Analysis of Single- and Double-bit Flip Errors on Palmetto

Eric Paulz, Matthew Lam, Jacary Richardson, Kyle Mcmindes
(epaulz, mlam, jacaryr, kmcmind) @clemson.edu

I. Introduction

The Palmetto Machine is an amazingly complex architecture that contains many nodes on which users can have virtual machine like capabilities with in the range of thousands of compute cores at their disposal. Of interest to us for this class however is the errors that the Palmetto cluster has encountered. This is a class in fault tolerance so this fits very nicely with what we have discussed and learned in class throughout the semester. We were given an entire set of logs collected from the Palmetto cluster's compute nodes and their respective GPUs. The purpose of this study was to analyze the data, crunch the numbers, and come up with meaningful results that might help CCIT learn more about errors affecting their systems.

II. Motivation

There were several reasons for the choice of this project for this class. First, it gave us invaluable experience evaluating real life data from an actual High Performance Computing system. We were given the actual data collected, and were expected to put our efforts together to analyze this data. Also, this gave us a chance to learn more about the cluster itself and how it operates. This includes its hardware, software, and even some fault tolerance techniques that Palmetto already has in place. Finally, this was a good opportunity to apply teamwork principles inside of a situation that could potentially be encountered in a workplace environment.

III. Contribution

The goal of our project was to help CCIT find weaknesses in the infrastructure of the Palmetto cluster. We tried to make it as easy as possible to analyze the data given. To do this we parse the data and created graphs.

IV. Results

Calculation of Total Single- and Double-bit Flip Errors

We created a Python script to traverse the log files given from Palmetto GPUs. After gathering relevant data and cleaning it up for easier analysis, we extracted bit-flip counts for a few different components. As we iterated through each day in the logs, we subtracted the aggregate bit-flips of the current day by that of the previous day and recorded any changes. This allowed us to calculate the true daily count of errors on each GPU and then add them all up for an accurate total for the date range provided in the logs. The results can be found in Table 1.

	Single Bit	*Double Bit*
Device Memory	84057	193
Register File	35	0
L1 Cache	58	0
L2 Cache	17664526616	0
Texture Memory	0	0
Texture Shared	0	0
CBU	0	0
Total	**17664610766**	**193**

