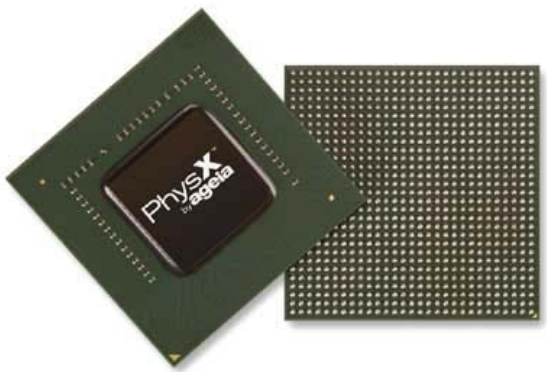
AGEIA PhysX Physics Processing Unit

EECS 573 Case Study

Joseph Lee Greathouse

March 21, 2007

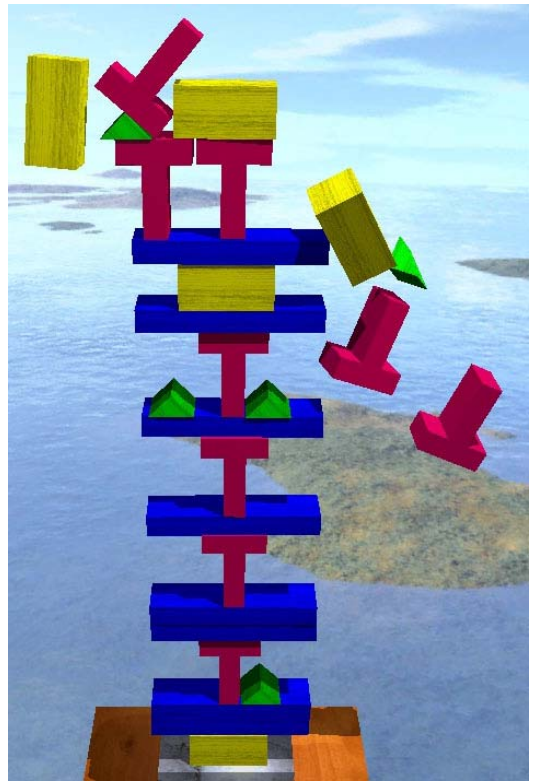


Overview Slide!!

- Physics!
  - What's that?
- Calculating Physics
  - Super-duper easy version
- Software versus Hardware physics
  - Why hardware physics is awesome
- PhysX Microarchitecture
  - Probably
- Comparison to other Devices
  - Alternate Title: PhysX is a lot like Cell & better than GPGPU

Physics in Modern Games

- Rigid Body Physics
  - Non-deformable objects
  - Newtonian physics effects
  - Most common type of game physics today
- Volumetric Fluids
  - Modeling how fluidic actions occur
  - Simulated with loads of individual particles
  - Other volumetric tasks: e.g. smoke
- Cloth
  - Deformable clothing, rugs, etc
  - Modeled by sheets of particles

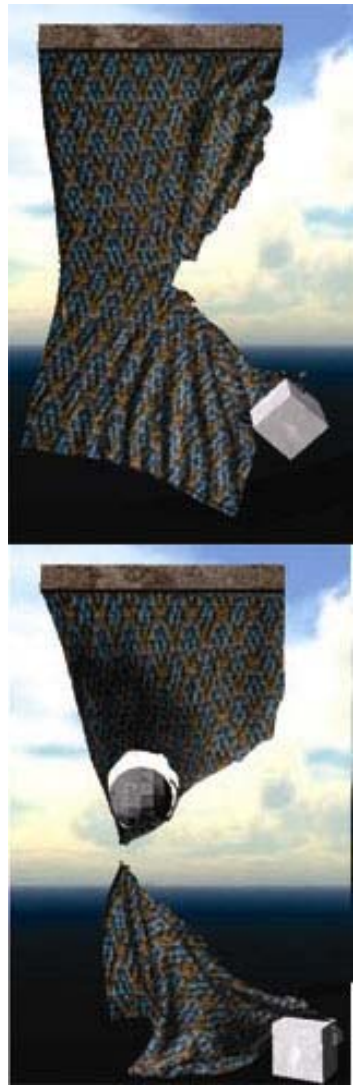


Physics in Modern Games

- Rigid Body Physics
  - Non-deformable objects
  - Newtonian physics effects
  - Most common type of game physics today
- Volumetric Fluids
  - Modeling how fluidic actions occur
  - Simulated with loads of individual particles
  - Other volumetric tasks: e.g. smoke
- Cloth
  - Deformable clothing, rugs, etc
  - Modeled by sheets of particles

Physics in Modern Games

- Rigid Body Physics
  - Non-deformable objects
  - Newtonian physics effects
  - Most common type of game physics today
- Volumetric Fluids
  - Modeling how fluidic actions occur
  - Simulated with loads of individual particles
  - Other volumetric tasks: e.g. smoke
- Cloth
  - Deformable clothing, rugs, etc
  - Modeled by sheets of particles

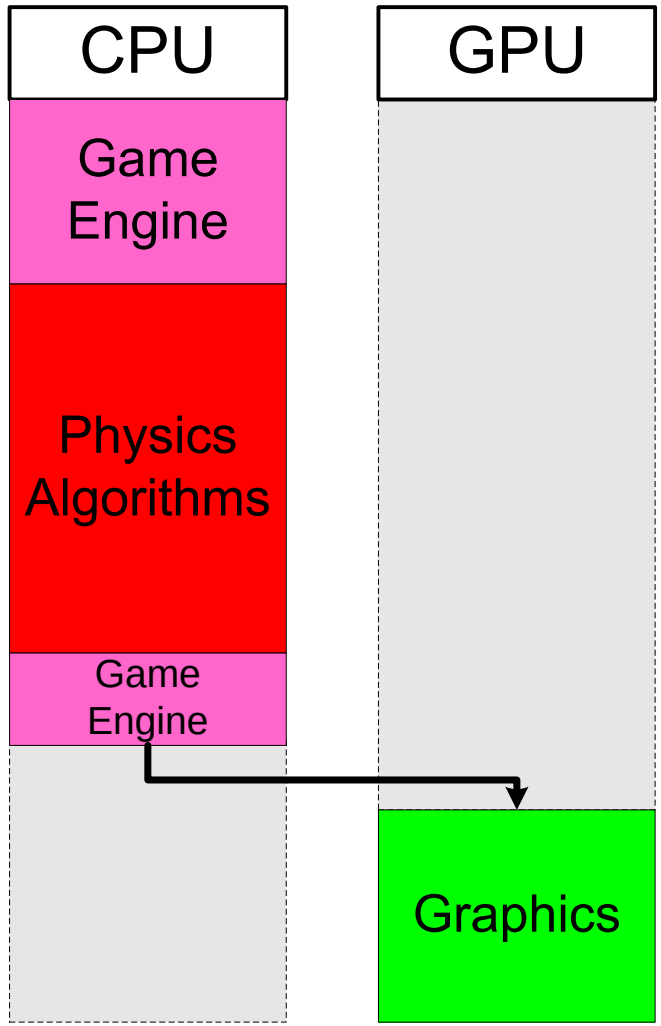


How is this stuff calculated??

- Lots and lots of matrix math
- LCP: Linear Complementarity Problem
  - It turns out that these are very easy to do in parallel
  - NP-Complete
- Requires a lot of communication between parallel processes
- Huge floating-point requirement

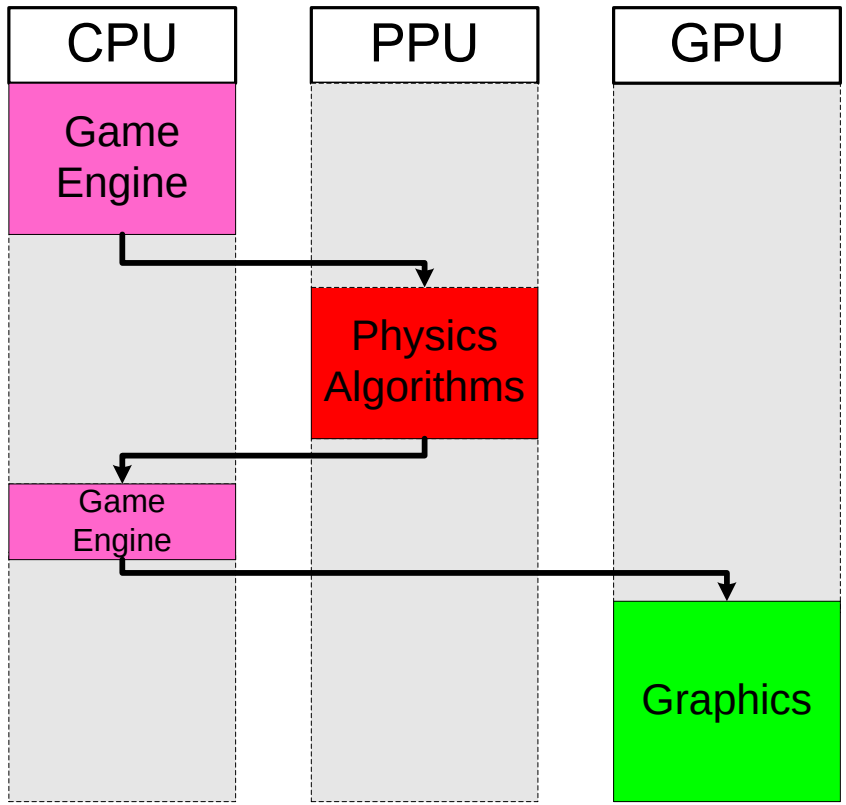
Physics on Modern Computers

- Processor runs game engine
- Game engine calls Physics API (e.g. Havok)
- Processor runs physics algorithms
  - This can take a really, really long time
- Processor returns result to game engine
- Processor sends data to graphics system



