Fine Grain Functional Parallelism

- Scalar pipeline : IPC ≤ 1

- Instruction Level Parallelism

Finding Instruction Level Paralleism

- Dynamic (Superscalar)
  - Using hardware
  - Expensive (hardware cost, power consumption)
  - Analyze instruction window
  - All delays and dependencies known
- Static (VLIW)
  - Using software (compiler)
  - Inexpensive
  - Analyze whole program
  - All delays and dependencies not known

How superscalars work

- Multiple pipelined EUs for parallel execution
- Fetch multiple instructions in a buffer, parallel decode – instruction window
- Dynamic scheduling - out of order
- Dependency check and resource check before execution
- Speculation
- Preserving the sequential consistency and exception processing - in order retirement

Limits on ILP

- Limited ILP in the application (available ILP)
- Limitation of HW and SW in exploiting given ILP (achieved ILP)
  - limited EUs
  - limited instruction window
  - limited issue capability
  - renaming limitations
  - imperfect branch/jump prediction
  - imperfect load/store speculation
  - imperfect cache

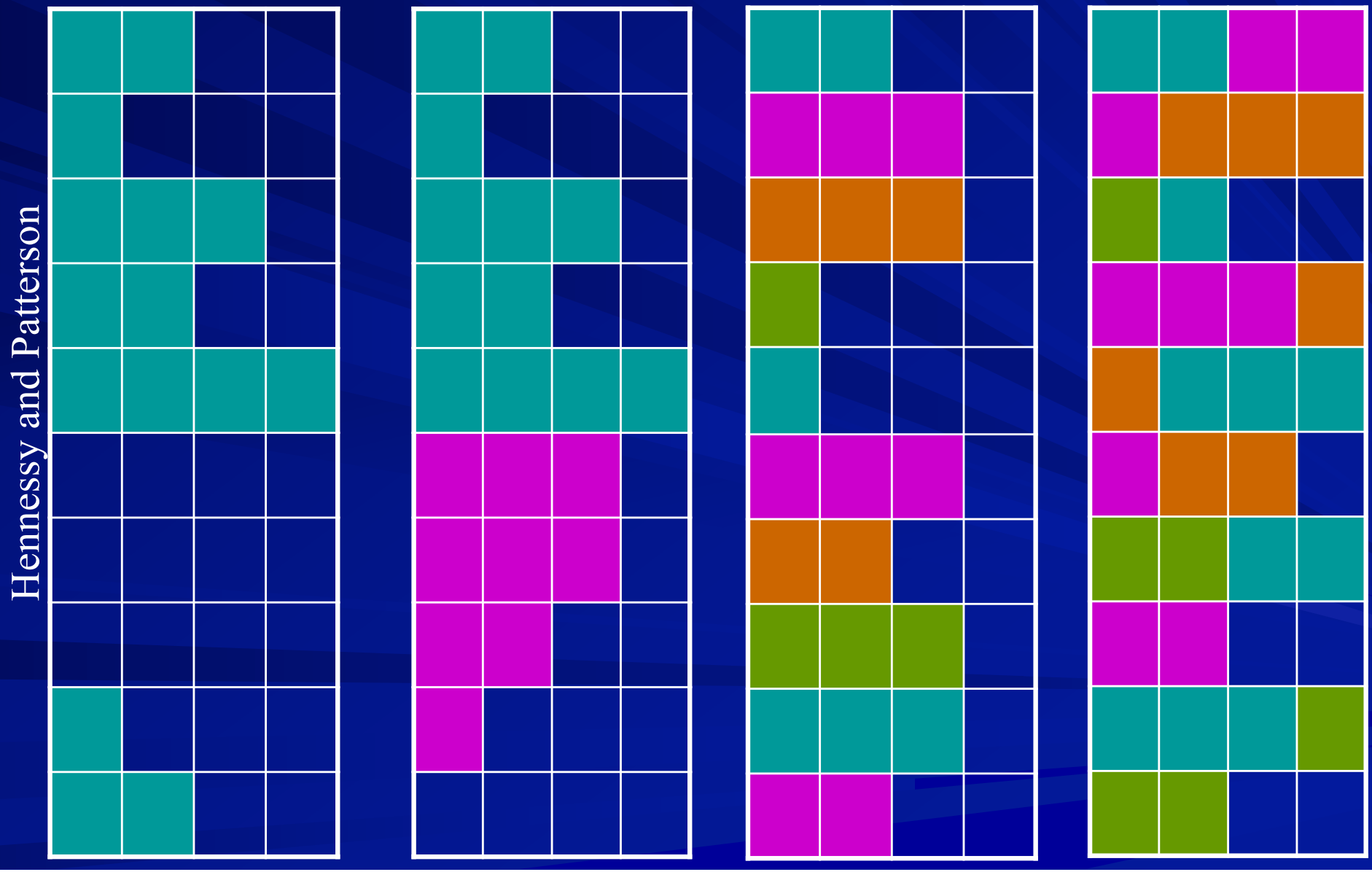
Beyond ILP

- Multithreading over ILP Datapath
- Multithreading over multiple cores

Multithreading over ILP datapath

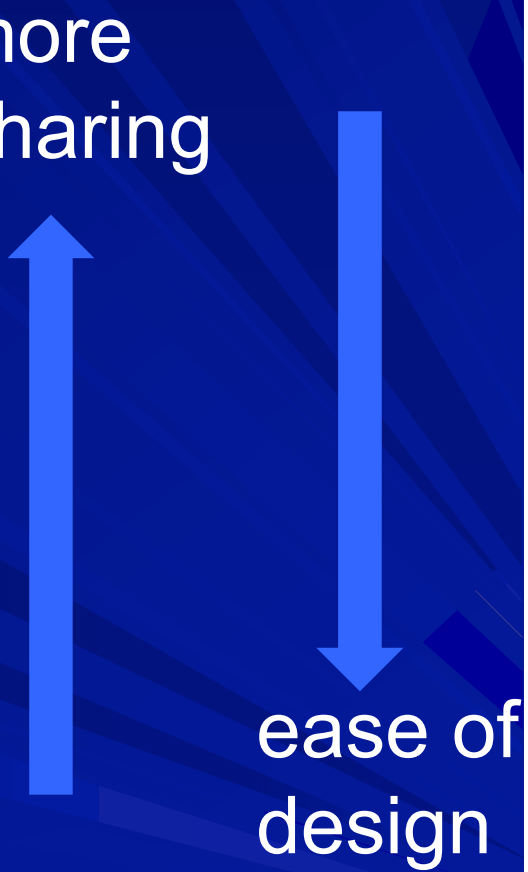
- The state needs to be replicated for each thread
  - program counter
  - registers
- Memory is shared
- Require ability to switch from one thread to other quickly

ILP and Multithreading

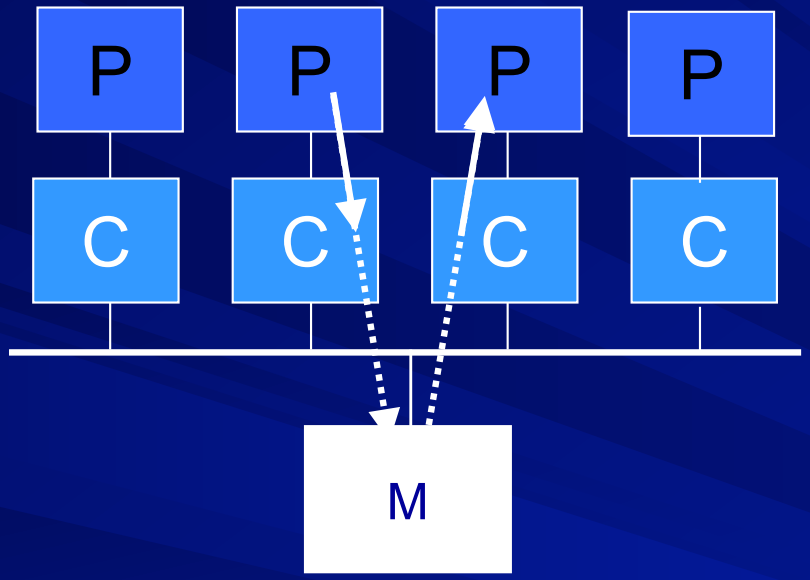


ILP, SMT and Multicores

- ILP
  - Multiple execution units
  - RF, caches and memory shared
- SMT
  - Multiple execution units and RFs
  - Caches and memory shared
- Multicore
  - Multiple EUs, RFs and caches (upper levels)
  - Caches (lower levels) and memory shared