Table 1

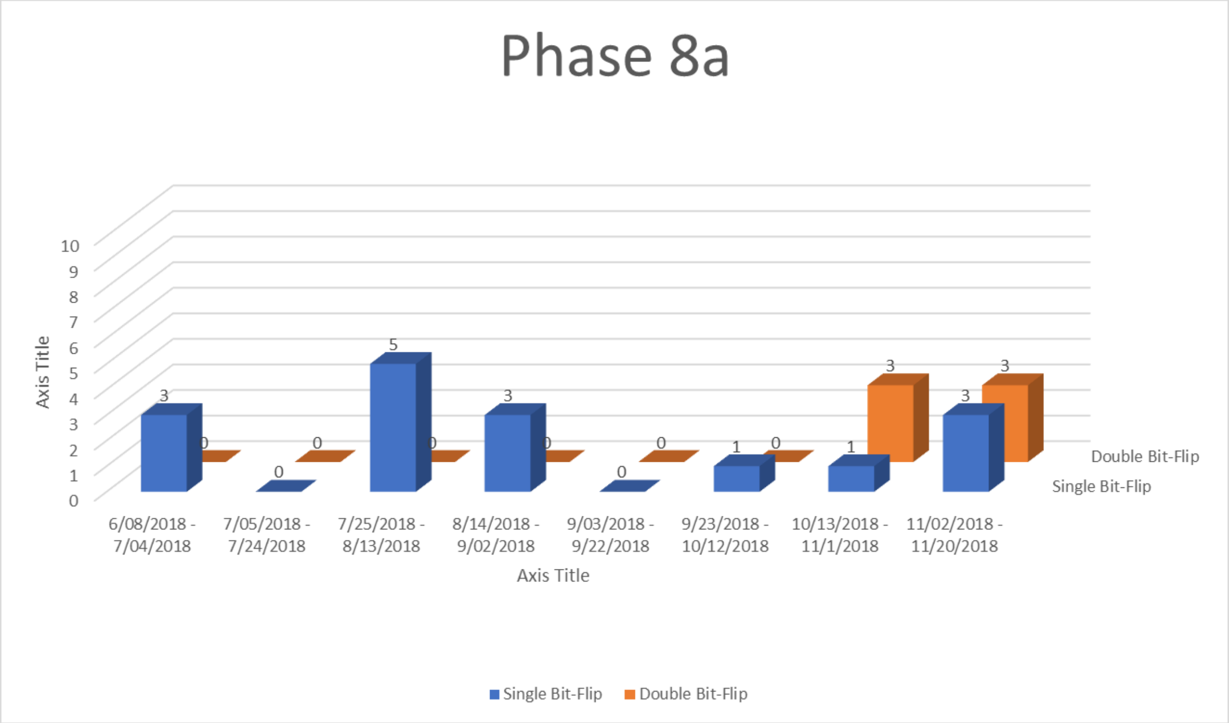
Mean Time Between Failure

We knew the each time a node was switched off or restarted, its volatile bit-flip count would reset to zero. Additionally, nodes are generally only restarted if there is a failure or if the volatile bit-flip count exceeded a certain maximum value. With this information, we decided that we could calculate the total number of failures represented within the logs by counting how many times a GPU's volatile flip count went from some value greater to zero on day back to zero on the next day. Using this method, we found that there were around 755 total failures over the course of around 3840 hours (160 days). This leaves us with a overall MTBF of ~5.086 hours. We were also able to break down the MTBF by phase to observe which phases tend to failure most frequently. This information can be found in Table 2. Note that only the phases in which we detected errors are represented in the table.

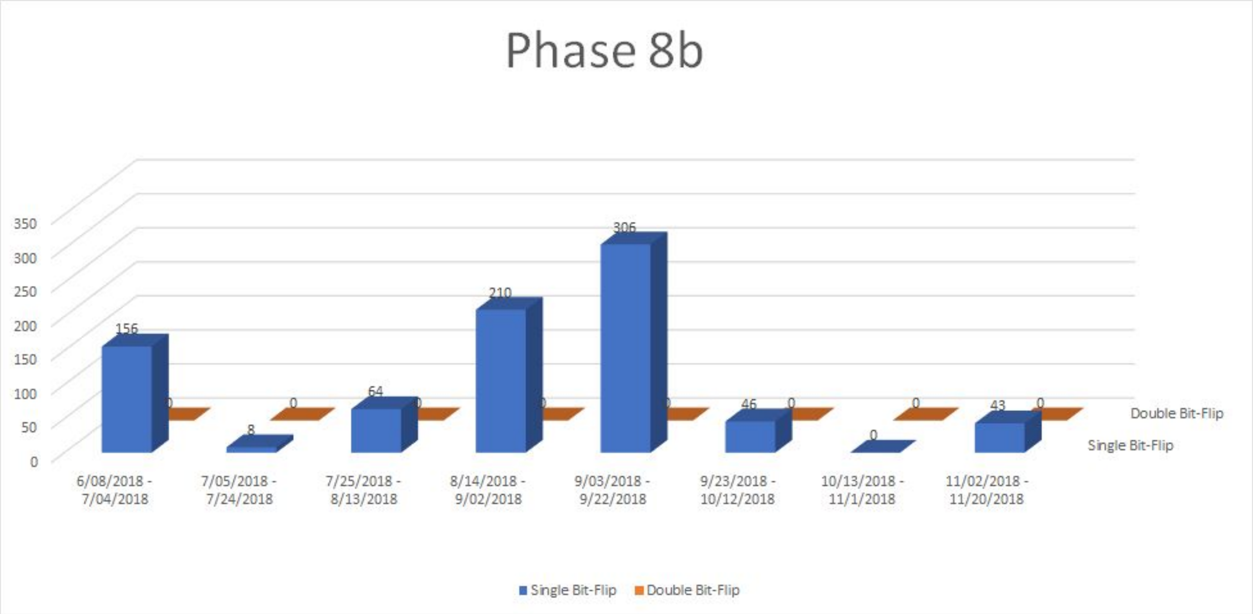
Phase	**MTBF**
Phase 08a	320 hours
Phase 08b	10 hours
Phase 16	46.829 hours
Phase 17	69.818 hours
Phase 18b	17.297 hours