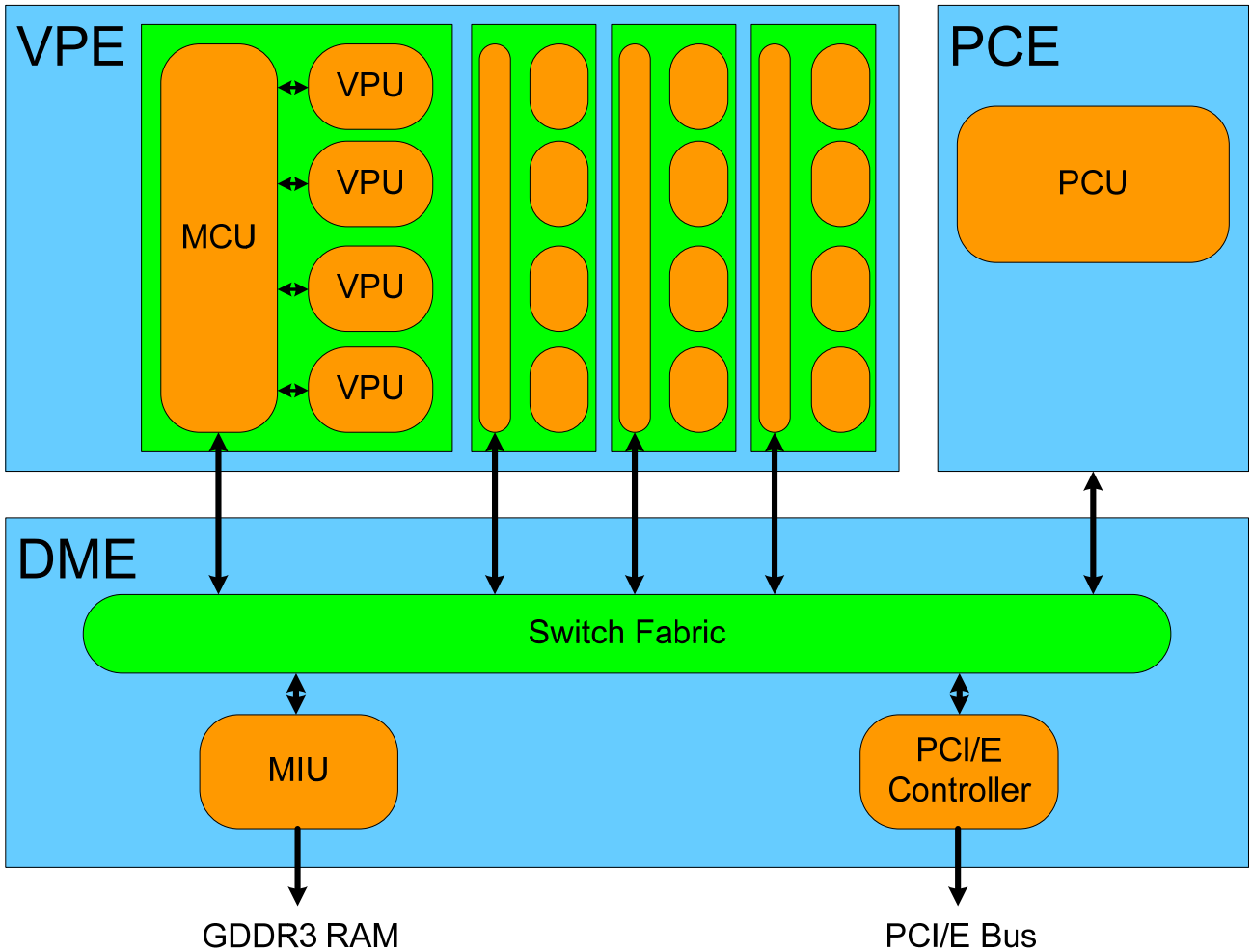
Physics with a Physics Accelerator

- Processor runs game engine
- Processor calls Physics API (e.g. PhysX/NovodeX)
- PPU runs physics algorithms
  - Faster at each physics calculation
  - Processor free to do other stuff now
- PPU returns result to game
- Processor sends data to graphics system



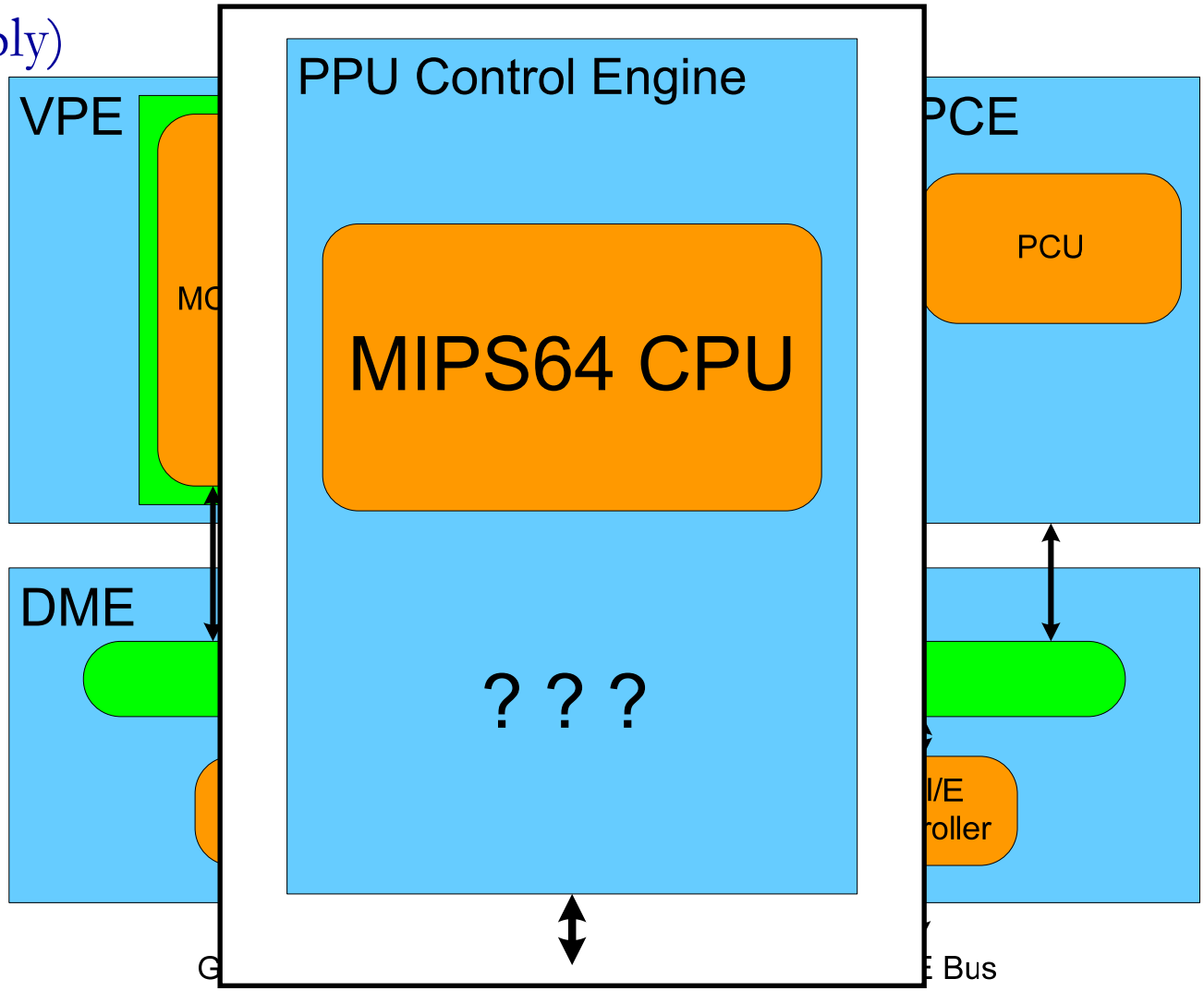
PhysX Microarchitecture

(Probably)

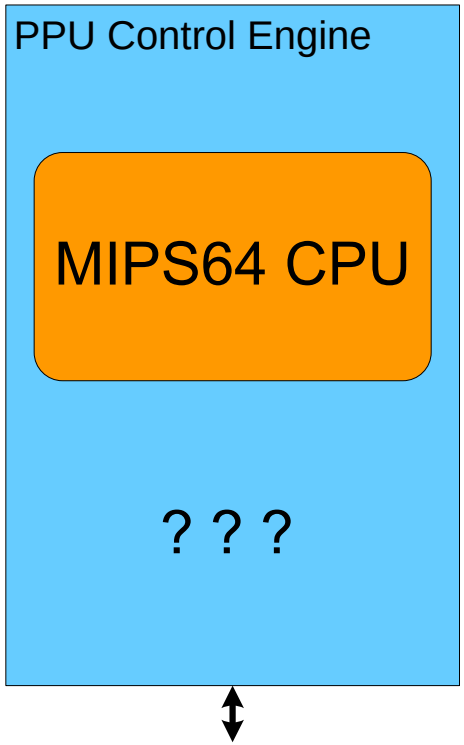


PhysX Microarchitecture

(Probably)



