AMD XpressIO (AMDXIO) User Guide (NDA)

Publication # **58280** Revision: **0.50**
Issue Date: **June 2023**

Trademarks

AMD, the AMD Arrow logo, AMD EPYC, and combinations thereof are trademarks of Advanced Micro Devices, Inc. Other product names used in this publication are for identification purposes only and may be trademarks of their respective companies.

Linux is a registered trademark of Linus Torvalds.

Table of Contents

List of Tables

List of Figures

Revision History

Date	Revision	Description
June 2023	0.50	Initial NDA release.

Chapter 1 Device Listing and Selection

The AMD Xpress IO (AMDXIO) device-listing command lists all AMD CPUs and dGPUs on the platform where the tool is executed. Each CPU die is given a unique ID. Similarly, each AMD dGPU also is given a unique ID.

The command to list AMD devices is "**-i**". Syntax for this command is:

```
amdxio -i
```

A sample output is below:

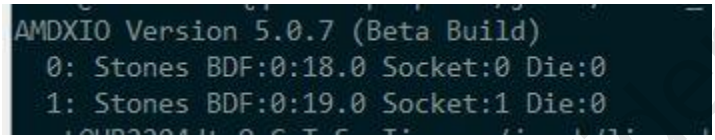


Figure 1. Device Listing Sample Output

As seen from this sample output, we have two AMD dies. The first die on socket 0 is given device ID 0, while the second die is given device ID 1.

For some commands (like xGMI margining, eye scan, etc.), the device ID is a required parameter to specify the device and lanes to be targeted.

For example, if the user wants to run xGMI margining on Socket 0 Die 0 in the example above, the command would be:

```
amdxio -i=0 -xgmi -margin …
```

If the user wants to run xGMI margining on Socket 1 Die 0 in the example, the command would be:

```
amdxio -i=1 -xgmi -margin …
```

Chapter 2 Links and Lanes

The AMDXIO tool supports listing all links and lanes in supported AMD CPUs and GPUs. The command to list links and lanes is:

```
amdxio -i=<device id> -laneinfo
```

A sample output is below:

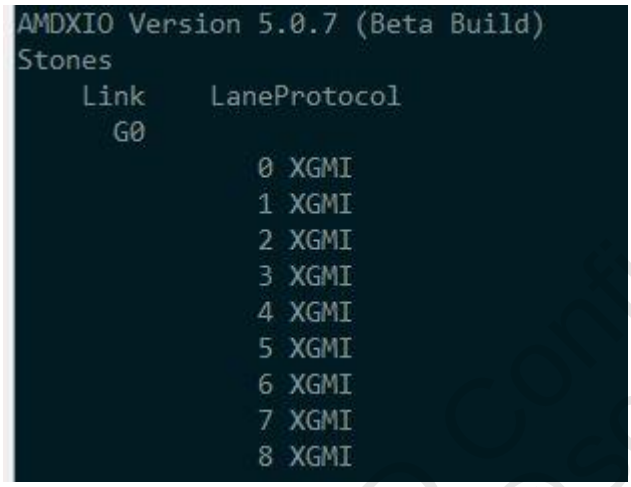


Figure 2. Links and Lanes Listing Sample Output

As seen in the sample output, there is a link named “G0” with lanes 0–8. (For brevity, additional lane output is not captured here.) All lanes are xGMI lanes. Note that in the case of PCIe lanes listed in this output, there need not be a PCIe device connected. To know which lanes have connected PCIe devices, use the PCIe port-listing command. (Refer to Chapter 3 PCIe Port Listing.)

For some commands (like xGMI margining, eye scan, etc.), the margin result of the target link and lane must meet the pass criteria. In the example above, if the user wants to run xGMI margining on link “XGMIPCS” and lane 0, the command would be:

```
amdxio -i=<device id> -xgmi -margin -lanes=XGMIPCS:0
```

To target multiple lanes, say lanes 3–5 on link “XGMIPCS” in the same example, the command would be:

```
amdxio -i=<device id> -xgmi -margin -lanes=XGMIPCS:3.5
```

Chapter 3 PCIe Port Listing

The AMDXIO tool supports listing AMD PCIe root ports and their SerDes mapping. The command is:

```
amdxio -list -pcie -ports
```

A sample output is below:

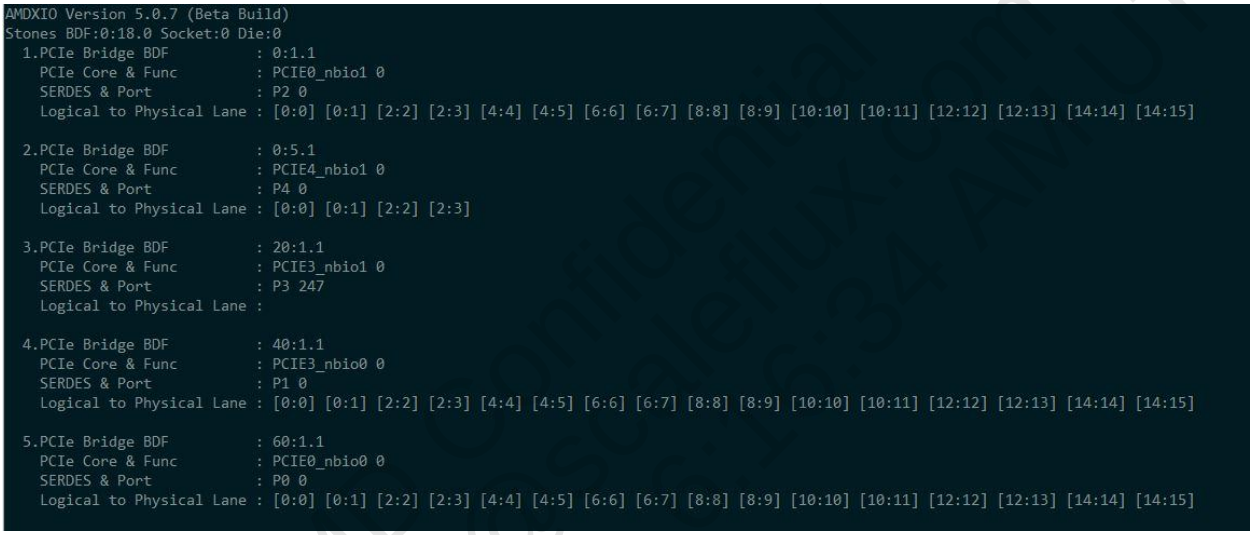


Figure 3. PCIe Root Port/SerDes Mapping Sample Output

As seen in the sample output, we found five PCIe root ports. Listed for each port are its PCIe core and function, as well as SerDes name and port. Additionally, the last row details how PCIe lane numbering matches to actual physical lane numbers.

The listed physical lanes can be used to identify which PCIe lanes have a PCIe device connected and can be used for running the eye scan command. For example, in the sample output's second listing, BDF 0:5.1 is mapped to P4, and lanes 0–3 are active, meaning a device is connected. Also, this port is for Socket 0 and Die 0, which was Device 0 in the listing command. To run the eye scan command, use:

```
amdxio -i=0 -eyescan -lanes=P4:0.3
```

Chapter 4 PCIe Margining

PCIe margining operations supported by the tool can be separated into three types:

- Margining of all PCIe devices (bridges and endpoints)
- Margining of a specific PCIe device (bridges, endpoints, and retimers)
- 4-point margining support (AMD bridges only)

4.1 Margining All PCIe Devices

The tool user can margin all PCIe bridges and endpoint devices using the margin "`-all`" switch from the command line. When this switch is invoked:

1. The tool will auto-enumerate and detect all PCIe devices on the system.
2. The tool will then split the device list into separate batches to ensure that both bridges and endpoints connected to those bridges are not in the same batch.
3. Then the tool will start margining all devices in parallel in one batch.
4. Once a batch is complete, the tool will move onto the next batch until all batches (and thus all devices) are margined.

4.1.1 Margin Command Invocation

```
./amdxio -pcie -margin -all
```

4.1.2 Console Output

```
./amdxio -pcie -margin -all
    Following PCIe devices to be margined in this batch.
      0. SBDF:0:0:7.1
      1. SBDF:0:20:7.1
      2. SBDF:0:40:1.1
      3. SBDF:0:42:0.0
      4. SBDF:0:60:1.1
      5. SBDF:0:65:0.0
      6. SBDF:0:80:7.1
      7. SBDF:0:a0:7.1
      8. SBDF:0:c0:7.1
      9. SBDF:0:e0:7.1
      10. SBDF:0:e2:0.0


    ..........................................................
```

Figure 4. Console Output for Margining PCIe Bridges/Endpoint Devices

4.1.3 File Output

The CSV file output will be dumped into a timestamped folder, which will have margin data for all devices margined.

4.2 Margining a Single PCIe Device

4.2.1 Getting a List of PCIe Devices

To get a list of all PCIe devices, including both bridges and endpoints, use this command and then select 4G and 5G devices from the list:

```
./amdxio -pcie -list -devices
```

```
AMDXIO Version 5.0.7 (Beta Build)
Bridges
  AMD Bridges
    BDF=00:01.001  32GT/s x16 AMD-PCI-to-PCI bridge Sec:Sub=[1:1]
    BDF=00:07.001  32GT/s x16 AMD-PCI-to-PCI bridge Sec:Sub=[3:3]
    BDF=20:07.001  32GT/s x16 AMD-PCI-to-PCI bridge Sec:Sub=[22:22]
    BDF=40:01.001  16GT/s x16 AMD-PCI-to-PCI bridge Sec:Sub=[41:41]
    BDF=40:07.001  32GT/s x16 AMD-PCI-to-PCI bridge Sec:Sub=[42:42]
    BDF=60:01.001  16GT/s  x4 AMD-PCI-to-PCI bridge Sec:Sub=[61:61]
    BDF=60:07.001  32GT/s x16 AMD-PCI-to-PCI bridge Sec:Sub=[65:65]
    BDF=80:07.001  32GT/s x16 AMD-PCI-to-PCI bridge Sec:Sub=[81:81]
    BDF=a0:07.001  32GT/s x16 AMD-PCI-to-PCI bridge Sec:Sub=[a1:a1]
    BDF=c0:07.001  32GT/s x16 AMD-PCI-to-PCI bridge Sec:Sub=[c1:c1]
    BDF=e0:07.001  32GT/s x16 AMD-PCI-to-PCI bridge Sec:Sub=[e1:e1]
    BDF=e0:07.002  32GT/s x16 AMD-PCI-to-PCI bridge Sec:Sub=[e2:e2]


EndPoints
  AMD EndPoints
    BDF=03:00.000  32GT/s x16 AMD-Non-Essential Instrumentation
    BDF=03:00.001  32GT/s x16 AMD-Other system peripheral
```

Figure 5. Listing All PCIe Devices Sample Output

4.2.2 Margin Command Invocation

Use the following format to invoke the generic PCIe margining module:

```
./amdxio -pcie -margin -bdf=<Bus:Device:Function> -receiver=<receiver
number> -ber=<ber value> -errcnt=<Error Threshold>
```

