AN 872: Thermal and Power Guidelines

For Intel® Programmable Acceleration Card with Intel® Arria® 10 GX FPGA
1. Introduction

1.1. About this Document

This document provides methods to estimate and validate the power and thermal performance of your AFU design using the Intel® Programmable Acceleration Card with Intel Arria® 10 GX FPGA in the target server platform.

1.2. Power Specification

The board management controller monitors and manages thermal and power events on the Intel FPGA PAC. When the board or FPGA is overheating or drawing excessive current, the board management controller shuts down the FPGA power for protection. Subsequently, it also brings down the PCIe link which may cause an unexpected system crash.

Refer to Auto-Shutdown for more details about the criteria that triggers board shutdown. In normal cases, the FPGA temperature and power are by far the leading cause of shutdown. To minimize downtime and ensure system stability, Intel recommends that the total board power does not go beyond 66 W and FPGA power does not go beyond 45 W.

Individual components and board assemblies have power variability. Therefore, the nominal values are lower than the limits to ensure that the board does not experience a random shutdown in a system with varying workloads and inlet temperatures.

Table 1. Power Specification

<table>
<thead>
<tr>
<th>System</th>
<th>Total Board Power (watts)</th>
<th>FPGA Power (watts)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A system with an FPGA Interface Manager (FIM) and AFU that runs with worst-case throttling workload for minimum 15 minutes at the core temperature of 95°C.</td>
<td>66</td>
<td>45</td>
</tr>
</tbody>
</table>

The total board power varies depending on your Accelerator Functional Unit (AFU) design (amount and frequency of logic toggling), inlet temperature, system temperature and airflow of the target slot for the Intel FPGA PAC. To manage this variability, Intel recommends you meet this power specification to prevent power shutdown by the Board Management Controller.

**Related Information**

Auto-Shutdown on page 5
1.3. Prerequisites

The server original equipment manufacturer (OEM) must validate that each Intel FPGA PAC interfacing to a PCIe slot in a target server platform can stay within the thermal limits even when the board consumes the maximum allowed power (66 W). For more information, refer to the Intel PAC with Intel Arria 10 GX FPGA Platform Qualification Guidelines(1).

1.4. Tools Requirements

You must have the following tools to estimate and evaluate the power and thermal performance.

- **Software:**
  - Intel Acceleration Stack for Development
  - BWtoolkit
  - AFU Design(2)
  - Tcl script (download) - Required to format the programming file for analysis
  - Early Power Estimator for Intel Arria 10 devices
  - Intel FPGA PAC Power Estimator Sheet (download)

- **Hardware:**
  - Intel FPGA PAC
  - Micro-USB cable(3)
  - Target Server for Intel FPGA PAC(4)

Intel recommends you to follow the Intel Acceleration Stack Quick Start Guide for Intel Programmable Acceleration Card with Intel Arria 10 GX FPGA for software installation.

**Related Information**

Intel Acceleration Stack Quick Start Guide for Intel Programmable Acceleration Card with Intel Arria 10 GX FPGA

---

(1) Contact your Intel support representative to access this document.
(2) The build_synth directory is created after you compile your AFU.
(3) In Acceleration Stack 1.2, the board monitoring is performed over PCIe.
(4) Ensure that your OEM has validated the targeted PCIe slot(s) in accordance to the Platform Qualification Guidelines for your Intel FPGA PAC.
2. Using the Board Management Controller

2.1. Auto-Shutdown

The Board Management Controller monitors and controls resets, different power rails, FPGA and board temperatures. When the Board Management Controller senses conditions that can potentially damage the board, it automatically shuts down board power for protection.

*Note:* When the FPGA loses power, the PCIe link between the Intel FPGA PAC and host is down. In many systems, the PCIe link-down may cause a system crash.

