



an URI / NEU collaboration



# output trace data from LevoSim

student **David Morano**  
advisors **Professor David Kaeli**  
**Professor Augustus Uht**

NUCAR talk 01/10/05

# Outline

---



- **main event trace**
- **detailed event trace**
- **bus traces**
  - monitoring of buses
  - logging bus master transactions
- **XML state trace**
  - XML
  - Levo state dump

# main event trace

---

- **only records the most "interesting" events**
  - clock transitions
  - AS snarfs
  - RFU snoops
  - RFU snarfs
  - AS retirements
  - i-fetch loads
  - deadlocks !
- **register file dumps**

# some main events

---



CLOCK ===== 7

regfilter\_lookup: id=0 rftt=0 tr=0 p=0 tt=1 a=29 dp=15 dv=00000000

regfilter\_writebuses: id=0 rftt=0 F tr=0 p=0 tt=-9:0 a=29 dp=1 dv=7fff2fd0

regfilter\_writebuses: id=1 rftt=2 B tr=0 p=0 tt=3:0 a=5 dp=0 dv=00000000

C7,ASID0,TT0,FA0x400150=>successful write to MBWB at 0; mem(0x7fff2fd0)

C7,ASID1,TT1,FA0x400154=>successful write to RFWB at 1; reg(5)=0x7fff2fd4

C7,ASID3,TT3,FA0x40015c=>successful write to RFWB at 3; reg(6)=0x4

C7,ASID10,TT6,FA0x400168=>successful write to RBWB at 10; reg(2)

CLOCK ===== 8

regfilter\_lookup: id=0 rftt=0 tr=0 p=0 tt=3 a=5 dp=0 dv=00000000

regfilter\_update: id=1 rftt=2 tr=0 p=0 tt=-9:0 a=29 dp=1 dv=7fff2fd0

regfilter\_lookup: id=3 rftt=4 tr=0 p=0 tt=4 a=4 dp=15 dv=00000000

regfilter\_lookup: id=4 rftt=6 tr=0 p=0 tt=6 a=6 dp=15 dv=00000000

regfilter\_writebuses: id=0 rftt=0 F tr=0 p=0 tt=-9:0 a=29 dp=1 dv=7fff2fd0

C8,ASID1,TT1,FA0x400154=>Snoop on RBWB at 1, data from tt=3;reg(5)

# main events (continued)

---



CLOCK ===== 27

AS ID=0, TT=0,FA=0x400150, ,s1(29)=0x7fff2fd0 ,dt(4)=0x1 ,mem(0x7fff2fd0)=0x0 ,cnst(0x0)=0x0 OP=46

pin(0)=1, cpin(-1)=0, pout(-1)=-2, cpout(-1)=-2

Inst. Committed

AS ID=1, TT=1,FA=0x400154, ,s1(29)=0x7fff2fd0 ,dt(5)=0x7fff2fd4 ,cnst(0x0)=0x4 OP=4

pin(0)=1, cpin(-1)=0, pout(-1)=-2, cpout(-1)=-2

Inst. Committed

AS ID=2, TT=2,FA=0x400158, ,dt(28)=0x10010000 ,cnst(0x0)=0x1001 OP=25

pin(0)=1, cpin(-1)=0, pout(-1)=-2, cpout(-1)=-2

Inst. Committed

AS ID=3, TT=3,FA=0x40015c, ,s1(5)=0x7fff2fd4 ,dt(6)=0x7fff2fd8 ,cnst(0x0)=0x4 OP=4

pin(0)=1, cpin(-1)=0, pout(-1)=-2, cpout(-1)=-2

Inst. Committed

Reg[4] = 0x1

Reg[5] = 0x7fff2fd4

# **detailed events**

---

- **compile-time chosen set of events from one or more software components**
- **generally includes most all of the information that is recorded in the main event output**
- **can get exceedingly large**
- **primary contributor to disk space usage during debugging**
- **although event types are determined at compile-time through compile-time switches, the level of detail is determined at runtime**