Table 1. Generic PCIe Margining Command-Line Switches

Switch	Description	Sample
`-bdf`	Provides the bus, device, and function of the PCIe device. Inputs are in hex.	For example, to target a PCIe device with Bus 0x1, Device 0xA, and Function 0x0, use the following syntax: `-bdf=1:A.0`

`-receiver`	Specifies which device is to be targeted. Use 1 for root port, 6 for endpoint device, and 2–5 for retimers.	`-receiver=1`
		
		`-receiver=6`

Switch	Description	Sample
`-ber`	Specifies the bit error rate. This value is taken as a decimal.	`-ber=9`
`-errcnt`	Specifies the error threshold for margin.	`-errcnt=5`
`-lanes`	(Optional) Specifies the lanes that need to be margined. If not specified, all lanes of the device are margined.	`-lanes=12-15`

Notes:

1. *When targeting a root port, use the BDF for the root port and receiver number 1.*
2. *When targeting an endpoint, use the BDF for the endpoint and receiver number 6.*
3. *For retimers 2 and 3, the BDF is the same as the root port's, and the receiver number is 2 or 3 respectively.*

4.2.3 Examples

1. To invoke margining for the root port at Bus:0x2 Device:0x17 Function:0x1 with BER of 9, error count of 5, the invocation would be:

   ```
   amdxio -margin -pcie -bdf=2:17.1 -ber=9 -errcnt=5 -receiver=1
   ```

2. To invoke margining for an endpoint device at Bus:0x3 Device:0x17 Function:0x0 with BER of 9, error count of 5 the invocation would be:

   ```
   amdxio -margin -pcie -bdf=3:17.0 -ber=9 -errcnt=5 -receiver=6
   ```

3. To invoke margining for retimer 2 on root port at Bus:0x2 Device:0x17 Function:0x1 with BER of 9, error count of 5 the invocation would be:

   ```
   amdxio -margin -pcie -bdf=2:17.1 -ber=9 -errcnt=5 -receiver=2
   ```

4.2.4 Console Output

```
AMDXIO Version 5.0.7 (Beta Build)

Device Characteristics
  Speed is: 32GT/s
  Width is: x16

Checking PCIe device margin capabilities...
  Device supports margining.
  Timing Steps         : 0x10  Max Time Offset      : 0x19
  Volt Steps           : 0x3f  Max Volt Offset      : 0xd
  IndErrorSampler      : 0
  SampleReportingMethod: 0
  IndLeftRightTiming   : 1
  IndUpDownVoltage     : 1
  VoltageSupported     : 1
  Device Error threshold set to:63

Input Parameters
  Bus             : 0x0
  Device          : 0x1
  Function        : 0x1
  BER             : 9
  Dwell Time(ms)  : 149
  Error Threshold: 1
  Receiver No     : 1


Starting Vref Margining...

Dwell Time:50
  Vref Negative
    LaneID:  0 [Offset:39 ErrCnt:2]
    LaneID:  1 [Offset:43 ErrCnt:6]
    LaneID:  2 [Offset:40 ErrCnt:2]
```

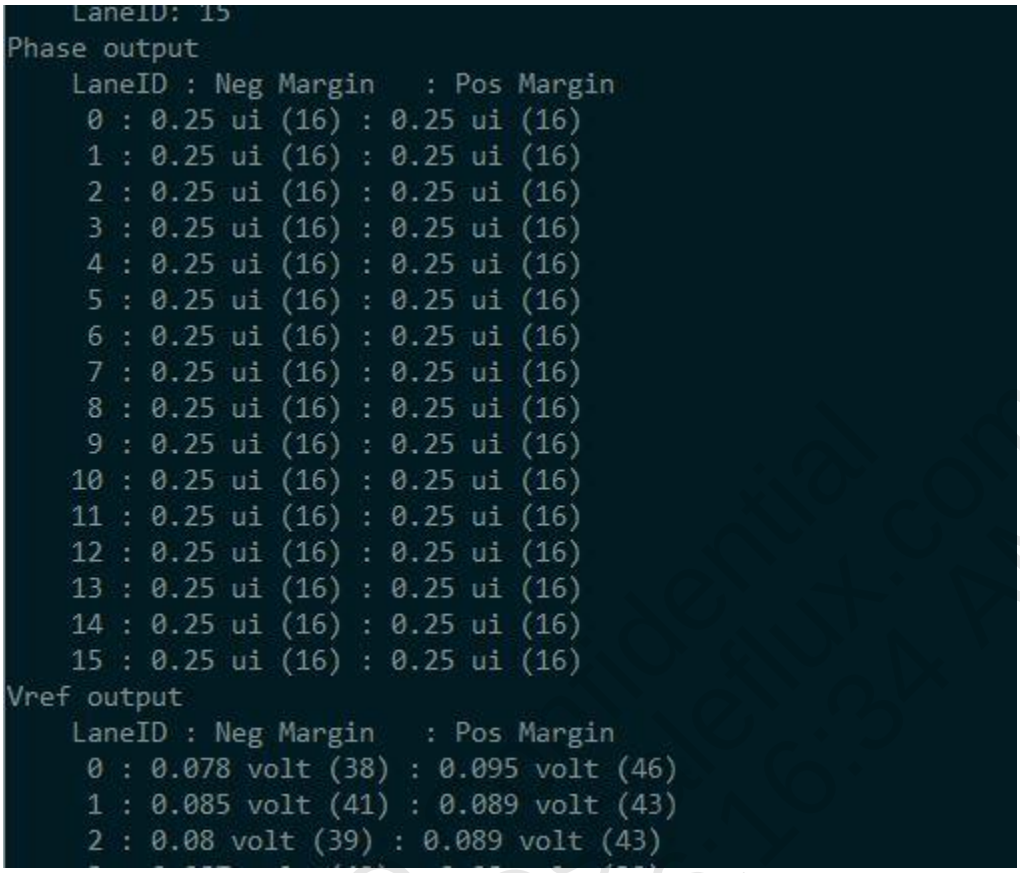


Figure 6. PCIe Margining Console Output

4.3 PCIe 4-Point Margining

The PCIe 4-point margining module supports only the command-line interface (no GUI).

Margining output is dumped in two formats:

- Console output
- CSV file

4.3.1 Getting a List of PCIe Devices

To get a list of all PCIe devices, both bridges and endpoints, use the command below and then select 4G and 5G devices from the list:

```
./amdxio -pcie -list -devices
```

```
AMDXIO Version 5.0.7 (Beta Build)
Bridges
  AMD Bridges
    BDF=00:01.001  32GT/s x16 AMD-PCI-to-PCI bridge Sec:Sub=[1:1]
    BDF=00:07.001  32GT/s x16 AMD-PCI-to-PCI bridge Sec:Sub=[3:3]
    BDF=20:07.001  32GT/s x16 AMD-PCI-to-PCI bridge Sec:Sub=[22:22]
    BDF=40:01.001  16GT/s x16 AMD-PCI-to-PCI bridge Sec:Sub=[41:41]
    BDF=40:07.001  32GT/s x16 AMD-PCI-to-PCI bridge Sec:Sub=[42:42]
    BDF=60:01.001  16GT/s  x4 AMD-PCI-to-PCI bridge Sec:Sub=[61:61]
    BDF=60:07.001  32GT/s x16 AMD-PCI-to-PCI bridge Sec:Sub=[65:65]
    BDF=80:07.001  32GT/s x16 AMD-PCI-to-PCI bridge Sec:Sub=[81:81]
    BDF=a0:07.001  32GT/s x16 AMD-PCI-to-PCI bridge Sec:Sub=[a1:a1]
    BDF=c0:07.001  32GT/s x16 AMD-PCI-to-PCI bridge Sec:Sub=[c1:c1]
    BDF=e0:07.001  32GT/s x16 AMD-PCI-to-PCI bridge Sec:Sub=[e1:e1]
    BDF=e0:07.002  32GT/s x16 AMD-PCI-to-PCI bridge Sec:Sub=[e2:e2]


EndPoints
  AMD EndPoints
    BDF=03:00.000  32GT/s x16 AMD-Non-Essential Instrumentation
    BDF=03:00.001  32GT/s x16 AMD-Other system peripheral
```

Figure 7. Listing All PCIe Devices Sample Output (4-Point Margining)

4.3.2 Margin Command Invocation

Use the following format to invoke the PCIe 4-point margining module:

```
./amdxio -margin -pcie -bdf=<bus:Device.Function> -ber=<BER>/-
dwelltime_ms=<Dwell Time in milliseconds> -receiver=<number> -
errcnt=<Error Threshold> -4pt -phaserange=<step:start.end> -
dacrange=<step:start.end>
```

Table 2. PCIe 4-Point Margining Command-Line Switches

Switch	Description	Sample
`-bdf`	Provides the bus, device, and function of the PCIe device. Inputs are in hex.	For example, to target a PCIe device with Bus 0x1, Device 0xA, and Function 0x0, use the following syntax: `-bdf=1:A.0`

Switch	Description	Sample

`-receiver`	Specifies which device is to be targeted. Use 1 for root port, 6 for endpoint device, and 2–5 for retimers.	`-receiver=1`
		
		`-receiver=6`

`-ber`	Specifies the bit error rate. This value is taken as a decimal.	`-ber=9`
`-dwelltime_ms`	Specifies the dwell time in milliseconds. This value is taken as a decimal.	`-dwelltime_ms=658`
`-errcnt`	Specifies the error threshold for margin.	`-errcnt=5`
`-lanes`	(Optional) Specifies the lanes that need to be margined. If not specified, all lanes of the device are margined.	`-lanes=12.15`
`-4pt`	Specifies that request is for 4-point lane margining.	
`-phaserange=<step:start.end>`	Input for timing margining where: step = margin offset step size start = start margin offset end = end margin offset All values are in decimals.	`-phaserange=2:6.10`
`-dacrange =<step:start.end>`	Input for voltage margining where: step = margin offset step size start = start margin offset end = end margin offset All values are in decimals.	`-dacrange =5:10.30`

Notes:

1. *When targeting a root port, use the BDF for the root port and receiver number 1.*
2. *When targeting an endpoint, use the BDF for the endpoint and receiver number 6.*
3. *For retimers 2 and 3, BDF is the same as the root port's, and the receiver number is 2 or 3 respectively.*
4. *You can provide either "-ber" or "-dwelltime_ms" as input.*

4.3.3 Examples

1. To invoke 4-point margining on all lanes of a given port:

-margin -pcie -bdf=<bus:Device.Function> -ber=<BER>/-dwelltime_ms=<Dwell Time in milliseconds> -receiver=<number> -errcnt=<Error Threshold> -4pt -phaserange=<step:start.end> -dacrange=<step:start.end>

Examples:

```
sudo ./amdxio -margin -pcie -bdf=40:1.1 -ber=9 -errcnt=5 -receiver=1
-4pt -phaserange=2:6.10 -dacrange=5:10.30

sudo ./amdxio -margin -pcie -bdf=40:1.1 -dwelltime_ms=100 -errcnt=5 -
receiver=1 -4pt -phaserange=2:6.10 -dacrange=5:10.30
```

2. To invoke 4-point margining on a single lane of a given port:

-margin -pcie -bdf=<bus:Device.Function> -ber=<BER>/-dwelltime_ms=<Dwell Time in milliseconds> -receiver=<number> -errcnt=<Error Threshold> -4pt - phaserange=<step:start.end> -dacrange=<step:start.end> -lanes=<single lane>

