



BMC Sensor Data

Table of contents

SDR Sensor List

Sensor Redfish Commands

Getting List of Support Sensors

Configuring Sensor Thresholds

Getting Data for Specific Sensor

Sensor IPMI Commands

Displaying Sensor Data

Displaying Sensor Data Records Specified by Sensor ID

Displaying Extended Sensor Data

Displaying All Records from SDR Repository of Specific Type

Displaying Data for Sensors Specified by Name

Displaying Sensors and Thresholds

Displaying Readings for Sensors Specified by Name (Only for Numeric Sensors)

SDR Sensor List

The following is a list of the available sensors maintained by the BMC including their type and name.

Sensor Name	Sensor Type	Source	Description
p0_link	Discrete	IPMB	Uplink port 0 link status <ul style="list-style-type: none"> 0x100 – connection OK 0x200 – connection error
p1_link	Discrete	IPMB	Uplink port 1 link status <ul style="list-style-type: none"> 0x100 – connection OK 0x200 – connection error
bluefield_temp	Temperature	IPMB	NVIDIA® BlueField® temperature
p0_temp	Temperature	IPMB	Uplink port 0 SFP temperature
p1_temp	Temperature	IPMB	Uplink port 1 SFP temperature
ddr_temp	Temperature	IPMB	DDR temperature
rtc_voltage	Voltage	IPMB	RTC battery voltage
power_envelope	Power	IPMB	<ul style="list-style-type: none"> This sensor indicates the maximum power consumption allowed for BlueField-3 This sensor is incompatible with unsecured Linux
soc_power	Power	IPMB	<ul style="list-style-type: none"> This sensor indicates the current power consumption of the BlueField-3 DPU This sensor is incompatible with unsecured Linux
power_envelope_deviation	Power	Synthesized Sensor	Measures the deviation of <code>soc_power</code> sensor value from <code>power_envelope</code> sensor value

Sensor Name	Sensor Type	Source	Description
			<p><code>power_envelope_de</code> <code>= soc_power -</code></p> <ul style="list-style-type: none"> <code>power_envelope</code> The sensor value should be negative for normal conditions. If the sensor value is positive, then SoC power has exceeded the allowed power envelope. If the value of <code>soc_po</code> is <code>power_envelope</code> or <code>NaN</code>, then the <code>power_envelope_de</code> sensor value will also be
<code>1V_BMC</code>	Voltage	BMC ADC	
<code>1_2V_BMC</code>	Voltage	BMC ADC	
<code>1_8V</code>	Voltage	BMC ADC	
<code>1_8V_BMC</code>	Voltage	BMC ADC	
<code>2_5V</code>	Voltage	BMC ADC	
<code>3_3V</code>	Voltage	BMC ADC	
<code>3_3V_RGM</code>	Voltage	BMC ADC	
<code>5V</code>	Voltage	BMC ADC	
<code>12V_ATX</code>	Voltage	BMC ADC	Input power rail from ATX (power from gold fingers in case of when ATX power is off)
<code>12V_PCIE</code>	Voltage	BMC ADC	Input power rail from gold fi
<code>DVDD</code>	Voltage	BMC ADC	
<code>HVDD</code>	Voltage	BMC ADC	
<code>VDD</code>	Voltage	BMC ADC	
<code>VDDQ</code>	Voltage	BMC ADC	

Sensor Name	Sensor Type	Source	Description
VDD_CPU_L	Voltage	BMC ADC	
VDD_CPU_R	Voltage	BMC ADC	

Note

IPMB sourced sensors are supported when operating in DPU mode only.

Sensor Redfish Commands

Getting List of Support Sensors

BlueField sensors are stored within the Sensors schema under the Chassis schema. To retrieve the list of supported sensors, execute the following command:

```
curl -k -u root:'<password>' -H 'Content-Type: application/json'
-X GET https://<bmc_ip>/redfish/v1/Chassis/Card1/Sensors
```