# detailed events

---



iw\_comb: doing REGFILTER(2)=00dad288  
regfilter\_comb: id=2 rftt=-4 entered ck=282 ph=1  
regfilter\_comb: i=0 lfgf\_comb() rs=0  
regfilter\_comb: i=1 lfgf\_comb() rs=0  
regfilter\_comb: regfilter\_writebuses()  
regfilter\_writebuses: id=2 rftt=-4 f\_fownbus=0 f\_fwdv=0  
regfilter\_writebuses: f\_bownbus=0 f\_bwdv=0  
regfilter\_comb: regfilter\_writebuses() rs=0  
regfilter\_checkholds: F FIFO=0 minfree=5  
regfilter\_checkholds: remaining entries=15  
regfilter\_checkholds: nhold=0 maxcount=0  
regfilter\_comb: regfilter\_checkholds() rs=0  
regfilter\_comb: regfilter\_fifoshifts() rs=0  
regfilter\_comb: exiting rs=0  
iw\_comb: regfilter\_comb() REGFILTER=00dad288 rs=0

# detailed events (continued)

---



iw\_xmlout: entered

iw\_xmlout: LIFETCH

iw\_xmlout: LLB

iw\_xmlout: sharing groups

iw\_xmlout: AS asid=0

las\_xmlout: ia=00400a3c instr=1120001f beq r9,r0,0x400abc

iw\_xmlout: AS asid=1

las\_xmlout: ia=00400a40 instr=00000000 nop

iw\_xmlout: AS asid=4

iw\_xmlout: AS asid=5

iw\_xmlout: AS asid=2

las\_xmlout: ia=00400a44 instr=80aa0002 lb r10,2(r5)

iw\_xmlout: AS asid=3

las\_xmlout: ia=00400a48 instr=00000000 nop

iw\_xmlout: AS asid=6



# bus traces

---

- **bus transaction traces**
- **captured by special components monitoring (reading) the buses all of the time**
- **the monitoring components also accumulate some bus usage statistics**
- **bus hardware components also accumulate bus usage statistics (not discussed)**
- **all execution window buses are traced**

# bus traces (transactions)

---



clock	bus	mid t p	tt	addr	data
00000007	000da800	23 0 0	-9:0	0000001d	7fff2fd0
00000008	000da800	23 0 0	-9:0	0000001d	7fff2fd0
00000008	000da8d0	13 0 0	2:0	0000001c	10010000
00000009	000da800	14 0 0	1:0	00000005	7fff2fd4
00000009	000da8d0	12 0 0	3:0	00000006	00000004
00000009	000da9a0	20 0 0	-9:0	00000004	00000000
0000000a	000da800	23 0 0	-9:0	00000005	00000000
0000000a	000da8d0	21 0 0	-9:0	0000001d	7fff2fd0
0000000a	000da9a0	7 0 0	4:0	00000002	00000000
0000000a	000daa70	5 0 0	6:0	00000006	00000000
0000000b	000da8d0	21 0 0	1:0	00000005	7fff2fd4
0000000b	000da9a0	6 0 0	5:0	0000001c	00001af0
0000000b	000daa70	4 0 0	7:0	00000001	10010000
0000000c	000da800	14 0 0	1:1	00000005	7fff2fd4

# bus traces (statistics)

---



<b>clocks</b>	<b>4222</b>
<b>buses</b>	<b>4</b>
<b>bus slots</b>	<b>16888 (0.016-Mi)</b>
<b>total transactions</b>	<b>4579 ( 27.1% utilization)</b>
<b>mean transactions per bus</b>	<b>1144.8 (stddev 129.5)</b>
<b>STORE transactions (total)</b>	<b>4579</b>
<b>STORE transactions (data)</b>	<b>4579</b>
<b>STORE transactions (no data)</b>	<b>0</b>
<b>NULLIFY transactions (total)</b>	<b>0</b>
<b>NULLIFY transactions (data)</b>	<b>0</b>
<b>NULLIFY transactions (no data)</b>	<b>0</b>