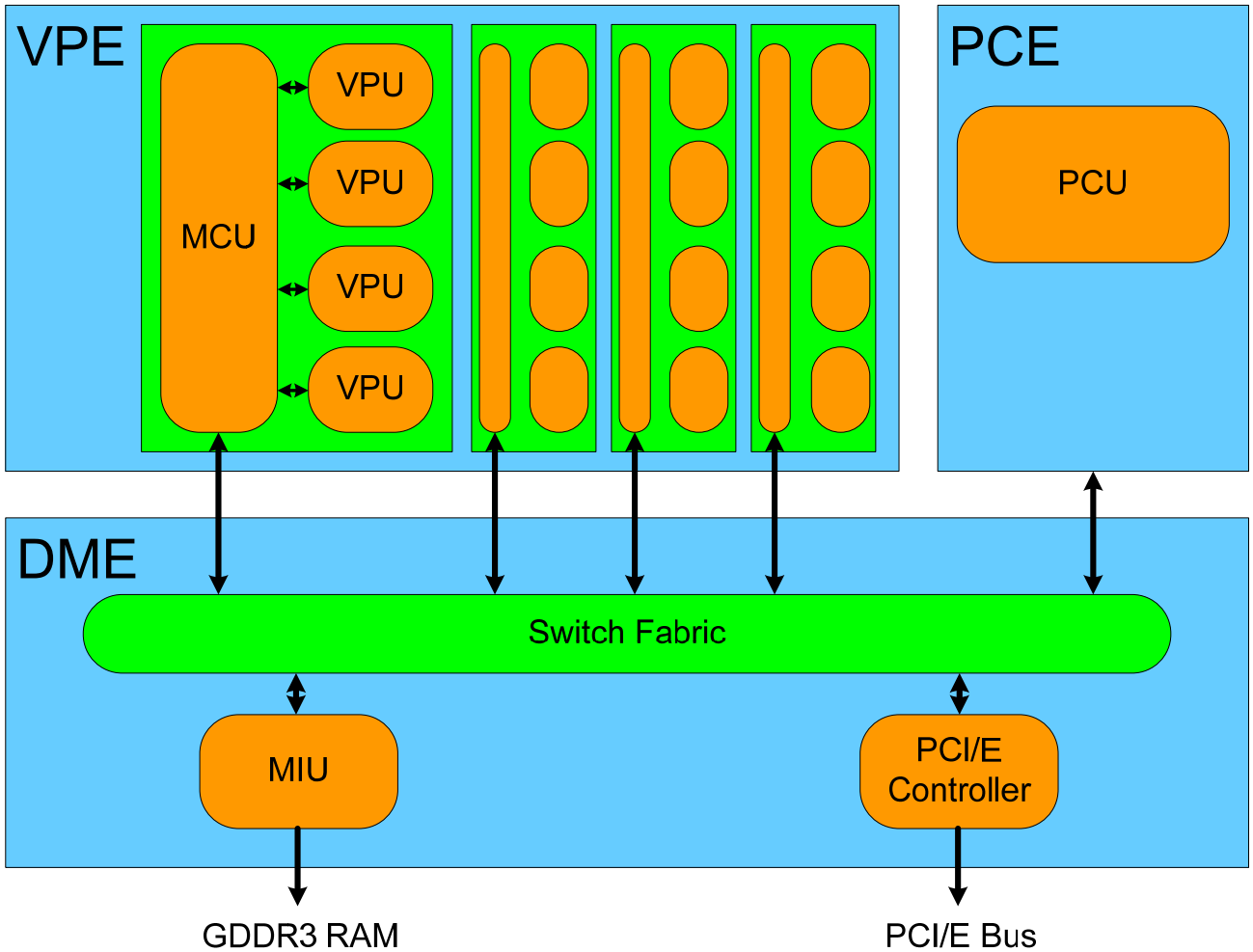
PCE: The PPU Control Engine



- PCU (probably) a MIPS64 5Kf RISC CPU
  - PCU means PPU Control Unit
- Controls physics “programs”
- Mostly scalar, integer, and control data
- Assigns tasks to DME/VPE
- Controls communication to PC and driver
- No other info about PCE architecture
- One option:
  - Maybe connects to VPE directly
  - Might put data into place for the MCU

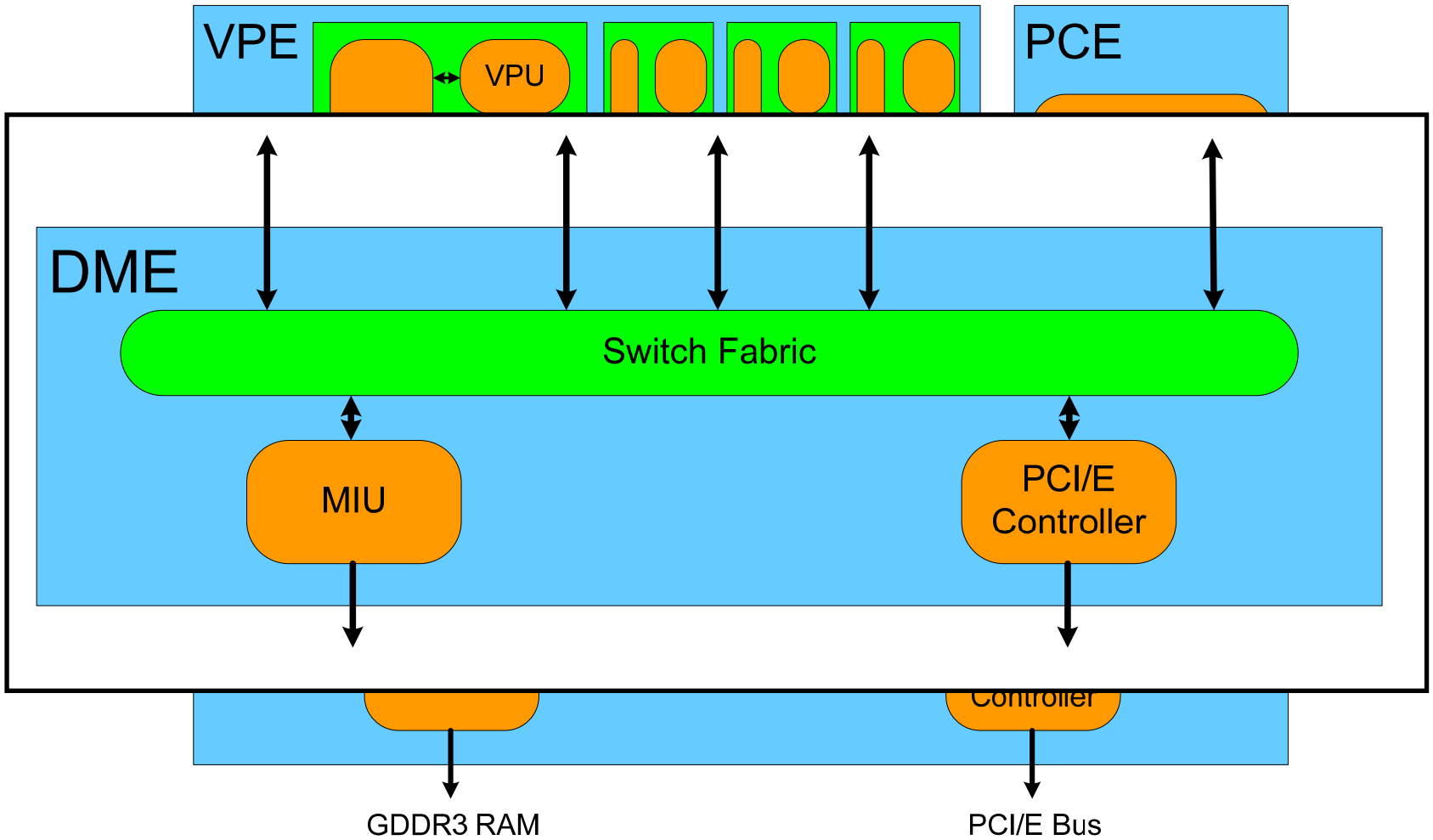
PhysX Microarchitecture

(Probably)

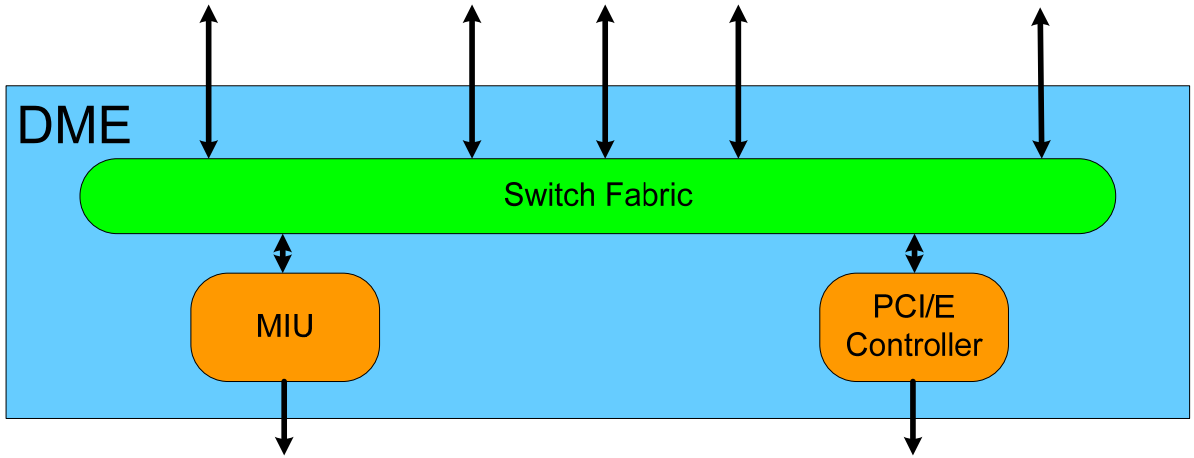


PhysX Microarchitecture

(Probably)



