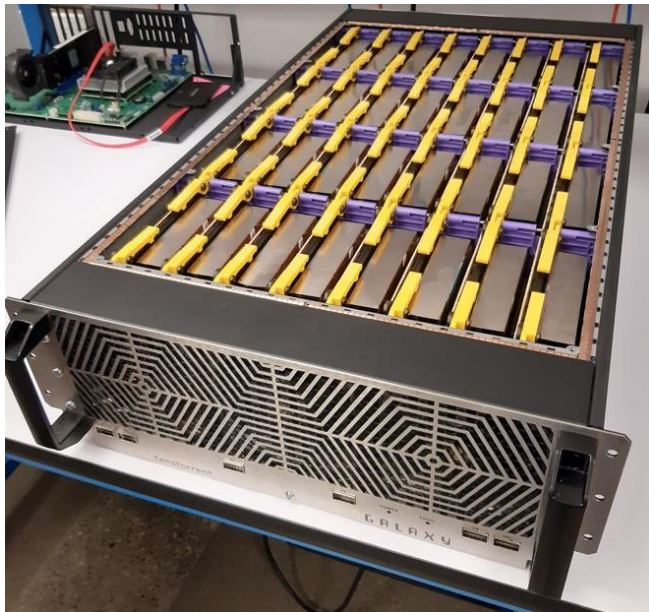


An overview of the Tenstorrent architecture



Scaling out rather than up

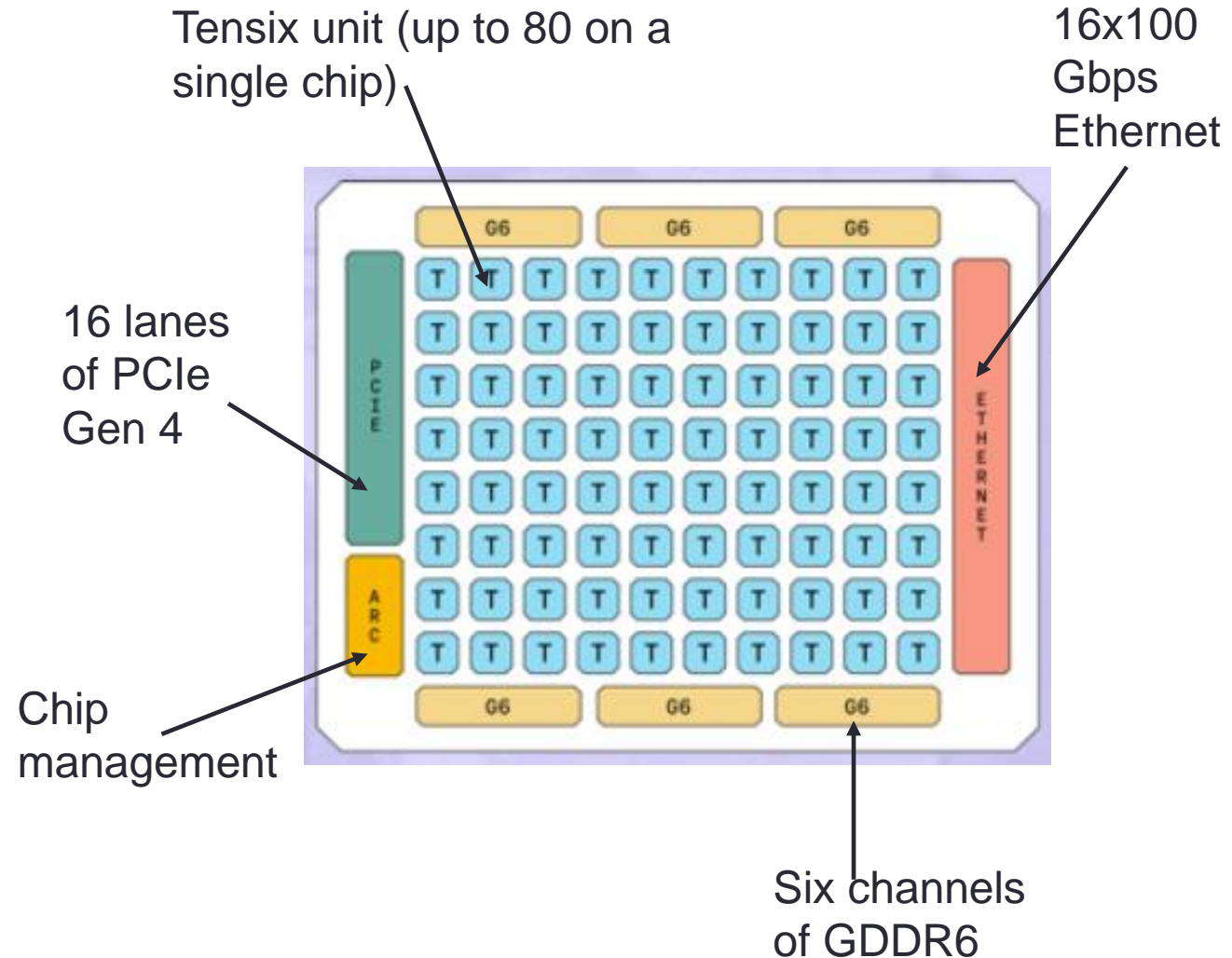
- The Tenstorrent approach is to scale out a (fairly) simple initial compute unit across a chip and multiple chips
 - This simple unit is known as a Tensix unit (more on this soon....)
 - PCIe accelerator cards contain one or more chips, each with many Tensix units



