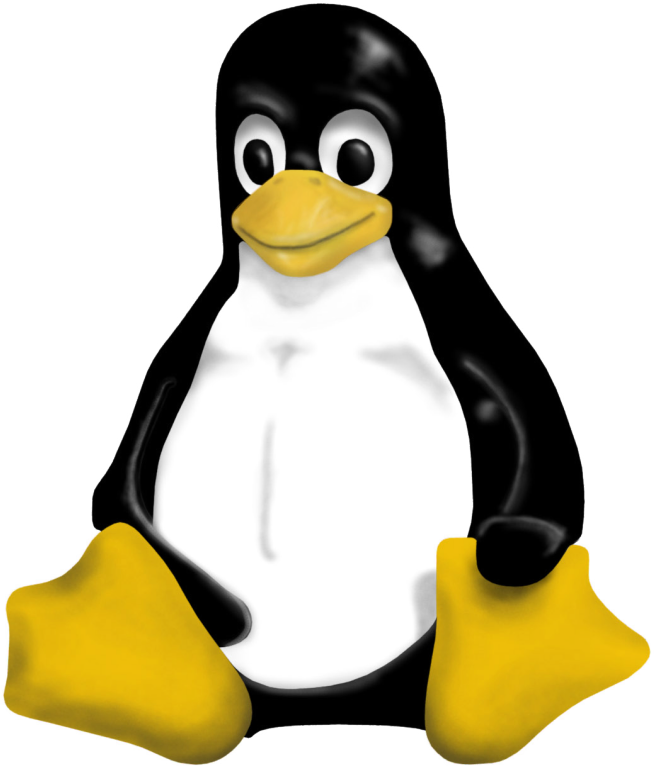
Linux ROCs



Bryan Moffit

Hall-D Online – 11 Nov. 2009

GE FANUC -V7865 VME CPU

- 2 GHz Intel Core Duo Processor (667 MHz Bus)
- 1-3 GB DDR2 SDRAM
- Dual GigE Network ports
- Bootable Compact Flash port (up to 4GB)
- USB 2.0 (2ports)
- 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)
- Additional Hardware Extensions
  - 4 timers (2 microsec resolution)
  - Watchdog timer
  - 32KB User accessable NVRAM
  - Thermal Probes

V7865 versus MV6100

Interrupt Response:	**V7856**	**MV6100**
Time from external signal in the TI to the IACK signal on the VME Bus:	20-23 s	6 us
Time from IACK cycle to execution of Callback (ISR):	14-15 us	1.5 us
TOTAL	**36-38 us**	**7.5 us**
DMA Transfers		
Overhead to move data to User accessable buffer:	**45-75 us**	**0 us**

Linux Drivers/Software

Tempe/TSI148 Linux Driver

- Linux 2.6.18 (RHEL5 2.6.18-128)
- GE Fanuc Proprietary Driver & API
- Some mods made for JLab use:
  - Remap already opened VME windows (Bug Fix)
  - Write to Tempe Chip REGs directly
  - Disable interrupts on BERR (for DMA)
- RPMs created for easy install/distribution

Linux Drivers/Software

jlabgef

Read/Write access to Tempe REGs

Open default a16/a24/a32 VME windows

Some code to mimic vxWorks routines/types

Module Libraries

TI, FADC, F1TDC

DMA Library

Buffer Library, DMA thread manager

DMA

GE Fanuc API/Driver has significant overhead

e.g. (2eSST - 250MB/s) 1624 32bit words -> ~25 µs

GE Fanuc API -> ~90 +/- 10 µS

dmaPollLib (uses pthreads)

One thread to start the transfer,

Second thread to poll memory for EOB

Result (so far) -> ~70 +/- 10 µS

Possibly more to be gained by optimizing the setup.

DMA/Linux Performance

Transfer time jitter (5-100 μs)

