Technical Computing for the Earth Sciences, Lecture 10:

# More Parallel: MPI, GPUs

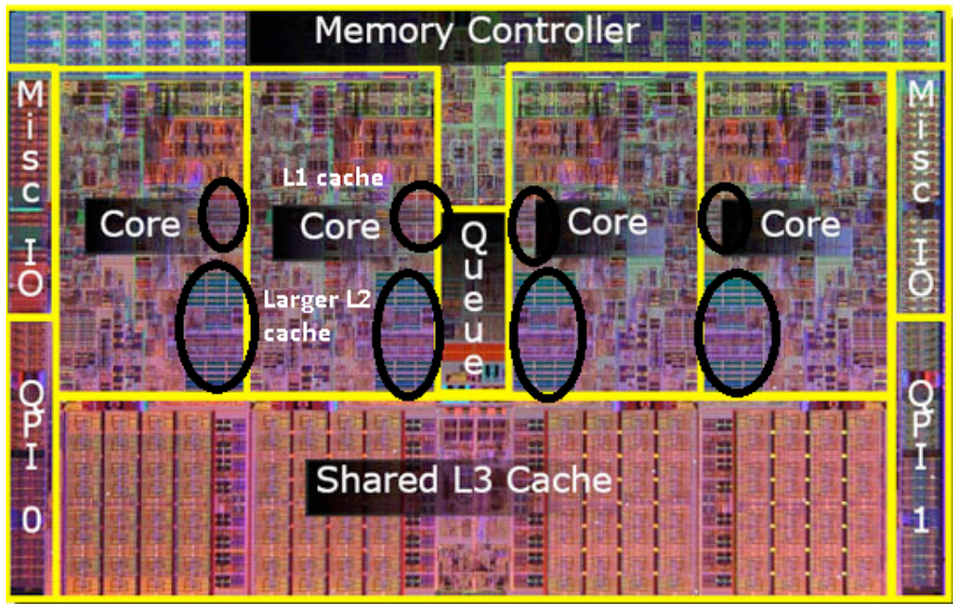
EARS 80.03

# MPI, the Message Passing Interface

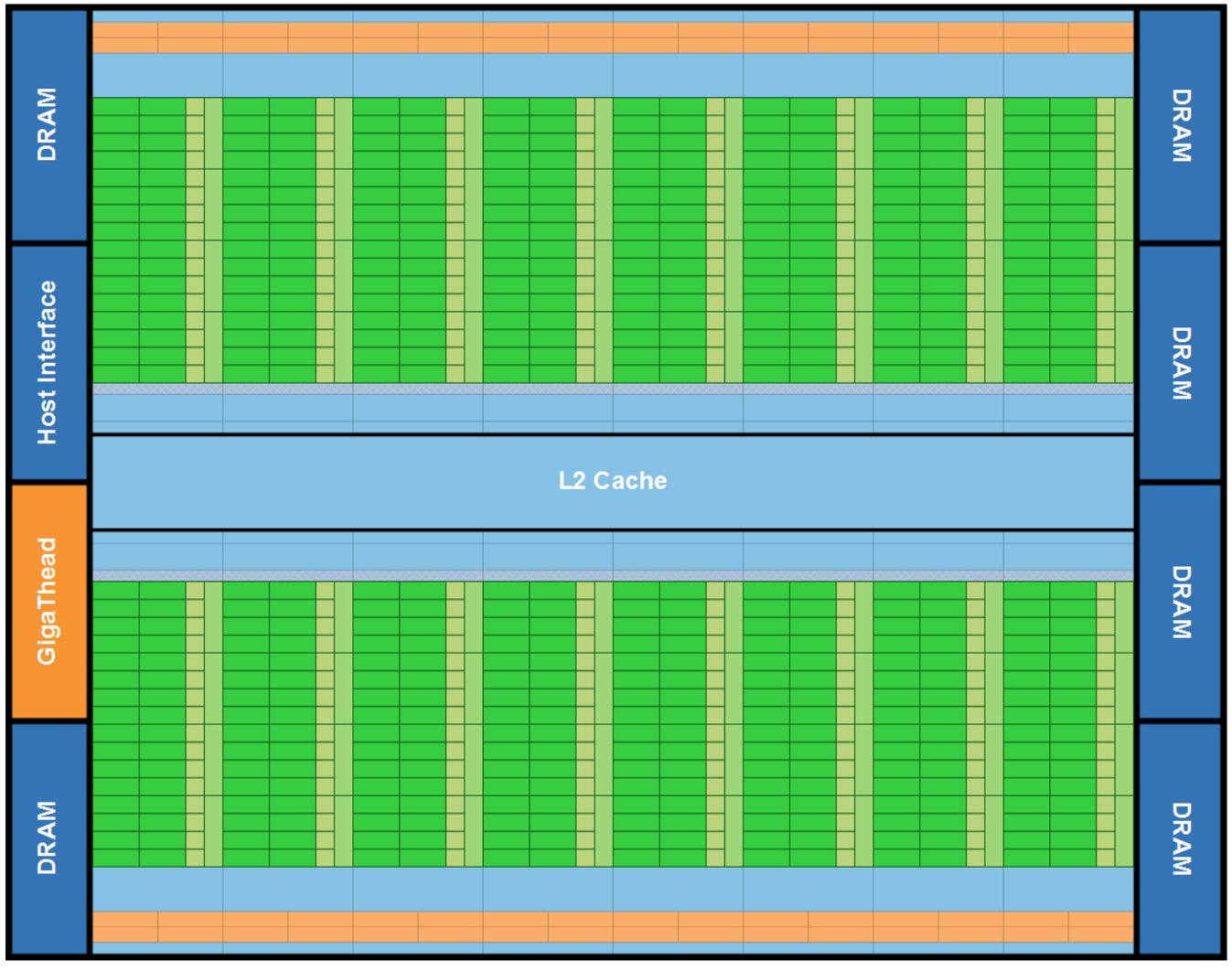
- MPI itself is actually a standard, not a piece of software
- There are two main implementations of this standard, OpenMPI and MPICH. Because they both implement the standard, they're effectively interchangeable.
- These days, almost all HPC / supercomputing uses a distributed memory model at least *between nodes*
    - Almost all distributed memory parallel programming is done with message passing, and most specifically with MPI.
        - Message passing is pretty much exactly what it sounds like. You manually write what information ("message") you want to send from which task to which other task
- Mostly called from C, C++, and Fortran (since these languages still make up 90% of HPC / supercomputer workloads)
- In Julia, use MPI.jl

