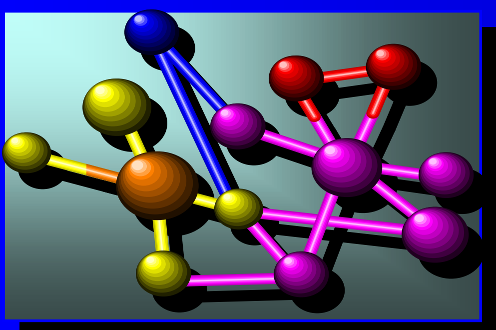
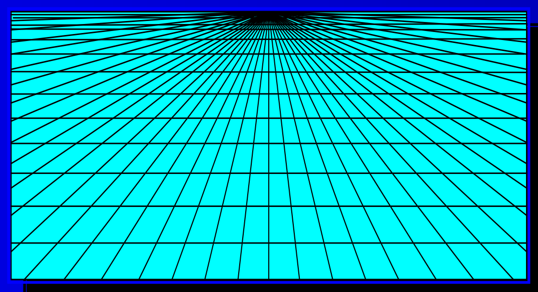
Need of more Computing Power: Grand Challenge Applications

Solving technology problems using computer *modeling*, *simulation* and *analysis*

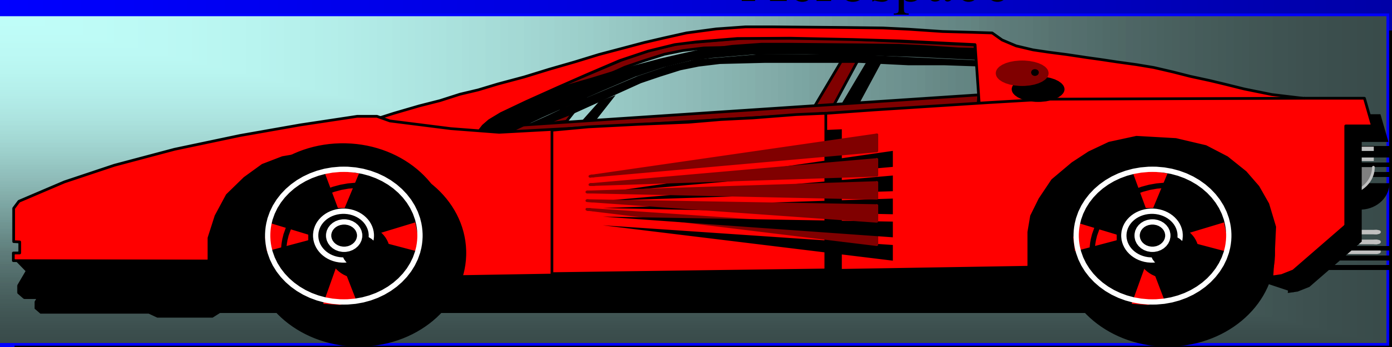


Life Sciences



Aerospace

Geographic Information Systems



Mechanical Design & Analysis (CAD/CAM)