Table 2. Auto-Shutdown Criteria

The following table lists the criteria beyond which the Board Management Controller shuts down board power.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Threshold Limit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Board Power</td>
<td>66 W</td>
</tr>
<tr>
<td>12v Backplane Current</td>
<td>6 A</td>
</tr>
<tr>
<td>12v Backplane Voltage</td>
<td>14 V</td>
</tr>
<tr>
<td>1.2v Current</td>
<td>16 A</td>
</tr>
<tr>
<td>1.2v Voltage</td>
<td>1.4 V</td>
</tr>
<tr>
<td>1.8v Current</td>
<td>8 A</td>
</tr>
<tr>
<td>1.8v Voltage</td>
<td>2.04 V</td>
</tr>
<tr>
<td>3.3v Current</td>
<td>8 A</td>
</tr>
<tr>
<td>3.3v Voltage</td>
<td>3.96 V</td>
</tr>
<tr>
<td>FPGA Core Voltage</td>
<td>1.08 V</td>
</tr>
<tr>
<td>FPGA Core Current</td>
<td>60 A</td>
</tr>
<tr>
<td>FPGA Core Temperature</td>
<td>100°C</td>
</tr>
<tr>
<td>Core Supply Temperature</td>
<td>120°C</td>
</tr>
<tr>
<td>Board Temperature</td>
<td>80°C</td>
</tr>
<tr>
<td>QSFP Temperature</td>
<td>90°C</td>
</tr>
<tr>
<td>QSFP Voltage</td>
<td>3.7 V</td>
</tr>
</tbody>
</table>

2.2. Recovering After Auto-Shutdown

The Board Management Controller holds power off until the next power cycle. Therefore, when an Intel FPGA PAC card power is shut down, you must power cycle the server to return power to the Intel FPGA PAC.
The common cause of power shutdown is the FPGA overheating (when the core temperature is over 100°C), or the FPGA drawing excessive current. This typically happens when the AFU design exceeds the Intel FPGA PAC defined power envelopes or there is insufficient airflow. In this case, you must reduce power consumption in your AFU.

### 2.3. Monitor On-Board Sensors Using OPAE

Use the `fpgainfo` command line program to gather the temperature and power sensor data from the Board Management Controller. You can use this program with the Acceleration Stack 1.2 and beyond. For Acceleration Stack 1.1 or older, use the BWMonitor tool as described in the next section.

- **To gather the temperature data:**
  
  ```bash
  bash-4.2$ fpgainfo temp
  
  Sample output:
  
  Board Management Controller, microcontroller FW version 26889
  Last Power Down Cause: POK_CORE
  Last Reset Cause: None
  //****** TEMP ******/
  Object Id                       : 0xF300000
  PCIe s:b:d:f                    : 0000:04:00:0
  Device Id                       : 0x09C4
  Socket Id                       : 0x00
  Ports Num                       : 01
  Bitstream Id                    : 0x121000200000161
  Bitstream Version               : 0x10201
  Fr Interface Id                 : 93abeb6a-30c8-5f77-8172-d828c3a699ca
  (11) FPGA Core TEMP             : 73.00 °C
  (12) Board TEMP                 : 47.00 °C
  (14) QSFP TEMP                  : No reading (reading state unavailable)
  (15) Core Supply Temp           : 75.96 °C
  ```

- **To gather the power data:**
  
  ```bash
  bash-4.2$ fpgainfo power
  
  Sample output:
  
  Board Management Controller, microcontroller FW version 26889
  Last Power Down Cause: POK_CORE
  Last Reset Cause: None
  //****** POWER ******/
  Object Id                     : 0xF300000
  PCIe s:b:d:f                  : 0000:04:00:0
  Device Id                     : 0x09C4
  Socket Id                     : 0x00
  Ports Num                     : 01
  Bitstream Id                  : 0x121000200000161
  Bitstream Version             : 0x10201
  Pr Interface Id               : 93abeb6a-30c8-5f77-8172-d828c3a699ca
  ( 0) Total Input Power        : 23.00 Watts
  ( 1) PCIe 12V Current          : 1.93 Amps
  ( 2) PCIe 12V Voltage          : 11.60 Volts
  ( 3) 1.2V Voltage              : 1.22 Volts
  ( 4) 1.2V Current              : 2.66 Amps
  ( 5) 1.8V Voltage              : 1.83 Volts
  ( 6) 1.8V Current              : 2.91 Amps
  ( 7) 3.3V Mgmt Voltage         : 3.36 Volts
  ( 8) 3.3V Current              : 0.72 Amps
  ( 9) FPGA Core Voltage         : 0.90 Volts
  (10) FPGA Core Current         : 7.65 Amps
  (13) QSFP P3V3                 : No reading (reading state unavailable)
  ```
2.4. Monitor On-Board Sensors Using BWMonitor

BWMonitor is a BittWare tool that allows you to measure FPGA/board temperature, voltage, and current.

