



X2000 Advanced Avionics Project
Performance Characterization Study
August 7, 2002

Len Day

Don Meyer, Nick Nicolich, Rob Steele, Igor Uchenik, Ken Vines



X2000 Avionics Performance Characterization Study Agenda



- Overview and Purpose
- Approach
- Results summary
- Conclusions Summary



- The characterization testing was undertaken with two primary purposes:
 - The X2000 AAP is a new development and it was desired to characterize the reality of what we have built
 - In the new world of computers dependent on cache it is much more difficult to predict software performance. An attempt has been made to provide future users with useful information characterize the use of the X2000 system in their application
- Testing will be repeated in the early '04 timeframe using the enhanced SFC



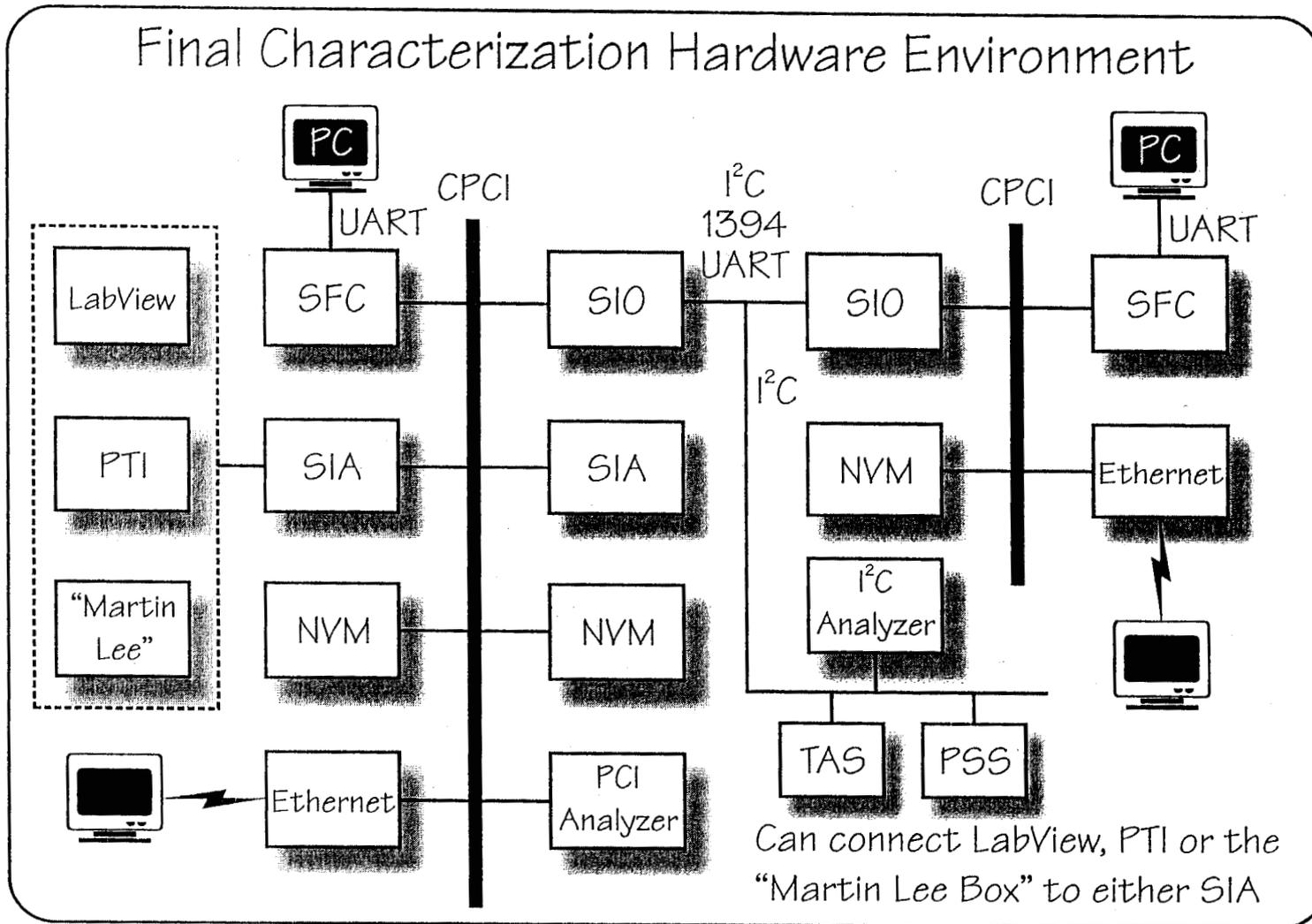
X2000 Avionics Performance Characterization Study Approach