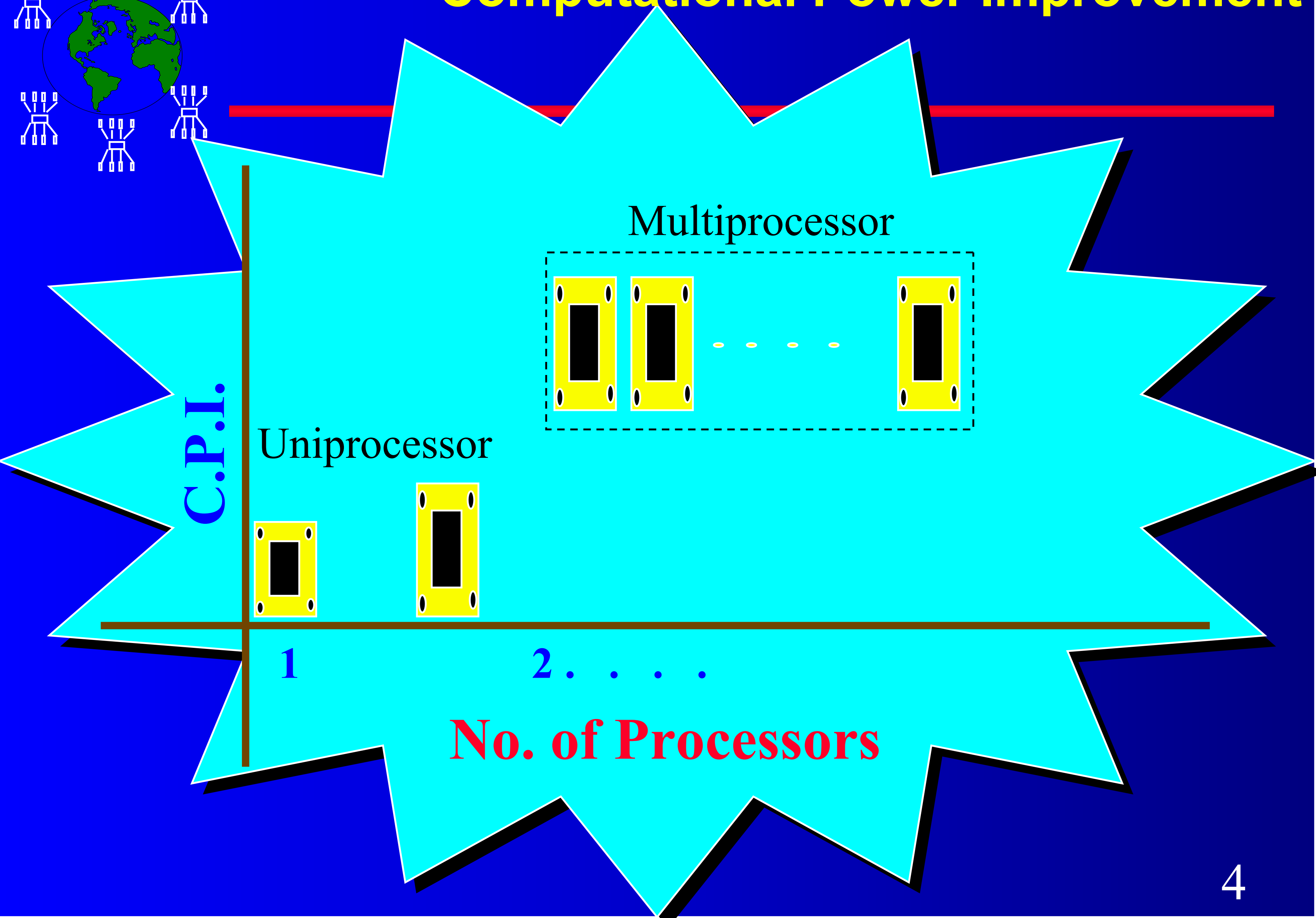
How to Run App. Faster ?

- There are 3 ways to improve performance:
  - 1. Work Harder
  - 2. Work Smarter
  - 3. Get Help
- Computer Analogy
  - 1. Use faster hardware: e.g. reduce the time per instruction (clock cycle).
  - 2. Optimized algorithms and techniques
  - 3. Multiple computers to solve problem: That is, increase no. of instructions executed per clock cycle.

