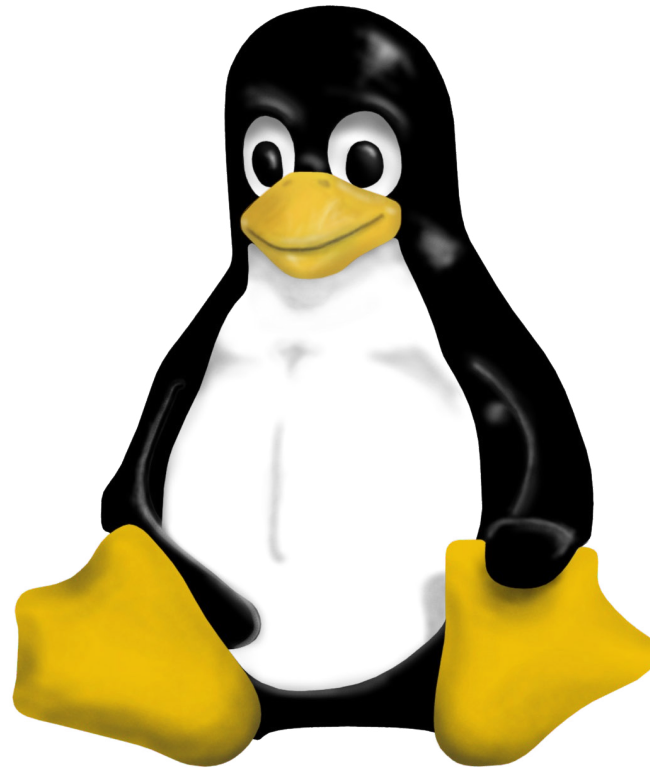


# Linux ROCs



Bryan Moffit  
*Trigger Workshop- 8 July 2010*

 Jefferson Lab

# GE V7865/V7875 SBC

## v7865

Intel Core Duo

2 GHz (667 MHz FSB)

1-3 GB DDR2 SDRAM

## v7875

Intel Core 2 Duo

2.5 GHz (1066 MHz FSB)

1-4 GB DDR3 SDRAM

USB 2.0

2 ports

4 ports

Dual GbE Network ports

Bootable Compact Flash port (up to 4GB)

Optional Transition Module

2 USB, 2 SATA, DVI-D

Optional VITA 41.3 (2 ethernet ports via P0)

VME 320 (Tempe chip -support for 2eVME and 2eSST)

4 timers (2 microsec resolution)

Watchdog timer

32KB User accessible NVRAM

Thermal Probes



**Bryan Moffit**

*Trigger Workshop- 8 July 2010*

**Jefferson Lab**

# Linux ROCs - Performance

V7865 (2 GHz, DDR2)      V7875 (2.5 GHz, DDR3)      MV6100 (1.3 GHz, DDR)

## Time from external signal in the TI to the User ISR/Callback:

Interrupt Response	44 $\mu$ s	29 $\mu$ s	7.5 $\mu$ s
Polling Response	4.5 $\mu$ s	2.5 $\mu$ s	2.5 $\mu$ s

## VME Single Cycle Transfer:

Write	370 ns	370 ns	460 ns
Read	2.6 $\mu$ s	1.5 $\mu$ s	1.0 $\mu$ s

## DMA Latency:

Setup/Start	9 $\mu$ s	5 $\mu$ s	4 $\mu$ s
Finish/Error Check	25 $\mu$ s	18 $\mu$ s	22 $\mu$ s
“Break Even” size	60 bytes	64 bytes	108 bytes

## Network:

Max Throughput	(12% CPU) 116 MB/s	(5% CPU) 116 MB/s	(100% CPU) 79 MB/s
ROC -> EB	75 MB/s	75 MB/s	35 MB/s



# VME Bridge Driver

## jvme

- JLab implementation of VME bridge API (kernel and userspace)
- Written to easily switch APIs (currently uses GEFanuc Proprietary v4.0)
- Routines and arguments mirror those of vxWorks
  - e.g. sysBusToLocalAdrs -> vmeBusToLocalAdrs

## Functionality

- Maps VME windows (A16/A24/A32) to USERSPACE
- Maps VME Bridge registers to USERSPACE
  - Provides lower latency for setup and finish of DMA
  - Maps System Memory to USERSPACE for DMA data to Readout Lists
- Provides USERSPACE routine links to kernel level VME interrupts

## Available Module Drivers:

- FADC250, F1TDC, FlexIO, TI, TS, c1190/1290, vmeDSC



# Future Developments

- ◆ Performance enhancements with more up-to-date kernels?
  - ◆ Mostly worked with 2.6.18 (RHEL5)
  - ◆ Some testing with 2.6.32 (FC12+RT)...  
but no significant improvement in performance
  - ◆ Benchmarking underway with home-built 2.6.34
- ◆ New CPUs (Intel Core i7, DDR3) available soon from several vendors
  - ◆ 2-2.5 GHz (1066 MHz FSB), 64bit, hyperthreading
    - ◆ GE Intelligent Platforms XVB601
    - ◆ Emerson iVME7210
    - ◆ Concurrent VP717



# Discussion Points - VME data

## User Access to VME Module Data

### ★ Process access to VME Bus

- ★ Must obey API (semaphore) that prevents readout during Trigger Routine
- ★ Modules must support this. A single FIFO probably won't work.
- ★ If the process crashes, it will deadlock the Trigger Routine (and vice versa).

### ★ Process access to CODA Readout List Buffers

- ★ Accumulate data from primary/secondary readout list
  - ★ Must know data format (headers, masks, etc)
  - ★ Too much data handling may slow down readout list and cause deadtime.
  - ★ No CODA run... No data.



# Discussion Point - Filesystem

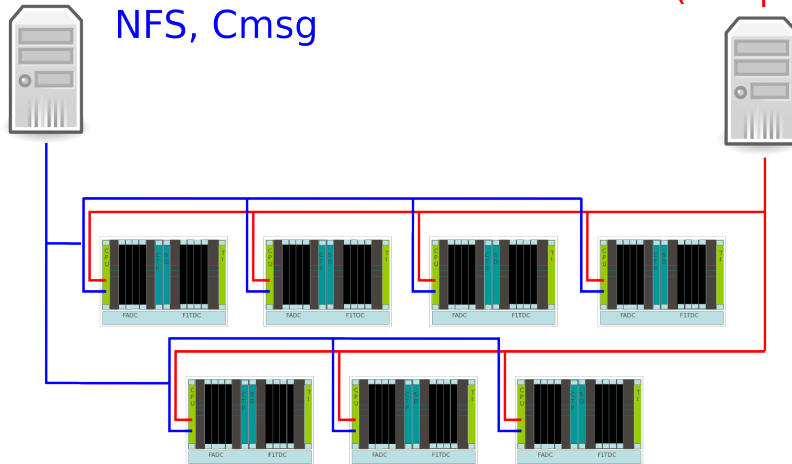
## Onboard Flash or attached Harddisk

- Small scale DAQ
- Kernel and OS locally installed
- Optional (Could use NFS):
  - CODA locally installed
  - Data Stored locally



"Control" Network  
Servers: DHCP, TFTP,  
NFS, Cmsg

"Data" Network  
(Simple)



## Network Boot and NFS

- PXEboot kernel and mount OS over NFS
  - Read-Only: /boot, /usr, /lib, /var ...
  - Write: /etc, /home

Available software (under review):

- System-config-netboot (RedHat)
- DRBL