- Based on the above purposes a variety of tests have been run
 - A series of I/O tests exercising all the various interfaces alone and in combination
 - A series of tests involving the exercising of snippets of actual FSW
 - The combination of the I/O tests and the FSW snippets to begin simulating a FSW environment
 - The Deep Impact optical navigation code
 - The DS1 FSW running on the X2000 processor against hardware simulators
 - The Fourier transform benchmarks (www.fftw.org)



- A locally written monitoring application was written to collect
 - CPU %
 - Instruction and data cache miss rate
 - Measured instruction execution rate (MIPs)
- Each statistic collected on the VxWorks task level
 - This allows the isolation of performance of different parts of the system at the same time
- Above statistics and interrupt counts separately collected for interrupt service routines for X2000 devices

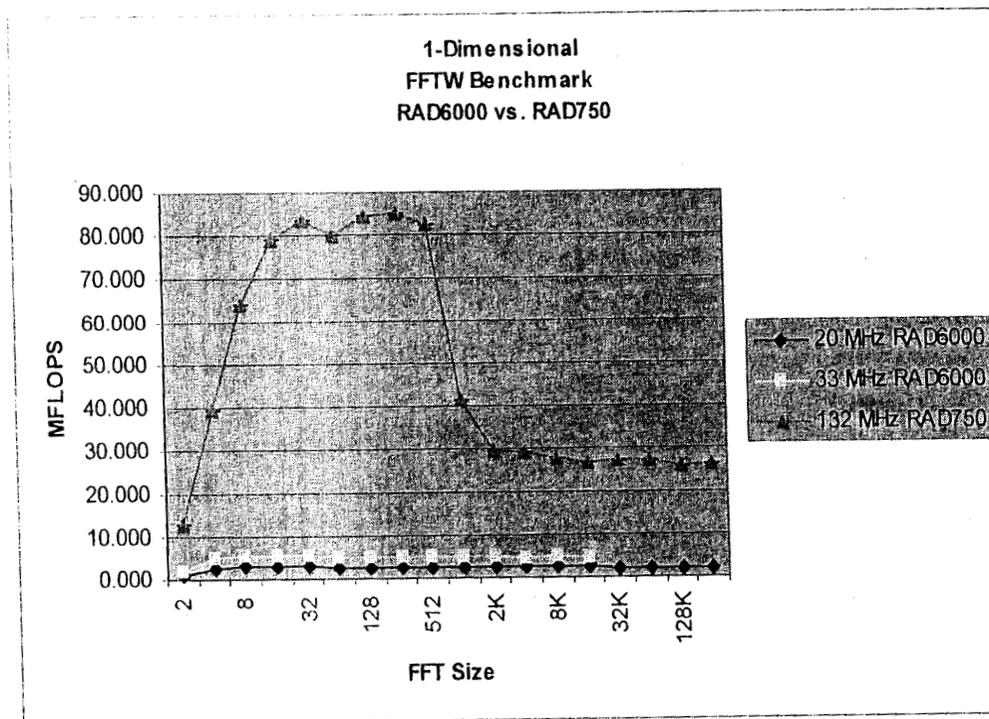




- One of the reasons this study is necessary is that performance can vary widely based on the specifics of what is being done by software
- Therefore it is difficult to summarize results
- Nevertheless, some samples of the results...