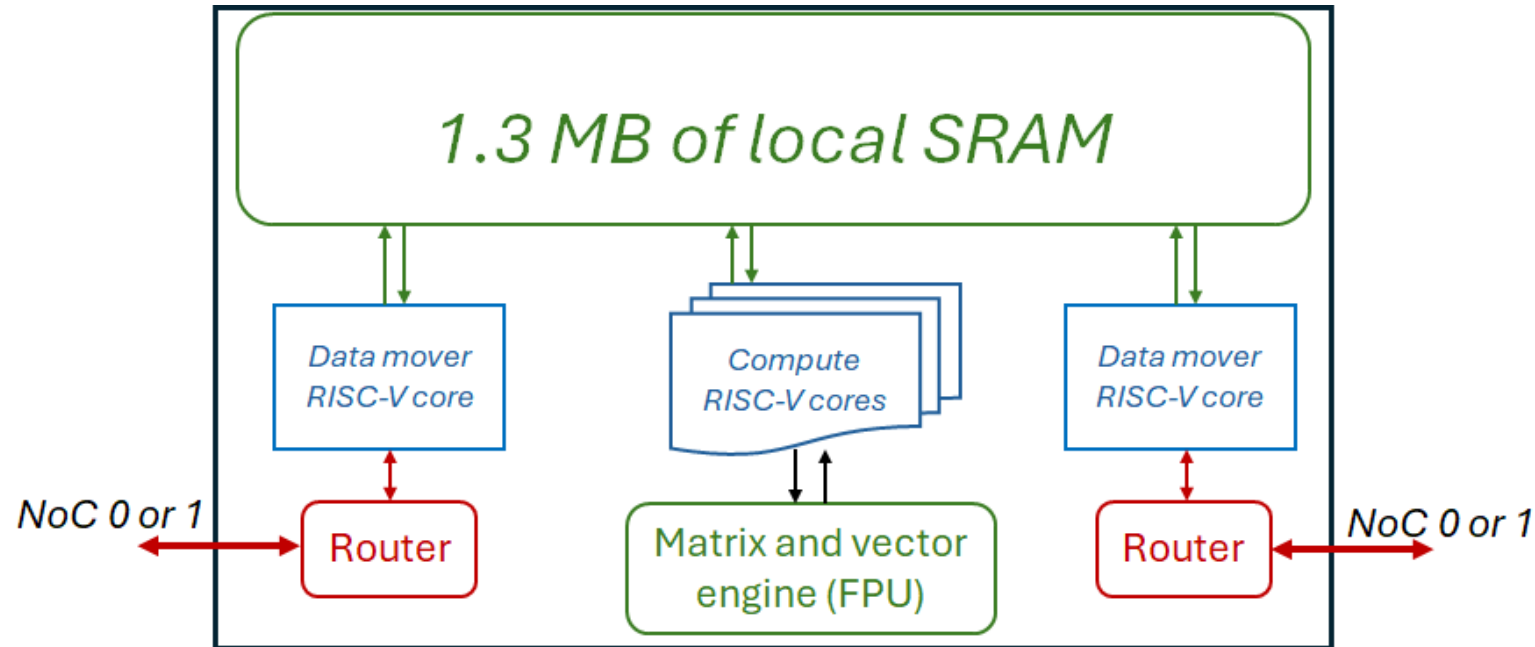
- Cards then scale by being interconnected together
 - These then all appear as a very large (virtual) chip
 - Can do this yourself with the correct cables and connectors
 - The Galaxy contains 32 Wormholes

A single Wormhole chip....

