

## **ECE599 Supercomputer Design and Analysis**

**Code Name – RP3 Titan (until the team comes up with a better code name)**

**Summer 2016**

**Dept. EECS, University of Tennessee, Knoxville**

**Project** – Design and Analysis of a Raspberry Pi 3 based Supercomputer (code named RP3 Titan)

### **Project/class Description –**

- Design, build, test and analyze a 128/256 core (32/64 node) ARM 64-bit based supercomputer using Raspberry Pi 3's as nodes.
- The primary goals are to establish the operating environment, application frameworks, application development tools, and libraries needed to support the effective operation of a supercomputer for students and faculty in the EECS department.
- The system will be an upgraded version of the Tiny Titan system constructed by ORNL for demonstration of how highly parallel supercomputers work.
- The students will work as a team to complete the project:
  - as a sustained computing resource for the department,
  - demonstrate the effectiveness of a set of applications (one per student),
  - complete a report on the design,
  - implementation and test of the system,
  - include an operations section in the report describing the operating environment, libraries, tools, and development framework best suited for application development and optimization.
- A successful 32-node system demonstration and companion report (with all target elements covered) is minimum for the class. A successful 64-node system demonstration and companion report (with all target elements covered) is the primary objective.

### **The system will be constructed and tested in multiple phases:**

1. Develop project plan and make assignments of responsibility to individual students to cover each major part of the project. The students will coordinate assignment of responsibilities based on project needs and individual interest.
2. 8 node system (+1 for cluster control/management and/or as an input node) that duplicates the operational characteristics of the ORNL Tiny Titan. Test must be run to demonstrate performance, power consumption, system utilization, and operating efficiencies of the RP3 Titan as compared to Tiny Titan. Establishment of the operating environment, libraries, tools and application development framework must start at this phase and carry forward for all subsequent phases of the project. Absolute performance analysis as well as scalability analysis must be done on the suite of applications selected for the project (one application per student). Also, scalability analysis of the operating environment components (e.g. drivers, libraries, file system, etc.) and hardware elements (e.g. network, cooling, power distribution, etc.) must be completed.
3. 16 node system (+1 for control/mngnt or input) with the same system evaluation test as #1.
4. 32 node system (+1 for control/mngnt or input) with the same system evaluation test as #1.
5. 48 node system (+1 for control/mngnt or input) with the same system evaluation test as #1.
6. 64 node system (+1 for control/mngnt or input) with the same system evaluation test as #1.

### **Application porting/development, testing and measurements:**

One application per student must be ported, developed and tested on the system. Scalability, operating efficiency, and system resource utilization must be measured on each application. Given the class is a multi-disciplinary team project, all applications may be developed by a subset of the class with other key aspects of the system (e.g. hardware design, build and test) handled by other groups of the class. The class must organize itself into subgroups with specific assignments in support of successfully completing the class project.

### **Class Deliverables:**

1. Project plan including dates when each phase and major parts of the project will be completed.
2. List of key software components targeted for the systems, including a component diagram of the operating environment for the full system (firmware, OS, libraries, middleware, tools, protocols, etc.)
3. List of key hardware components targeted for the system, including a system diagram showing how the system will be constructed.
4. Demonstration of the system's basic operation (show that all major resources of the system can be accessed and leveraged.)
5. List and detailed description of target applications. Details include type and structure of the application, libraries and system calls needed, operating environment needed, the purpose or intent of the application, etc.
6. Demonstrate the scope, problem addressed, and operational characteristics of each application selected for the project (one application for each student), including performance, scalability, system constraints/bottlenecks, possible improvements, etc.
7. Final report including the above deliverables as well as chapters covering: HW/SW design, HW/SW operational characteristics and test for each major subsystem, operating efficiencies and scalability, and a chapter for each application ported and tested on the system covering key detail characteristics of the application (e.g. application structure, scalability, system utilization, libraries used, bottlenecks, optimizations and potential improvements for future consideration).
8. Operations and How-To Manual (could be appendix or chapter in final report) which includes sufficient detail for students and faculty to: replicate the system, maintain the system, add new system functionality, and construct additional applications.