Memory hierarchy in Multicores



- Multiple copies of data in caches may exist
- ⇒ Problem of cache coherence

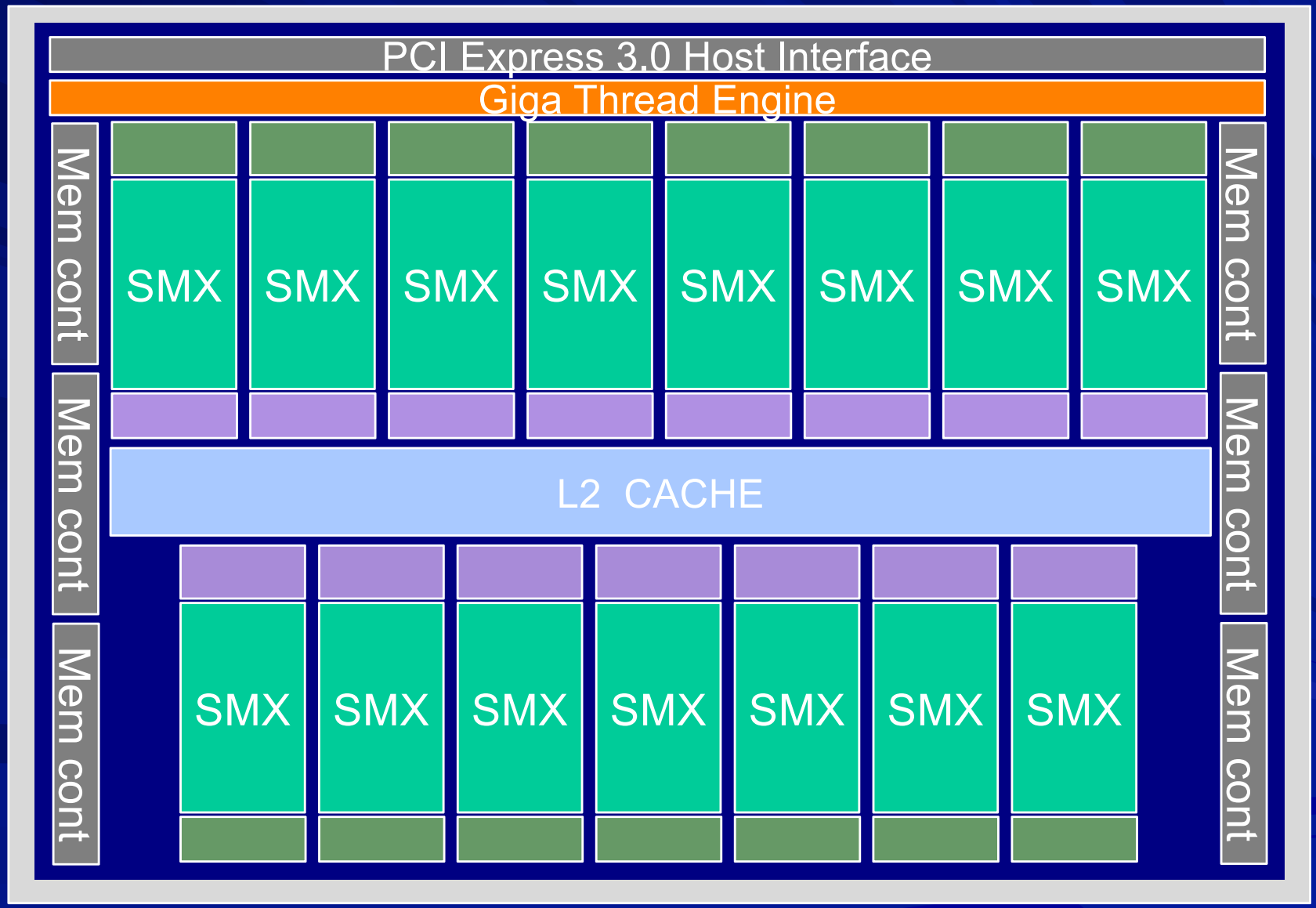
Solution:

- Cache controllers follow “coherence protocol”
- Interaction and communication in a mutually agreed manner

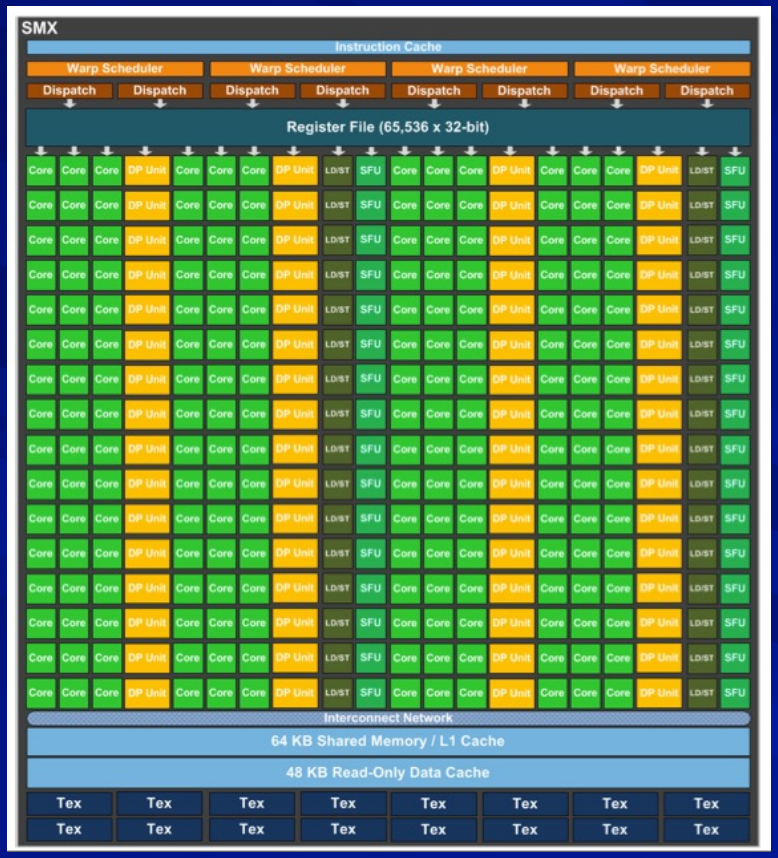
Graphics processors

- Developed originally for accelerating graphics
- Now used for General Purpose computing as well
- Major manufacturers : Nvidia, AMD, Intel
- Support for programming : CUDA (for Nvidia devices), OpenCL (general)

Nvidia’s Kepler GK110



SMX: Streaming Multiprocessor



16 Rows

192 CUDA cores, 64 DPUs

32 LSUs,32 SFUs

Thanks