- Built on a 12 nm process
- 12GB GDDR6 on the board
- Draws up to 300 Watts
- Performance:
 - 292 TFLOPS (FP8)
 - 164 TFLOPS (BLOCKFP8)



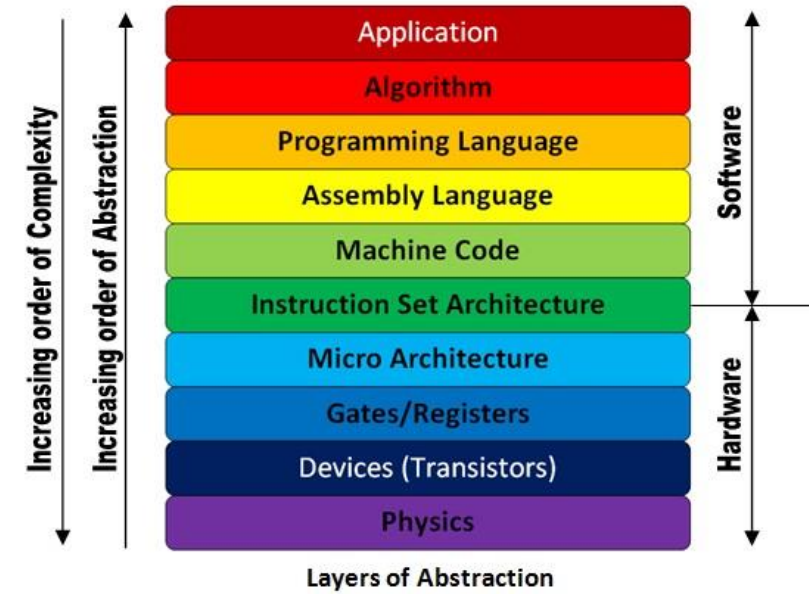
A Tensix unit



- Each Tensix unit contains:
 - Five “baby” RISC-V CPU cores
 - Two of these are for data movement, three drive the compute side by driving the matrix and vector engine
 - A matrix and vector engine (FPU)
 - 1.3MB of local fast SRAM (a bit like a cache)
 - Two routers (one connected to each data mover core)

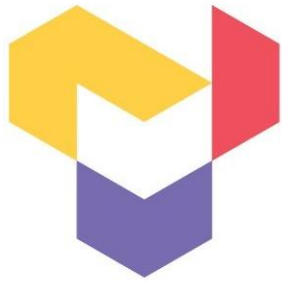
What is RISC-V anyway?

- Started out by Berkley in 2012
- An open Instruction Set Architecture (ISA) which is overseen by RISC-V International
 - Standardisation activities driven by expert members
 - Numerous areas of focus ranging from HPC & ML to the data centre to embedded computing



- RISC-V is an Open Standard Instruction Set Architecture (ISA)
 - Software uses the ISA to tell the hardware what to do.
 - At the base level, the RISC-V ISA and extensions ratified by RISC-V International are royalty free and open base building blocks for anyone to build their own solutions and services on

Lots of serious players here



tenstorrent



InspireSemi™

Qualcomm



SiFive



Why for HPC?

- Modularity and freedom to design bespoke hardware such as the Tensix is the key advantage
- Especially as we see an increased focus on energy efficiency

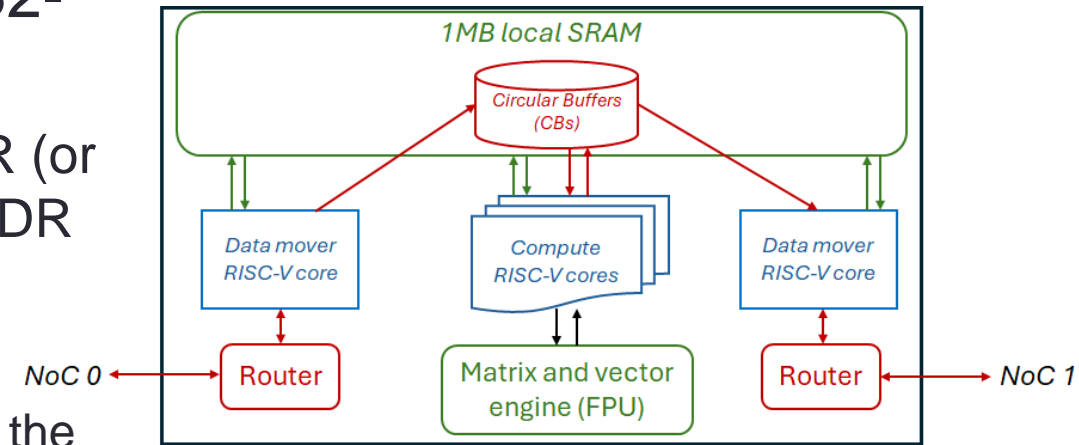
ISA	Chips?	Architecture License?	Commercial Core IP?	Add Own Instructions?	Open-Source Core IP?
x86	Yes, <i>three</i> vendors	No	No	No	No
ARM	Yes, <i>many</i> vendors	Yes, <i>expensive</i>	Yes, <i>one</i> vendor	No (Mostly)	No
RISC-V	Yes, <i>many</i> vendors	Yes, <i>free</i>	Yes, <i>many</i> vendors	Yes	Yes, <i>many</i> available