# GPU basics

## CPU




## GPU



# GPU basics

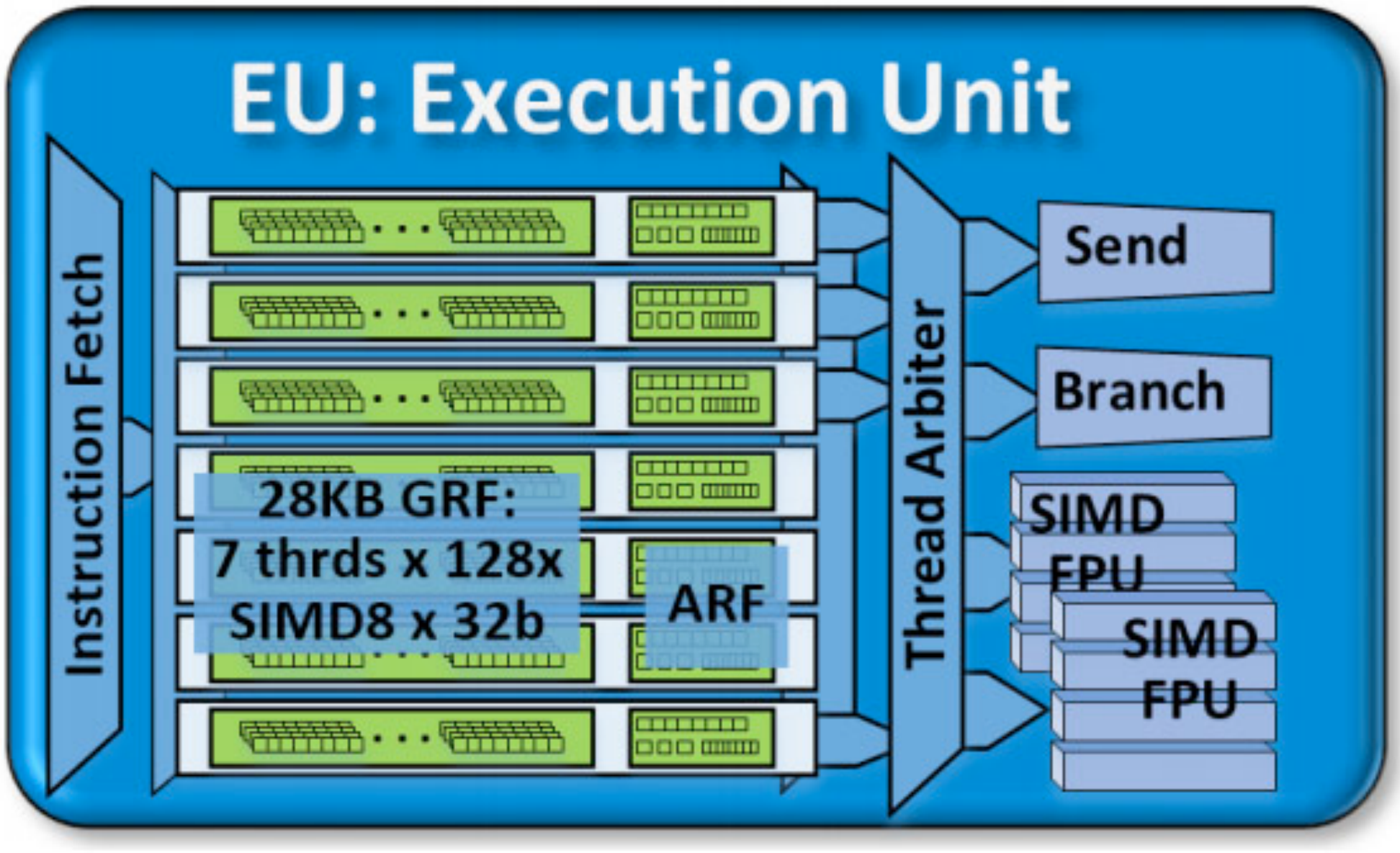



*Figure 3: The Execution Unit (EU). Each gen9 EU has seven threads. Each thread has 128 SIMD-8 32-bit registers (GRF) and supporting architecture specific registers (ARF). The EU can co-issue to four instruction processing units including two FPUs, a branch unit, and a message send unit.*

# GPUs in Julia

- CUDA.jl
  - CUDA.jl provides custom types (CuArray) extends the base functions (+, *, etc., etc.) such that you can use normal Julia functions and normal Julia code on the GPU
  - On the back end, it does this using CUDA, a GPU programming language developed by NVIDIA
  - NVIDIA and CUDA unfortunately has a bit of a monopoly on the scientific “general purpose gpu” programming market
    - AMD and Intel GPUs can use OpenCL instead, but the future of OpenCL is unclear since it is now officially no longer supported by Apple, who started it.
      - Apple is now trying to get people to use something they’re calling “Metal” instead now, but no one uses that for scientific computing (yet)
- AMDGPU.jl uses AMD’s ROCm to let you run Julia on AMD GPUs — but ROCm only works on Linux currently