Table 2

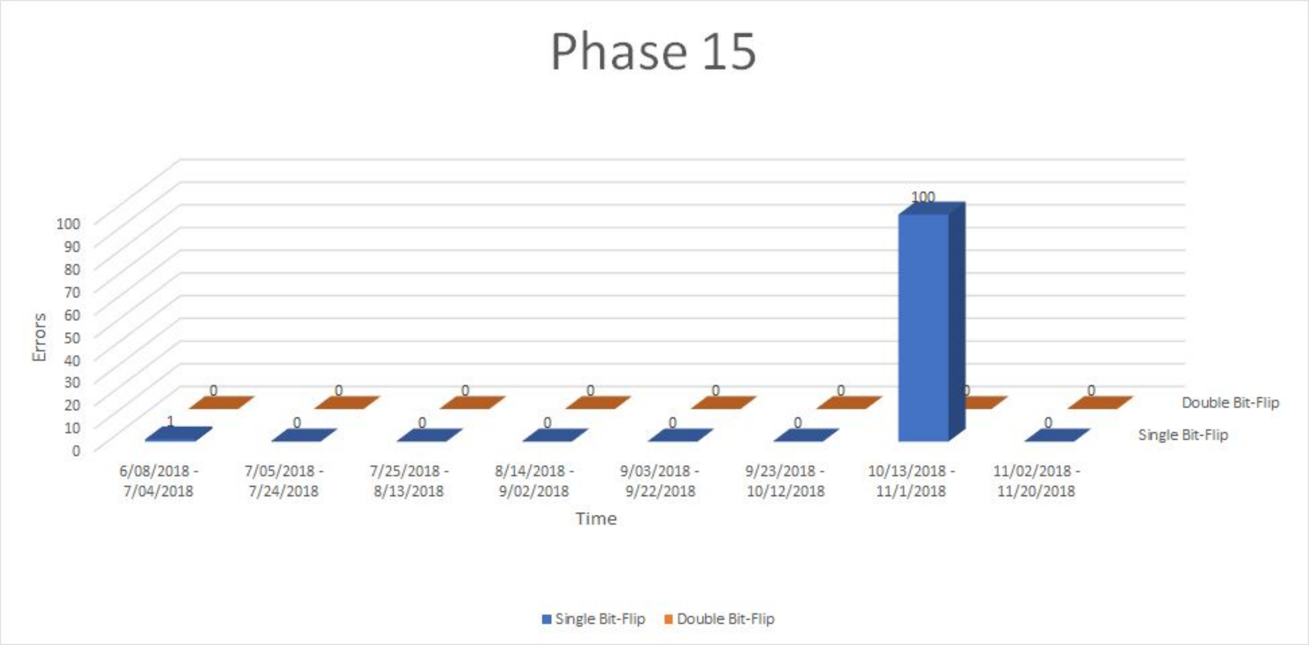
We admit that this may not be the most accurate or complete way of counting failures on Palmetto, but we found that it was the most effective method using the information we were given.



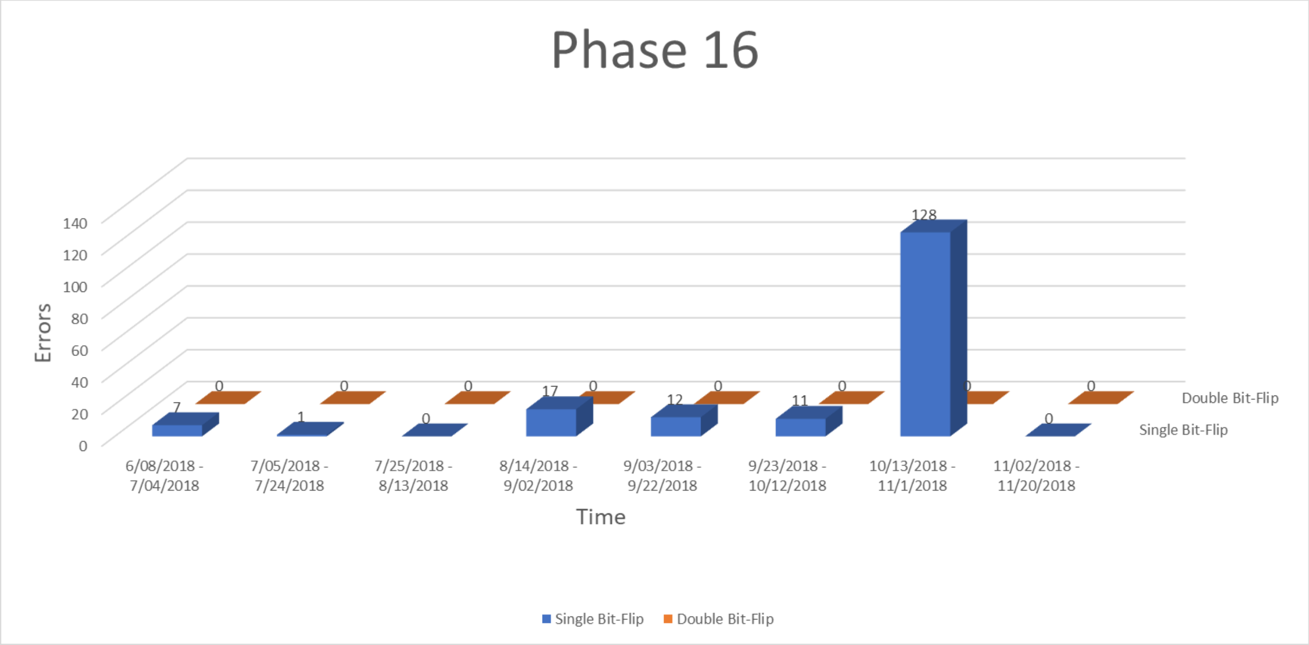
Graph 1: Single and Double bit flips of Palmetto Cluster Phase 8a