Example:

```
sudo ./amdxio -margin -pcie -bdf=40:1.1 -ber=9 -errcnt=5 -receiver=1
-4pt -phaserange=2:6.10 -dacrange=5:10.30 -lanes=0
```

3. To invoke 4-point margining on a custom lane range of a given port

-margin -pcie -bdf=<bus:Device.Function> -ber=<BER>/-dwelltime_ms=<Dwell Time in milliseconds> -receiver=<number> -errcnt=<Error Threshold> -4pt - phaserange=<step:start.end> -dacrange=<step:start.end> -lanes=<start lane>.<end lane>

Example:

```
sudo ./amdxio -margin -pcie -bdf=40:1.1 -ber=9 -errcnt=5 -receiver=1
-4pt -phaserange=2:6.10 -dacrange=5:10.30 -lanes=0.3
```

4.3.4 Console Output

```
root@UB2204dtcQqGeTaSmpIipxe:~/sonu/amdxio-5/cpp/linux_build# sudo ./amdxio -margin -pcie -bdf=40:1.1 -ber=9 -errcnt=5 -receiver=1 -4pt -phaserange=2:6.10 -dacrange=5:20.40
AMDXIO Version 5.1.5.0(Beta Build)
Input Information:
  Lane Number      : 0
  BER              : 9
  Err Cnt          : 5
  Dwell Time(ms)   : 658
Margin Capabilities:
  NumTimingSteps: 16
  MaxTimingOffset: 25
  NumVoltageSteps: 63
  MaxVoltageOffset: 13
PCIe Device Details:
  Segment    : "0"
  Bus        : "40"
  Device     : "1"
  Function   : "1"
  Speed      : "16GT/s"
  Width      : "x16"
Starting Vref Margining ...
Vref Negative
Exiting Lane: 0 Offset: 30 ErrCnt: 63
Exiting Lane: 1 Offset: 30 ErrCnt: 63
Exiting Lane: 2 Offset: 30 ErrCnt: 63
Exiting Lane: 3 Offset: 30 ErrCnt: 63
Exiting Lane: 4 Offset: 30 ErrCnt: 63
Exiting Lane: 5 Offset: 30 ErrCnt: 63
Exiting Lane: 6 Offset: 30 ErrCnt: 63
Exiting Lane: 7 Offset: 30 ErrCnt: 63
Exiting Lane: 8 Offset: 30 ErrCnt: 63
Exiting Lane: 9 Offset: 30 ErrCnt: 63
Exiting Lane: 10 Offset: 30 ErrCnt: 63
Exiting Lane: 11 Offset: 30 ErrCnt: 63
Exiting Lane: 12 Offset: 30 ErrCnt: 63
Exiting Lane: 13 Offset: 30 ErrCnt: 63
Exiting Lane: 14 Offset: 30 ErrCnt: 63
Exiting Lane: 15 Offset: 30 ErrCnt: 63
Vref Positive
Exiting Lane: 14 Offset: 25 ErrCnt: 29
Exiting Lane: 0 Offset: 30 ErrCnt: 63
Exiting Lane: 1 Offset: 30 ErrCnt: 63
Exiting Lane: 2 Offset: 30 ErrCnt: 63
Exiting Lane: 3 Offset: 30 ErrCnt: 63
Exiting Lane: 4 Offset: 30 ErrCnt: 63
Exiting Lane: 5 Offset: 30 ErrCnt: 63
Exiting Lane: 6 Offset: 30 ErrCnt: 63
Exiting Lane: 7 Offset: 30 ErrCnt: 63
Exiting Lane: 8 Offset: 30 ErrCnt: 63
Exiting Lane: 9 Offset: 30 ErrCnt: 63
Exiting Lane: 10 Offset: 30 ErrCnt: 63
Exiting Lane: 11 Offset: 30 ErrCnt: 63
Exiting Lane: 12 Offset: 30 ErrCnt: 63
Exiting Lane: 13 Offset: 30 ErrCnt: 63
Exiting Lane: 15 Offset: 30 ErrCnt: 63
Starting Phase Margining ...
Phase Negative
Exiting Lane: 0 Offset: 10 ErrCnt: 8
```

Figure 8. 4-Point Margining Console Output

4.3.5 File Output

Segment	Bus	Device	Func	LaneSeri	LanePCIe	LanePhy	BER	ErrCnt	NumTimir	MaxTimin	NumVolta	MaxVolta	Total Phas	Total Volt	Margin Le	Margin Ri	Margin To	Margin Bo	Margin Le	Margin Ri	Margin To	Margin Bott	Margin Lef	Margin Rig	Margin Top	Margin Bott	TxPreset	LCBestPre	LCLocalPri	USPTxPreset
0	40	1	1	0	0	9	5	16	25	63	13	0.390625	0.111429	0.15625	0.234375	0.053651	0.057778	10	15	26	28	39	0	8	7	7	9	6	4
0	40	1	1	1	1	9	5	16	25	63	13	0.375	0.107302	0.140625	0.234375	0.051587	0.055714	9	15	25	27	16	0	9	7	7	9	6	4
0	40	1	1	2	2	9	5	16	25	63	13	0.390625	0.111429	0.15625	0.234375	0.053651	0.057778	10	15	26	28	61	0	8	7	7	9	6	4
0	40	1	1	3	3	9	5	16	25	63	13	0.390625	0.111429	0.15625	0.234375	0.053651	0.057778	10	15	26	28	59	0	9	7	7	9	6	4
0	40	1	1	4	4	9	5	16	25	63	13	0.390625	0.111429	0.15625	0.234375	0.053651	0.057778	10	15	26	28	41	0	9	7	7	9	6	4
0	40	1	1	5	5	9	5	16	25	63	13	0.375	0.111429	0.140625	0.234375	0.053651	0.057778	9	15	26	28	22	0	8	16	7	9	6	4
0	40	1	1	6	6	9	5	16	25	63	13	0.390625	0.111429	0.15625	0.234375	0.053651	0.057778	10	15	26	28	40	0	8	7	7	9	6	4
0	40	1	1	7	7	9	5	16	25	63	13	0.390625	0.111429	0.15625	0.234375	0.053651	0.057778	10	15	26	28	45	0	8	8	7	9	6	4
0	40	1	1	8	8	9	5	16	25	63	13	0.390625	0.111429	0.15625	0.234375	0.053651	0.057778	10	15	26	28	46	0	8	7	7	9	6	4
0	40	1	1	9	9	9	5	16	25	63	13	0.390625	0.111429	0.15625	0.234375	0.053651	0.057778	10	15	26	28	63	0	7	7	7	9	6	4
0	40	1	1	10	10	9	5	16	25	63	13	0.375	0.109365	0.140625	0.234375	0.051587	0.057778	9	15	25	28	12	0	5	8	7	9	6	4
0	40	1	1	11	11	9	5	16	25	63	13	0.390625	0.111429	0.15625	0.234375	0.053651	0.057778	10	15	26	28	62	0	7	7	7	9	6	4
0	40	1	1	12	12	9	5	16	25	63	13	0.390625	0.111429	0.15625	0.234375	0.053651	0.057778	10	15	26	28	37	0	7	7	7	9	6	4
0	40	1	1	13	13	9	5	16	25	63	13	0.375	0.111429	0.140625	0.234375	0.053651	0.057778	9	15	26	28	8	0	6	7	7	9	6	4
0	40	1	1	14	14	9	5	16	25	63	13	0.34375	0.107302	0.109375	0.234375	0.049524	0.057778	7	15	24	28	14	0	12	7	7	9	6	4
0	40	1	1	15	15	9	5	16	25	63	13	0.390625	0.111429	0.15625	0.234375	0.053651	0.057778	10	15	26	28	34	0	6	7	7	9	6	4

Figure 9. 4-Point Margining File Output

Chapter 5 xGMI Margining

xGMI margining operations supported by the tool can be separated into three types:

- Margining of all xGMI links
- Margining of a specific xGMI link
- 4-point margining support

5.1 Margining All xGMI Links

When this command is invoked, the tool will auto-enumerate all xGMI links on the system and margin them in parallel. However, when doing this, the tool will take care to margin only the endpoint of a link at a given time.

For example, when margining G0 of Socket 0, the tool will not margin G2 of Socket 1, since G0 of Socket 0 is connected to G2 of Socket 1. But the tool can margin G1 of Socket 1 in the same batch because it is a separate link. Therefore, links will be split into two batches so that both endpoints of a link are not present in a single batch. Then all links in that batch will be margined in parallel.

5.1.1 Margin Command Invocation

```
sudo ./amdxio -xgmi -margin -all
```

5.1.2 Console Output

```
./amdxio -xgmi -margin -all -ber=8 -errcnt=1
   AMDXIO Version 5.1.10.0(Beta)
   Device selected for margining:
     Socket:0 Die:0 Link:G0
     Socket:0 Die:0 Link:G1
     Socket:0 Die:0 Link:G2
     Socket:0 Die:0 Link:G3
     Socket:1 Die:0 Link:G0
     Socket:1 Die:0 Link:G1
     Socket:1 Die:0 Link:G2
     Socket:1 Die:0 Link:G3

   Starting Margining on:
     Socket:0 Die:0 Link:G0
     Socket:0 Die:0 Link:G1
     Socket:0 Die:0 Link:G2
     Socket:0 Die:0 Link:G3
   ........................................
```

Figure 10. Margining All xGMI Links Console Output

5.1.3 File Output

The CSV file output will be dumped into a timestamped folder that will have margin data for all devices margined.

5.2 Margining a Specific xGMI Link

The tool can be invoked to margin a specific xGMI link. A user can specify which socket and link to margin. The option to specify a set of lanes also is available.

5.2.1 Margin Command Invocation

Use the following format to invoke xGMI margining for one lane:

```
amdxio -i=<device id> -xgmi -margin -lanes=<Link Name: Lane no>
```

Use the following format to invoke xGMI margining for a lane range:

```
amdxio -i=<device id> -xgmi -margin -lanes=<Link Name: Start Lane
no. End Lane No>
```

5.2.2 Console Output

```
./amdxio -i=0 -xgmi -margin -lanes=G0:0
AMDXIO Version 5.1.10.0(Beta)
Device selected for margining:
  Socket:0 Die:0 Link:G0

Starting Margining on:
  Socket:0 Die:0 Link:G0
Input Information:
  Socket         : 0
  Die            : 0
  Device Name    : "Stones BDF:0:18.0 Socket:0 Die:0"
  Link           : "G0"
  Lane Number    : 0
  BER            : 9
  Err Cnt        : 5
  Dwell Time(ms): 421
Margin Capabilities:
  NumTimingSteps: 16
  MaxTimingOffset: 25
  NumVoltageSteps: 63
  MaxVoltageOffset: 13
....Results:
  Top   Margin: 0.07 volt
  Bot   Margin: 0.068 volt
  Left  Margin: 0.19 ui
  Right Margin: 0.25 ui
```

Figure 11. Margining a Specific xGMI Link Console Output

5.2.3 File Output

Notes:

- *For device ID details, refer to Chapter 1 Device Listing and Selection.*
- *For lane and link listing and selection, refer to Chapter 2 Links and Lanes.*

5.2.4 Examples

1. To invoke xGMI margining for link XGMIPCS for Lane 3 on Device 0:

   amdxio -i=0 -xgmi -margin -lanes= XGMIPCS:3

2. To invoke xGMI margining for link XGMIPCS for Lanes 3 to 5 on Device 0:

   amdxio -i=0 -xgmi -margin -lanes= XGMIPCS:3.5

5.3 xGMI 4-Point Margining

The tool supports xGMI 4-point lane margining on AMD platforms. Margining output is dumped in two formats:

- Console output
- CSV file

Data are shown per lane with UI and voltage data.

