trigger Tomato



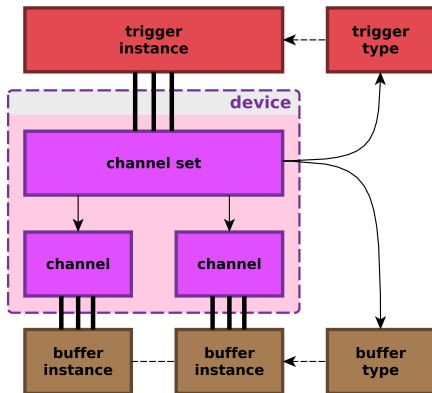
Periph. Purple



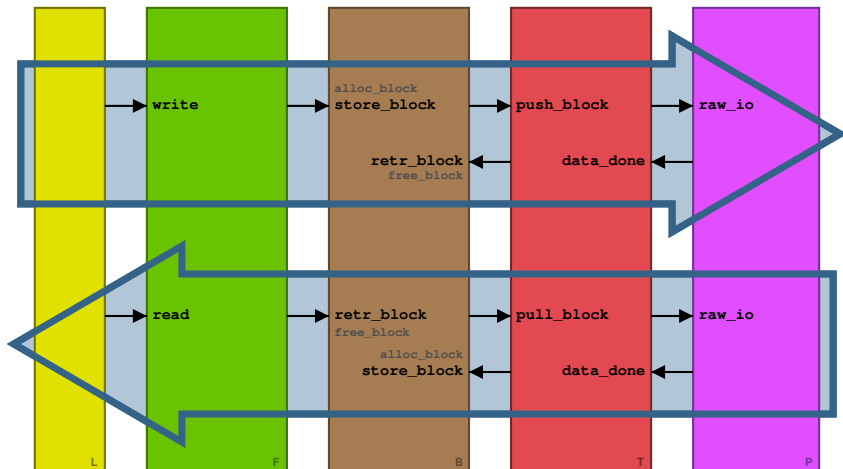
Network Neutral

Modules

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



The Char Device Pipeline



DEMO

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

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

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

[http://www.ohwr.org/\[...\]/zio-manual-130121-v1.0.pdf](http://www.ohwr.org/[...]/zio-manual-130121-v1.0.pdf)