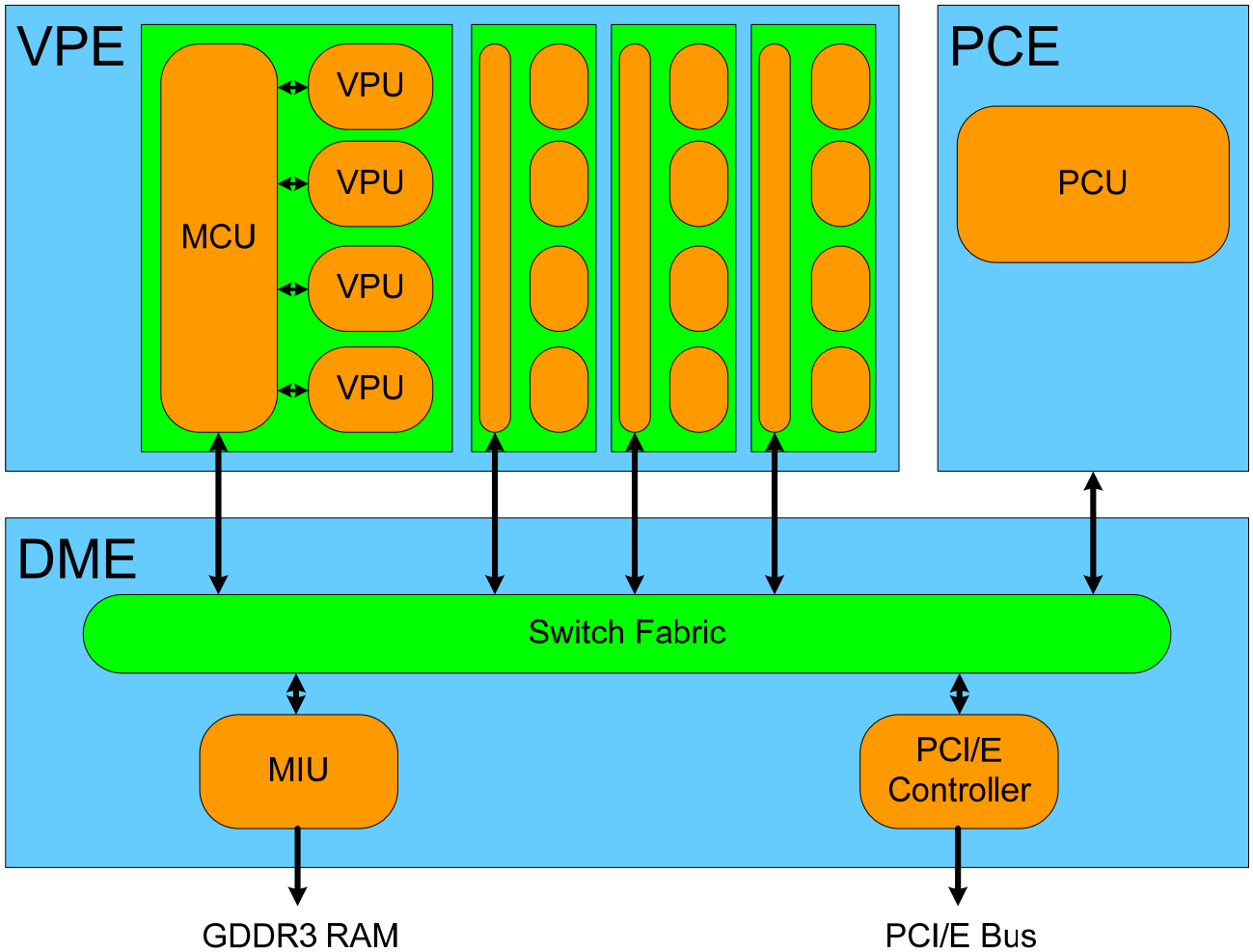
DME: Data Movement Engine



- MIU Connects to Board's 128MB GDDR3 RAM
- Two (or more) options:
  - Dumb switching fabric to connect everything together
  - May be partially programmable to optimize interconnections
- PCI & PCI-E Currently Supported
  - Patents mention using Firewire, USB, etc.

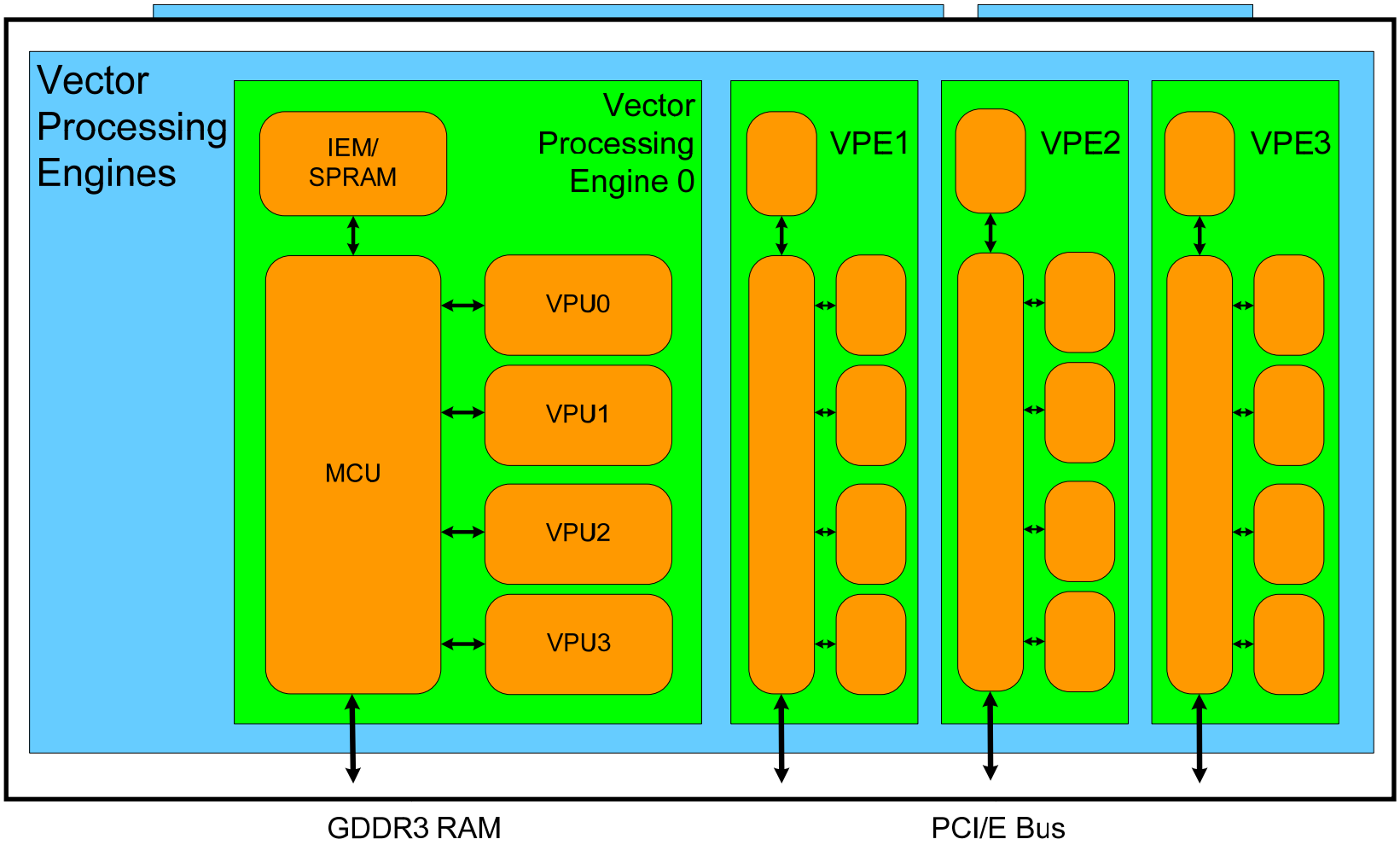
PhysX Microarchitecture

(Probably)

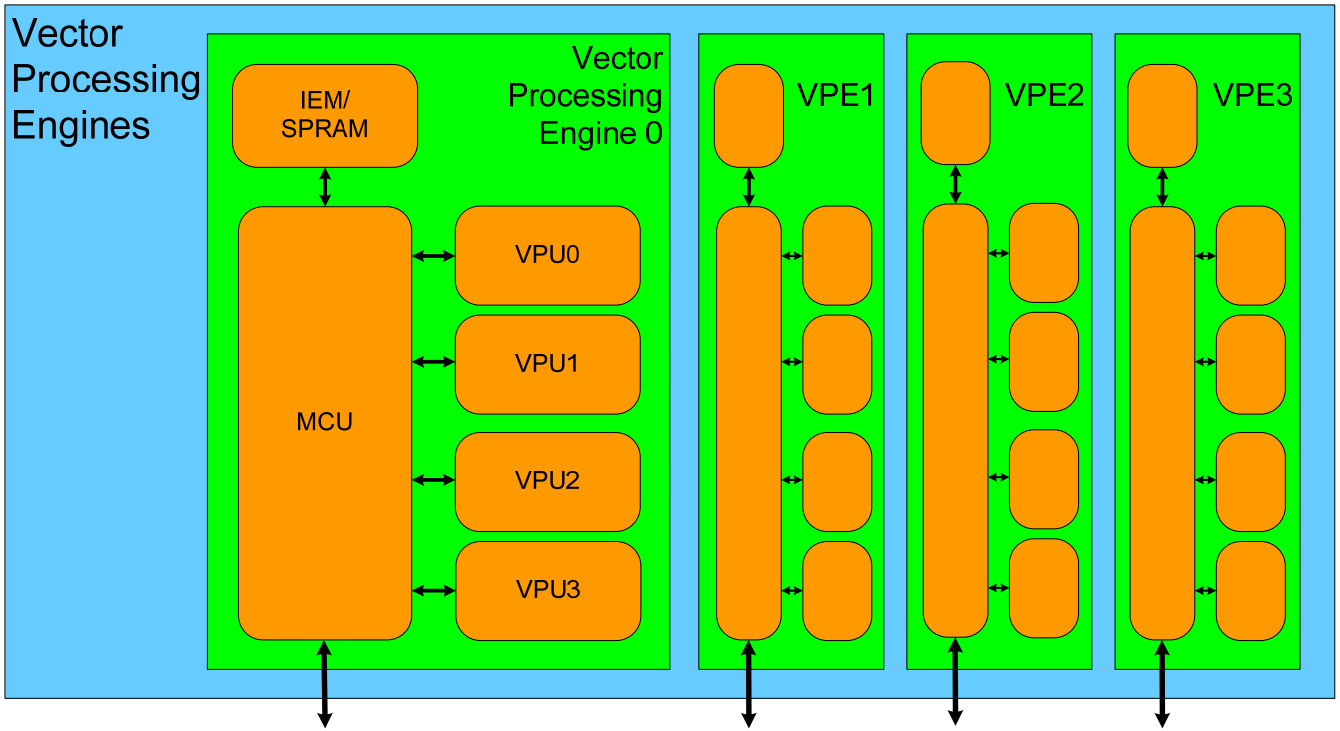


PhysX Microarchitecture

(Probably)