5.3.1 Margin Command Invocation

Use the following format to invoke xGMI margining for one lane:

```
amdxio -i=<device id> -xgmi -margin -lanes=<Link Name: Lane no> -
dacrange=<step:start_offset.end_offset> -
phaserange=<step:start_offset.end_offset>
```

Use the following format to invoke xGMI margining for a lane range:

```
amdxio -i=<device id> -xgmi -margin -lanes=<Link Name: Start Lane
no. End Lane No> -dacrange=<step:start_offset.end_offset> -
phaserange=<step:start_offset.end_offset>
```

Notes:

- *For device ID details, refer to Chapter 1 Device Listing and Selection.*
- *For lane and link listing and selection, refer to Chapter 2 Links and Lanes.*

5.3.2 Examples

1. To invoke xGMI margining for link XGMIPCS for Lane 3 on Device 0:

   ```
   amdxio -i=0 -xgmi -margin -lanes= XGMIPCS:3 -dacrange=1:7.7 -
   phaserange=1:5.5
   ```

2. To invoke xGMI margining for link XGMIPCS for Lanes 3 to 5 on Device 0:

   ```
   amdxio -i=0 -xgmi -margin -lanes= XGMIPCS:3.5 -dacrange=1:7.7 -
   phaserange=1:5.5
   ```

Chapter 6 xGMI BER Polling

The AMDXIO tool supports polling of xGMI error counters. The user can specify the poll period as well as the poll interval for this test.

Command syntax:

```
./amdxio -i=<Device> -xgmi -pollber -pollinterval=<milliseconds> -
pollperiod=<seconds>
```

The tool will poll all xGMI links in the specified device for the specified poll-period duration, which is defined by the poll interval switch parameter. Sample output shown below:

```
# ./amdxio -i=0 -xgmi -pollber -pollinterval=1 -pollperiod=10
AMDXIO Version 5.1.13.0
Going to start XGMI BER polling with polling period of 10 secs and poll interval of 1ms.
Initalizing error registers
Checking error registers
Cleaning up.
Stones BDF:0:18.0 Socket:0 Die:0
    G0 Width:x16 Speed:32Gts Error Count:000 Packet Error Rate=000
    G1 Width:x16 Speed:32Gts Error Count:000 Packet Error Rate=000
    G2 Width:x16 Speed:32Gts Error Count:000 Packet Error Rate=000
    G3 Width:x16 Speed:32Gts Error Count:000 Packet Error Rate=000
```

Figure 12. Sample Output from xGMI BER Polling

Chapter 7 Eye Scan Command

The tool supports eye scan on AMD platforms. The eye scan command has the following outputs:

- Console output
- CSV file
- Eye scan image

7.1 How to Run Eye Scan

7.1.1 Identifying PCIe Lanes

To identify PCIe lanes, we can run an eye scan using the command below to get lane ranges. This will give the SerDes name and status for each lane:

```
amdxio -pcie -list -ports
```

For PCIe port listing and selection, refer to Chapter 3 PCIe Port Listing.

7.1.2 Identifying xGMI lanes

To identify xGMI lanes, run an eye scan using the command below to get lane ranges. This will give the SerDes name and status for each lane:

```
amdxio -i=<device id> -laneinfo
```

For lane and link listing and selection, refer to Chapter 2 Links and Lanes.

7.1.3 Running Eye Scan Command

Use the following format to invoke the eye scan for one lane:

```
amdxio -i=<device id> -eyescan -lanes=<Link Name: Lane no>
```

Use the following format to invoke the eye scan for a lane range:

```
amdxio -i=<device id> -eyescan -lanes=<Link Name: Start Lane no. End Lane No>
```

For device ID details, refer to Chapter 1 Device Listing and Selection.

7.1.4 Sample Commands

1. To invoke an eye scan for link XGMIPCS for Lane 3 on Device 0:

   amdxio -i=0 -eyescan -lanes= XGMIPCS:3

2. To invoke an eye scan for link XGMIPCS for Lanes 3 to 5 on Device 0:

   amdxio -i=0 -eyescan -lanes= XGMIPCS:3.5

7.2 Sample Outputs

7.2.1 Console Output

```
AMDXIO Version 5.0.7 (Beta Build)
Going to start eye scan
Link:          SERDESA
Lane:          0
Phase start:   -32
Phase end:     32
Phase step:    4
Voltage start: -200
Voltage end:   200
Voltage step:  10
nbits:         23

Data collection complete

PuTTY X11 proxy: Unsupported authorisation protocol
Results and images are available in folder: 2022_12_15-05_53_24
```

Figure 13. Eye Scan Console Output

7.2.2 File Outputs

Two files will be created: a PNG file of the eye scan image and a CSV file with information on top, bottom, left, right, etc.

```
analysis.csv  socket_0_die_1_SERDESA_0_eye.png
```

Figure 14. Eye Scan CSV

An eye scan sample image is shown here:

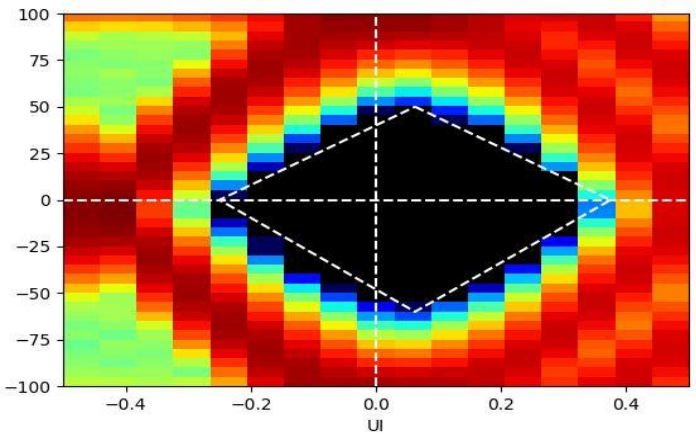
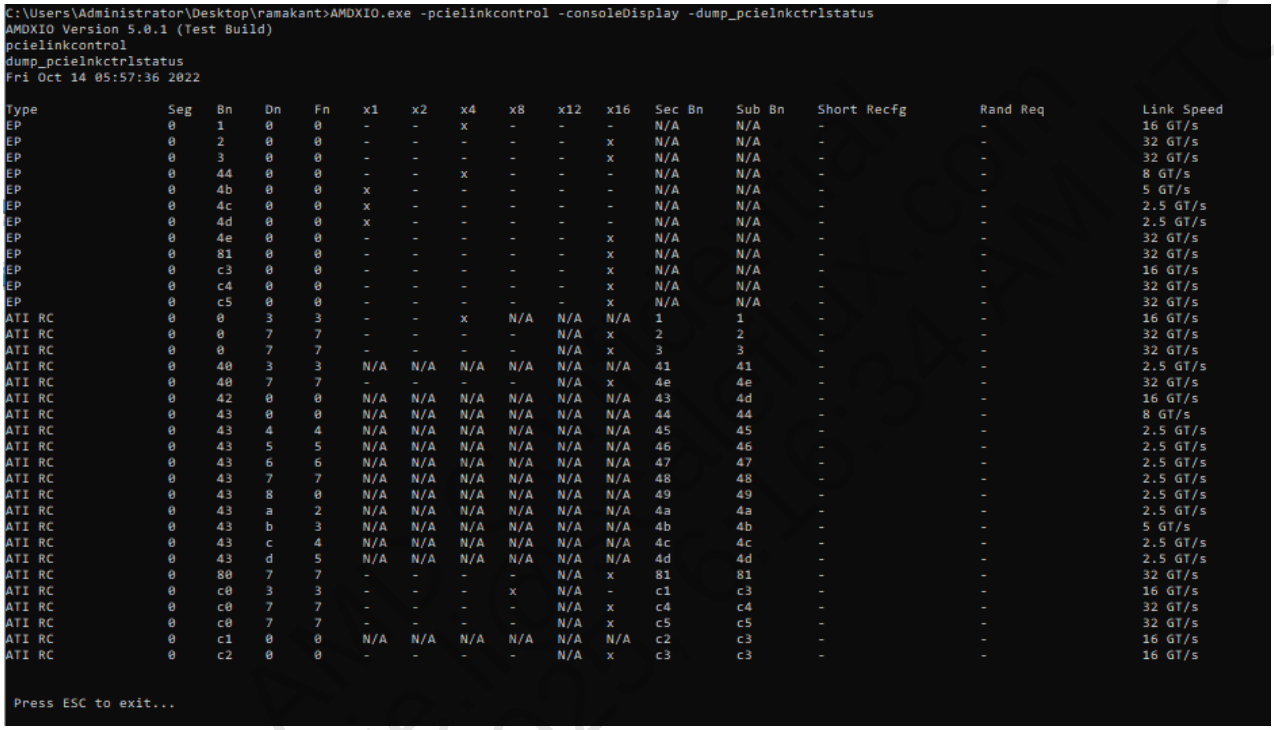