# bus master requests traces

---

- **records requests for bus transactions from bus masters that use the bus management agent component (LBUSINT)**
  - Active Station
  - Write Queue
- **very useful for tracking bus master related activity**

# bus master trace

---



clock	masterid	bus	op	tr	p	tt	addr	val
00000003	000e0d00	000db400	15	W	0 0	4:0	0000001d	00000000
00000003	000e1000	000db400	14	W	0 0	5:0	0000001d	00000000
00000003	000e1600	000db4d0	12	W	0 0	7:0	00000005	00000000
00000005	000e0d00	000db400	15	R	0 0	0:0	0000001d	00000000
00000005	000e1000	000db400	14	R	0 0	0:0	0000001d	00000000
00000005	000e1300	000db4d0	13	R	0 0	3:0	00000005	00000000
00000005	000e1600	000db4d0	12	R	0 0	3:0	00000005	00000000
00000005	00d9e500	000db5a0	7	W	0 0	4:0	00000004	00000000
00000005	00d9e800	000db5a0	6	W	0 0	5:0	0000001c	00000000
00000005	00d9eb00	000db670	5	W	0 0	6:0	00000006	00000000
00000006	000e0d00	000db400	15	R	0 0	1:0	0000001d	00000000
00000006	000e1000	000db400	14	R	0 0	1:0	0000001d	00000000
00000006	000e1200	000da8d0	13	W	0 0	2:0	0000001c	10010000
00000007	000e0c00	000da800	15	R	0 0	-9:0	0000001d	7fff2fd0

