

# Help Topic: Assessing the Specs of your Machine

Spring 2022

Michael Benjamin, mikerb@mit.edu  
Department of Mechanical Engineering  
MIT, Cambridge MA 02139

---

## Assessing the Specs of your Machine

The vast majority of machines built in the last 5 years or so are probably sufficient for use in our lab, or in MIT 2.680. But let's discuss some of the metrics we use for those who may not be familiar, or for those who may be considering upgrading their machine:

- Number of CPUs, or "cores"
- CPU Speed
- RAM memory
- Hard disk memory

For all of the above information, there is both a GUI based method to find the info you need, and several command line methods. We encourage you to begin learning the command line versions, because:

- You may not always have access to the GUI version if you are logging remotely into a machine (when it is on a robot for example).
- GUIs can vary between Linux distros, where the command line tools tend to vary much less.
- Knowing how to get system info from the command line allows one to use this info in a script or other tools you may encounter or build yourself later on.
- Command line mastery tracks very closely to mastery of field robotics. We encourage you to learn the command line version generally, and rely less on a GUI generally. Some system GUI tools are indeed handy and powerful and easier to use, but nevertheless, the rule of thumb holds: know how to get your info from the command line.

## Determining the Number of CPUs

Your computer is actually comprised of several computers, or CPUs, or Central Processing Units. There are probably several dozen pieces of software (jobs) running on your computer at any given time, and your operating system time-shares these across CPUs. Later when we run course autonomy software, you will be launching even more jobs that the operating system has to handle. The more CPUs, the better.

How many CPUs does your machine have? If you're running linux, here's one way to find out:

```
$ nproc --all  
4
```

Other Linux utilities that may give you similar information are `lscpu`, and examining the contents of `/proc/cpuinfo`. For a graphical tool, try launching `hardinfo` from the command line.

If you're on a MacOS machine, here's one way to find out:

```
sysctl machdep.cpu.core_count
machdep.cpu.core_count: 8
```

On the Mac, a really handy tool is just to pull down the Apple menu, always in the upper left-hand corner of every window, and select "About This Mac", which should be the very first selection. It will bring up a pop-up window similar to:



Figure 1: The About This Mac utility.

In this window you can see (a) the number of processors, 8 in this case, (b) processor speed, 2.4Ghz in this case, and (c) the amount of system RAM, 32Gb in this case.

## Determining the CPU Clock Speed

The CPU clock speed reflects the speed at which each CPU can handle the many processes or jobs it may be executing at any one time. In terms of overall performance in our lab or in MIT 2.680, it is probably more important to have more CPUs (cores) than the actual clock speed. But nevertheless, it is a factor. The clock speed can be obtained from both of the GUIs discussed above for the Mac (About this Computer pull-down menu) and through the `hardinfo` GUI if it is available in your Linux distro.

In Linux, to get Clock Speed info on the command line:

```
$ cat /proc/cpuinfo | grep MHz
cpu MHz          : 2793.155
cpu MHz          : 2793.155
```

The two lines of output above, also indicate that this machine has two CPUs.

On the Mac, to get Clock Speed info on the command line, there are a couple options, both using the `sysctl` utility:

```
$ sysctl machdep.cpu.brand_string
machdep.cpu.brand_string: Intel(R) Core(TM) i9-9980HK CPU @ 2.40GHz

$ sysctl hw.cpufrequency
hw.cpufrequency: 2400000000
```

## Determining the Amount of System RAM, and Hard Disk Storage

Your computer has two primary types of memory. Random Access Memory (RAM) refers to the memory used by active processes (programs, applications), running on your computer. The second type of primary memory refers to your hard drive space, which holds static files stored on your computer like the PDF you were reading last week. When you open a PDF to read it, likely you are launching a PDF reader application which loads the PDF from the hard disk into RAM while you are using the application and scrolling through the document. When you close the PDF reader, the RAM it was using is free to be used by other applications, and the PDF document is retained on your hard disk.

The amount of RAM on your computer is very relevant to your ability to get things done, especially since newer versions of software (like your PDF reader, or Web browser), use more and more RAM as newer versions are released. Hard disk space is becoming less relevant with the increased adoption of cloud storage. For example, many people have all their personal photos stored in the cloud, or on their mobile device. Many students now compose their entire thesis on cloud based services like Overleaf. Of course, this does create a dependency on a strong network connection, the nevertheless for these reasons people tend care less about the size of their hard disk. That being said, another trend is that the "basic software" that comes pre-installed on your computer, is also growing. If you buy a computer with 256Gb of hard disk space, you may find that almost all of that is used up on day 1.

In Linux, to get the system RAM info on the command line:

```
$ free -m
```

	total	used	free	shared	buff/cache	available
Mem:	7942	1210	508	12	6223	6421
Swap:	7627	463	7164			

This indicates that there is 7942Mb, or roughly 8Gb of RAM on this computer.

On the Mac, to get the system RAM info on the command line, here are a couple options using the `system_profiler` or `sysctl` utilities:

```
$ system_profiler SPHardwareDataType | grep "Memory:"
Memory: 32 GB

$ sysctl hw.memsize
hw.memsize: 34359738368
```

The second result is given in bytes, rather than gigabytes (GB) as with the first utility.

## Determining the Amount of Hard Disk Storage

In Linux, to get the hard disk capacity info on the command line:

```
$ df -H /
Filesystem      Size  Used Avail Use% Mounted on
/dev/sdd5       829G  711G   77G  91% /
```

This indicates that the hard disk is roughly 830Gb on this computer.

On the Mac, the hard disk space and current capacity can be found in the "About this Mac" GUI under the tab "Storage".

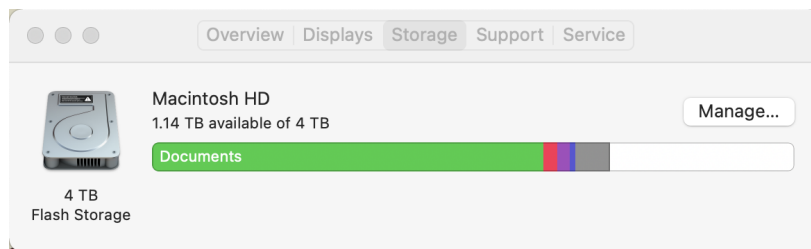


Figure 2: The About This Mac utility and the Storage Tab

To get the system hard disk capacity info on the command line, the same command used for Linux can be used:

```
$ df -H /
Filesystem      Size  Used Avail Capacity iused      ifree %iused  Mounted on
/dev/disk1s5s1  4.0T   15G   1.1T    2% 553788 39080576172    0% /
```

This indicates that the hard disk is roughly 4 Terabytes on this computer.