- FFTW benchmark shows a factor of 10+ improvement over RAD6000 for this compute / data bound benchmark. Also shows clearly the results of exceeding the cache



- Even in the worst case, difference in performance is greater than difference in clock speed
- The sudden dropoff is due to exceeding L1 cache. If an L2 cache were present a substantially higher level of performance would extend to the right
- Enhanced SFC available in '04 will double the performance in the worst case



- Provides an example of the influence of activity on performance
 - Transform only
 - 54 MIPs, I cache miss .003%, D cache miss 7%
 - Bit plane compression, narrow image space
 - 130 MIPs, I cache miss .002%, D cache miss 0.9%
 - Bit plane compression, wide image space
 - 60 MIPs, I cache miss .002%, D cache miss 6%

These results show the importance of implementing functionality in a “cache friendly” way



X2000 Avionics Performance Characterization Study

Set of tests



Test suite	Number of tests	Comments
1394 Bus	4	Stand-alone tests of 1394
I2C	6	Stand-alone tests of I2C
SIA Instrument Interface	3	Stand-alone tests if SIA instrument interfaces
SIA Telecom	5	Stand-alone tests of SIA telecom interface
DIO UART	3	Stand-alone tests of DIO UART
DIO Custom Logic	1	Timers and other DIO custom logic
NVM	5	Stand-alone tests of mass storage
PSS	2	In Progress, done on Synergy, waiting for PSS on SFC
TRIO	1	Stand-alone test of TRIO
PCI-PCI DMA	1	Data transfer directly between NVM and 1394
Bus Activity	8	Various combinations of above stand-alone tests
FSW	2	DS1 and Deep Impact Op Nav
Flight Algorithm	4	"Snippets" of FSW code
Flight Algorithm Combinations	3	Snippets above run in combination with I/O tests
Totals	48	

Most test cases above have a number of variations such as data rate



X2000 Avionics Performance Characterization Study Report



- Report contains results for all tests and variations mentioned earlier
- Intended to aid in the sizing of any system using the X2000 hardware
 - Could have application to any similar architecture
- Answer questions like
 - Camera emits 1024 byte frames 5 times per second, what processor load does this generate?
 - This EDL algorithm operates on N bytes of data taken at X rate, what processing speed can I expect?
 - ACS requires 10 Hz with input from this set of sensors and this processing, do I have a fit?
 - Etc
- Data are intended to provide enough variation so that specific project cases can be predicted



- I/O operations are expensive when done by program I/O, reading is more expensive than writing.
 - This is not unique to this architecture, any system needs to watch I/O but it may be more of an issue in PCI where there is more speed to lose
 - PCI max throughput is approximately 3 times VME
 - This bottleneck is eliminated by using DMA (Direct Memory Access), either through DMA capability on the PCI peripheral or through the DMA controller in the enhanced SFC. Using DMA it is possible to closely approach the 132 MB/sec PCI theoretical limit
 - The report has the numbers to support trades of performance vs cost of implementing DMA



- Wide variations in performance can be expected based on application and software implementation
 - Software designers and implementers will need to be aware of the “cache friendliness” of implementations
 - In the past we have allocated MIPs to functions
 - This is no longer possible because of processor speeds much faster than memory speeds
 - Traditional approach can no longer be used
 - This report should provide a substantial leg up in sizing an application



- Current SFC meets Europa Orbiter bus bandwidth requirements but as predicted margin is not good
- Enhancement currently underway, units in early '04
 - Double speed memory bus
 - Mitigates the effects of exceeding cache and doubles the performance for any cache-limited case such as hazard avoidance or data compression
 - General-purpose DMA controller
 - Allows DMA to be used with peripherals that do not natively support DMA greatly increasing throughput while greatly decreasing processor load for I/O bound case like imaging.
 - Above are the primary improvements, some other performance improvements implemented