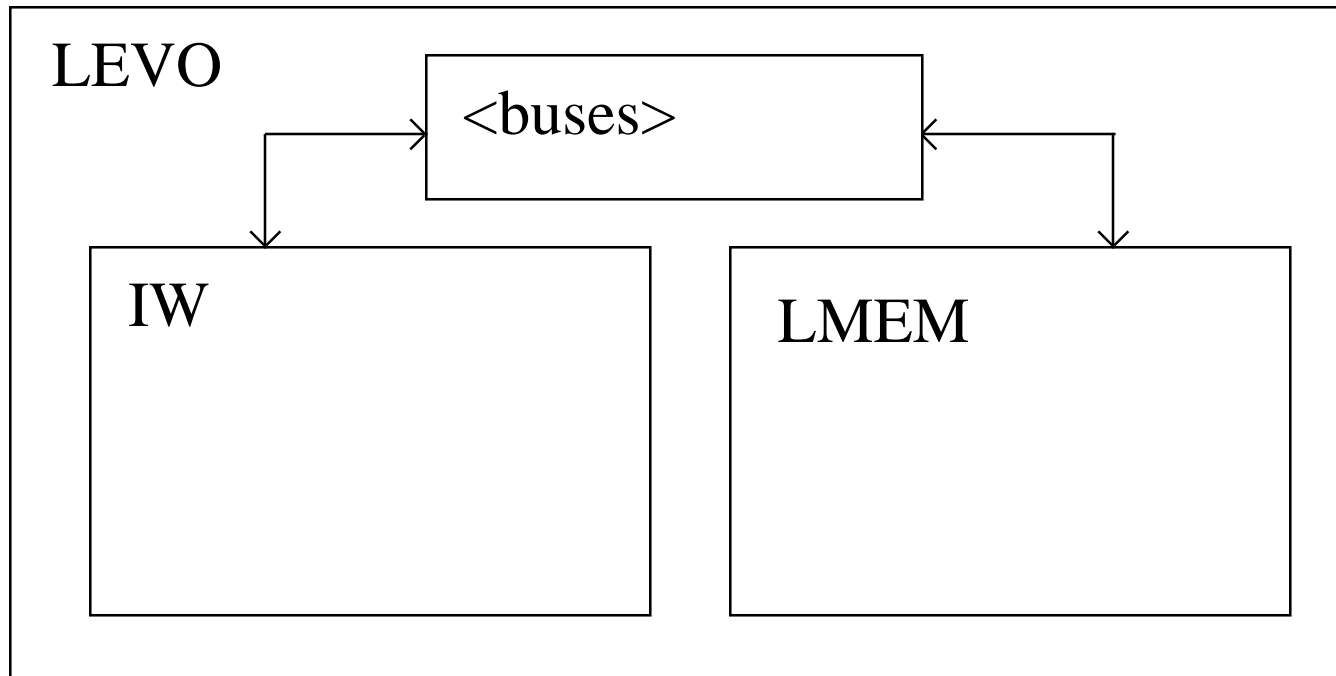
# XML state output

---

- contains the state of Levo machine component registers, clock by clock
- uses XML as the file structuring format
- an index file is also created that is used to index the XML file by clock number
- the scheme is designed to allow for the incremental dump of state information (only state that changed from the previous clock)
- the index file contains a means to find the last clock record with a full (synchronized) state
- state is dumped hierarchically according to machine component arrangement and connections