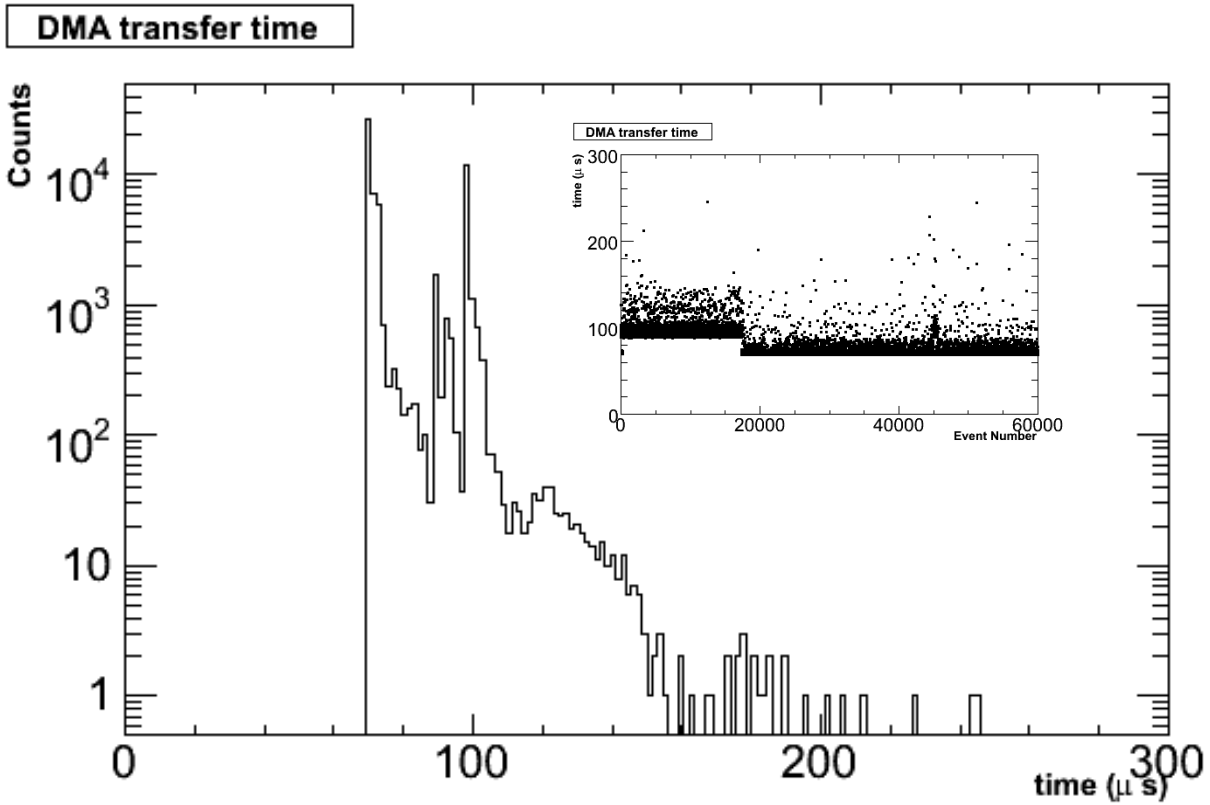
Due to Linux default scheduler and choice of core

Much more stable with some optimization:

- Hardcoding threads to specific CPU cores
- Running SUID "root" (SCHED_FIFO/RR)

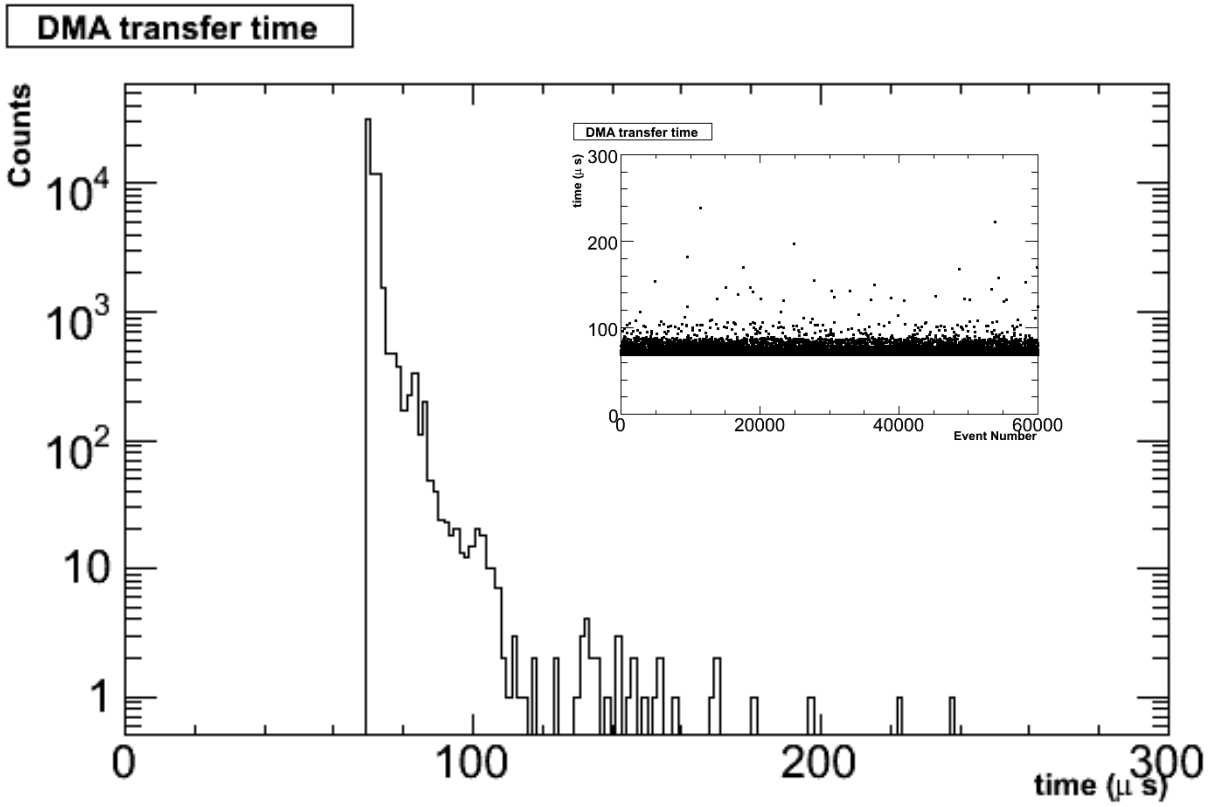
Jitter plots (SUID)

