* A task is defined as a completely separate set of instructions that can be executed by a processor.
  + Pipelining is defined as breaking up a program into specific sets of task that can be computed by specific units within the processor.
  + Shared Memory is a computer architecture that allows for any processor in a system to access the same main memory over a bus.
  + Communications in concurrent and parallel processes and threads is used to be able to express how each process is to collaborate with each other. Also, it allows for light weight parallel threads to not be executing their code in a critical section at the same time as another thread. This communication can happen over shared memory or via dedicated sockets via a network in distributed computing applications.
  + Synchronization is used
* The shared memory model is similar to the Von Neumann architecture in that parallelized processes share the same data from within main memory. These processes are independent of each other but share the same critical sections of data in main memory. The threads model is based on another layer of abstraction where there is shared memory between the sub-process/threads of a main heavy weight process which is responsible for forking and joining sub-parallel threads as it sees fit.
* Parallel programming is programming that allows for multiple simultaneous instructions on the same set of data to allow for faster overall computing.
* System on a Chip integrates the CPU, GPU, RAM, and other aspects onto a single PCB to allow for computing on a single compact package.
* SOC is cheaper, optimized, portable, lower power. Having separate components can cause problems with compatibility and performance. With a SOC you can guarantee every component won’t bottleneck the other component and that they all work well together.