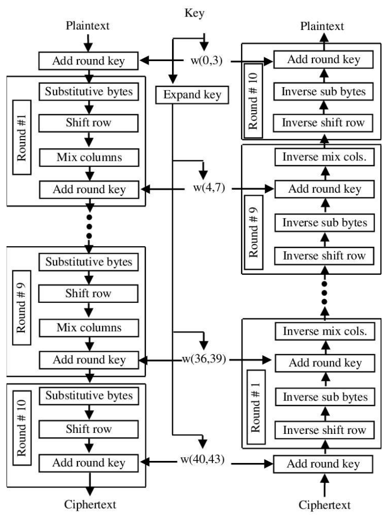
Nolan Anderson | Project Proposal | CPE 613 | Janruary 29 2023

1. What do you propose to do for this assignment?

In CPE 512 (Parallel Programming), I implemented AES Encryption using OpenMP. While it was good performance, I believe it can be much better using CUDA. In the assignment, I created a serial and parallel version of AES (encryption). I compared the speed up, cost, and performance when encrypting 100, 200, and 300 files. I want to expand on this previous project by also including decryption into it, and hopefully increase the performance of AES even further. Note that I already have input data for this assignment, so results are repeatable. There is an interesting article about this from nvdia developer:



https://developer.nvidia.com/gpugems/gpugems3/part-vi-gpu-computing/chapter-36-aes-encryption-and-decryption-gpu

My hope is that the resulting program will take in predefined data, encrypt using AES, then feed right into decryption. The resulting data should match the input data that I fed to the encryption. And, it should be faster than my OpenMP implementation.

2. Why is this problem useful to industry practitioners or you in your career? How will you justify this via your implementation results and report?

Additional CUDA experience aside, I believe I may be able to apply this implementation at work. Many encryption techniques companies use generally take a very long time to complete. While they may already have good encryption algorithms, I would like to try my own (and potentially make theirs better). Comparing my CUDA results to my previous implementations in serial and using OpenMP will show how much performance (and time) is being left on the table.

3. Provide a detailed schedule of your planned development process, including time for each of the steps listed below (and iterations of steps 4-7) as well as preparation of the final report.

1. Since I already have a C++ implementation, I will review this and make sure it still works in the first sprint (2 weeks). I also need to implement the decryption portion of AES.

2. The next sprint, I will create a simple CUDA implementation by just throwing the serial section into a kernel and running. Ensuring that all the data is the same. This may not take very long, so I am including a buffer for the first sprint since the decryption may take some time.
3. The next sprint, I will start by testing the two different implementations to get a baseline.
4. Next, I will devise a plan to make the parallel version even better.
5. The next sprints, up until the end of the semester, will be devoted to making the code better and better optimized
6. Optimization included in these sprints.
7. Continue to test as I add in changes in these sprints.
8. Repeat as necessary.

4. How do you plan to quantify and demonstrate success?

I plan to compare my results to the OpenMP implementation from last semester. This will include timing, speed up, cost, efficiency etc. This may take some additional testing, especially since I am adding the decryption portion to this assignment. If you would like to seem my first project submission, it can be found here:

https://docs.google.com/document/d/1tAJCauOs9sSHIpvLvP-PmSMnrSsds3Qc/edit?usp=share_link&ouid=117655861274293519793&rtpof=true&sd=true

Shoot me an email if this link doesn't work. I will likely us a similar format to this to compare my results, but obviously make it better. There are a few areas I would like to improve.

5. What risks or potential for failure do you see and how do you plan to mitigate them?

I think actually implementing the decryption portion of the assignment may be a bit difficult. I would like to make this a loop, where the encryption feeds into the decryption. So making sure I actually have good data from the encryption side will be crucial. To mitigate this extra programming time, I am allowing for extra buffer time in the first and second sprints. I believe this extra time will give me a good base to start from before attempting the CUDA implementation.