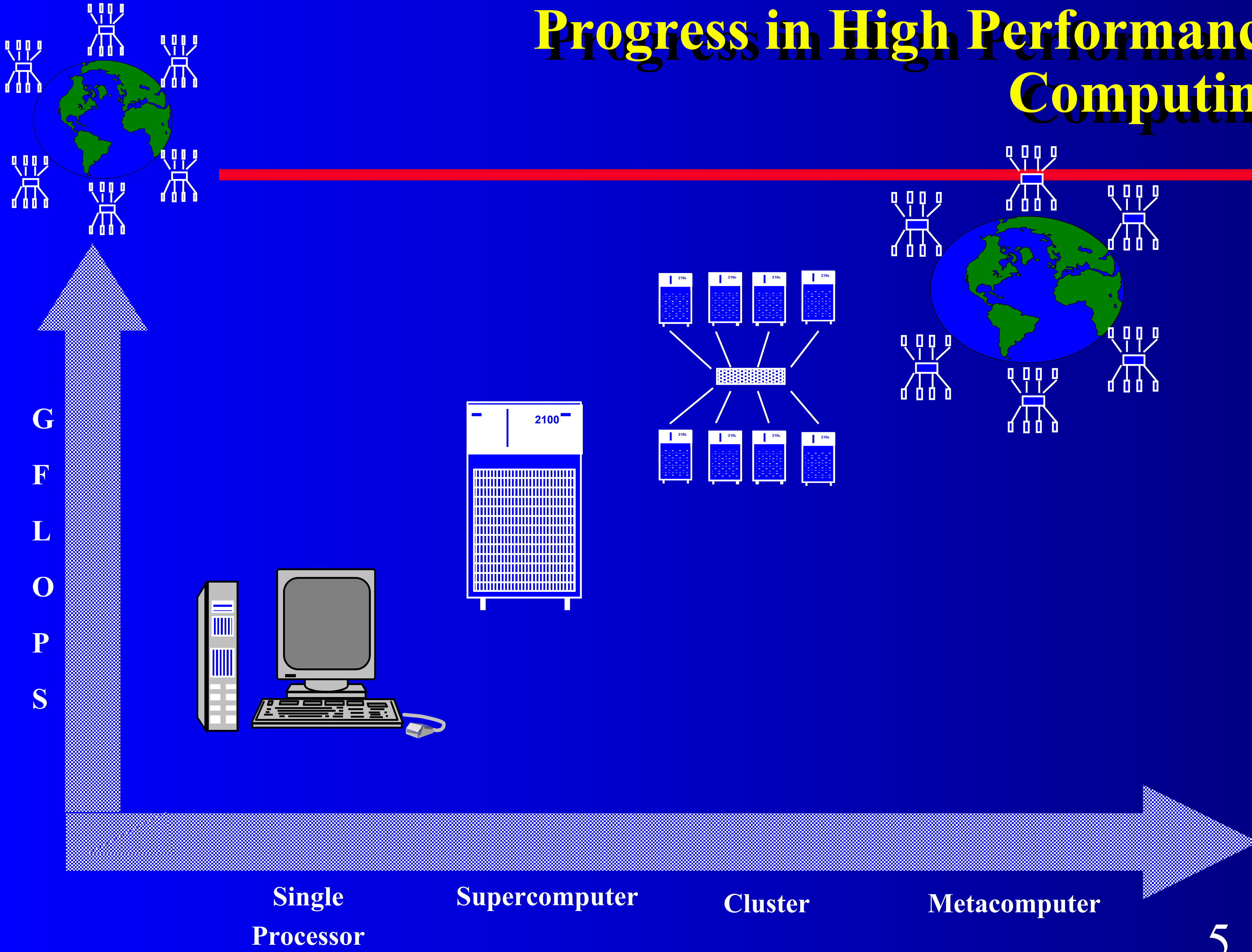
Sequential Architecture Limitations

- **Sequential architectures reaching physical limitation (speed of light, thermodynamics)**
- **Hardware improvements like pipelining, Superscalar, etc., are non-scalable and requires sophisticated Compiler Technology.**
- **Vector Processing works well for certain kind of problems.**

Computational Power Improvement



Progress in High Performance Computing



Why Parallel Processing NOW?

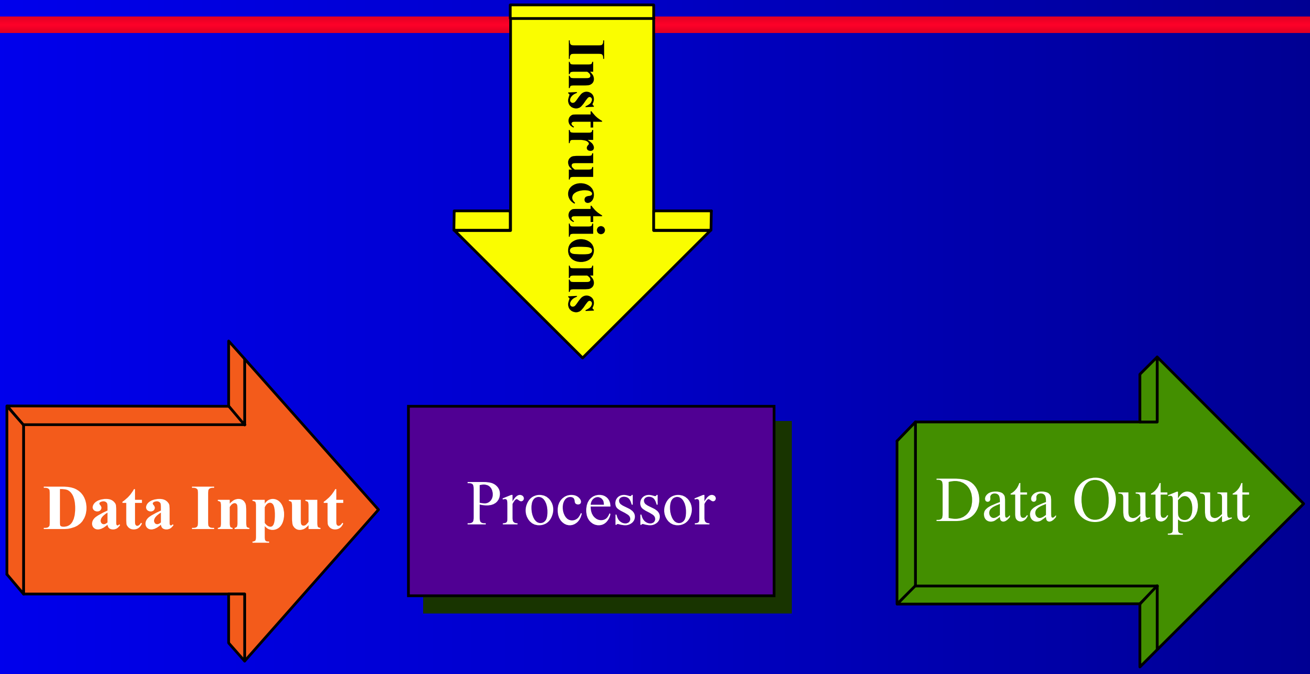
- **The Tech. of PP is mature and can be exploited commercially; significant <u>R & D</u> work on development of tools & environment.**