Figure 15. Eye Scan Sample Image

Chapter 8 PCIe Link Control

The PCIe Link Control tab is used to monitor the status of PCIe devices. You can monitor the root complex (RC) as well as the endpoint (EP) of supported devices. The monitoring includes PCIe device number, bus number, link speed, and link width for each device. A user also has the option to set certain parameters on supported devices, such as link speed and link width.



```
C:\Users\Administrator\Desktop\ramakant>AMDXIO.exe -pcielinkcontrol -consoleDisplay -dump_pcielnkctrlstatus
AMDXIO Version 5.0.1 (Test Build)
pcielinkcontrol
dump_pcielnkctrlstatus
Fri Oct 14 05:57:36 2022

Type            Seg  Bn  Dn  Fn  x1  x2  x4  x8  x12 x16 Sec Bn Sub Bn Short Recfg Rand Req Link Speed
EP              0    1   0   0   -   -   x   -   -   -   N/A    N/A    -          -        16 GT/s
EP              0    2   0   0   -   -   -   -   -   x   N/A    N/A    -          -        32 GT/s
EP              0    3   0   0   -   -   -   -   -   x   N/A    N/A    -          -        32 GT/s
EP              0    44  0   0   -   -   x   -   -   -   N/A    N/A    -          -        8 GT/s
EP              0    4b  0   0   x   -   -   -   -   -   N/A    N/A    -          -        5 GT/s
EP              0    4c  0   0   x   -   -   -   -   -   N/A    N/A    -          -        2.5 GT/s
EP              0    4d  0   0   x   -   -   -   -   -   N/A    N/A    -          -        2.5 GT/s
EP              0    4e  0   0   -   -   -   -   -   x   N/A    N/A    -          -        32 GT/s
EP              0    81  0   0   -   -   -   -   -   x   N/A    N/A    -          -        32 GT/s
EP              0    c3  0   0   -   -   -   -   -   x   N/A    N/A    -          -        16 GT/s
EP              0    c4  0   0   -   -   -   -   -   x   N/A    N/A    -          -        32 GT/s
EP              0    c5  0   0   -   -   -   -   -   x   N/A    N/A    -          -        32 GT/s
ATI RC          0    0   3   3   -   -   x   N/A N/A N/A 1      1      -          -        16 GT/s
ATI RC          0    0   7   7   -   -   -   -   N/A x   2      2      -          -        32 GT/s
ATI RC          0    0   7   7   -   -   -   -   N/A x   3      3      -          -        32 GT/s
ATI RC          0    40  3   3   N/A N/A N/A N/A N/A N/A 41     41     -          -        2.5 GT/s
ATI RC          0    40  7   7   -   -   -   -   N/A x   4e     4e     -          -        32 GT/s
ATI RC          0    42  0   0   N/A N/A N/A N/A N/A N/A 43     4d     -          -        16 GT/s
ATI RC          0    43  0   0   N/A N/A N/A N/A N/A N/A 44     44     -          -        8 GT/s
ATI RC          0    43  4   4   N/A N/A N/A N/A N/A N/A 45     45     -          -        2.5 GT/s
ATI RC          0    43  5   5   N/A N/A N/A N/A N/A N/A 46     46     -          -        2.5 GT/s
ATI RC          0    43  6   6   N/A N/A N/A N/A N/A N/A 47     47     -          -        2.5 GT/s
ATI RC          0    43  7   7   N/A N/A N/A N/A N/A N/A 48     48     -          -        2.5 GT/s
ATI RC          0    43  8   0   N/A N/A N/A N/A N/A N/A 49     49     -          -        2.5 GT/s
ATI RC          0    43  a   2   N/A N/A N/A N/A N/A N/A 4a     4a     -          -        2.5 GT/s
ATI RC          0    43  b   3   N/A N/A N/A N/A N/A N/A 4b     4b     -          -        5 GT/s
ATI RC          0    43  c   4   N/A N/A N/A N/A N/A N/A 4c     4c     -          -        2.5 GT/s
ATI RC          0    43  d   5   N/A N/A N/A N/A N/A N/A 4d     4d     -          -        2.5 GT/s
ATI RC          0    80  7   7   -   -   -   -   N/A x   81     81     -          -        32 GT/s
ATI RC          0    c0  3   3   -   -   -   x   N/A -   c1     c3     -          -        16 GT/s
ATI RC          0    c0  7   7   -   -   -   -   N/A x   c4     c4     -          -        32 GT/s
ATI RC          0    c0  7   7   -   -   -   -   N/A x   c5     c5     -          -        32 GT/s
ATI RC          0    c1  0   0   N/A N/A N/A N/A N/A N/A c2     c3     -          -        16 GT/s
ATI RC          0    c2  0   0   -   -   -   -   N/A x   c3     c3     -          -        16 GT/s


 Press ESC to exit...
```

Figure 16. Link Control Status

8.1 Features

The monitoring window in Figure 16. Link Control Status is a table type with each row indicating a device and columns indicating values for that device.

- Bn and Dn indicate the corresponding Bus Number and Device Number of the device.
- For the link width columns (x1, x2, x4, x8, x12, x16), a "-" indicates it is available to be toggled at that specific speed, while "X" means the device is currently at that speed. "N/A" means that speed is not available to set.
- Sec Bn and Sub Bn display the corresponding Secondary Bus Number and Sub Bus Number of the device.

- Link Speed is the current PCIe speed of the device, which can be set using the speed change command. When you change the link speed through the command, the changed link speed will show on the monitoring column.

8.2 PCIe Link Control Options

Table 3. PCIe Link Control Options

Command Options	Description
`-seg=<#>`	Segment number (Default = 0)
`-bn=<#>`	Bus number
`-dn=<#>`	Device number
`-fn=<#>`	Function number
`-linkspeed=<#>`	1. 2.5 GT/s 2. 5 GT/s 3. 8 GT/s 4. 16 GT/s 5. 32 GT/s
`-dump_pcielnkctrlstatus`	Log link control status to LinkControlStatus.txt
`-consoleDisplay`	Console display
`-logToFile`	Log to file
`-continuousRunInterval=#`	1. 50 ms 2. 500 ms 3. 1000 ms 4. 5000 ms
`-continuousRun`	Continuous run flag
`-loopCount=#`	Continuous run loop count
`-continuousWidthChange`	Continuous link width change. Used with `-continuousrun` flag.
`-continuousSpeedChange`	Continuous link width change. Used with `-continuousrun` flag.
`-continuousLinkSpeedLower=#`	1. Gen1, 2. Gen2, 3. Gen3, 4. Gen4, 5. Gen5
`-continuousLinkSpeedUpper=#`	1. Gen1, 2. Gen2, 3. Gen3, 4. Gen4, 5. Gen5
`-continuousLinkWidthLower=#`	1. X1, 2. X2, 4. X4, 8. X8, 12. X12, 16. X16
`-continuousLinkWidthUpper=#`	1. X1, 2. X2, 4. X4, 8. X8, 12. X12, 16. X16
`-onlySelectedIndex`	Display runtime information of only selected index

Chapter 9 PCIe Performance Counters

PCIe IP performance counters can be used to measure overall system performance or to help debug and diagnose hardware problems.

9.1 PCIe Performance Counter Options

Table 4. PCIe Performance Counter Options

Command Options	Description
`-PerfCounterStart`	Start performance counter
`-dump_pciePerfCounter`	Log PCIe performance counter options to PCIePerformanceCounterOptions.txt
`-consoleDisplay`	Console display
`-enableDecimalDisplay`	Enable decimal display
`-disableDecimalDisplay`	Disable decimal display
`-enableLogging`	Enable logging
`-disableLogging`	Disable logging
`-enableChangesOnly`	Enable changes-only flag
`-disableChangesOnly`	Disable changes-only flag
`-LogFile<=fileName>`	Log file (default filename is count.txt)
`-enableControlsInitCounter`	Enable controls initialization counters
`-disableControlsInitCounter`	Disable controls initialization counters
`-enableControlsReadCounter`	Enable controls read counters
`-disableControlsReadCounter`	Disable controls read counters
`-enableControlsStartMonitor`	Enable controls start monitor
`-enableCounter`	Enable counter for port/counter selected index

Command Options	Description
`-disableCounter`	Disable counter for port/counter selected index
`-spacePortCounterSelIndex [=<#1>-<#2>-<#3>-<#4>,<#1>-<#2>-<#3>-<#4>]`	Performance counter selection index. Default is 0 for all <#>. Individual option selection using comma operator (<#1>-<#2>-<#3>-<#4>,<#1>-<#2>-<#3>-<#4>). <#1> = SpaceIndex, <#2> = CounterIndex, <#3> = PortSelectionIndex, <#4> = EventSelectionIndex

9.2 PCIe Performance Counter Sample Commands

PCIe performance counter help:

```
sudo ./amdxio -pcieperfcounter -help
```

```
root@UB2204dtcQqGeTaSmpIipxe:~/amdxio-5# sudo ./amdxio -pcieperfcounter -help
AMDXIO Version 5.1.6.0(Release)

Performance Counters

<options>
  -PerfCounterStart             Start perf counter
  -dump_pciePerfCounter         Log PCIe Performance counter options to PCIePerformanceCounterOptions.txt
  -consoleDisplay               Console display.
  -enableDecimalDisplay         Enable decimal display.
  -disableDecimalDisplay        Disable decimal display.
  -enableLogging                Enable logging.
  -disableLogging               Disable logging.
  -enableChangesOnly            Enable changes only flag.
  -disableChangesOnly           Disable changes only flag.
  -LogFile<=fileName>           Log File. Default filename is count.txt
  -enableControlsInitCounter    Enable Controls Init Counters
  -disableControlsInitCounter   Disable Controls Init Counters
  -enableControlsReadCounter    Enable Controls Read Counters
  -disableControlsReadCounter   Disable Controls Read Counters
  -enableControlsStartMonitor   Enable Controls Start Monitor
  -enableCounter                Enable Counter for port/counter selected index.
  -disableCounter               Disable Counter for port/counter selected index.
  -spacePortCounterSelIndex[=<#1>-<#2>-<#3>-<#4>,<#1>-<#2>-<#3>-<#4>].
                                Perf counter Selection index. Default is 0 for all <#>
                                Individual option selection using comma operator(<#1>-<#2>-<#3>-<#4>,<#1>-<#2>-<#3>-<#4>).
                                <#1> = SpaceIndex, <#2> = CounterIndex, <#3> = PortSelectionIndex, <#4> = EventSelectionIndex,

  Sample command.
  -pcieperfcounter -consoleDisplay -dump_pciePerfCounter
  -pcieperfcounter -consoleDisplay -spacePortCounterSelIndex=0-4-0-0,0-5-0-0 -enableControlsReadCounter -enableCounter -PerfCounterStart -enableLogging
```

Figure 17. PCIe Performance Counter Help Sample Output

Display performance counter lists on console:

```
sudo ./amdxio -pcieperfcounter -consoleDisplay
```

```
root@UB2204dtcQqGeTaSmpIipxe:~/amdxio-5# sudo ./amdxio -pcieperfcounter -consoleDisplay
AMDXIO Version 5.1.6.0(Release)


Tue Apr 25 14:46:15 2023

Space                      Name                             Port(BN:DN.FN) Value    Rate (/s)    Max Rate (/s)  Enabled
PCIE0_nbio0 NAK            0. PCIE Rx number of NACK received    60:1.1                                          No
PCIE0_nbio0 NAK            0. PCIE Rx number of NACK generated   60:1.1                                          No
PCIE0_nbio0 LC TXCLK       0. Disabled                      60:1.1                                              No
PCIE0_nbio0 LC TXCLK       0. Disabled                      60:1.1                                              No
PCIE0_nbio0 LC TXCLK       0. Disabled                      60:1.1                                              No
PCIE0_nbio0 LC TXCLK       0. Disabled                      60:1.1                                              No
PCIE0_nbio0 RX TXCLK       0. Disabled                      60:1.1                                              No
PCIE0_nbio0 RX TXCLK       0. Disabled                      60:1.1                                              No
PCIE0_nbio0 RX TXCLK       0. Disabled                      60:1.1                                              No
PCIE0_nbio0 RX TXCLK       0. Disabled                      60:1.1                                              No
PCIE0_nbio0 RX TXCLK       0. Disabled                      60:1.1                                              No
PCIE0_nbio0 RX TXCLK       0. Disabled                      60:1.1                                              No
PCIE0_nbio0 RX TXCLK       0. Disabled                      60:1.1                                              No
PCIE0_nbio0 RX TXCLK       0. Disabled                      60:1.1                                              No
PCIE0_nbio0 TX TXCLK       0. Disabled                      60:1.1                                              No
PCIE0_nbio0 TX TXCLK       0. Disabled                      60:1.1                                              No
PCIE0_nbio0 TX TXCLK       0. Disabled                      60:1.1                                              No
PCIE0_nbio0 TX TXCLK       0. Disabled                      60:1.1                                              No
PCIE0_nbio0 TX TXCLK       0. Disabled                      60:1.1                                              No
PCIE0_nbio0 TX TXCLK       0. Disabled                      60:1.1                                              No
PCIE0_nbio0 TX TXCLK       0. Disabled                      60:1.1                                              No
PCIE0_nbio0 TX TXCLK       0. Disabled                      60:1.1                                              No
PCIE0_nbio0 RX SCLK        0. Disabled                      60:1.1                                              No
PCIE0_nbio0 RX SCLK        0. Disabled                      60:1.1                                              No
PCIE0_nbio0 RX SCLK        0. Disabled                      60:1.1                                              No
PCIE0_nbio0 RX SCLK        0. Disabled                      60:1.1                                              No
PCIE0_nbio0 RX SCLK        0. Disabled                      60:1.1                                              No
PCIE0_nbio0 RX SCLK        0. Disabled                      60:1.1                                              No
PCIE0_nbio0 RX SCLK        0. Disabled                      60:1.1                                              No
```