The following is an example of the anticipated output:

```
{
  "@odata.id": "/redfish/v1/Chassis/Card1/Sensors",
  "@odata.type": "#SensorCollection.SensorCollection",
  "Description": "Collection of Sensors for this Chassis",
  "Members": [
    {
      "@odata.id":
"/redfish/v1/Chassis/Card1/Sensors/power_envelope"
    },
  ],
}
```

```
{
  "@odata.id" :
"/redfish/v1/Chassis/Card1/Sensors/power_envelope_deviation"
},
{
  "@odata.id" : "/redfish/v1/Chassis/Card1/Sensors/soc_power"
},
{
  "@odata.id" :
"/redfish/v1/Chassis/Card1/Sensors/bluefield_temp"
},
{
  "@odata.id" : "/redfish/v1/Chassis/Card1/Sensors/ddr_temp"
},
{
  "@odata.id" : "/redfish/v1/Chassis/Card1/Sensors/p0_temp"
},
{
  "@odata.id" : "/redfish/v1/Chassis/Card1/Sensors/p1_temp"
},
{
  "@odata.id" : "/redfish/v1/Chassis/Card1/Sensors/12V_ATX"
},
{
  "@odata.id" : "/redfish/v1/Chassis/Card1/Sensors/12V_PCIE"
},
{
  "@odata.id" : "/redfish/v1/Chassis/Card1/Sensors/1V_BMC"
},
{
  "@odata.id" : "/redfish/v1/Chassis/Card1/Sensors/1_2V_BMC"
},
{
  "@odata.id" : "/redfish/v1/Chassis/Card1/Sensors/1_8V"
},
{
```

```

    "@odata.id": "/redfish/v1/Chassis/Card1/Sensors/1_8V_BMC"
  },
  {
    "@odata.id": "/redfish/v1/Chassis/Card1/Sensors/2_5V"
  },
  {
    "@odata.id": "/redfish/v1/Chassis/Card1/Sensors/3_3V"
  },
  {
    "@odata.id": "/redfish/v1/Chassis/Card1/Sensors/3_3V_RGM"
  },
  {
    "@odata.id": "/redfish/v1/Chassis/Card1/Sensors/5V"
  },
  {
    "@odata.id": "/redfish/v1/Chassis/Card1/Sensors/DVDD"
  },
  {
    "@odata.id": "/redfish/v1/Chassis/Card1/Sensors/HVDD"
  },
  {
    "@odata.id": "/redfish/v1/Chassis/Card1/Sensors/VDD"
  },
  {
    "@odata.id": "/redfish/v1/Chassis/Card1/Sensors/VDDQ"
  },
  {
    "@odata.id": "/redfish/v1/Chassis/Card1/Sensors/VDD_CPU_L"
  },
  {
    "@odata.id": "/redfish/v1/Chassis/Card1/Sensors/VDD_CPU_R"
  },
  {
    "@odata.id":
"/redfish/v1/Chassis/Card1/Sensors/rtc_voltage"
  }

```

```
],  
  "Members@odata.count": 24,  
  "Name": "Sensors"  
}
```

Getting Data for Specific Sensor

```
curl -k -u root:'<password>' -H 'Content-Type: application/json'  
-X GET  
https://<bmc_ip>/redfish/v1/Chassis/Card1/Sensors/<sensor_name>
```

The following is an example of a temperature sensor BlueField reading:

```
curl -k -u root:'<password>' -H 'Content-Type: application/json'  
-X GET  
https://<bmc_ip>/redfish/v1/Chassis/Card1/Sensors/bluefield_temp  
{  
  "@odata.id":  
  "/redfish/v1/Chassis/Card1/Sensors/bluefield_temp",  
  "@odata.type": "#Sensor.v1_2_0.Sensor",  
  "Id": "bluefield_temp",  
  "Name": "bluefield temp",  
  "Reading": 43.0,  
  "ReadingRangeMax": 255.0,  
  "ReadingRangeMin": 0.0,  
  "ReadingType": "Temperature",  
  "ReadingUnits": "Cel",  
  "RelatedItem": [  
    {  
      "@odata.id": "/redfish/v1/Systems/Bluefield"  
    }  
  ]  
}
```

```

],
"Status": {
  "Conditions": [],
  "Health": "OK",
  "HealthRollup": "OK",
  "State": "Enabled"
},
"Thresholds": {
  "LowerCaution": {
    "Reading": 5.0
  },
  "LowerCritical": {
    "Reading": 0.0
  },
  "UpperCaution": {
    "Reading": 95.0
  },
  "UpperCritical": {
    "Reading": 105.0
  }
}
}
}

```

Configuring Sensor Thresholds

The following commands set the thresholds for sensors that support setting a threshold:

```

curl -k -u root:'<password>' -X PATCH
https://<bmc_ip>/redfish/v1/Chassis/Card1/Sensors/<sensor name>/
-d '{"Thresholds":{"<Threshold name>": {"Reading":<value>}}}'

```

The following is an example of how to set the upper critical threshold for the BlueField temperature sensor:

```
curl -k -u root:'<password>' -X PATCH
https://<bmc_ip>/redfish/v1/Chassis/Card1/Sensors/bluefield_temp
-d '{"Thresholds":{"UpperCritical":{"Reading":100}}}'
{
  "@Message.ExtendedInfo": [
    {
      "@odata.type": "#Message.v1_1_1.Message",
      "Message": "The request completed successfully.",
      "MessageArgs": [],
      "MessageId": "Base.1.15.0.Success",
      "MessageSeverity": "OK",
      "Resolution": "None"
    }
  ]
}
```

Sensor IPMI Commands

BMC software supports reading chassis sensor information using the IPMITool.

The following subsections list commands which allow reading SDR data.