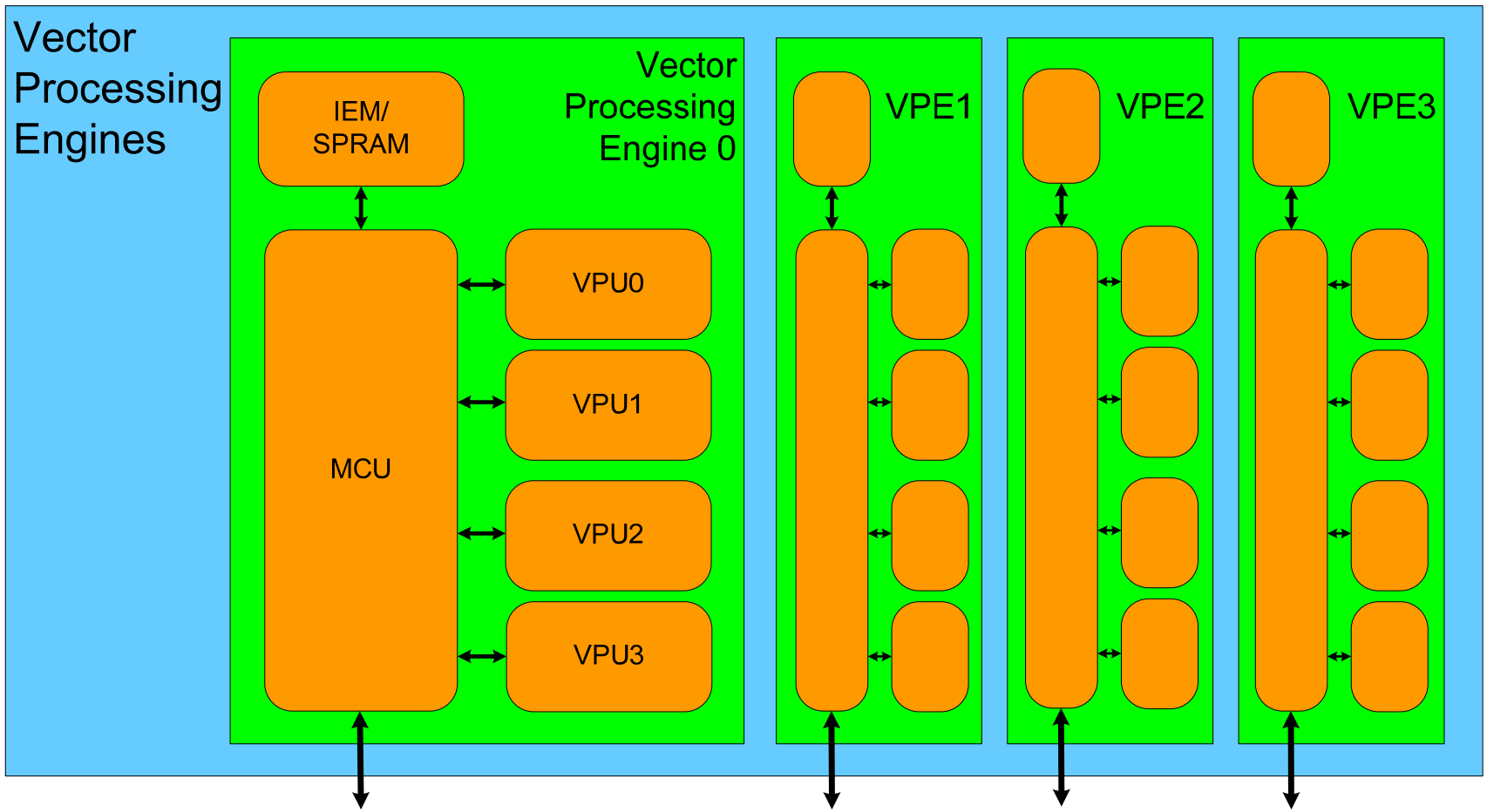


VPE: Vector Processing Engines

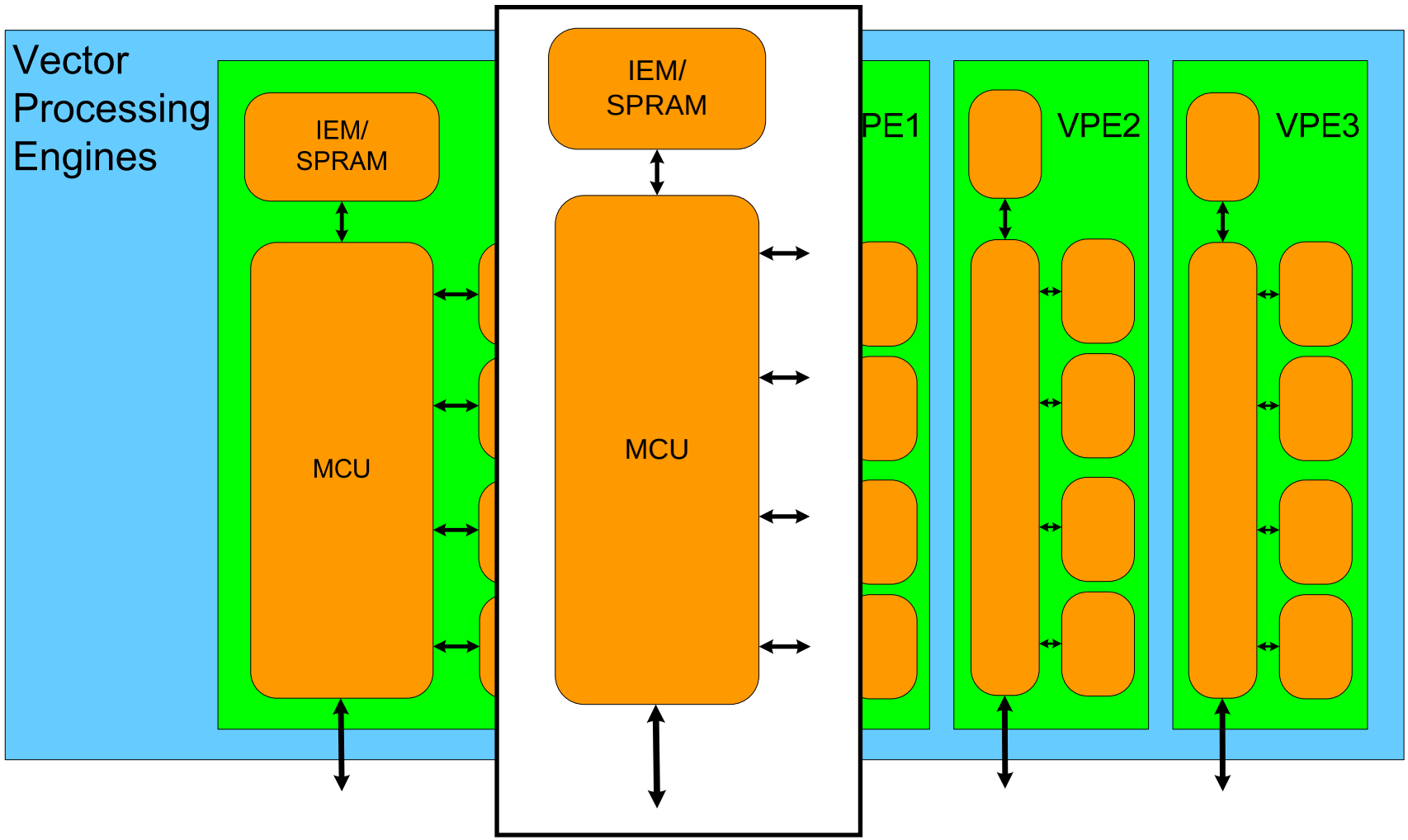


- Probably four VPE in current PhysX implementation
- VPEs communicate with each other through DME

