

1 Conclusion

1.1 ~~Introduction~~

The conclusion will start with a summary of what has happened in the previous sections. Then it will move on to answer the research questions that ~~was~~ introduced in the beginning of the thesis. After the research questions have been answered there will be a section with reflections of what could have been done better or different. The thesis will end with a section that ~~consider~~ what further work could be.

1.2 Summary

This thesis started in chapter 1 ~~with~~ explaining what persistent memory is, what are the advantages and disadvantages. Four research questions ~~where~~ introduced in the end of the section that will be answered later in this section.

The thesis continued with explaining the basic when programming with NVDIMM in chapter 2.

In chapter 3 I created a modified version of the Stream benchmark that could be used to measure the speed of the NVDIMM. I also created three other ~~benchmark~~ ~~benchmark that is~~ similar to each other where the threads are competing with each other for bandwidth. I used the benchmarks to measure the performance of NVDIMM on the server and made observation based on those performance measurements.

~~Chapter are~~ about DRAM and NVIDMM working together because the amount of data exceeds the total capacity of the DRAM. I created a formula that calculated how much should be allocated from DRAM to NVDIMM. I also created two ~~version~~ of a new test that could test if the formula is correct. The purpose of the two ~~version~~ was to which programming decision is the fastest.

~~The last chapter are~~ about NVDIMM and DRAM doing two different types of jobs. I created a new test where DRAM ~~are~~ working on generating data while the NVDIMM ~~are~~ transferring the last set of generated data to from DRAM to NVDIMM and analyzing the data afterward.

1.3 Research questions


1.3.1 Question 1

What is the data transfer speed of NVDIMM compared to DRAM?


The expectation before starting this thesis was that NVDIMM would be slower than DRAM. What interesting is how fast the speed of the NVDIMM is compared to DRAM. This question is possible to answer by comparing the results Stream benchmark int section ?? and the result from modified Stream benchmark in section ?. By comparing the speed from these two benchmarks there is which has been done in ?? in section ?? it possible to see how much faster the DRAM is.

1.3.2 Question 2

In an competitive environment, in what way will NVDIMM and DRAM affect each other?

When both DRAM and NVDIMM are competing for bandwidth the speed will be slower than what they would be if were alone with the bandwidth. When using all cores in a socket the sum of the DRAM speed and NVDIMM speed will be lower then the speed of sixteen DRAM threads running alone, this can be determined by comparing table ?? and ??. 

1.3.3 Question 3

When the size of the data is higher than the capacity of the DRAM, How much data should be transferred to NVDIMM? How many threads should be allocated to work on the data on NVDIMM? 

In chapter four I created a formula that can be used to find out how much data should be placed on NVDIMM based on what the speed of DRAM and NVDIMM is. By deciding first on how many threads should be allocated the formula will tell how much data should be placed on NVDIMM.

1.3.4 Question 4

While DRAM is working on a task, is it possible for NVDIMM to be working on a different type of task? In chapter four I created a program where a group of DRAM threads working on one task while a group of NVDIMM threads transfer the the result of that work to NVDIMM and do another type of task on that data. The program is slower then the

DRAM version of the program, but it is possible to have DRAM and NVDIMM do different kinds of tasks simultaneously.

1.4 Reflections

The benchmark section ??, ??, ?? had unstable NVDIMM threads that dropped their speed for some reason. More time could have been spent trying to find out why this is occurring.

While working on this thesis I could have saved a lot of time by having a greater attention to details. By looking at the results more critically I could have noticed mistakes more easier and it would not be necessary for other to point it out.

1.5 Further work

Further work would be to find out what causes the NVDIMM threads to drop their speeds sometimes that was shown in competing benchmark section in chapter three. If ~~its~~ possible to explain what causes this it might be possible to make programming choices that avoid the sudden drop from happening. The result is that with no drop in speed it might be possible to get more performance out of NVDIMM.