Figure 18. Display Performance Counter Lists on Console Sample Output

Dump PCIe performance counter details into a file:

```
sudo ./amdxio -pcieperfcounter -dump_pciePerfCounter
```

```
root@UB2204dtcQqGeTaSmpIipxe:~/amdxio-5# sudo ./amdxio -pcieperfcounter -dump_pciePerfCounter
AMDXIO Version 5.1.6.0(Release)
root@UB2204dtcQqGeTaSmpIipxe:~/amdxio-5#
root@UB2204dtcQqGeTaSmpIipxe:~/amdxio-5# ls
PCIePerfCounterOptions.txt  amdxio
```

Figure 19. Dump PCIe Performance Counter Details Sample Output

This creates a “PCIePerfCounterOptions.txt” file, which contains detailed information.

Monitor events:

1. To monitor an event, first dump the details into a file using this switch: “`-dump_pciePerfCounter`”

2. The switch to select events to monitor is "`-spacePortCounterSelIndex[=<#1>-<#2>-<#3>-<#4>,<#1>-<#2>-<#3>-<#4>]`". This takes four arguments:

   <#1> = SpaceIndex [Socket Index]
   <#2> = Perf Counter Index
   <#3> = Port Selection Index
   <#4> = Event Selection Index

   This information will be available in the "PCIePerfCounterOptions.txt" file.

3. Select the required details and use the following command:

   `sudo ./amdxio -pcieperfcounter -consoleDisplay -spacePortCounterSelIndex=[<#1>-<#2>-<#3>-<#4>,>,<#1>-<#2>-<#3>-<#4>]" <#enableControlsReadCounter -enableCounter -PerfCounterStart -enableLogging`

   For example:
   `sudo ./amdxio -pcieperfcounter -consoleDisplay -spacePortCounterSelIndex=0-2-0-41 -enableControlsReadCounter -enableCounter -PerfCounterStart -enableLogging`

```
Tue Apr 25 15:20:07 2023
PCIE0_nbio0 LC TXCLK: 41. LC_PERF_SpdChg_A on 60:1.1: 0x0000000000 Rate: 0x00000000 Max Rate: 0x00000000

Press ESC to exit...
Tue Apr 25 15:20:08 2023
PCIE0_nbio0 LC TXCLK: 41. LC_PERF_SpdChg_A on 60:1.1: 0x0000000001 Rate: 0x00000001 Max Rate: 0x00000001

Press ESC to exit...
Tue Apr 25 15:20:09 2023
PCIE0_nbio0 LC TXCLK: 41. LC_PERF_SpdChg_A on 60:1.1: 0x0000000002 Rate: 0x00000001 Max Rate: 0x00000001

Press ESC to exit...
Tue Apr 25 15:20:10 2023
PCIE0_nbio0 LC TXCLK: 41. LC_PERF_SpdChg_A on 60:1.1: 0x0000000002 Rate: 0x00000000 Max Rate: 0x00000001
```

Figure 20. Monitoring Single Event Sample Output

This is monitoring "LC_PERF_SpdChg_A" event for PCIE0_nbio0 60:1.1 port.

Event information is logged every second in the "count.txt" file.

4. To log event if only some events occur, we can use the switch "`-enableChangesOnly`".

   For example:
   ```
   sudo ./amdxio -pcieperfcounter -consoleDisplay -
   spacePortCounterSelIndex=0-2-0-41 -enableControlsReadCounter -
   enableCounter -PerfCounterStart -enableLogging -enableChangesOnly
   ```

5. To monitor multiple events together, we can pass multiple arguments in:

   `-spacePortCounterSelIndex[=<#1>-<#2>-<#3>-<#4>,<#1>-<#2>-<#3>-<#4>,<#1>-<#2>-<#3>-<#4>]`

   For example:
   ```
   sudo ./amdxio -pcieperfcounter -consoleDisplay -
   spacePortCounterSelIndex=0-2-0-3,0-  3-0-41 -
   enableControlsReadCounter -enableCounter -PerfCounterStart -
   enableLogging -enableChangesOnly
   ```

```
Tue Apr 25 15:41:58 2023

PCIE0_nbio0 LC TXCLK: 3. LC_PERF_Retrain_A on 60:1.1: 0x0000000032 Rate: 0x00000032 Max Rate: 0x00000032
PCIE0_nbio0 LC TXCLK: 41. LC_PERF_SpdChg_A on 60:1.1: 0x0000000000 Rate: 0x00000000 Max Rate: 0x00000000

Tue Apr 25 15:42:27 2023

PCIE0_nbio0 LC TXCLK: 41. LC_PERF_SpdChg_A on 60:1.1: 0x0000000001 Rate: 0x00000001 Max Rate: 0x00000001

Tue Apr 25 15:42:28 2023

PCIE0_nbio0 LC TXCLK: 41. LC_PERF_SpdChg_A on 60:1.1: 0x0000000002 Rate: 0x00000001 Max Rate: 0x00000001

Tue Apr 25 15:42:50 2023

PCIE0_nbio0 LC TXCLK: 3. LC_PERF_Retrain_A on 60:1.1: 0x0000000033 Rate: 0x00000001 Max Rate: 0x00000032
```

Figure 21. Monitoring Multiple Events Sample Output

Chapter 10 PCIe Port Status

Table 5. PCIe Port Status Options

Command Options	Description
`-seg=<#>`	Segment number (default = 0)
`-bn=<#>`	Bus number
`-dn=<#>`	Device number
`-fn=<#>`	Function number
`-dump_pcieportstatus[=<#>]`	Log port status to PCIEPortStatus.log. # is the port index.
`-dump_pcieportoptions`	Log port status to PCIEPortOptions.txt
`-consoleDisplay`	Console display of status/option. Used along with: `-dump_pcieportstatus/-dump_pcieportoptions`.
`-logToFile`	Save status/option to file. Used along with: `-dump_pcieportstatus/-dump_pcieportoptions`.
`-pcieportindex=<#>`	Port index
`-advErrorEnabled`	Advanced error enabled
`-advErrorDisabled`	Advanced error disabled
`-hidePortEnabled`	Hide port enabled
`-hidePortDisabled`	Hide port disabled
`-linkDisabled`	Link disabled
`-linkEnabled`	Link enabled
`-secBusResetEnable`	Secondary bus reset enabled
`-secBusResetDisable`	Secondary bus reset disabled
`-clearAdvErrorRegisters`	Clear advanced error registers
`-contLoopDisableEnable`	Continuous loop options enable/disable
`-contLoopsecBusReset`	Continuous loop options: secondary bus reset
`-contLoopLinkRetrain=#`	Continuous loop options: retrain link (speed # is 1: Gen1, 2: Gen2, 3: Gen3, 4: Gen4, 5: Gen5)

Command Options	Description
`-contLoopStopOnError`	Continuous loop options: stop on error
`-contLoopContRun=<#>`	Continuous loop options ContinuousRun default wait interval is 0 (5 s). 0: 5 s, 1: 4 s, 2: 3 s, 3: 2 s, 4: 1 s, 5: 750 msec, 6: 500 msec, 7: 300 msec, 8: 50 msec
`-contLoopCount=<#>`	Continuous loop options. (ContinuousRun default value is 0xFFFF)
`-contLoopStopOnError`	Continuous loop options: stop on error
`-commandFile=#`	Command file as input for all options
`-createSampleCommandFile[=#]`	Create sample command file for continuous run (# for type of command file, default 0)

Chapter 11 PCIe AER Error Monitor

AER Error Monitor is used for reporting correctable, uncorrectable, and recovery counter errors. The AER Error Monitor feature is enabled with the `-pcie -aermonitor =< DisplayTimer in seconds >, < PollingTimer in seconds>` flag.

Description of inputs:

- DisplayTimer: Time interval at which the tool reports AER status
- PollingTimer: Time interval at which the tool polls errors from the system

Example Command:

```
./amdxio -i=0,1 -pcie -aermonitor=60,5
```

Chapter 12 AMDXIO USR

12.1 USR Link Status

```
-usr -linkstatus
```

This command can be invoked either on a particular device using the "`-i`" switch or on all devices without the "`-i`" switch.

Sample commands:

1. On a specific device:

   ```
   ./amdxio -i=<device id>  -usr  -linkstatus
   ```

2. On all devices:

   ```
   ./amdxio -usr  -linkstatus
   ```

Output:

The output will list each USR instance and its link state. Additionally, output will include the BER count per instance, if available.

12.2 USR Performance Counter

USR instances have a performance counter that can be used to monitor certain events. The AMDXIO tool supports monitoring these events.

Two main commands are supported for USR performance: one to dump all USR performance counters and associated events, and another for actual monitoring of events.

12.2.1 USR Performance Dump

```
-usr -perfdump
```

This command can be invoked either on a particular device using the "`-i`" switch or on all devices without the "`-i`" switch.

Sample commands:

1. On a specific device:

```
./amdxio -i=<device id>  -usr  - perfdump
```

2. On all devices:

```
./amdxio -usr  - perfdump
```

Output:

This command dumps all performance counter instances per USR instance in the selected device. Additionally, the tool will list the exact command syntax to invoke each event for that instance. Users can directly copy that syntax and invoke it.

12.2.2 USR Performance Control

```
-usr -pollperiod=<in seconds> -i = <device ID>  -perfcounter=
LinkName:Type.InstanceID - event0=<event value>
```

This command can be invoked on a particular device using the "`-i`" switch. To invoke it on multiple devices, append each device on the command line. Refer to the example below.

Example command:

1. On a specific device:

```
amdxio -usr -pollperiod=2 -i=0-perfcounter=4CH:Ctrl.0 - event0=12
```

2. On multiple devices or instances

```
amdxio -usr -pollperiod=2  -i=0-perfcounter=4CH:Ctrl.0 - event0=12 -
i=0-perfcounter=4CH:Ctrl.1  - event0=12 -i=1-perfcounter=4CH:Ctrl.0
- event0=12
```

Output:

The command will set the performance event selections. Then, in poll-period intervals (specified in the command line), it will poll performance counters and dump the value to console if it has changed.