VPE: Vector Processing Engines

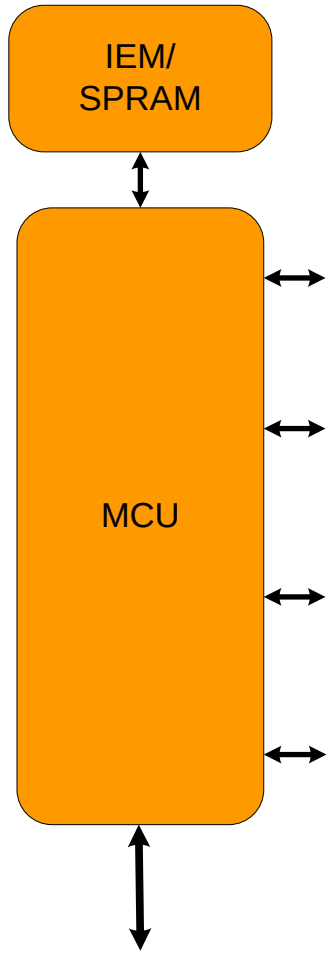


VPE: Vector Processing Engines

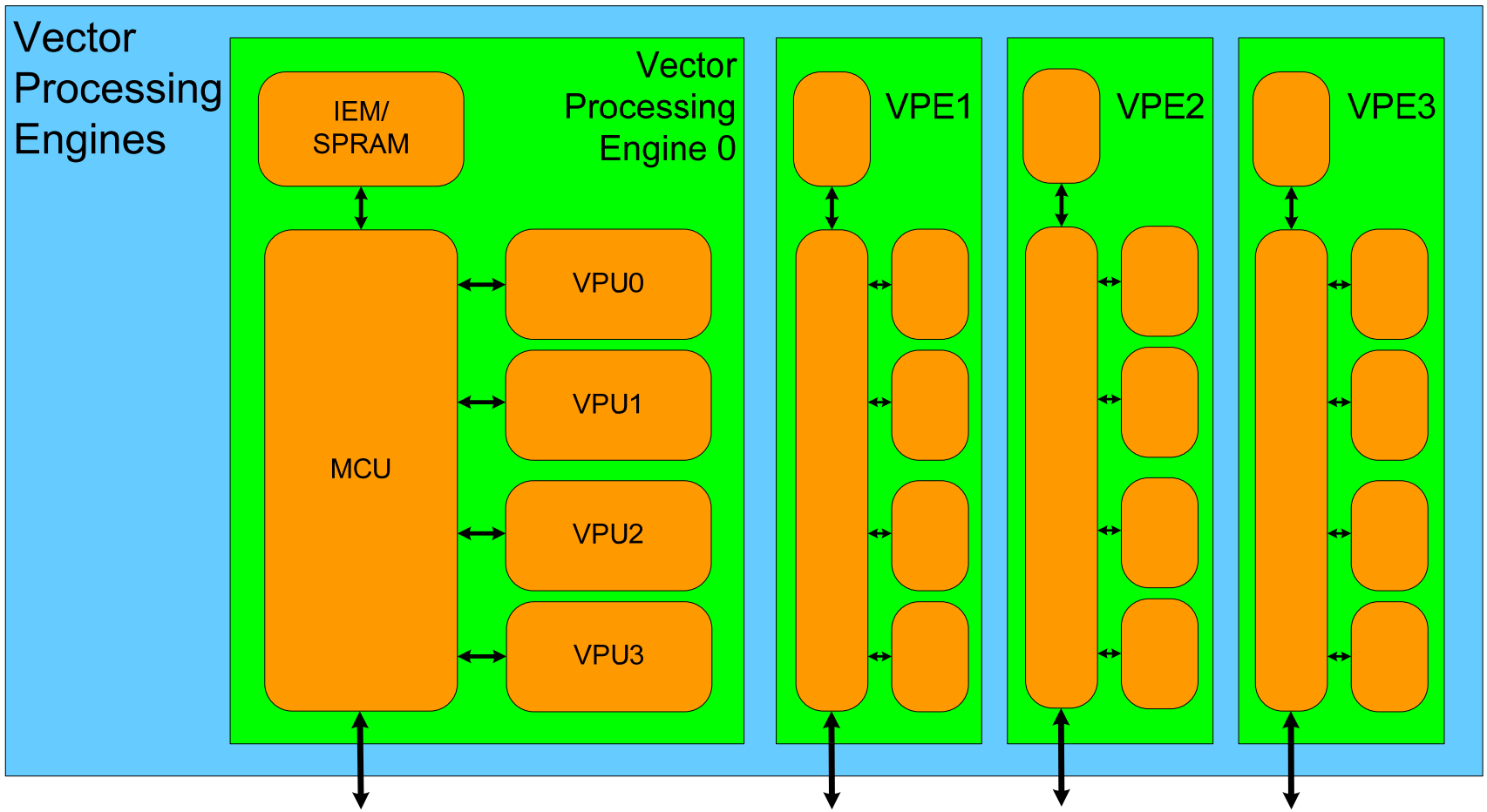


MCU: Memory Control Unit

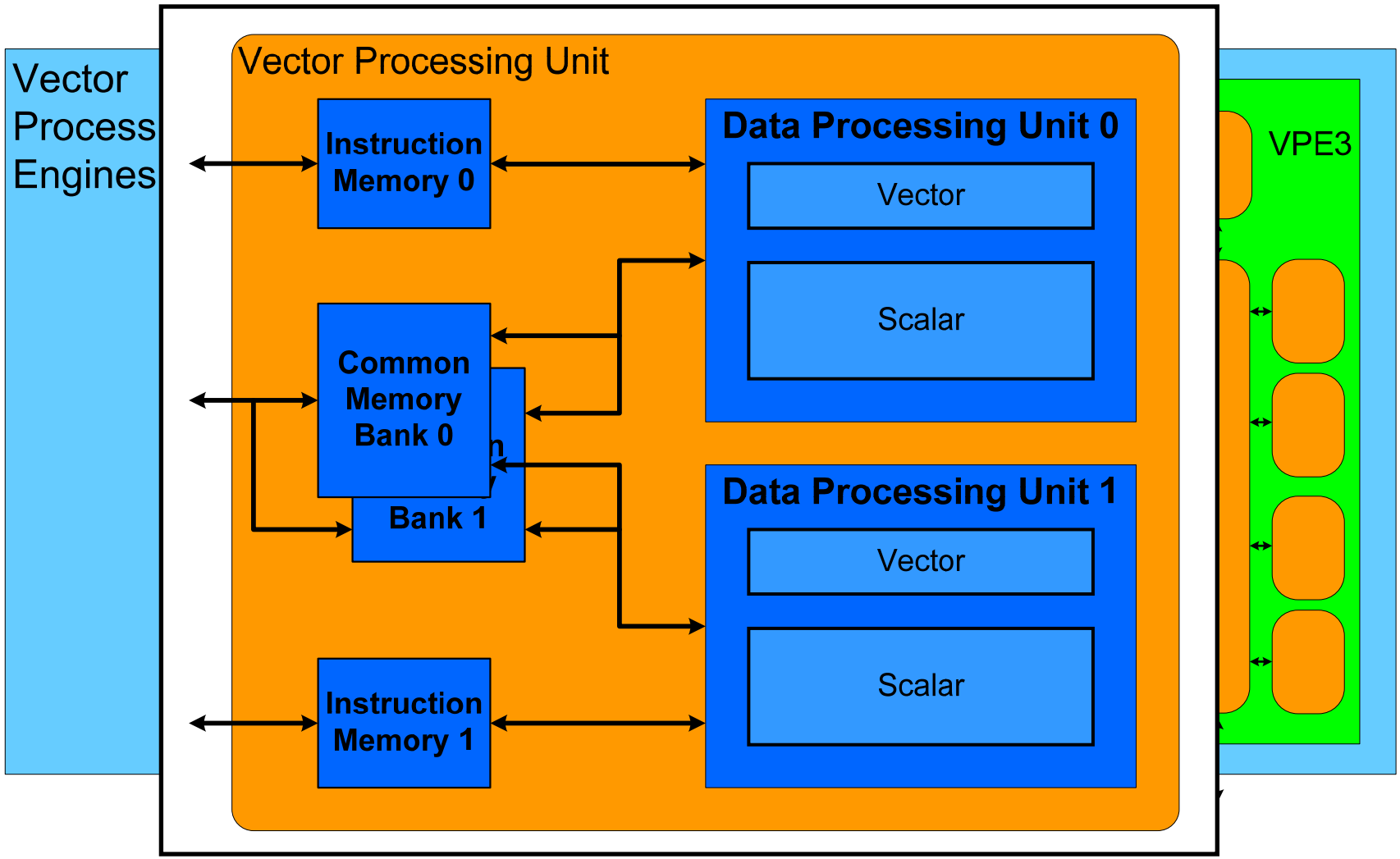
- Controls data transfer from PPU RAM to VPU
  - Received program orders from PCE
  - Can communicate with other MCUs as well (through DME)
- Can store data in an Inter-Engine Memory or Scratchpad RAM, etc.
  - Can also move data to/between individual VPUs
- IEM can be used as storage space to move data between VPUs



