



# Performance Counter Exercises

## OpenPOWER Tutorial, SC19, Denver

18 November 2019 | Andreas Herten | Forschungszentrum Jülich, Jülich Supercomputing Centre

# Outline

Introduction

Ascent

PMU Hands-On

Connect to Ascent

Task Setup

Task Organization

PMU Hands-On Description

# Ascent

- Supercomputer used for exercises: **Ascent**
- Ascent: Training machine for Summit; same node design
  - 18 nodes
  - IBM POWER9 + NVIDIA Tesla V100
  - POWER9: 2 sockets, each 22 cores
  - Tesla V100: 6 GPUs per node
  - CPU ↔ GPU, GPU ↔ GPU: connections via NVLink
- This tutorial: Make use of all of this!

# PMU Hands-On

## Overview

- Measure various counters with PAPI
- Characterize Jacobi application...
- ... in context of POWER9 architecture.
- **But first: Getting to Ascent**

**Connect to Ascent**

# Ascent Setup

URL to follow along <http://bit.ly/sc19-op-material>

## 1 Login to Ascent with `ssh user@login1.ascent.olcf.ornl.gov`

- Sign up at <http://bit.ly/sc19-op-signup> for **TRN003** project
- Get your login credentials via mail

## 2 Bootstrap your account

- Call `source /gpfs/wolf/trn003/world-shared/bootstrap.sh`
- This sets up scripts and loads the latest course material in your home directory

## 3 Re-login with SSH forwarding

- We are using Jupyter Notebooks for tasks!
- During login, a message is printed to forward SSH ports with your personal port  
Your personal SSH port forwarding command (for Jupyter Notebooks):  
`ssh -L 8888:localhost:12345 aherten@login1.ascent.olcf.ornl.gov`
- Windows: Configure PuTTY (/...) to forward remote port 12345 to your local port 8888

See also: `man instructions` on Ascent

# Task Organization

## Part 1

- Three hands-ons exercises

Session 2 Performance Counters (*PMU*)

Session 3 Performance Optimization

Session 4 GPU and Multi-GPU programming

- Each exercise in own folder
- All use Jupyter Notebooks to guide through flow of tasks → interactive computing with comments and visualization
- **Not able to use Jupyter Notebook?** Backup: Download converted Jupyter Notebook (HTML, PDF) and follow along analogously. Notebook has instructions for you as well!  
→ **Interactive Notebooks strongly recommended!**
- **Always:** Instructors there to help you!

# Task Setup

## 1 Start Jupyter Lab server

`jupyter lab`

## 2 Visit printed URL, click to open Notebook

Copy/paste this URL into your browser when you connect for first time, to login with token:  
`localhost:8888/?token=13dabbec6216514124ea3fef196b1622494b023b520e6e30`

## 3 Navigate to folder of exercise in Jupyter's side drawer, i.e. to

`SC19-Tutorial/2-Performance_Counters/Handson`

## 4 Hack away!

Jupyter 101: Execute a cell with *Shift+Enter*; everything is Python, except for `!`, which is Bash



# Task Organization

## Part 2

- Tasks have Makefiles to steer compilation but also to submit to batch system
- Batch system: IBM Spectrum LSF, with combination of `bsub` and `jsrun`
  - Shortcut submission command prepared in environment variable `$SC19_SUBMIT_CMD`
    - Use it with `eval $SC19_SUBMIT_CMD ls`
    - Study and adapt it with `echo $SC19_SUBMIT_CMD`
  - Usually, tasks have functionality to hide batch system interaction: `make run`
- Editing: Directly in Jupyter *or via text editors on Ascent, like vim*
- Solutions always given in separate directories. You decide when/if to look at them!

# PMU Hands-On Description

# PMU Hands-On Descriptions

Task 1 Measure cycles and instructions needed for Jacobi solver

Task 2 Measure loads and stores

A Normal loads and stores

B More loads and stores

C Compute bandwidth

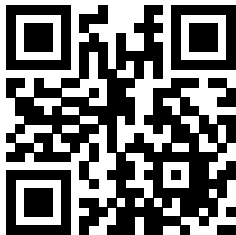
Task E1 Measure floating point operations; determine Arithmetic Intensity

Task E2 Extend range

# Solutions

Solutions are shown in a dedicated Jupyter Notebook.  
*You decide when to look at them!*

Please submit evaluation!



[bit.ly/sc19-eval](https://bit.ly/sc19-eval)

# Appendix

# References: Images, Graphics I

- [1] Oak Ridge National Lab. *Oak Ridge Launches Summit Supercomputer*. URL: <https://ornl.gov/news/ornl-launches-summit-supercomputer>.