### **References:**

<https://tinytitan.github.io/>

<https://github.com/TinyTitan>

<http://www.zdnet.com/article/build-your-own-supercomputer-out-of-raspberry-pi-boards/>

<http://www.southampton.ac.uk/~sjc/raspberrypi/>

[http://www.southampton.ac.uk/~sjc/raspberrypi/raspberry\\_pi\\_iridis\\_lego\\_supercomputer\\_paper\\_cox\\_Jun2013.pdf](http://www.southampton.ac.uk/~sjc/raspberrypi/raspberry_pi_iridis_lego_supercomputer_paper_cox_Jun2013.pdf)

[http://www.southampton.ac.uk/~sjc/raspberrypi/pi\\_supercomputer\\_southampton.htm](http://www.southampton.ac.uk/~sjc/raspberrypi/pi_supercomputer_southampton.htm)

[https://www.element14.com/community/community/raspberry-pi/raspberry\\_pi\\_projects/blog/2016/03/03/raspberry-pi-3-cooling-heat-sink-ideas](https://www.element14.com/community/community/raspberry-pi/raspberry_pi_projects/blog/2016/03/03/raspberry-pi-3-cooling-heat-sink-ideas)

<http://www.mpi-forum.org/docs/mpi-3.1/mpi31-report.pdf>

<http://likemagicappears.com/projects/raspberry-pi-cluster/>

[https://www.youtube.com/watch?v=i\\_r3z1jYHAc](https://www.youtube.com/watch?v=i_r3z1jYHAc)

<https://www.youtube.com/watch?v=ubUTTrbEckM>

<https://www.youtube.com/watch?v=eIax2FUV218>

<https://www.youtube.com/watch?v=13x90STvKnQ&list=PLQVvva0QuDf9IW-fe6No8SCw-aVnCfRi>

<https://www.raspberrypi.org/magpi/pi-spark-supercomputer/>

### **Benchmark with other SoC boards (RP3 compared with other SoC boards):**

<http://www.phoronix.com/scan.php?page=article&item=raspberry-pi-3&num=1>

Initial list of components available to construct a 64 node RP3 Titan. This list will be updated as needed during the course of the project.

<b>Quantity</b>	<b>Item</b>
64+4*	<a href="#">Raspberry Pi 3 Model B+</a>
16	<a href="#">USB 2.0 6ft cables, 5 pack</a>
8	<a href="#">USB 10 port Charging Station, 2.1A per port maximum</a>
16	<a href="#">GeauxRobot Raspberry Pi 3 Model B 4-layer Dog Bone Stack Clear Case Box</a>
16	<a href="#">5 pack, 6ft Flat Ethernet Cables</a>
1	<a href="#">Wireless Microsoft Xbox 360 Controller</a>
2	<a href="#">PDU - 12 outlet</a>
2	<a href="#">TP-LINK TL-SG1048 48-Port 10/100/1000Mbps Gigabit 19-inch Rackmount Switch, 96Gbps Switching Capacity</a>
64+4*	<a href="#">64GB SD Card Ultra</a>
68	<a href="#">Addicore Raspberry Pi Heatsink Set for B B+ 2 and 3 (Set of 3 Aluminum Heat Sinks)</a>
8	<a href="#">AC Infinity MULTIFAN S3, Quiet 120mm USB Fan for Receiver DVR Playstation Xbox Computer Cabinet Cooling</a>

\* Additional nodes may be needed to support: customer control/management, storage management, visualization and/or network management.

### **Project Assumptions:**

- Raspberry Pi 3 Nodes (Pine64 Nodes may be made available during the project to evaluate effect of a more powerful node).
- Raspbian Linux – 64bit (32bit version used by ORNL Tiny Titan and Univ. of South Hampton)
- MPI 3.1
- Python Bindings for MPI and PIP
- Hadoop and Fortran support
- 64GB SD Card per node for storage and operating system kernel.
- Gbit Ethernet interconnect using Ethernet switch (router may be needed to support dynamic IP address assignment if static assignment is not used)
- Display interface provided through one node (could be dedicated node, beyond 64, for system visualization).
- Cluster control and management provided through one node (could be dedicated node, beyond 64).
- External interface (e.g. joy stick or game controller) provided through one node (could be dedicated node, beyond 64).