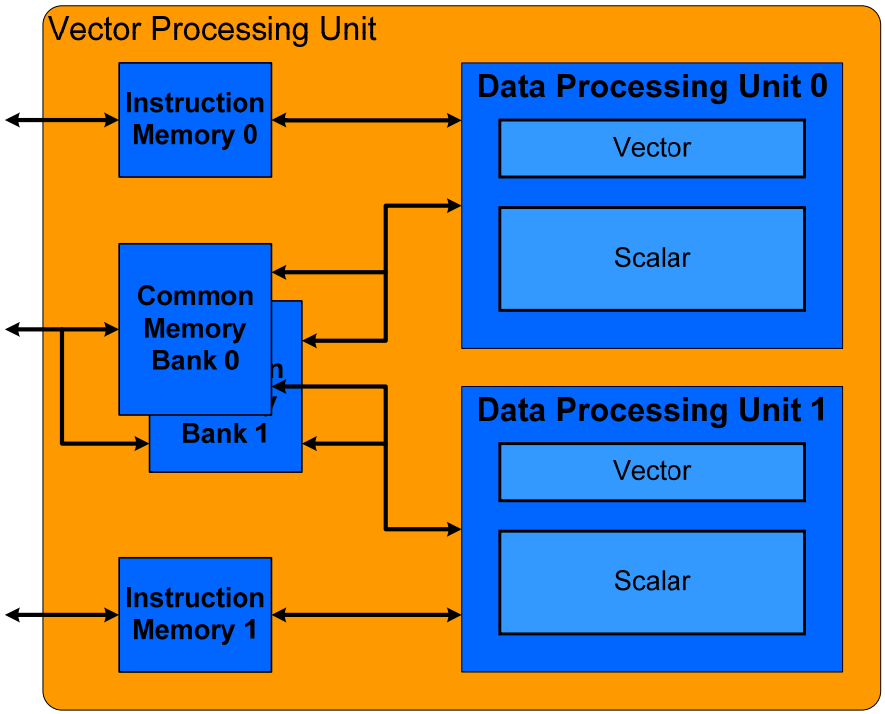
VPE: Vector Processing Engines



VPE: Vector Processing Engines

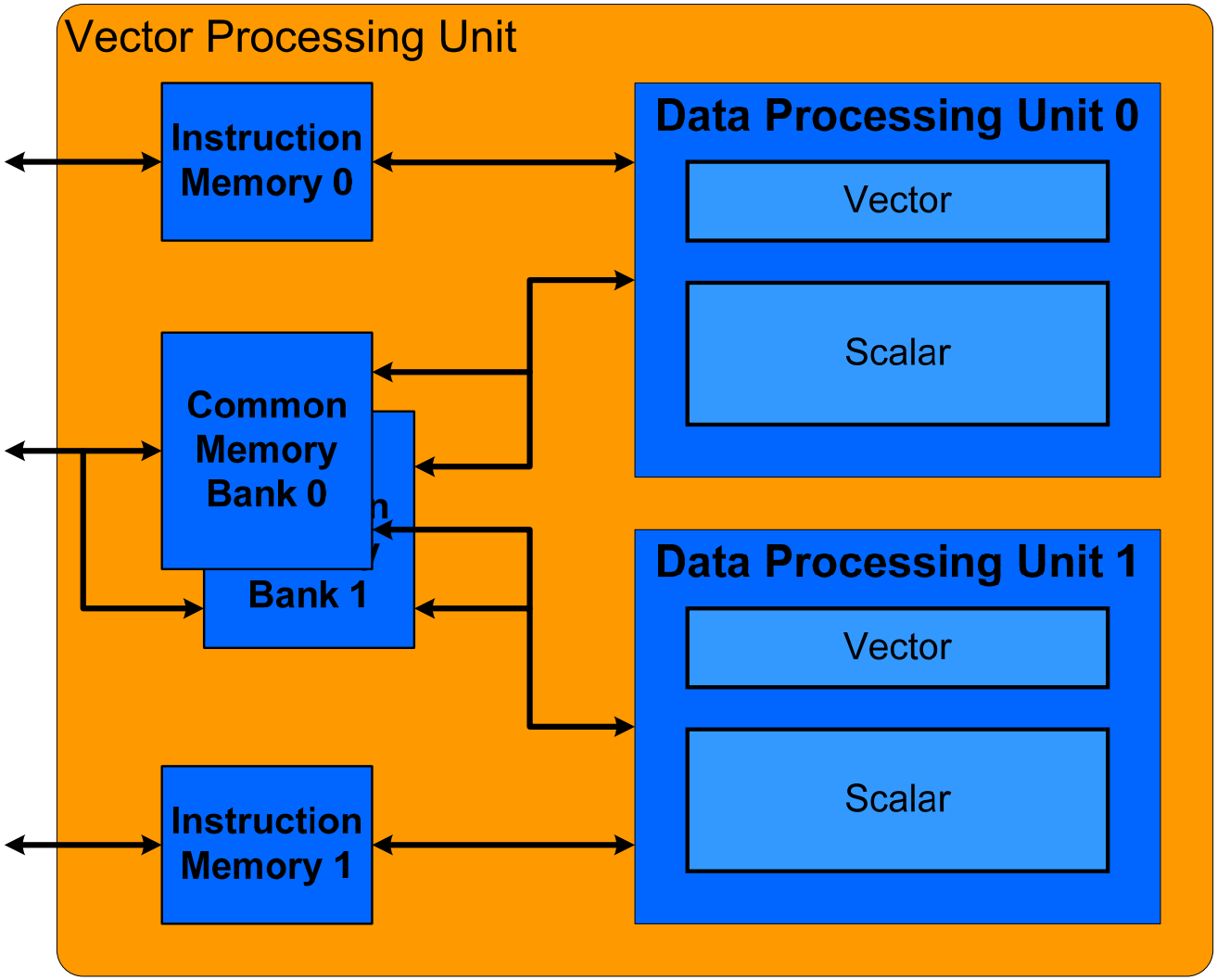


VPU: Vector Processing Unit

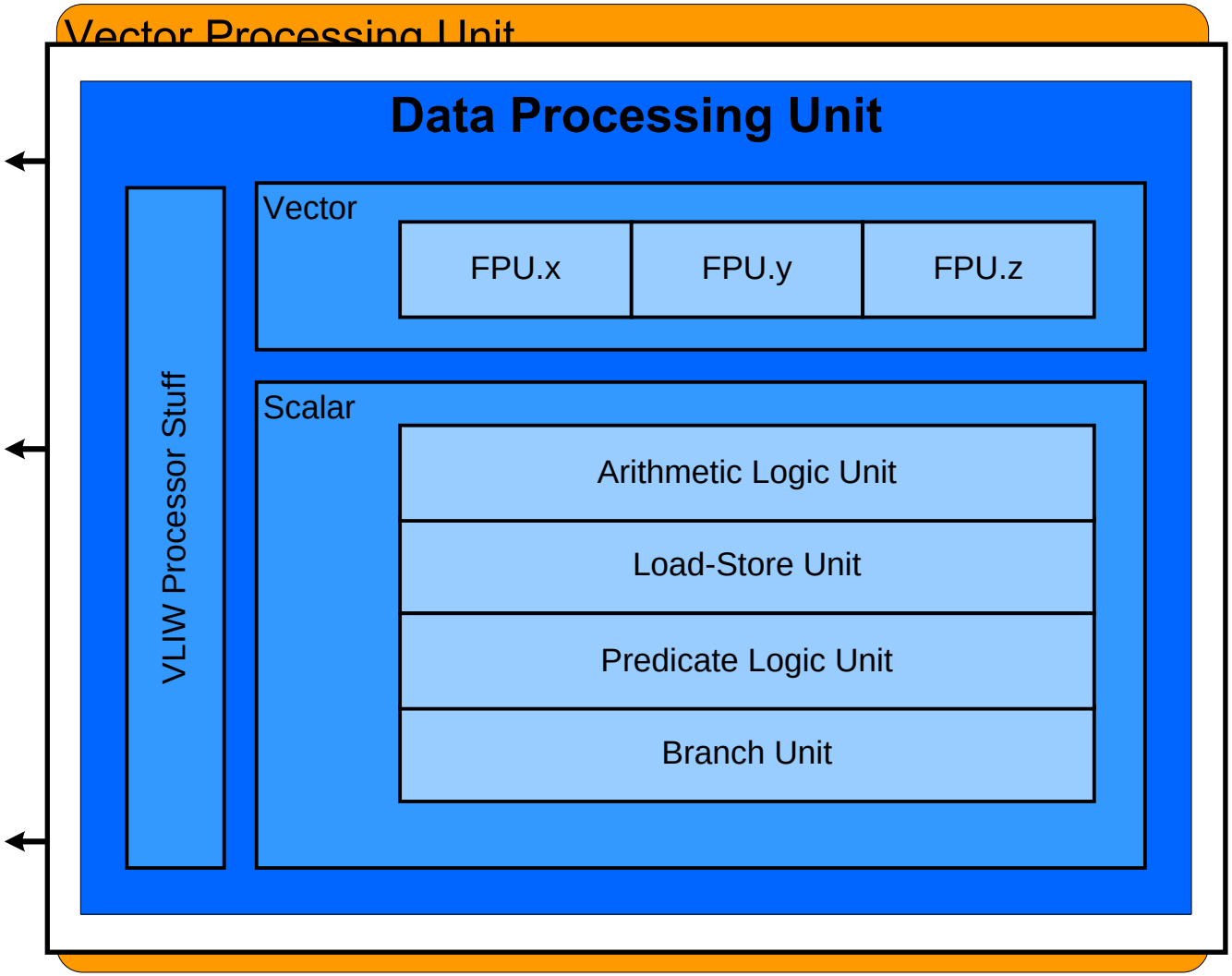


- Does the actual physics calculations
- Units communicate through shared memory & registers
  - Multithreaded
- Shared memory is banked:
  - MCU can access one block while units utilize the other, etc
- Shared memory might be 4Kbyte/bank

