Documentation

Before you start:

1. ssh [group\_58@schroedinger.soe.ucsc.edu](mailto:group_58@schroedinger.soe.ucsc.edu)
2. Pass: KevinMatt23!
3. Then type in /bin/bash
4. Ready to work!

--------------------------------------------------------------------------

1. ./configure --extra-cflags=-O[NUMBER used for optimization like 0 or 3] --enable-pic --enable-shared
2. make
3. ./x264 --threads=1 soccer\_4cif\_30fps.y4m -o output.fov
   1. Ensure x264 is the right name or if it’s x264-O0 or “”””-O3
4. perf stat -B time ./x264 --threads=1 soccer\_4cif\_30fps.y4m -o output.fov
   1. Ensure x264 is the right name or if it’s x264-O0 or “”””-O3
   2. Using perf stat gives you more in depth details than just time by itself
   3. Change threads=1 or 4 or any number if necessary
5. Use perf record ./<exe> to generate a perf.data file
   1. Use perf report perf.data to open generated file
6. Perf stat --topdown -a ./x264-O[number] <video name> -o output.fov

--------------------------------------------------------------------------

Video transfer instructions (I believe you only need to do this one time only which I did)

1. Go on your local account and run these commands
2. scp /Users/KevinLe/Downloads/soccer\_4cif\_30fps.y4m [group\_58@schroedinger.soe.ucsc.edu](mailto:group_58@schroedinger.soe.ucsc.edu):/home/group\_58/
3. It will ask you to log into group\_58 and then you’ll transfer it.
4. Move the video to your location accordingly

--------------------------------------------------------------------------

Notes:

* -o = output
* -o output.fov stores our output into a file called output.fov Apparently we don’t need to do anything with that output file though.
* -O0 IN gcc is no optimization
* -O3 turns on some optimization

For # 5 Notes:

1. Front end - time spent waiting for instructions to arrive - fetching instructions from memory
2. Back end - time spent waiting for operands to arrive
3. Bad speculation = time you spent recovering from a wrong speculation
4. Retiring - when you have completed your instruction and you have retired to memory or save to memory. This is when you executed it and have actual results.

* In bad scenarios, we are still waiting.
* In good/optimized scenario, we are not wasting any time because we are retiring instructions every time.
* Look up wiki page for CPI
* Cant calculate the theoretical optimal average CPI with percentages only (screenshots). We need more info. TA does not know how to get that yet since he/she hasn’t gotten that far.
* You can either do an average for all 4 cores put together or list all 4 cores individually.