**Prerequisite**: You must install a micro-USB cable between the Intel FPGA PAC and the server.

1. Install the appropriate BittWorks II Toolkit-Lite software, firmware, and bootloader.

### Table 3. OS-Compatible BittWorks II Toolkit-Lite Version

<table>
<thead>
<tr>
<th>Operating System</th>
<th>Release</th>
<th>BittWorks II Toolkit-Lite Version</th>
<th>Install Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>CentOS 7.4/RHEL 7.4</td>
<td>2018.6 Enterprise Linux 7 (64-bit)</td>
<td>bw2tk-lite-2018.6.el7.x86_64.rpm</td>
<td><code>sudo yum install bw2tk-lite-2018.6.el7.x86_64.rpm</code></td>
</tr>
<tr>
<td>Ubuntu 16.04</td>
<td>2016 Ubuntu 16.04 (64-bit)</td>
<td>bw2tk-lite-2016.6.0.1604.amd64.deb</td>
<td><code>sudo dpkg -i bw2tk-lite-2016.6.0.1604.amd64.deb</code></td>
</tr>
</tbody>
</table>

Refer the **Getting Started** webpage to download the BMC firmware and tools:

- BMC Firmware version: 26889
- BMC Bootloader version: 26879

Save the files to a known location on the host machine. The following script prompts for this location.

2. Add Bittware tool to PATH:

```bash
export PATH=/opt/bwtk/2018.6.0L/bin/:$PATH
```

You can launch the BWMonitor using:

```
/opt/bwtk/2018.6.0L/bin/bwmonitor-gui
```
## 2. Using the Board Management Controller

**AN-872 | 2019.08.30**

### Figure 1. Sample Measurements

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Board Management Controller</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Microcontroller</td>
<td>Version 26815</td>
<td>Powered on</td>
</tr>
<tr>
<td>SDR Sensors</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Board Power</td>
<td>65 Watts</td>
<td>OK</td>
</tr>
<tr>
<td>12v Backplane Current</td>
<td>5.40 Amps</td>
<td>OK</td>
</tr>
<tr>
<td>12v Backplane Voltage</td>
<td>12.01 Volts</td>
<td>OK</td>
</tr>
<tr>
<td>1.2v Current</td>
<td>2.66 Amps</td>
<td>OK</td>
</tr>
<tr>
<td>1.2v Voltage</td>
<td>1.22 Volts</td>
<td>OK</td>
</tr>
<tr>
<td>1.8v Current</td>
<td>3.45 Amps</td>
<td>OK</td>
</tr>
<tr>
<td>1.8v Voltage</td>
<td>1.83 Volts</td>
<td>OK</td>
</tr>
<tr>
<td>3.3v Current</td>
<td>1.27 Amps</td>
<td>OK</td>
</tr>
<tr>
<td>3.3v Voltage</td>
<td>3.36 Volts</td>
<td>OK</td>
</tr>
<tr>
<td>FPGA Core Voltage</td>
<td>0.93 Volts</td>
<td>OK</td>
</tr>
<tr>
<td>FPGA Core Current</td>
<td>39.32 Amps</td>
<td>OK</td>
</tr>
<tr>
<td>FPGA Core Temperature</td>
<td>72 degrees C</td>
<td>OK</td>
</tr>
<tr>
<td>Core Supply Temperature</td>
<td>94 degrees C</td>
<td>OK</td>
</tr>
<tr>
<td>Board Temperature</td>
<td>30 degrees C</td>
<td>OK</td>
</tr>
<tr>
<td>QSFP Temperature</td>
<td></td>
<td>Unavailable</td>
</tr>
<tr>
<td>QSFP Voltage</td>
<td></td>
<td>Unavailable</td>
</tr>
<tr>
<td>VCCR Voltage</td>
<td>1.05 Volts</td>
<td>OK</td>
</tr>
<tr>
<td>VCCT Voltage</td>
<td>1.04 Volts</td>
<td>OK</td>
</tr>
<tr>
<td>VCCR Current</td>
<td>2.00 Amps</td>
<td>OK</td>
</tr>
<tr>
<td>VCCT Current</td>
<td>0.37 Amps</td>
<td>OK</td>
</tr>
<tr>
<td>VPP Voltage</td>
<td>2.54 Volts</td>
<td>OK</td>
</tr>
<tr>
<td>VTT Voltage</td>
<td>0.61 Volts</td>
<td>OK</td>
</tr>
</tbody>
</table>

