Trigger Timing Tests
Zoran Milosevich and Eleanor G. Judd

Test	Setup	Experiment	Information Obtained
1	Trigger Only L2 Writes Off PedAsPhys 1 token All DSM CPUs included	Vary # pre/post buffers Measure event rate	Check that I am getting similar event rates to those obtained by Zoran and Tonko in March => I am not doing anything stupid and my method is OK
2	Trigger Only L2 Writes Off PedAsPhys 4095 tokens All DSM CPUs included	Set # pre/post = 5 Measure event rate	Check effect of maximizing number of tokens
3	Trigger Only L2 Writes Off PedAsPhys 1 token 5 pre/post buffers	Include one DSM CPU and measure the event rate. Repeat for each DSM CPU in turn.	Somewhat confusing. I was expecting the event rate to be a simple function of the number of DSMs read by the CPU that was included, but it was not. I am not sure now if this was evidence that the Myrinet overhead depends on the number of x-bars involved or something else. I left the results of this test out of further analysis. I would like to repeat it later when we have a better idea of how the number of x-bars affects things.
4	Trigger Only L2 Writes Off PedAsPhys 1 token CTB CPU included	Vary # pre/post buffers Measure event rate	Event time = 1/event rate The gradient of the "event time vs # pre/post buffers" graph is the time for the "DMA+Myrinet" loop in the CTB CPU
5	Trigger Only L2 Writes Off PedAsPhys 1 token CTB CPU included DMA call commented out	Vary # pre/post buffers Measure event rate	Event time = 1/event rate The gradient of the "event time vs # pre/post buffers" graph is the time for the "Myrinet only" loop in the CTB CPU
6	Trigger Only L2 Writes Off PedAsPhys 1 token CTB CPU included myriMemcpy call commented out	Vary # pre/post buffers Measure event rate	Event time = 1/event rate The gradient of the "event time vs # pre/post buffers" graph is the time for the "DMA only" loop in the CTB CPU
7	Trigger Only L2 Writes Off PedAsPhys 1 token EEC CPU included	Vary # pre/post buffers Measure event rate	Event time = 1/event rate The gradient of the "event time vs # pre/post buffers" graph is the time for the "DMA+Myrinet" loop in the EEC CPU
8	Trigger Only L2 Writes Off PedAsPhys 1 token EEC CPU included DMA call commented out	Vary # pre/post buffers Measure event rate	Event time = 1/event rate The gradient of the "event time vs # pre/post buffers" graph is the time for the "Myrinet only" loop in the EEC CPU

9	Trigger Only L2 Writes Off PedAsPhys 1 token EEC CPU included myriMemcpy call commented out	Vary # pre/post buffers Measure event rate	Event time = 1/event rate The gradient of the "event time vs # pre/post buffers" graph is the time for the "DMA only" loop in the EEC CPU
10	Tests 4:9 above	Extract DMA and myriMemcpy latency and data rates by comparing loop times with CTB and EEC CPUs incldued	See email "DMA and Myrinet Latencies" from Eleanor, Sunday 4th May 2003. NOTE: myriMemcpy information was obtained by assuming there was NO dependence on the number of x-bars involved. CTB and EEC have different numbers of x-bars between them and L1 and L2. If the assumption was wrong then these results may be wrong too (but hopefully not by too much).
11	Trigger Only L2 Writes Off PedAsPhys 1 token NO DSM CPUs included	Vary # pre/post buffers Measure event rate	Minimum times for a token to run around the system. It is unaffected by the # pre/post buffers since L1 only reads the triggered crossing and L2 writes and DAQ were off.
12	Trigger Only L2 Writes Off PedAsPhys 4095 tokens NO DSM CPUs included	Set # pre/post = 5 Measure event rate	Check effect of maximizing number of tokens
13	Trigger only L2 Writes turned ON PedAsPhys 1 token NO DSM CPUs included	Vary # pre/post buffers Measure event rate	Minimum time for a token to run around the system with the extra byte swapping included for the L2 write. This turns out to be very small
14	Trigger + DAQ L2 Writes Off PedAsPhys 1 token NO DSM CPUs included	Vary # pre/post buffers Measure event rate	Minimum time for a token to run around the system with DAQ. This turns out to be large. We do not know what DAQ is doing to slow things down so much.
15	Trigger + DAQ L2 Writes Off PedAsPhys 4095 tokens NO DSM CPUs included	Set # pre/post = 5 Measure event rate	Check effect of maximizing number of tokens
16	Trigger + DAQ L2 Writes Off PedAsPhys 4095 tokens CTB CPU included	Set # pre/post = 5 Measure event rate	Check effect of adding in one DSM CPU.
17	Trigger + DAQ L2 Writes Off PedAsPhys 4095 tokens All DSM CPUs included	Set # pre/post = 5 Measure event rate	Check effect of adding in all DSM CPUs.

18	Trigger + DAQ L2 Writes Off PedAsPhys 4095 tokens All DSM CPUs included EMC Included	Set # pre/post = 5 Measure event rate	Check effect of adding in a slow detector
19	Trigger + DAQ L2 Writes Off PedAsPhys 4095 tokens All DSM CPUs included EMC Included TOF Included	Set # pre/post = 5 Measure event rate	Check effect of adding in a second slow detector

Comments

1) Eleanor only measured the time for the myriMemcpy call, not the send_message call. Zoran started measuring the send_message time. We need to finish measuring the send_message times. These messages go from VxWorks->VxWorks, VxWorks->Linux and Linux->VxWorks. We cannot assume the time for the difference message types is the same; we need to measure it. We also cannot assume the time for one message type is constant between all relevant CPUs; it may depend on our network architecture and the number of cross-bars in between the CPUs.
2) Eleanor's myriMempcy latency measurement was made under the assumption that our Myrinet network architecture (number of cross-bars etc…) was irrelevant. This assumption is potentially important since the CTB and EEC CPUs were used in these tests and they have different numbers of cross-bars between them and L2 (1 and 2 cross-bars respectively I believe). It would be good to redo these measurements using CPUs with the same numbers of cross-bars.
3) All these tests were done with 1 or 4095 tokens in the system. The effect of multiple tokens can be small or large, depending on the system configuration. It would be nice to slowly add more tokens and see where the rate saturates for different system configurations (DAQ, no DAQ, DSM CPUs, no DSM CPUs, etc…).