VPU: Vector Processing Unit

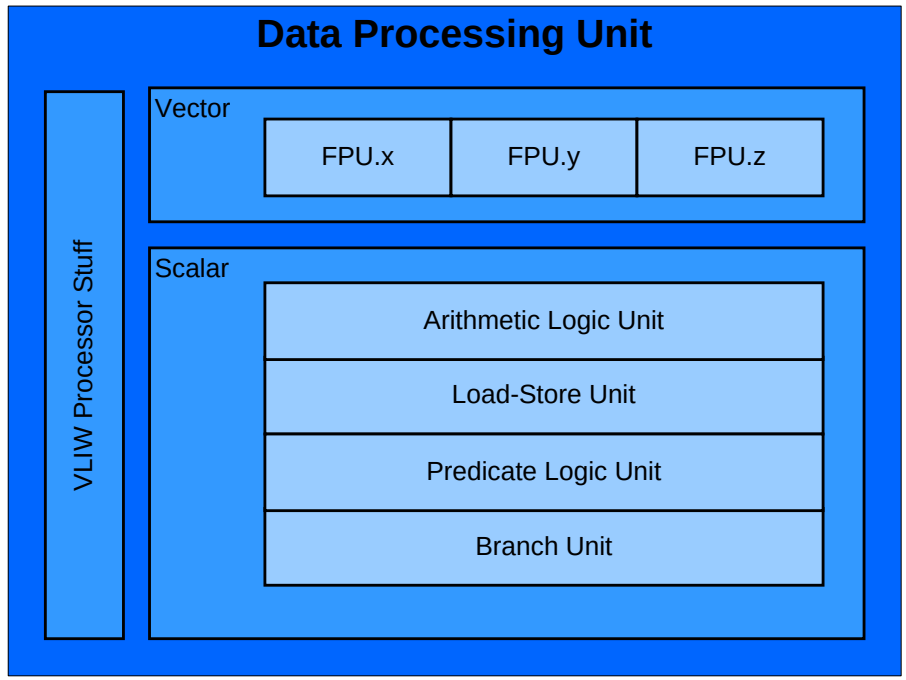


VPU: Vector Processing Unit



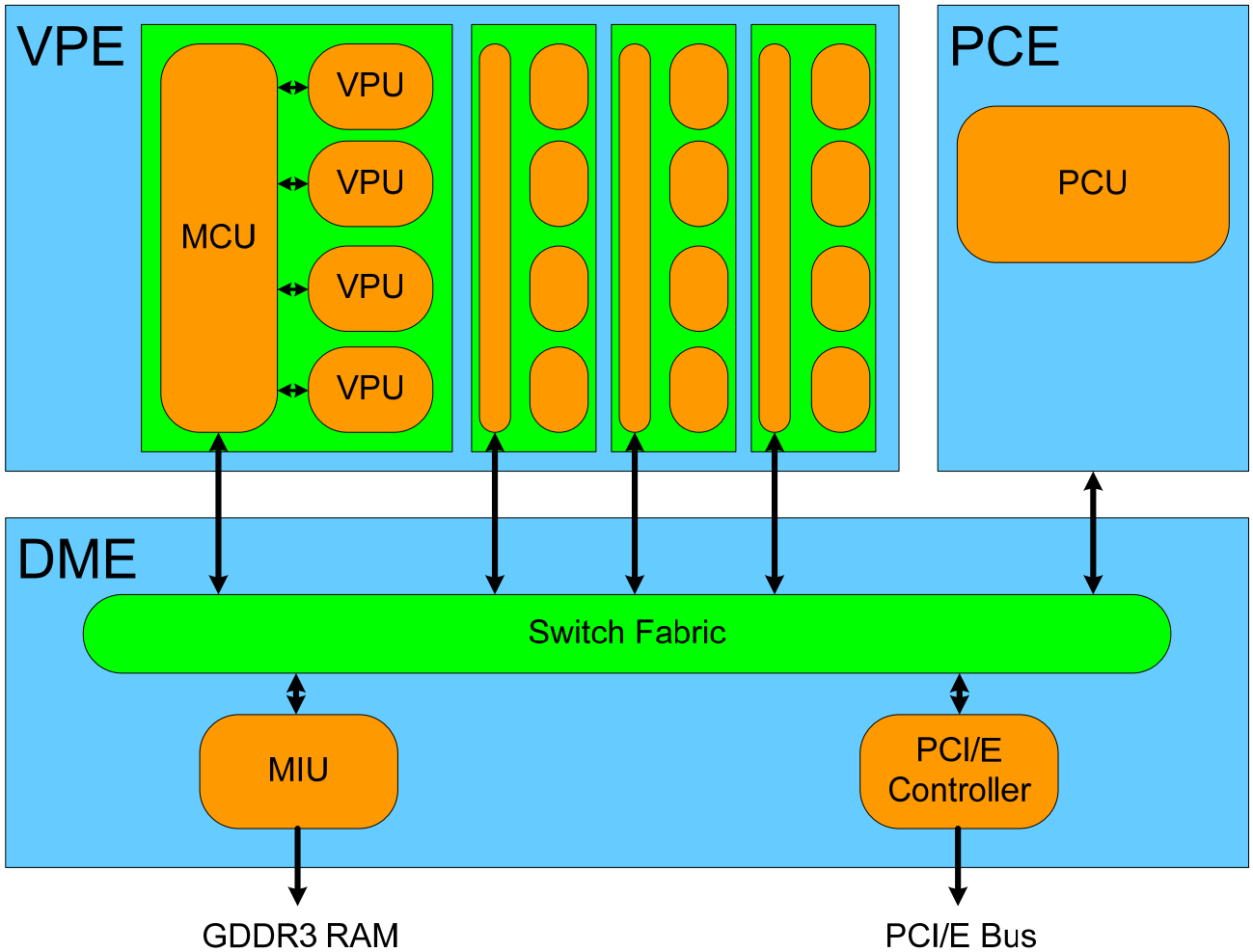
Data Processing Unit

- Specialized VLIW Processor
- Three FPU Calculations at once in vector processor
  - e.g. FMADD, FMSUB, FDOT, etc.
- Integer, load/store, branching done in scalar form
- MCU puts program for each data processing unit in instruction memory bank



PhysX Microarchitecture

(One last time)



Some interesting PhysX Facts

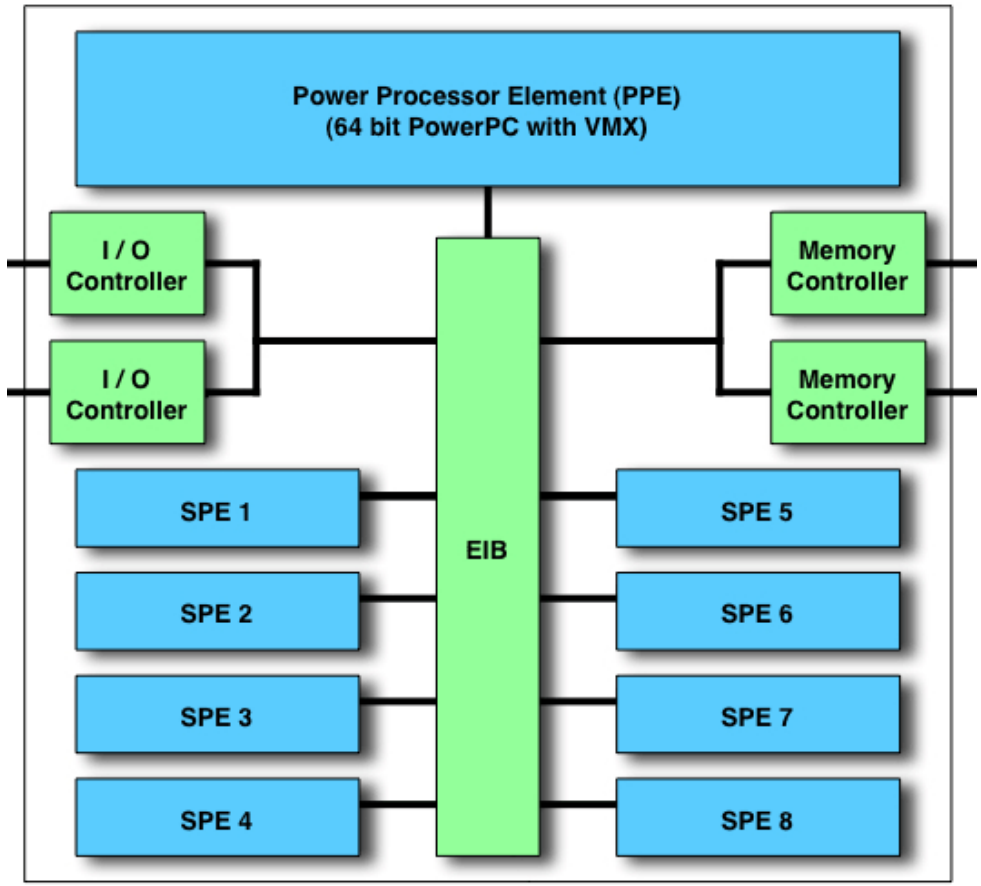
- No traditional cache
  - Physics calculations don't display good locality
  - Memory is controlled explicitly by MCU and PPE
- TSMC manufactured: AGEIA is fabless
  - 125 million transistors
  - 182 mm^2
  - 28W (for card?) / 20W (for chip?)
- Up to 96 floating point ops (multiply-add, multiply-sub, etc) per clock cycle
  - 4 VPE x 4 VPU x 2 DPU x 3 FP ops = 96 ops

Real Performance Figures

- Hard to come by
  - Not a particularly large number of games support PhysX yet
- Some games slow down
  - When PhysX is plugged in, particle count skyrockets
  - You need to graphically render all those neat particles now!
- No industry-standard Physics benchmarks
  - Makes it hard for review sites to press button and get charts

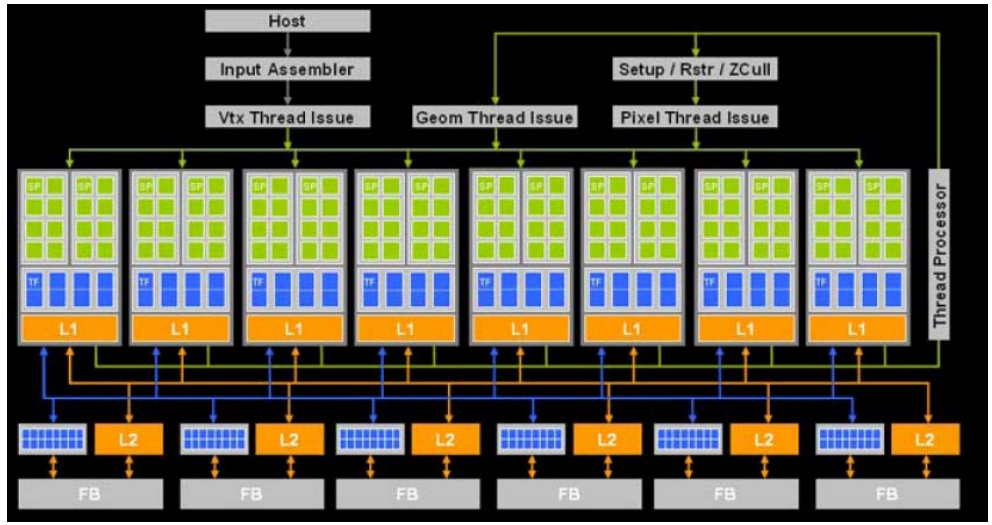
PhysX versus Cell

- PhysX looks a lot like the Cell processor!
  - Central RISC Processor
  - Vector processing elements
  - No classical caches
- It turns out that Cell is pretty good at physics too.
  - PS3 Dev kit includes PhysX SDK at no cost.
- Memory subsystem, says AGEIA, is limited, though.
  - EIB is a ring network.
  - SPE can't talk freely.



PhysX versus GeForce8

- Large number of parallel stream processors
- NVIDIA hopes to use this for GPGPU – Including physics
- AGEIA engaged in PR war over this stuff.



- Primary argument: Memory System
  - Extremely large LCP require moving a lot of data between processors
  - Memory arrangement in GeForce8 can't effectively move this data
- Might be good enough for the majority of users, though!

Questions? Discussion?

- Future Features?
  - AGEIA is only saying “smaller process: more transistors” right now.
- How do you measure performance?
  - Harder to quantify than framerate and resolution.
- Will it survive?
  - Will GPGPU or highly-multicore CPU kill it?
  - Will it end up integrated into some other product?
- How can you take this online?
  - My physics is different from your physics... Whose world is right?