- **Significant development in <u>Networking</u> technology is paving a way for heterogeneous computing.**

Taxonomy of Architectures

- **Simple classification by Flynn:**
  (No. of instruction and data streams)
  - **SISD** - conventional
  - **SIMD** - data parallel, vector computing
  - **MISD** - systolic arrays
  - **MIMD** - very general, multiple approaches.
- **Current focus is on MIMD model, using general purpose processors or multicomputers.**

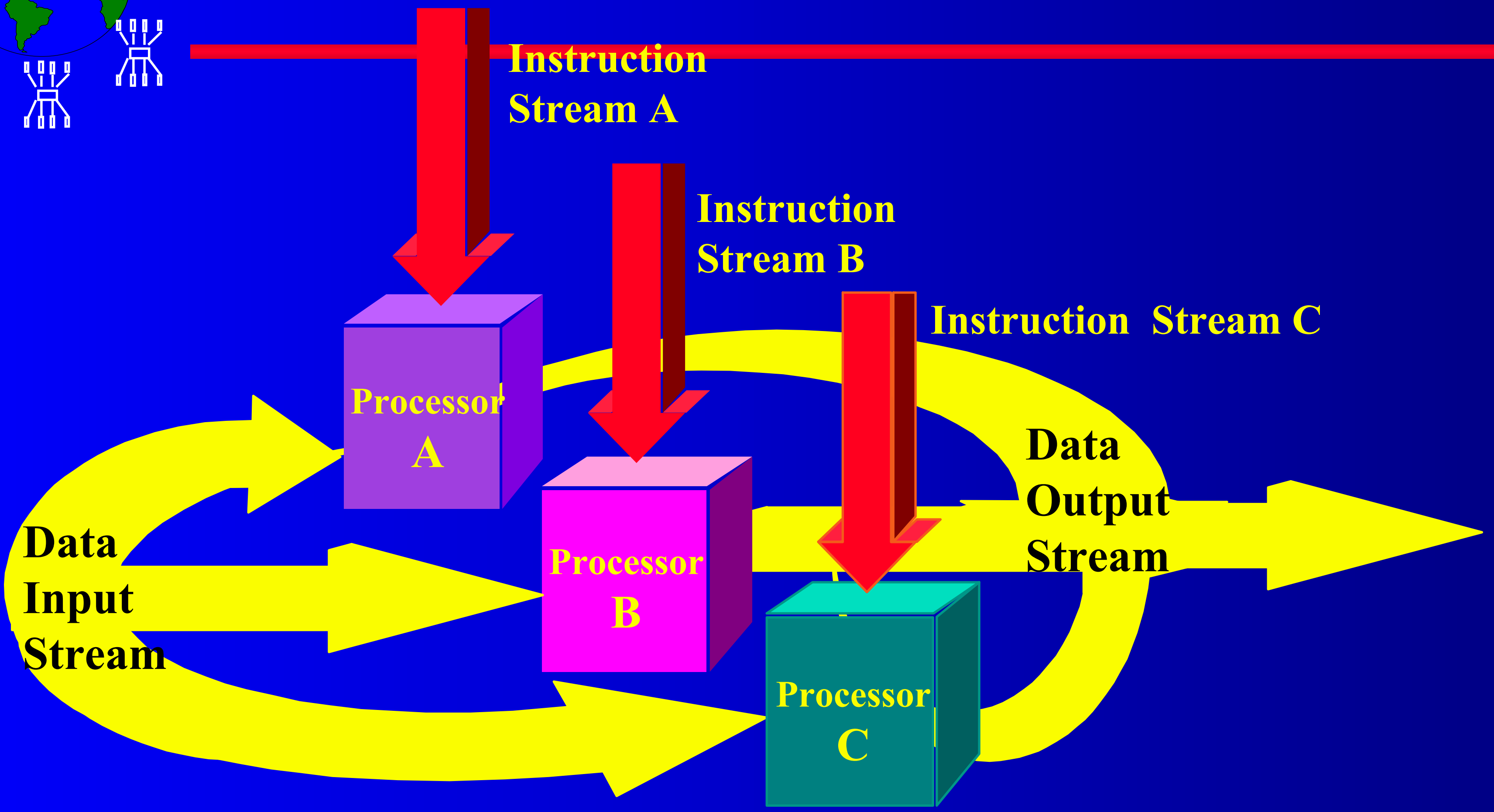
SISD : A Conventional Computer



➔ **Speed is limited by the rate at which computer can transfer information internally.**

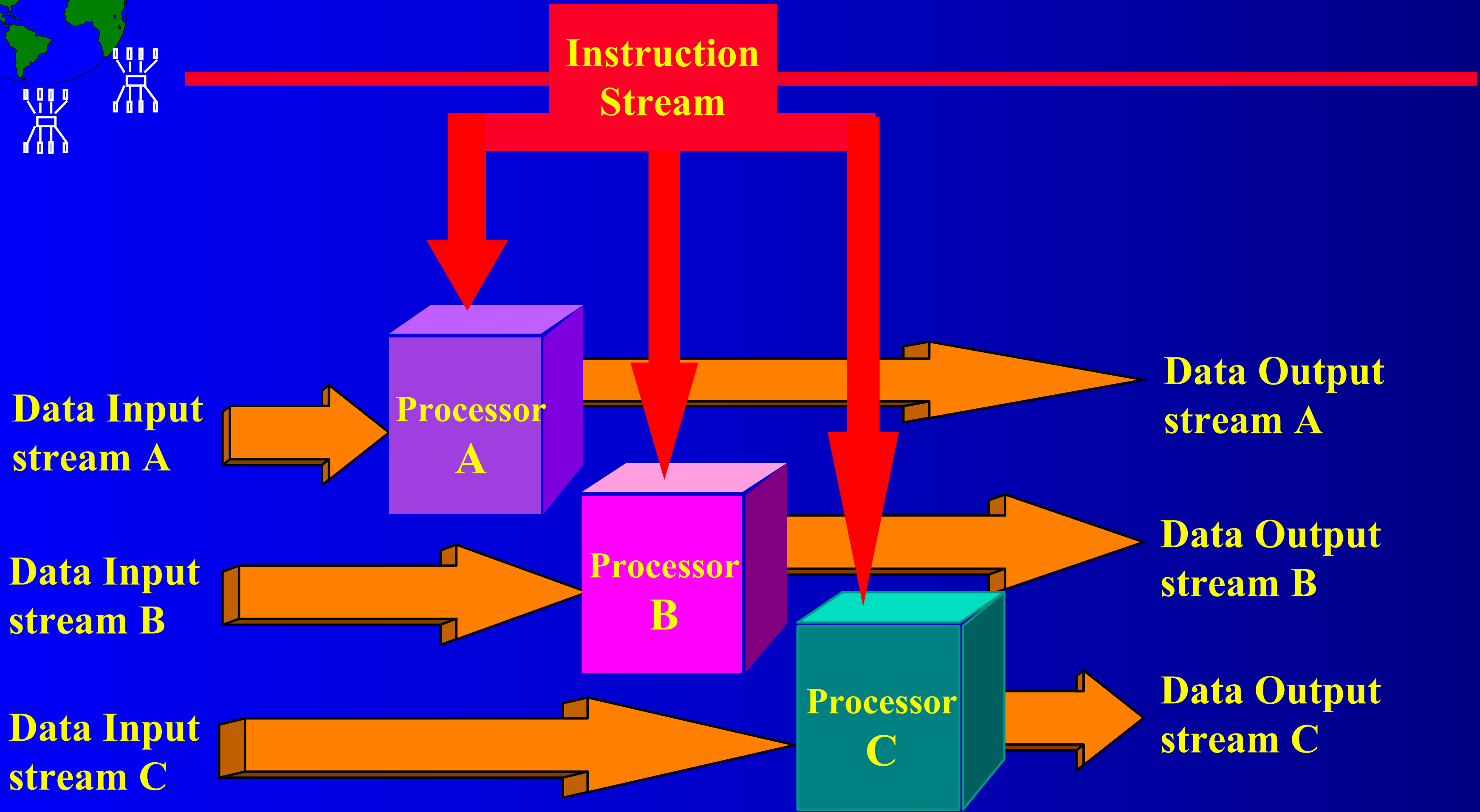
Ex:PC, Macintosh, Workstations

The MISD Architecture



➔ More of an intellectual exercise than a practical configuration. Few built, but commercially not available