SUID: Regular User Parent Thread CPU Mask: 3



Jitter plots (SUID)

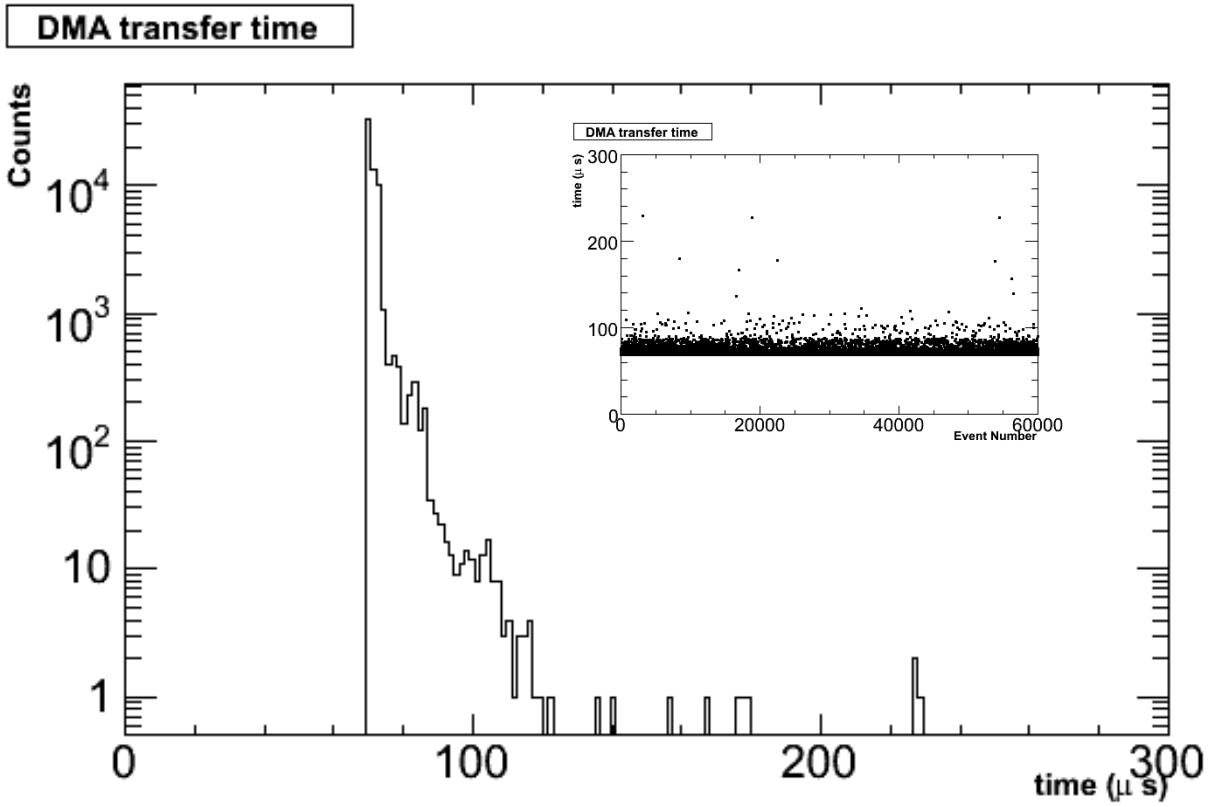
SUID: Regular User Parent Thread CPU Mask: 1