Displaying Sensor Data

Displays sensor data repository entry readings and their status.

```
ipmitool -C 17 -I lanplus -H <bmc_ip> -U ADMIN -P ADMIN sdr list
```

Displaying Extended Sensor Data

Displays extended sensor information.

```
ipmitool -C 17 -I lanplus -H <bmc_ip> -U ADMIN -P ADMIN sdr elist
```

Displaying Sensors and Thresholds

Displays sensors and thresholds in a wide table format.

```
ipmitool -C 17 -I lanplus -H <bmc_ip> -U ADMIN -P ADMIN sensor  
list
```

Displaying Sensor Data Records Specified by Sensor ID

Displays sensor data records specified by sensor ID.

```
ipmitool -C 17 -I lanplus -H <bmc_ip> -U ADMIN -P ADMIN sdr get  
<name>
```

Displaying All Records from SDR Repository of Specific Type

Displays all records from the SDR repository of a specific type.

```
ipmitool -C 17 -I lanplus -H <bmc_ip> -U ADMIN -P ADMIN sdr type  
<type>
```

Displaying Data for Sensors Specified by Name

Displays information for sensors specified by name.

```
ipmitool -C 17 -I lanplus -H <bmc_ip> -U ADMIN -P ADMIN sensor  
get <sensor_name>
```

Displaying Readings for Sensors Specified by Name (Only for Numeric Sensors)

```
ipmitool -C 17 -I lanplus -H <bmc_ip> -U ADMIN -P ADMIN sensor  
reading <name>...<name>
```

Notice

This document is provided for information purposes only and shall not be regarded as a warranty of a certain functionality, condition, or quality of a product. NVIDIA Corporation ("NVIDIA") makes no representations or warranties, expressed or implied, as to the accuracy or completeness of the information contained in this document and assumes no responsibility for any errors contained herein. NVIDIA shall have no liability for the consequences or use of such information or for any infringement of patents or other rights of third parties that may result from its use. This document is not a commitment to develop, release, or deliver any Material (defined below), code, or functionality.

NVIDIA reserves the right to make corrections, modifications, enhancements, improvements, and any other changes to this document, at any time without notice.

Customer should obtain the latest relevant information before placing orders and should verify that such information is current and complete.

NVIDIA products are sold subject to the NVIDIA standard terms and conditions of sale supplied at the time of order acknowledgement, unless otherwise agreed in an individual sales agreement signed by authorized representatives of NVIDIA and customer ("Terms of Sale"). NVIDIA hereby expressly objects to applying any customer general terms and conditions with regards to the purchase of the NVIDIA product referenced in this document. No contractual obligations are formed either directly or indirectly by this document.

NVIDIA products are not designed, authorized, or warranted to be suitable for use in medical, military, aircraft, space, or life support equipment, nor in applications where failure or malfunction of the NVIDIA product can reasonably be expected to result in personal injury, death, or property or environmental damage. NVIDIA accepts no liability for inclusion and/or use of NVIDIA products in such equipment or applications and therefore such inclusion and/or use is at customer's own risk.

NVIDIA makes no representation or warranty that products based on this document will be suitable for any specified use. Testing of all parameters of each product is not necessarily performed by NVIDIA. It is customer's sole responsibility to evaluate and determine the applicability of any information contained in this document, ensure the product is suitable and fit for the application planned by customer, and perform the necessary testing for the application in order to avoid a default of the application or the product. Weaknesses in customer's product designs may affect the quality and reliability of the NVIDIA product and may result in additional or different conditions and/or requirements beyond those contained in this document. NVIDIA accepts no liability related to any default, damage, costs, or problem which may be based on or attributable to: (i) the use of the NVIDIA product in any manner that is contrary to this document or (ii) customer product designs.

No license, either expressed or implied, is granted under any NVIDIA patent right, copyright, or other NVIDIA intellectual property right under this document. Information published by NVIDIA regarding third-party products or services does not constitute a license from NVIDIA to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property rights of the third party, or a license from NVIDIA under the patents or other intellectual property rights of NVIDIA.

Reproduction of information in this document is permissible only if approved in advance by NVIDIA in writing, reproduced without alteration and in full compliance with all applicable export laws and regulations, and accompanied by all associated conditions, limitations, and notices.

THIS DOCUMENT AND ALL NVIDIA DESIGN SPECIFICATIONS, REFERENCE BOARDS, FILES, DRAWINGS, DIAGNOSTICS, LISTS, AND OTHER DOCUMENTS (TOGETHER AND SEPARATELY, "MATERIALS") ARE BEING PROVIDED "AS IS." NVIDIA MAKES NO WARRANTIES, EXPRESSED, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE MATERIALS, AND EXPRESSLY DISCLAIMS ALL IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE. TO THE EXTENT NOT PROHIBITED BY LAW, IN NO EVENT WILL NVIDIA BE LIABLE FOR ANY DAMAGES, INCLUDING WITHOUT LIMITATION ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, PUNITIVE, OR CONSEQUENTIAL DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF

ANY USE OF THIS DOCUMENT, EVEN IF NVIDIA HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Notwithstanding any damages that customer might incur for any reason whatsoever, NVIDIA's aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms of Sale for the product.

Trademarks

NVIDIA and the NVIDIA logo are trademarks and/or registered trademarks of NVIDIA Corporation in the U.S. and other countries. Other company and product names may be trademarks of the respective companies with which they are associated.

© Copyright 2025, NVIDIA. PDF Generated on 11/20/2025