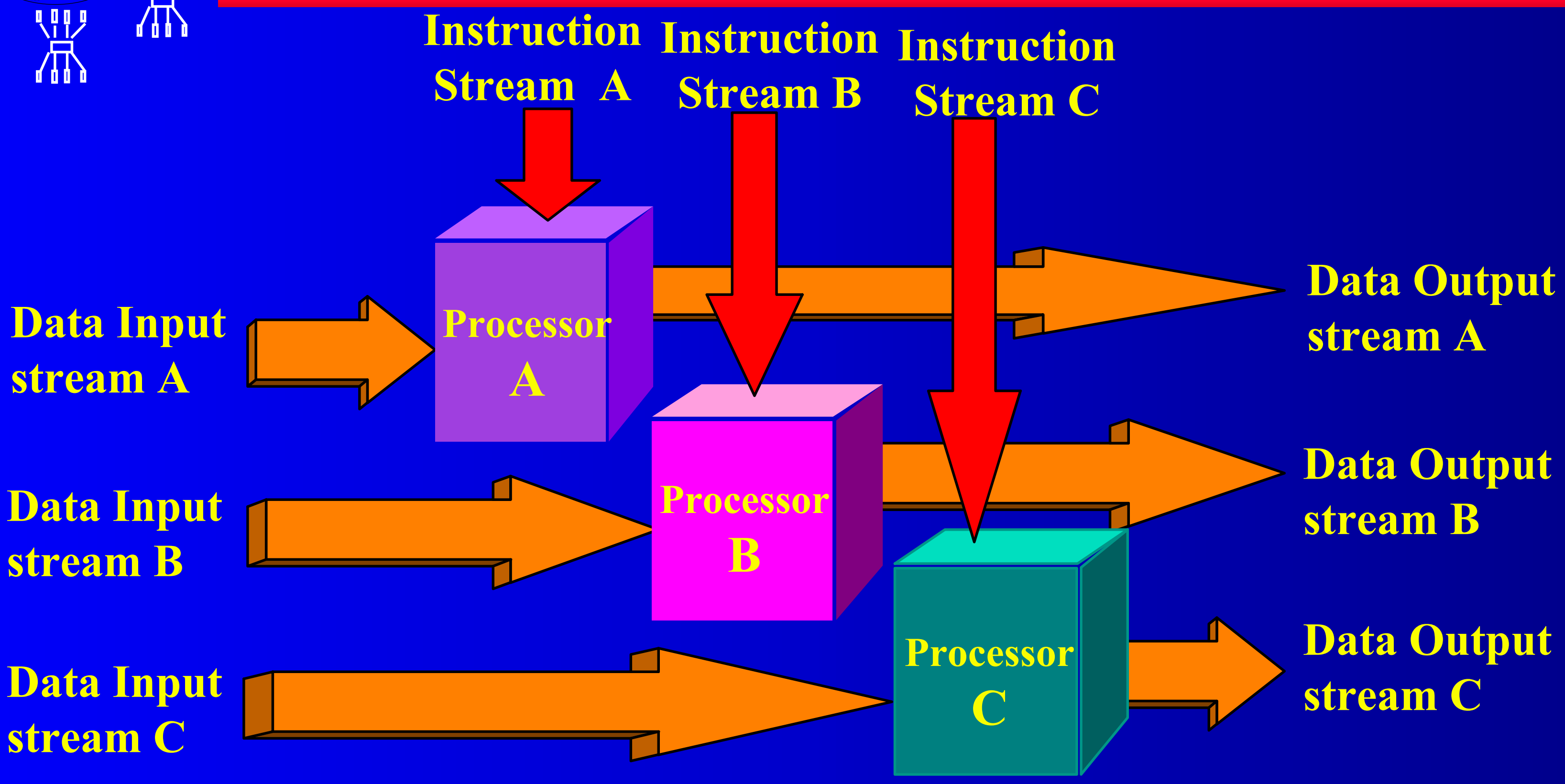
SIMD Architecture



$C_i <= A_i * B_i$

Ex: CRAY machine vector processing, Thinking machine cm*

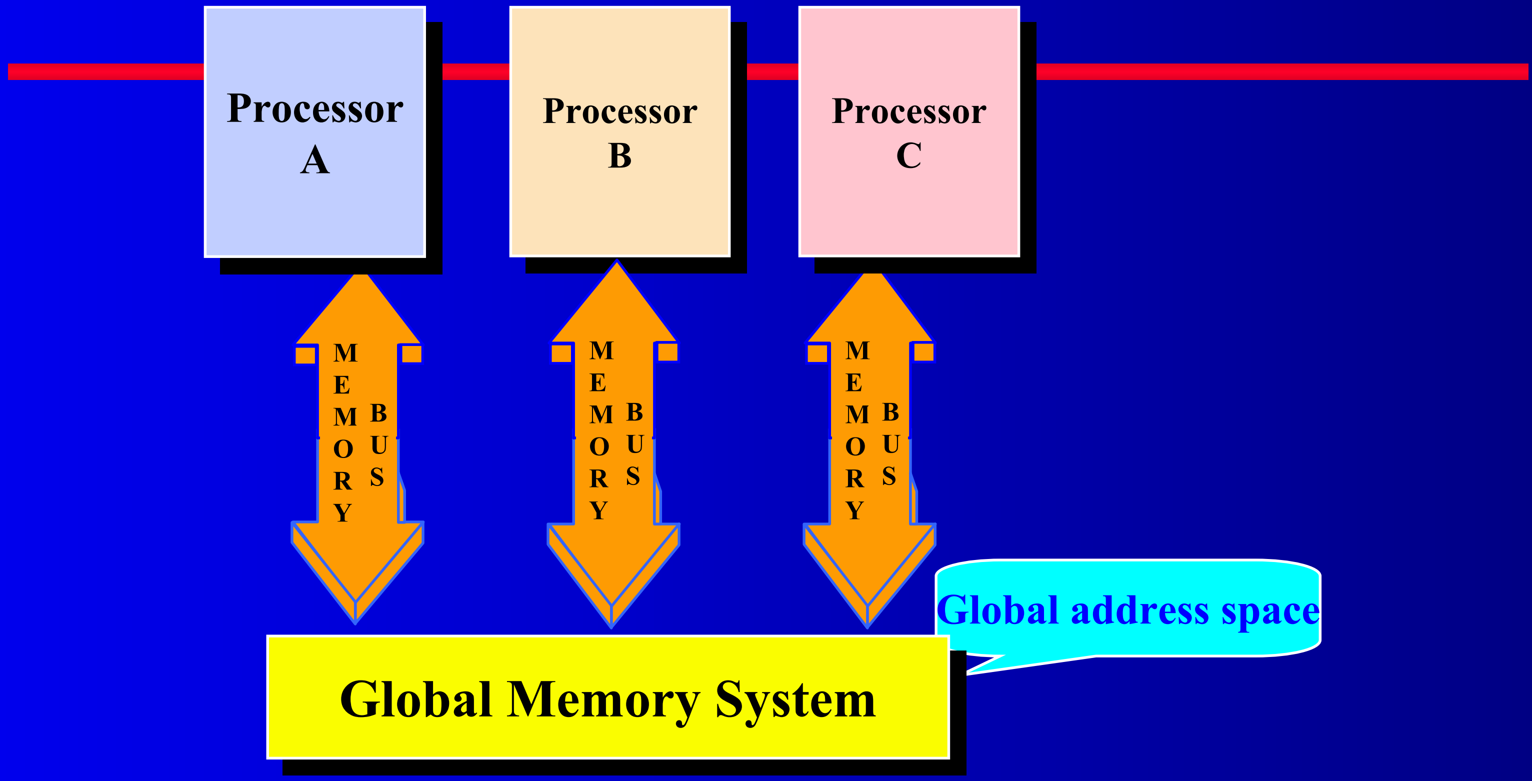
MIMD Architecture



MIMD computer works asynchronously.

- Shared memory (tightly coupled) MIMD
- Distributed memory (loosely coupled) MIMD

