**dimm-ras-linux**

DIMM RAS feature monitoring on Linux servers

**VME metrics**

* bin/
  + mem\_inspector.py - Memory Inpsector (collects data from MEM\_BW, CORRERRCNT, HWMON, DIMM temp)
* demos/ - set of standalone demos based on PMON,HWMON libraries
* services/
  + mem\_inspector.service - Systemd service, collecting mem\_inpsector output in CSV format
  + pcm\_memory\_bw.service - Systemd service, collecting memory bandwidth in CSV format

**Instalation**

cp services/\*.service /etc/systemd/system

systemctl enable mem\_inspector.service

systemctl enable pcm\_memory\_bw.service

Check the ./vme/bin/mem\_inspector.py for

* interval time values
* DID values

Command("read\_correrrcnt",

[

"read\_correrrcnt",

"0x6fb2", "0x6fb3", "0x6fb6", "0x6fb7", "0x6fd2", "0x6fd3", "0x6fd6","0x6fd7"

], 15)

interval time is 15 seconds  
  
DIDs are the list of PCI devices identified by DeviceID.  
  
If not sure please check "lspci -D -nn" output and search for Memory Controller the [VID:DID] sequence is  
  
present in every row. Intel E7 V4 has [8086:6f\*\*] where Intel E7 V3 might have [8086:2f\*\*].  
  
Please setup accordingly.  
At the end please launch:

systemctl start mem\_inspector.service

systemctl start pcm\_memory\_bw.service

and check the /var/log/mem\_inspector\*.log or /var/log/pcm\_memory\_bw.csv

**Read CSV data**

When memory inspector is controlled by mem\_inspector.service all standard output / error streams are redirected to files:

StandardOutput=file:/var/log/mem\_inspector\_stdout.log  
StandardError=file:/var/log/mem\_inspector\_stderr.log

File mem\_inspector\_stdout.log should include CSV data that can be view and edit in Excel environment.

**Header file PMON read\_correrrcnt**

Date ; Tool Name ; Host ID ; Device address ; correrrcnt\_0 ; correrrcnt\_1 ; correrrcnt\_2 ; correrrcnt\_3 ; correrrthrshld\_0 ; correrrthrshld\_1 ; correrrthrshld\_2 ; correrrthrshld\_3 ; correrrorstatus

"2022-11-22 15:52:55.487297";"pmon.read\_correrrcnt";"h03hcrbbm06";"0000:ff:14.3";"0";"0";"0";"0";"2147450879";"2147450879";"2147450879";"2147450879";"274432";

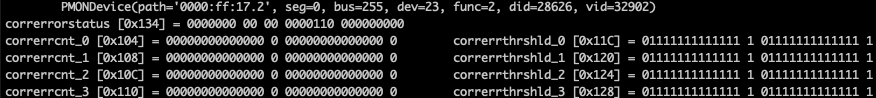
According to the 3 tables below that represent Correrrcnt\_{0,1,2,3}, Correrrthrshld\_{0,1,2,3} and Correrrorstatus all this 32bit dword should be interpreted in specific bit format

correrrorstatus [0x134] = 0000000 00 00 0000110 000000000

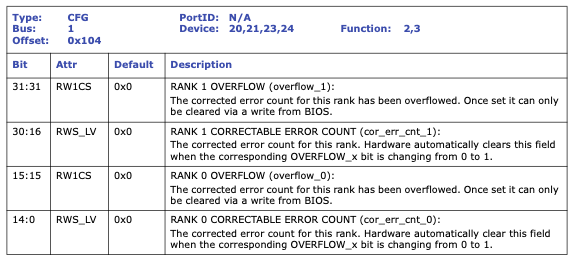
correrrcnt\_0 [0x104] = 00000000000000 0 00000000000000 0

correrrthrshld\_0 [0x11C] = 01111111111111 1 01111111111111 1

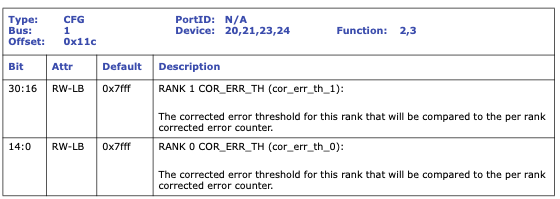
Sample from demo\_correrrcnt\_linux.py tool:



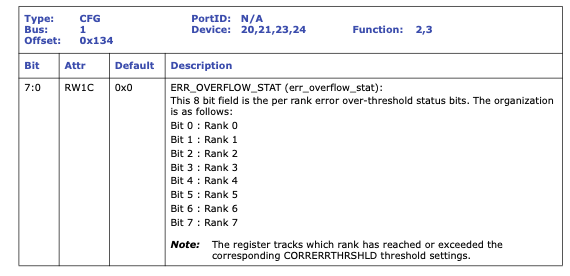
Correrrcnt\_{0,1,2,3}:

****

Correrrthrshld\_{0,1,2,3}:



Correrrstatus



**Header file PMON read\_dimm\_temp**

Date ; Tool Name ; Host ID ; Device address ; channel0\_max\_temp ; channel1\_max\_temp ; channel2\_max\_temp ; channel3\_max\_temp

"2022-11-22 15:52:55.564085";"pmon.read\_dimm\_temp";"h03hcrbbm06";"0000:3f:14.0";"0";"0";"0";"0";

**Header file PMON HWMON read\_temp**

Date ; Tool Name ; Host ID ; Temperature sensor name ; Sensor # ; Socket # ; Input temp ; Critical temp ; Maximum temp

"2022-11-22 15:52:55.569925";"hwmon.read\_temp";"h03hcrbbm06";"Package 0";"1";"0";"29.0";"98.0";"88.0";

"2022-11-22 15:52:55.569942";"hwmon.read\_temp";"h03hcrbbm06";"Core 0";"10";"0";"27.0";"98.0";"88.0";

"2022-11-22 15:52:55.569950";"hwmon.read\_temp";"h03hcrbbm06";"Core 1";"20";"0";"28.0";"98.0";"88.0";

"2022-11-22 15:52:55.569958";"hwmon.read\_temp";"h03hcrbbm06";"Core 2";"21";"0";"27.0";"98.0";"88.0";

Where Input temp, Critial temp, Maximum temp are represented by float values in Celcius degree i.e 27.0°C

**Reading /var/log/pcm\_memory\_bw.csv**

According to pcm\_memory\_bw.service output of Intel pcm-memory tool is redirected to:

ExecStart=/usr/sbin/pcm-memory 2 -csv=/var/log/pcm\_memory\_bw.csv

The output is simple and represent memory bandwidth over socket vs channel



It contains the overall Read / Write / Memory summary at the 3 last columns.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **SKT3** | **SKT3** | **SKT3** | **SKT3** | **System** | **System** | **System** |
| Mem Read (MB/s) | Mem Write (MB/s) | P. Write (T/s) | Memory (MB/s) | Read | Write | Memory |
| 69.00 | 69.08 | 1050863 | 138.09 | 275.40 | 275.53 | 550.93 |
| 68.63 | 68.72 | 1047943 | 137.35 | 273.03 | 273.01 | 546.04 |
| 68.67 | 68.77 | 1048686 | 137.44 | 272.76 | 272.85 | 545.61 |