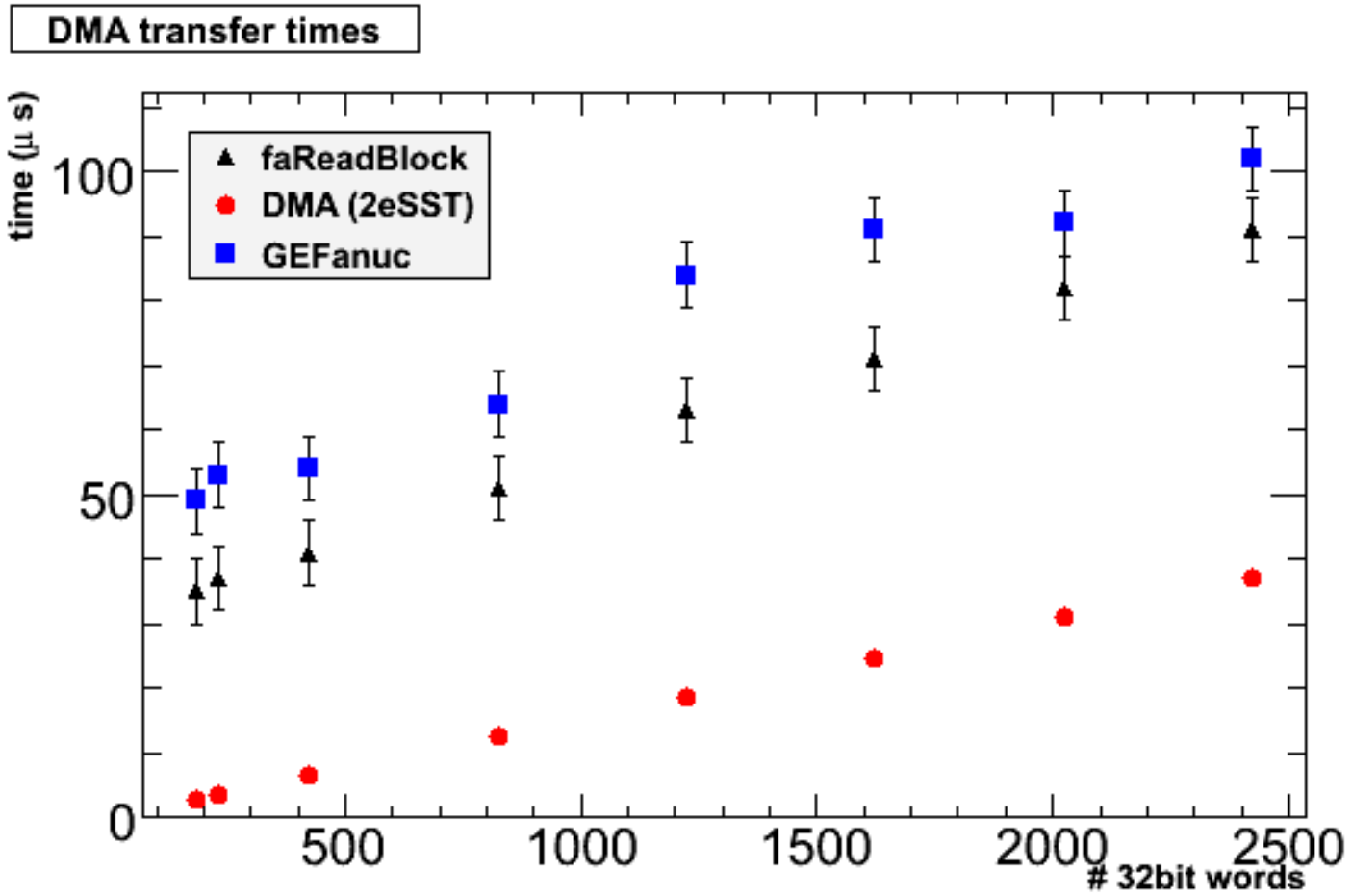
Jitter plots (SUID)

SUID: root

Parent Thread CPU Mask: 1



Transfer Word Count vs Time plot



Changes for CODA 2.6

Currently uses 3 readout lists

- DMA requires event buffers modified for GE Fanuc API
- Primary and Secondary required by CODA ROC 2.6

Mods to crl headers

- Mimic/translate from vxWorks implementation
- makelist -> Makefile-rol for compilation with shared libraries

2.6 extension available soon...