# high-level block diagram (1)

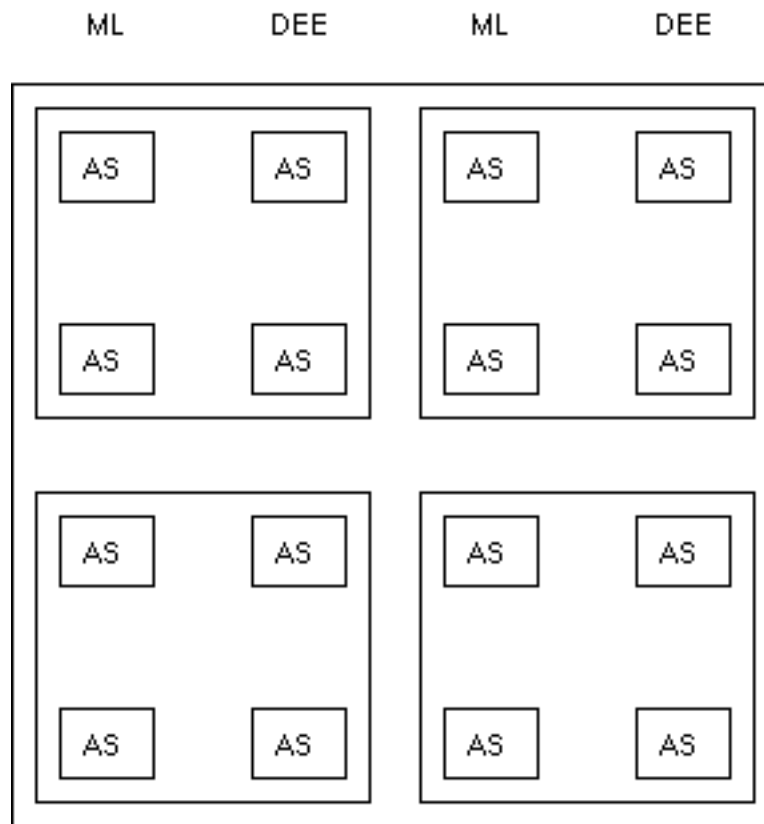
---



# high-level block diagram (2)



8 ML Active Stations  
8 DEE Active Stations  
2 ML columns  
2 DEE columns  
4 Sharing Groups



instruction execution window

I-fetch and  
branch prediction

branch tracking  
buffer

station load buffer

write store queue

memory caches

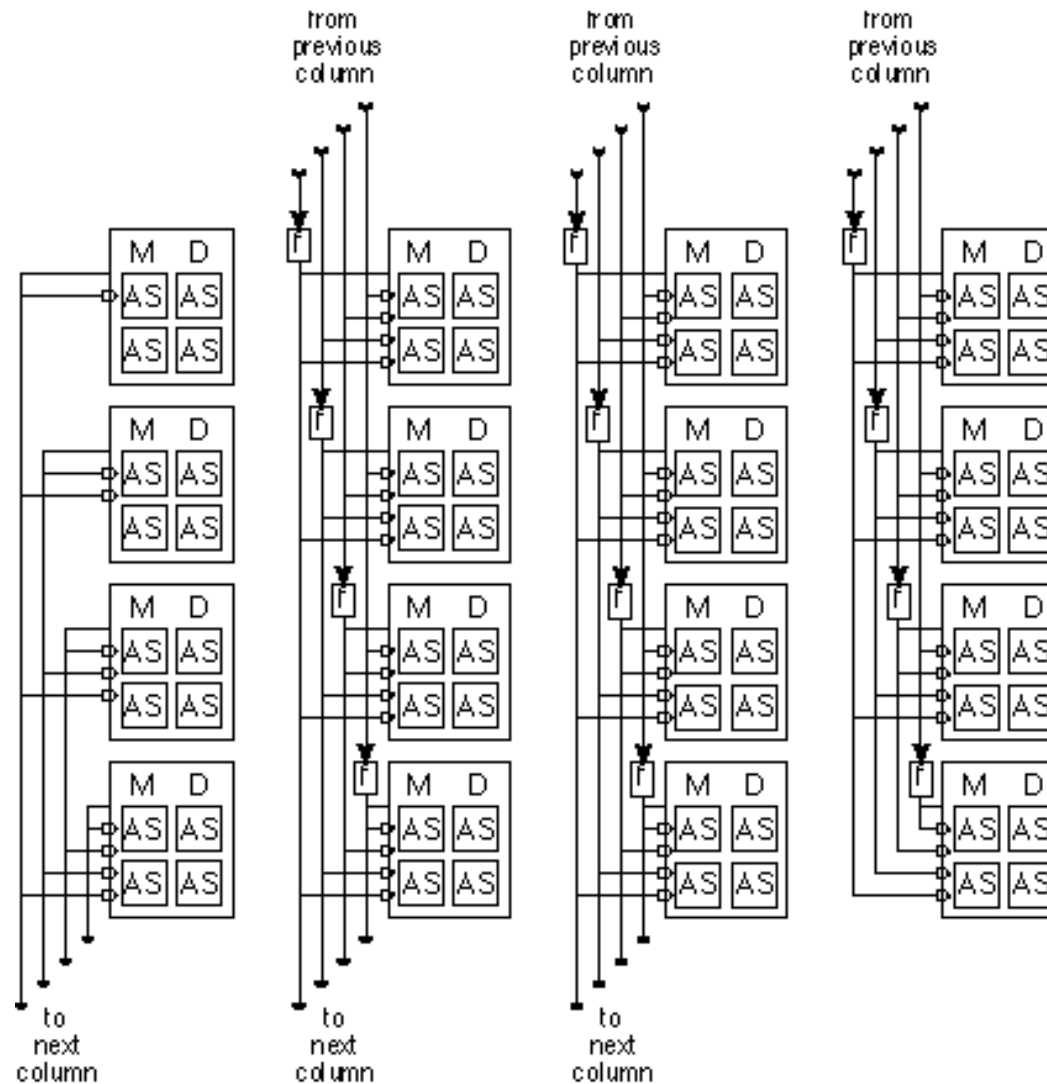
main memory

- **execution flow**

- fetch
- load
- issue
- dispatch
- execute
- re-execute !
- retire



# sharing group forwarding buses



# sample XML output

---



```
<configuration>
<nsgrows>2</nsgrows>
<nsgcols>2</nsgcols>
<nasrpsg>2</nasrpsg>
<nrfspan>1</nrfspan>
<nrbspan>1</nrbspan>
<npfspan>3</npfspan>
<nmfspan>0</nmfspan>
<nmbspan>0</nmbspan>
<ifetchwidth>2</ifetchwidth>
<mfinter>12</mfinter>
<mbinter>12</mbinter>
<mwinter>12</mwinter>
<wmfinter>12</wmfinter>
<wmbinter>12</wmbinter>
```

# sample XML output

---



```
<sg>
<sgid>0</sgid>
<ascol>
<id>0</id>
<aselem>
<id>0</id>
<asrow>0</asrow>
<asid>0</asid>
<as>
<uid>fe7000b0</uid>
<asid>00000000</asid>
<used>0</used>
</as>
</aselem>
<aselem>
```