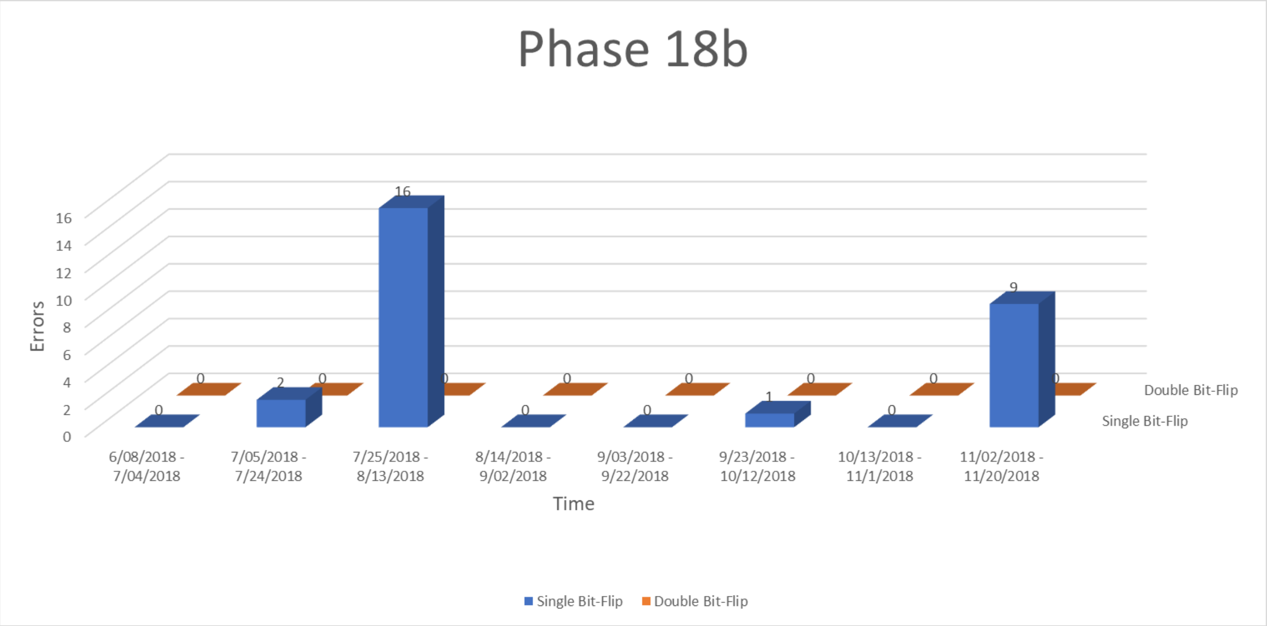
Graph 2: Single and Double bit flips of Palmetto Cluster Phase 8b