Chapter 13 GUI

When AMDXIO is installed, there will be two binaries in the installed folder:

- amdxio_ui → GUI binary
- amdxio → command-line binary

13.1 Launching the GUI

On the command line, invoke the GUI binary. It will launch a web server with a URL that should be opened:

```
root@ausquartz63d5:~/tmp# ./amdxio_ui
Started server at 10.138.132.33:9970
Open the link in any local/remote Web Browser (chrome recommended):  http://10.138.132.33:9970
```

13.2 PCIe Margin Tab

In the PCIe margin tab, users can do margining and eye scans of PCIe devices. Margin and eye scan of each device are displayed together.

Figure 22. PCIe Margin Tab

Users can select the BER, lanes to margin, error count, and receiver type for margining and start the test. Similarly, users also can select the BER lanes for the eye scan test.

Once a test is started, active tests will be shown in the active tests tab. When a test is complete, the results can be viewed in the results tab. Please see Section 13.4 “Results Tab” for more information.

13.3 xGMI Margin and Eye Scan

Margining and eye scans of xGMI links can be run in the xGMI margin tab. The options, margin, and eye scan of each device are displayed together.

Figure 23. xGMI Margin Tab

Users can select the BER, lanes to margin, error count, and receiver type for margin and start the test. Similarly, users can select the BER and lanes for the eye scan test.

Once the test is started, active tests will be shown in the active tests tab. When the test is complete, results can be viewed in the results tab. Please see Section 13.4 “Results Tab” for more information.

13.4 Results Tab

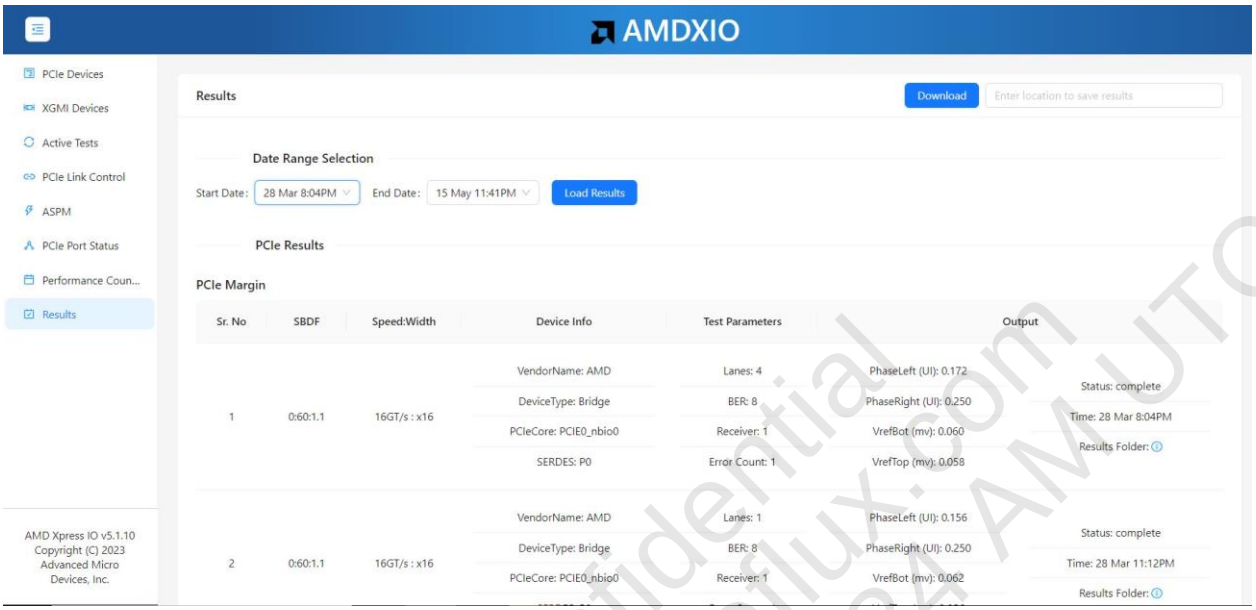


Figure 24. Results Tab

The results tab will display results of all PCIe margining, xGMI margining, and eye scan tests run on the machine from the GUI. Users can select the date range to view.

Left, right, top, and bottom values are listed in the GUI itself.

If more details are required, users can view the files in the output folder, whose location is shown when the mouse is hovered over the Results Folder field under the Output column.

13.5 PCIe Link Control Tab

The PCIe Link Control tab is used to monitor the status of PCIe devices. Users can monitor the root complex (RC) and endpoint (EP) of supported devices. Monitoring includes PCIe device number, bus number, link speed, and link width for each device. Users also have the option to set certain parameters on supported devices, such as link speed and link width.

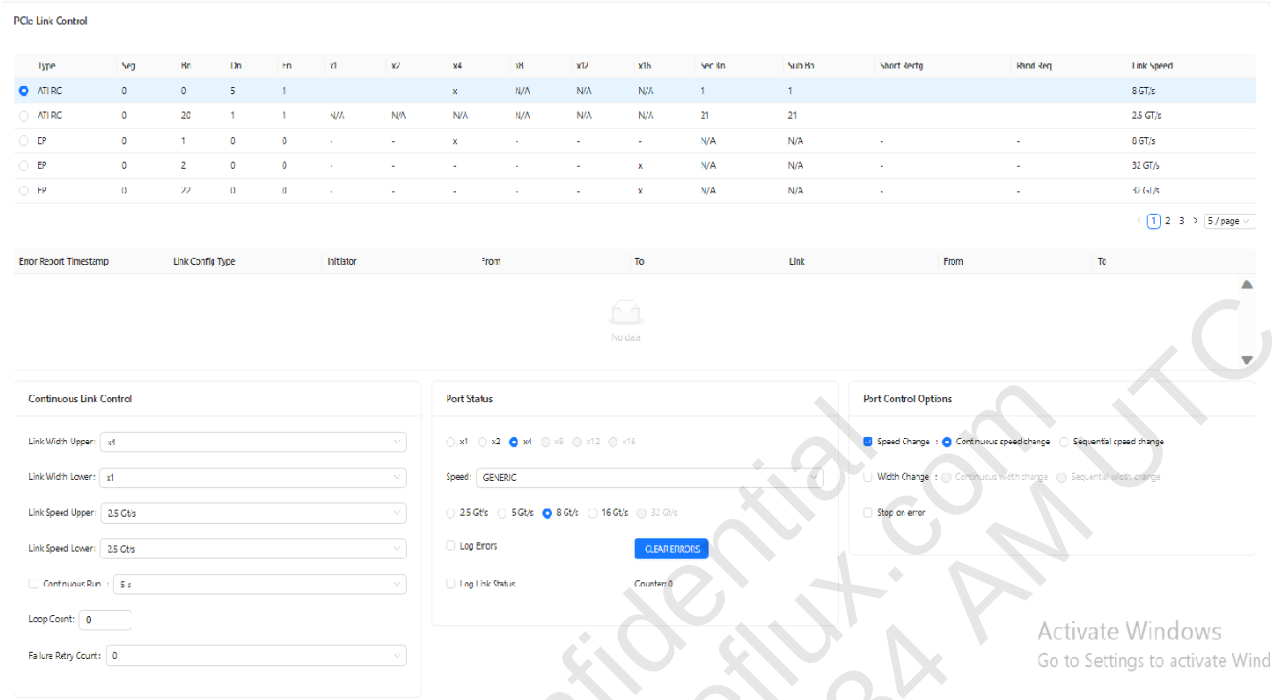


Figure 25. PCIe Link Control Tab

13.5.1 Features

13.5.1.1 PCIe Link Monitoring and Control

The monitoring window is a table type with each row indicating a device while columns indicate values for that device.

- Bn and Dn indicate the corresponding bus number and device number of the device.
- For the link width columns (x1, x2, x4, x8, x12, x16), a "-" indicates availability to be toggled at that specific speed, while "X" means the device is currently at that speed. "N/A" means that speed is not available to set.
- Sec Bn and Sub Bn display, respectively, the corresponding secondary bus number and sub bus number of the device.
- Link speed is the device's current PCIe speed, which can be set in the device setting box. When a user changes the link speed in the device setting box, the changed link speed will show in the monitoring column.

13.5.1.2 Port Control Options and Continuous Link Control

Width Change: To configure continuous width change:

- Enable the “Width Change” checkbox.
- Select a radio button for either Sequential Width Change or Continuous Width Change.
- In the “Continuous Link Control” menu, set upper and lower limits for the link width (“Link Width Upper” and “Link Width Lower”).
- Enable the “Continuous Run” checkbox.

Speed Change: To configure continuous speed change:

- Enable the “Speed Change” checkbox.
- Select a radio button for either Sequential Speed Change or Continuous Speed Change.
- In the “Continuous Link Control” menu, set upper and lower limits for the link speed (“Link Speed Upper” and “Link Speed Lower”).
- Enable the “Continuous Run” checkbox.

Other Options:

- Stop on error: Enable this checkbox if you want the continuous width/speed change stopped upon error.
- Sequential speed change/Sequential width change: The sequential speed and width change features test all speed and link width combinations supported by the PCIe device under test.

13.5.1.3 Error Logging and Device Setting

Users can set device link speed and width using the device control box. A user also can configure Continuous Link Control, which allows continuous switching in link speed and link width within certain values. When an error occurs, the error logging box will display it and explain where it occurred.

13.6 Active State Power Management (ASPM)

The Active State Power Management (ASPM) tab displays and controls devices' PCIe power management functionality and L0/L1 status and control.

The screen displays a PCIe device’s root complex (RC), endpoint (EP), and bridge (BR) in the system (including DGPU/SOC), as well as ASPM status and capability and L0 and L1 inactivity timers. (L0, L1 are user-control based.)