### **Considerations/Possible Issues (must be assessed in final report):**

- Memory capacity per node.
- Storage capacity per node and configuration (distributed versus centralized)
- Interconnect capacity, configuration, performance, bottlenecks
- Power, Packaging and Cooling
- Operating Environment
- Distributed Computing Programming Model (e.g. MPI versus others)
- File System (type, performance, bottlenecks, ...)
- Application framework and development tools.
- Scalability and performance – theoretical versus actual (applications, libraries, system calls, file system, tools, ...)
- GPU utilization
- Security

### **Major content to be included in Final Report and/or How-To Manual:**

- Hardware design description (including list of components, system/subsystem diagrams, , interfaces, signaling, and/or custom subsystems designs)
- Operating environment description (including OS, frameworks, libraries, protocols, interfaces, drivers, ...)
- Application development tools and frameworks description (including compilers, optimization tools, ...)
- Applications ported and tested (including type, structure, performance results, scalability, ...)
- Systems management environment

- Test results (including on all hardware and software components and applications, performance and scalability, issues found and resolved)
- Results of “consideration and possible issues” listed above (in the section with this name).

### Possible Application Areas:

- Data Analytics
- Thermo-dynamic simulations
- Fluid-dynamic simulations
- Computational Science (bio, nano, chemistry, ...)
- Monty Carlo simulations
- Circuit design simulations
- Natural Systems modeling
- Pattern/Image Recognition
- Deep learning using Caffe (might be way to leverage Nvidia GPUs) – [github.com/BVLC/caffe](https://github.com/BVLC/caffe) – can support multiple GPUs. On Linux and OS X platform. Used CUDA 7.0.



Figure 1: Raspberry Pi 3 Single Board Computer

### Raspberry Pi 3 Specification:

- SoC: Broadcom BCM2837
- CPU: 4× ARM Cortex-A53, 1.2GHz
- GPU: Broadcom VideoCore IV
- RAM: 1GB LPDDR2 (900 MHz)
- Networking: 10/100 Ethernet, 2.4GHz 802.11n wireless
- Bluetooth: Bluetooth 4.1 Classic, Bluetooth Low Energy
- Storage: microSD
- GPIO: 40-pin header, populated

- Ports: HDMI, 3.5mm analogue audio-video jack, 4× USB 2.0, Ethernet, Camera Serial Interface (CSI), Display Serial Interface (DSI)

### **SoC**

Built specifically for the new Pi 3, the Broadcom BCM2837 system-on-chip (SoC) includes four high-performance ARM Cortex-A53 processing cores running at 1.2GHz with 32kB Level 1 and 512kB Level 2 cache memory, a VideoCore IV graphics processor, and is linked to a 1GB LPDDR2 memory module on the rear of the board.

### **GPIO**

The Raspberry Pi 3 features the same 40-pin general-purpose input-output (GPIO) header as all the Pis going back to the Model B+ and Model A+. Any existing GPIO hardware will work without modification; the only change is a switch to which UART is exposed on the GPIO's pins, but that's handled internally by the operating system.

### **USB chip**

The Raspberry Pi 3 shares the same SMSC LAN9514 chip as its predecessor, the Raspberry Pi 2, adding 10/100 Ethernet connectivity and four USB channels to the board. As before, the SMSC chip connects to the SoC via a single USB channel, acting as a USB-to-Ethernet adaptor and USB hub.

### **Wireless radio**

So small, its markings can only be properly seen through a microscope or magnifying glass, the Broadcom BCM43438 chip provides 2.4GHz 802.11n wireless LAN, Bluetooth Low Energy, and Bluetooth 4.1 Classic radio support. Cleverly built directly onto the board to keep costs down, rather than the more common fully qualified module approach, its only unused feature is a disconnected FM radio receiver.