Graph 3: Single and Double bit flips of Palmetto Cluster Phase 15



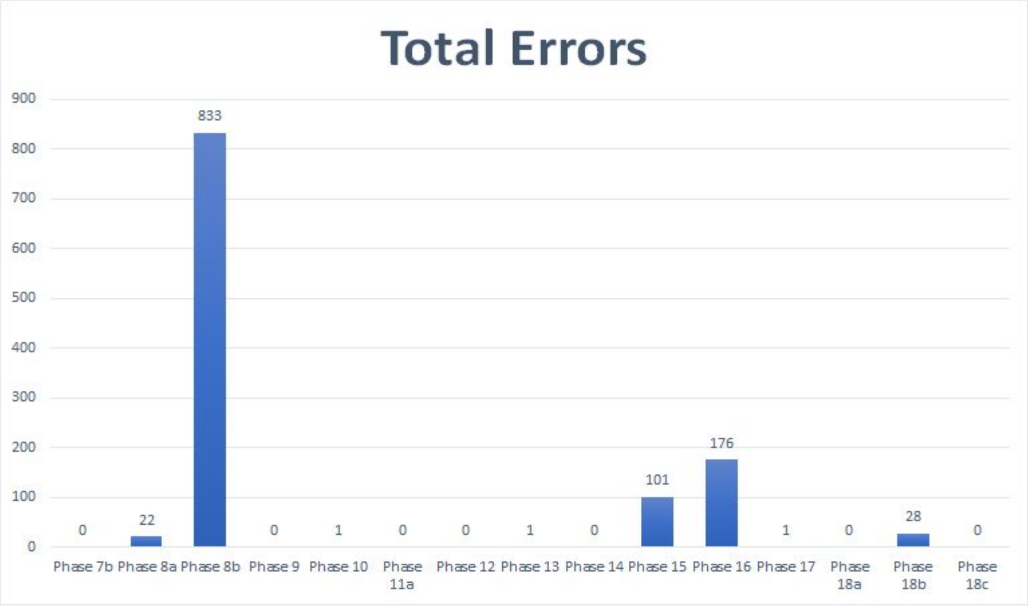
Graph 4: Single and Double bit flips of Palmetto Cluster Phase 16



Graph 5: Single and Double bit flips of Palmetto Cluster Phase 8b

Graphical result of the number of single and double bit flips.

These graphs are the results of the phases that had significant errors (more than 1). Each bar represents number of bit flips every 20 days. The other phases had little to no flips as shown in the Total Error graph below. These graphs are created from the the cumulative values of the logs. We made a script that took all the cumulative values from log. Then using excel we found the difference between the dates and determined the flips between the periods. Total Errors is the sum of the flips that happened. We found that only phase 8a had double bit flips and they were relatively recent. Overall, the flips were random over time with spikes when flips happen.



Graph 6: Total Errors

V. Discussion

As discussed in our presentation, one of our biggest uncertainties and main discussion point was the issue of the data spikes. Essentially what would happen is a node would turn off for a certain amount of time, but then when it would turn back on, those errors would be reintroduced, causing the number of errors to spike to an unreasonable amount. The discussion here was about whether this was an actual data reset of the nodes themselves so maybe they were recurring errors. Some of these data resets most went unnoticed as they were so small they would have no real effect on the overall data.

VII. Conclusions

We found that overall there were significantly more single-bit flip errors than double-bit flip errors. There are two reasons for this. The first is that it is more difficult to detect (and correct) double-bit flip errors. The second is that made double-bit flips are caught and result in retired pages. This affects the way we were detecting bit flip errors, so in many cases the doubles may not have been accounted for.

We were also surprised to see that there were a phases that did not contain any bit flips and/or failures than expected. Based on what we've learned this semester, we expected to see at least some soft errors or failures within a 5-6 month span. However, some of the phases showed no errors whatsoever. It would be interesting to investigate this further in the future.

We also noticed that the majority of bit-flip errors occurred between June 8 and July 4. At first we just believed that this may have been a time where old nodes were experiencing a lot of errors for whatever reason, and these spikes seemed somewhat strange to us. However, after speaking with some of the CCIT staff during and after our presentation, we learned that they were performing benchmark tests on the Palmetto system during this exact timespan. We were happy to hear this as explained the strange spikes in errors and it confirmed that our results were fairly accurate.

Overall, we believe that this project provided us with great experience in analyzing and understand a real HPC system.

VIII. Future Work

Looking forward, a few things that we could continue to look into for this project include an in depth look at the reliability of each node and even more specifically how each GPU factors into that. This research could include things like analyzing errors per node instead of per day. It could also include eliminating error prone nodes (the outliers in the data) and regraphing each phase to see if that phase was actually very reliable without that one node. Also, there are several instances where there could have been one GPU that was at fault, and it would be interesting and useful to go in depth with figuring out exactly which GPU needed to be fixed. This is such a broad topic that we obviously cannot cover all of the future analysis that could be done on this, but these are the few items that we came up with.