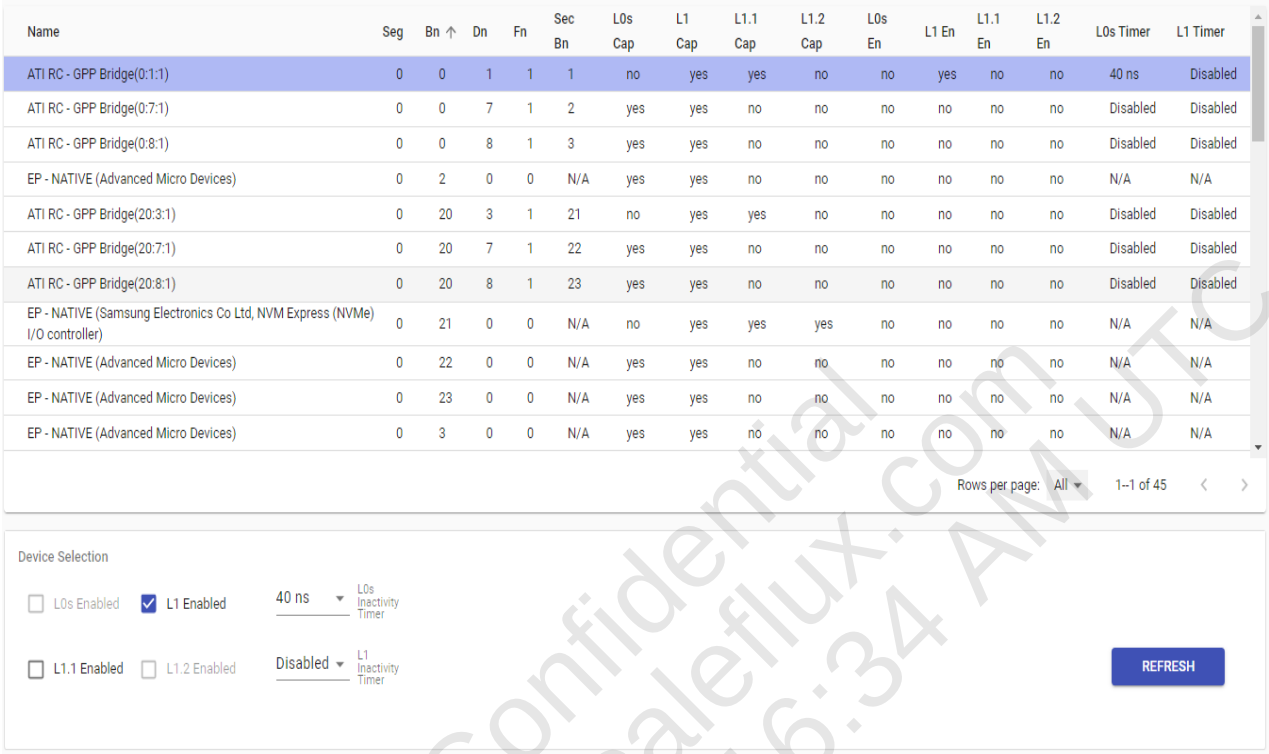


Figure 26. GUI ASPM Features

13.7 PCIe Port Status

The PCIe Port Status tab displays the status of registers for PCIe ports.

13.7.1 Features

13.7.1.1 PCIe Port Status Registers

The following registers for graphics PCIe device graphics (ATI EP GFX) on the system can be monitored by the PCIe port status registers window:

- PCIE_STATUS: PCIe status
- DEVICE_STATUS: Device status
- LINK_STATUS: Link status
- SLOT_STATUS: Slot status
- ROOT_STATUS: Root status
- PCIE_UNCORR_ERR_STATUS: PCIe uncorrectable error status
- PCIE_CORR_ERR_STATUS: PCIe correctable error status

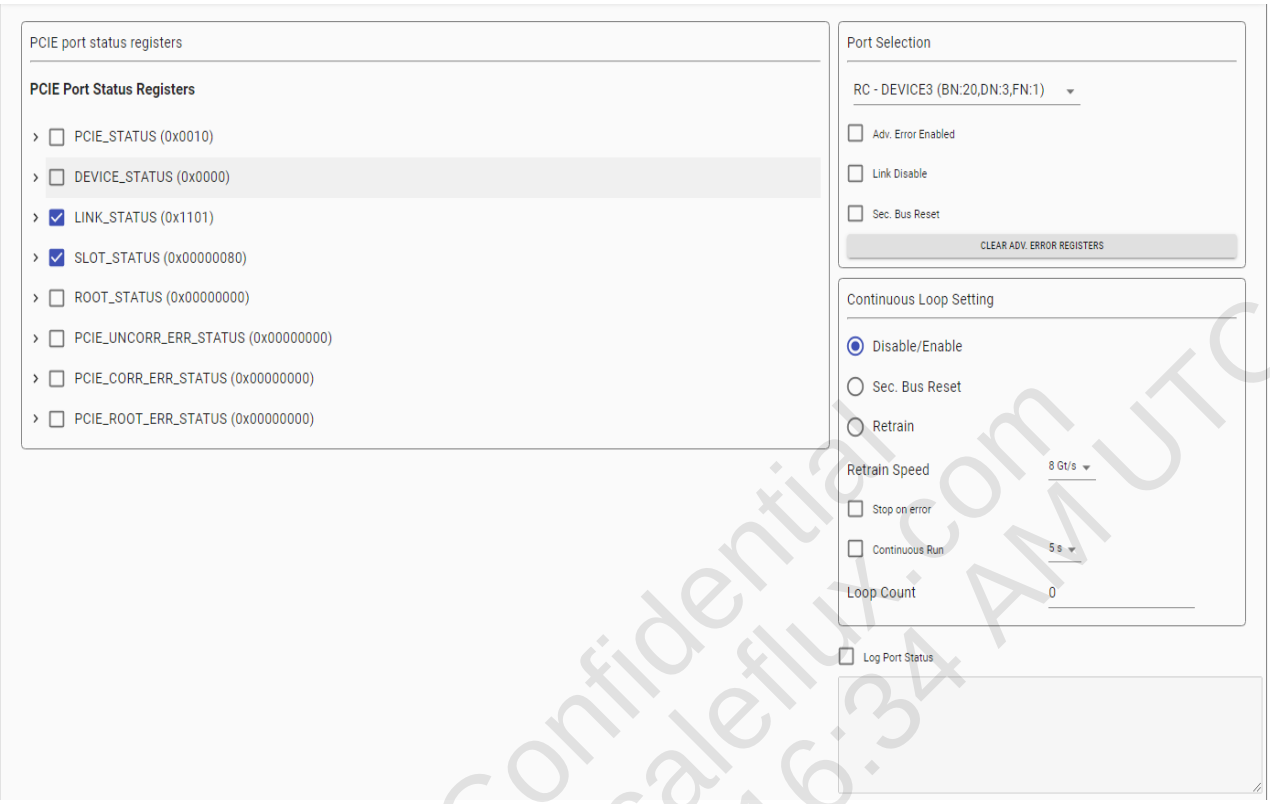


Figure 27. GUI Feature: PCIe Port Status Register

13.7.1.2 PCIe Port Selection

The ports used to monitor the registers can be selected from the drop-down list. The “Adv. Error Enabled” checkbox enables advanced error reporting for PCIe correctable and uncorrectable error status registers (PCIE_CORR_ERR_STATUS and PCIE_CORR_ERR_STATUS). “Clear Adv. Error Registers” clears these registers.

13.7.1.3 PCIe Link Continuous Test Executions

Few specific test conditions are supported to be run in a loop. The following features are currently supported:

- Link enable/disable
- Secondary bus reset
- Link retrain (“Rome” and later, only for PCIe Gen3, Gen4, and Gen5)

When doing these tests, especially for secondary bus reset, disabling the device driver first is recommended.

Chapter 14 Performance Counters

The Performance Counters tab monitors available counters for supported devices to assist in debugging, including features that log counter values and can store or load counter configurations.

Counters are detected and displayed based on the number of cores addressable either on the system's APU or discrete GPU. The user-provided counter can be loaded (based on system support).

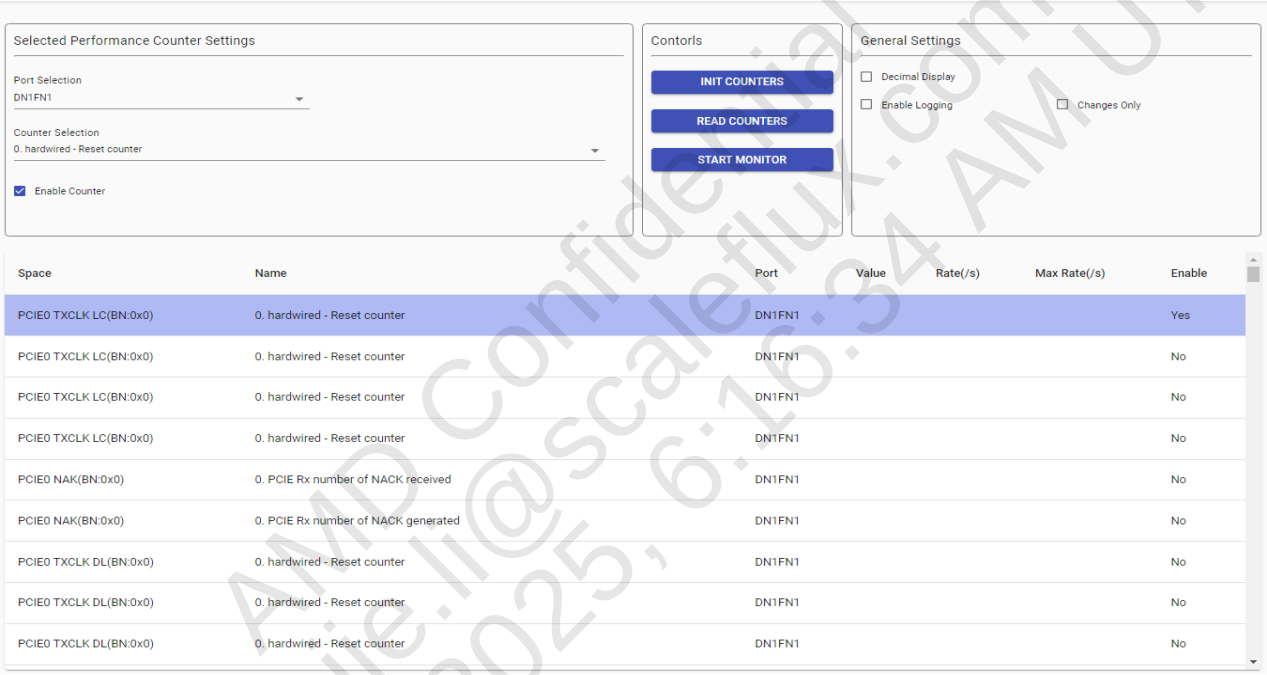


Figure 28. GUI Features: Performance Counters

Chapter 15 PHY Margin

The tool supports PHY-based margin on AMD platforms. The PHY-based margin command has the following outputs:

- Console output
- CSV file

15.1 How to Run PHY Margin

Use the following format to invoke PHY Margin for one lane:

```
amdxio -i=<device id> -margin -phy -lanes=<LinkName:Lane No> -
ber=<BER value>, -errcnt=<Error threshold, default is 1>
```

Use the following format to invoke the PHY margin for a lane range:

```
amdxio -i=<device id> -margin -phy -lanes=<LinkName:Lane start. Lane
end> -ber=<BER value>, -errcnt=<Error threshold, default is 1>
```

For device ID details, refer to Chapter 1 Device Listing and Selection.

15.1.1 Sample Commands

To invoke an eye scan for link XGMIPCS for Lane 3 on Device 0:

```
amdxio -i=0 -margin -phy -lanes= XGMIPCS:3
```

To invoke an eye scan for link XGMIPCS for Lanes 3 to 5 on Device 0:

```
amdxio -i=0 -margin -phy -lanes= XGMIPCS:3.5
```

15.2 Sample Outputs

15.2.1 Console Output

```
AMDXIO Version 5.1.12.0
. . . . . .
Results      :
Socket       :  0
Link Name    :  G3
Lane Number  :  0
BER          :  8
Error Count  :  1
NBITS        :  25
Top   Margin :  0.065 volt
Bot   Margin :  0.065 volt
Left  Margin :  0.23 ui
Right Margin :  0.3 ui
```

Figure 29. PHY Margin Console Output

15.2.2 File Outputs

The CSV file output will be dumped into a timestamped folder that will have margin data for all devices margined.

```
Output file present in: 2023_05_30-06_57_05
```

Figure 30. PHY Margin CSV Folder