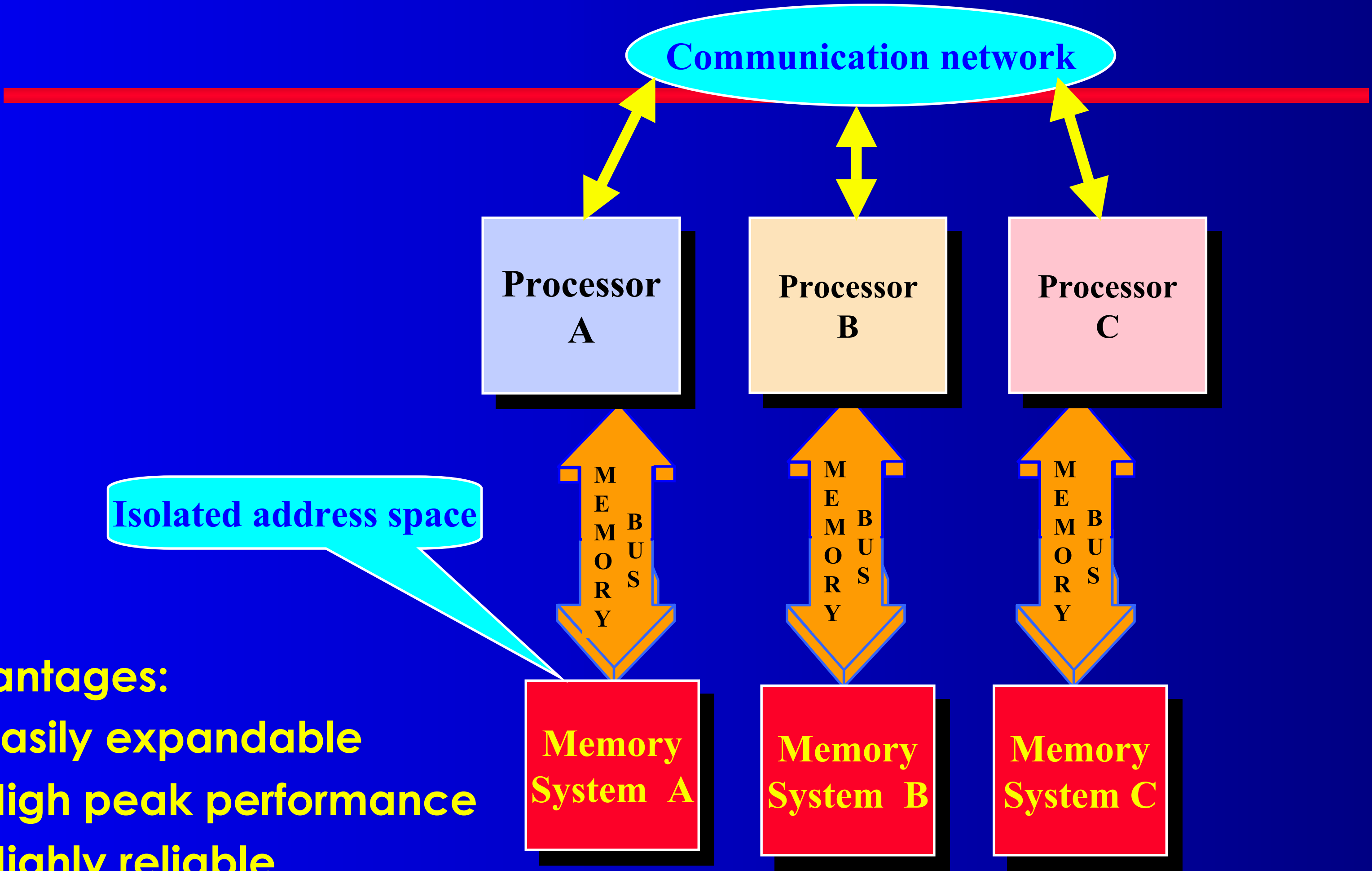
Shared Memory MIMD machine



Comm: Source PE writes data to GM & destination retrieves it

- Easy to program
- High sustained performance
- Limitation 1 : Increase of processors leads to memory contention.
- Limitation 2 : reliability & expandability. A memory component or any processor failure affects the whole system.

Distributed Memory MIMD



- Advantages:
  - ➔ Easily expandable
  - ➔ High peak performance
  - ➔ Highly reliable
- Drawbacks:
  - ➔ difficult load balancing, mapping
  - ➔ communication is more costly

Main HPC Architectures

- SISD - mainframes, workstations, PCs.
- SIMD Shared Memory - Vector machines, Cray...
- MIMD Shared Memory - Sequent, KSR, Tera, SGI, SUN.
- SIMD Distributed Memory - DAP, TMC CM-2...
- MIMD Distributed Memory - Cray T3D, Intel, Transputers, TMC CM-5, plus recent workstation clusters (IBM SP2, DEC, Sun, HP).

Main HPC Architectures

- **NOTE:** **Modern sequential machines are not purely SISD - advanced RISC processors use many concepts from**
  – vector and parallel architectures (pipelining, parallel execution of instructions, prefetching of data, etc) in order to achieve one or more arithmetic operations per clock cycle.