**FPGA metrics**

**TDP (total board power)**
3. AFU Design Power Verification

3.1. Power Measurement Flow

To evaluate the power for your AFU design, capture the following metrics:

- Total board power and FPGA temperature
  (after running the worst-case data patterns on your design for 15 minutes)
- Static Power and Temperature
  (using a static power measurement design)
- Worst Case Static Power
  (predicted values using the Early Power Estimator for Intel Arria 10 devices)

Then, use the Intel FPGA PAC Power Estimator Sheet (download) with these recorded metrics to verify if your AFU design meets the specification.

3.2. Measuring the Total Board Power

Follow these steps:

1. Install the Intel PAC with Intel Arria 10 GX FPGA into a qualified PCIe slot in the server. If you are using BWMonitor for measurement, connect the Micro-USB cable from back of the card to any USB port of the server.
2. Load your AFU and run at its maximum power.
   a. If the AFU uses Ethernet, then ensure that the network cable or module is inserted and connected to the link partner and network traffic is turned on in the AFU.
   b. If appropriate, run DMA continuously to exercise on-board DDR4.
   c. Run your applications on the host to feed the AFU the worst-case traffic as well as to fully exercise FPGA. Ensure that you stress the FPGA with the most stressful data traffic.

Run this step for minimum 15 minutes to allow the FPGA core temperature to settle.

Note: During testing, monitor the total board power, FPGA power, and FPGA core temperature value to ensure they stay within specification. If 66 W, 45 W, or 100°C limits are reached, stop the test immediately.

3. After the FPGA core temperature becomes stable, use the fpgainfo program or BWMonitor tool to record the total board power and FPGA core temperature. Input these values in row Step 1: Total board power measurement of the Intel FPGA PAC Power Estimator Sheet.
3.3. Measuring the Real Static Power

Leakage current is a leading cause of board-to-board power consumption variation. The power measurements from the above section include power due to leakage current (static power) and power due to the AFU logic (dynamic power). In this section, you will measure the static power of the board-under-test in order to understand the dynamic power.

Before measuring the FPGA static power, use the `disable-gpio-input-buffer-intel-pac-arria10-gx.tcl` script (download) to process the FPGA programming file, (*.sof file) which contains a FIM and AFU design. The tcl script disables all FPGA input pins to ensure that there is no toggling inside the FPGA (which means no dynamic power). Refer to the Minimal Flow Example to compile a sample AFU. The generated *.sof file is located at:

```bash
cd $OPAE_PLATFORM_ROOT/hw/samples/<afu name>
$ OPAE_PLATFORM_ROOT/hw/samples/<afu name>build_synth/build/output_files/afu_*.sof
```

You must save the `disable-gpio-input-buffer-intel-pac-arria10-gx.tcl` in the above directory and then run the following command:

```bash
# quartus_asm -t disable-gpio-input-buffer-intel-pac-arria10-gx.tcl afu_*.sof
```

Sample output:

```
Info: *******************************************************************
Info: Running Quartus Prime Assembler
Info: Version 17.1.1 Build 273 12/19/2017 SJ Pro Edition
```

**Figure 2. Intel FPGA PAC Power Estimator Sheet Sample**
Upon successful execution of the tcl script, the afu_*.sof file is updated and ready for FPGA programming.

Follow these steps to measure the real static power:

1. Use the Intel Quartus® Prime programmer to program the *.sof file. Refer to the Using the Intel Quartus Prime Programmer on page 12 for detailed steps.

2. Monitor the FPGA core temperature, voltage, and current using the BWMonitor tool. Enter these values in row Step 2: FPGA core static power measurement of the Intel FPGA PAC Power Estimator Sheet.

Related Information

- Intel Acceleration Stack Quick Start Guide for Intel Programmable Acceleration Card with Intel Arria 10 GX FPGA
- Monitor On-Board Sensors Using BWMonitor on page 7
### 3.3.1. Using the Intel Quartus Prime Programmer

You must have the micro USB cable connected between the Intel FPGA PAC and the server to execute these steps:

1. Find the Root Port and Endpoint of the Intel FPGA PAC card:

   ```bash
   $ lspci -tv | grep 09c4
   ```

   Example output 1 shows that the Root Port is `d7:0.0` and the Endpoint is `d8:0.0`:

   ```
   --+-[0000:d7]-+-00.0-[d8]----00.0 Intel Corporation Device 09c4
   ```

   Example output 2 shows that the Root Port is `0:1.0` and the Endpoint is `3:0.0`:

   ```
   +-01.0-[03]----00.0 Intel Corporation Device 09c4
   ```

   Example output 3 shows that the Root Port is `85:2.0` and the Endpoint is `86:0.0` and:

   ```
   +-[0000:85]-+-02.0-[86]----00.0 Intel Corporation Device 09c4
   ```

   **Note:** No output indicates a PCIe* device enumeration failure and that flash is not programmed.

   # Mask uncorrectable errors and correctable errors of FPGA

   ```bash
   $ sudo setpci -s d8:0.0 ECAP_AER+0x08.L=0xFFFFFFFF
   $ sudo setpci -s d8:0.0 ECAP_AER+0x14.L=0xFFFFFFFF
   ```

   # Mask uncorrectable errors and Mask correctable errors of RP

   ```bash
   $ sudo setpci -s d7:0.0 ECAP_AER+0x08.L=0xFFFFFFFF
   $ sudo setpci -s d7:0.0 ECAP_AER+0x14.L=0xFFFFFFFF
   ```

2. Run the following Intel Quartus Prime Programmer command:

   ```bash
   sudo $QUARTUS_HOME/bin/quartus_pgm -m JTAG -o 'pvbi;afu_*.sof'
   ```
3. AFU Design Power Verification

3. To unmask uncorrectable errors and mask correctable errors, run the following commands:

   # Unmask uncorrectable errors and mask correctable errors of FPGA
   $ sudo setpci -s d8:0.0 ECAP_AER+0x08.L=0x00000000
   $ sudo setpci -s d8:0.0 ECAP_AER+0x14.L=0x00000000

   # Unmask uncorrectable errors and mask correctable errors of RP:
   $ sudo setpci -s d7:0.0 ECAP_AER+0x08.L=0x00000000
   $ sudo setpci -s d7:0.0 ECAP_AER+0x14.L=0x00000000

4. Reboot.

Related Information

Intel Acceleration Stack Quick Start Guide for Intel Programmable Acceleration Card with Intel Arria 10 GX FPGA
3.4. Estimating the Worst Case Core Static Power

Follow these steps to estimate the worst case static power:

1. Refer to the Minimal Flow Example to compile a sample AFU located at:
   \[<\text{inteldevstack}/hw/samples/<\text{AFU name}>/\]

2. In the Intel Quartus Prime Pro Edition software, click File > Open Project and select your .qpf file to open the AFU synthesis project from the following path:
   \[<\text{Acceleration Stack Directory}/hw/samples/<\text{AFU name}>/build_synth/build\]

3. Click Project > Generate EPE File to create the required .csv file.

**Figure 3. Step 2 Illustration**

4. Open the Early Power Estimator tool\(^{(5)}\) and click Import CSV icon. Select the above generated .csv file.

   \[Note: You can ignore the warning while importing the .csv file.\]

5. Inputs parameters are filled out automatically.

\(^{(5)}\) Download Information: PowerPlay Early Power Estimator for Intel Arria 10 Devices.
3. AFU Design Power Verification

- Change the value to **User Entered** in the Junction Temp. $T_J$ field. And set the Junction Temp. $T_J$ (°C) field to **95**
- Change the **Power Characteristics** field from **Typical** to **Maximum**.
- In the EPE Tool, the $P_{\text{STATIC}}$ is the total static power in Watts. You can calculate the worst case core static power from the **Report** tab

**Figure 4. EPE Tool Sample Output**

**Figure 5. Report Tab**

In the example shown above, the total FPGA core static current is the sum of all static current and standby current at 0.9V ($V_{\text{CC}}, V_{\text{CCP}}, V_{\text{CCERAM}}$).

Enter these value in row **Step 3: Worst static power from EPE** of the **Intel FPGA PAC Power Estimator Sheet**.

Observe the **Calculated output** row for the maximum power consumption of your AFU.
4. Document Revision History for Thermal and Power Guidelines for Intel PAC with Intel Arria 10 GX FPGA

<table>
<thead>
<tr>
<th>Document Version</th>
<th>Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>2019.08.30</td>
<td>Initial release.</td>
</tr>
</tbody>
</table>