- We will be running on the EPCC RISC-V testbed throughout this tutorial
 - Provides free access to RISC-V so people can experiment with their workloads for HPC
 - Is one of three officially sanctioned RISC-V labs by the standards body
 - Contains Tenstorrent Wormhole accelerators

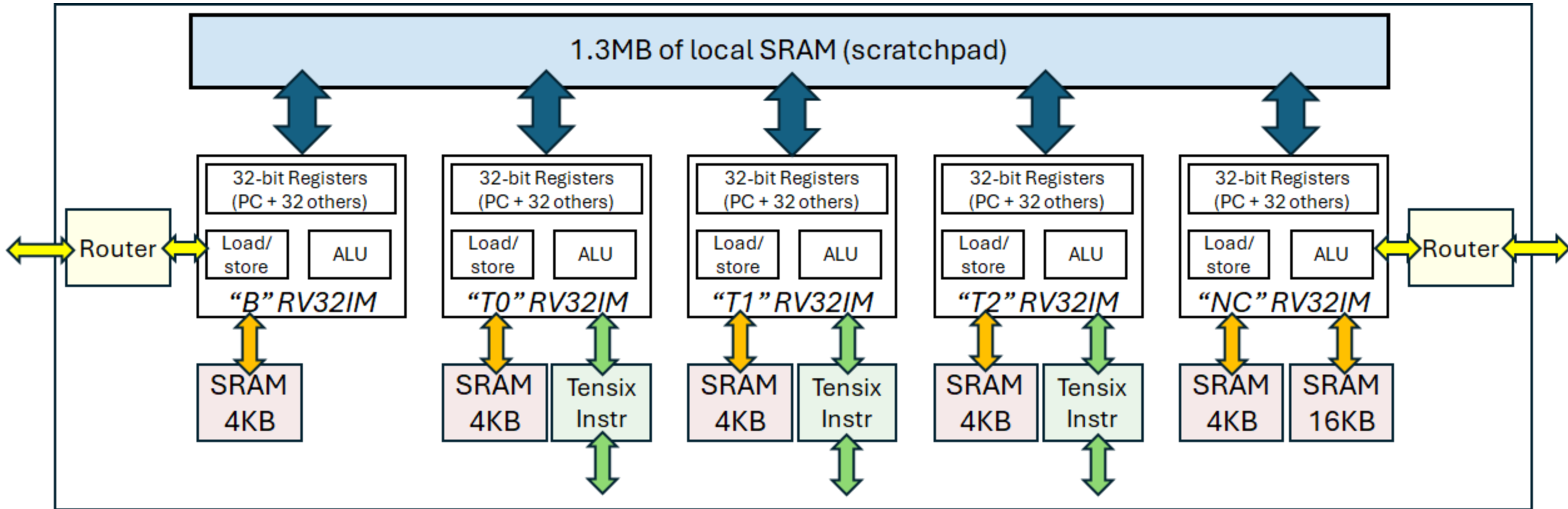


RISC-V CPU cores

- The five “baby” CPU cores are very simple (32-bit RISC-V with Integer support)
 - Two data movers one to get data from external DDR (or another Tensix unit) in, and one to write results to DDR or another Tensix
 - Three compute cores that interact with the FPU
 - One packs data into registers of the FPU, one controls the FPU compute, and the third unpacks from FPU result registers to SRAM.
- RISC-V cores communicate with each other via Circular Buffers (CBs)
 - CBs contain pages of memory, each is a configurable size and follows a producer-consumer approach.
 - Producers will wait until there is a free page, fill this and push to make it available
 - Consumers will block for a page to be pushed and made available, read the data and then pop it



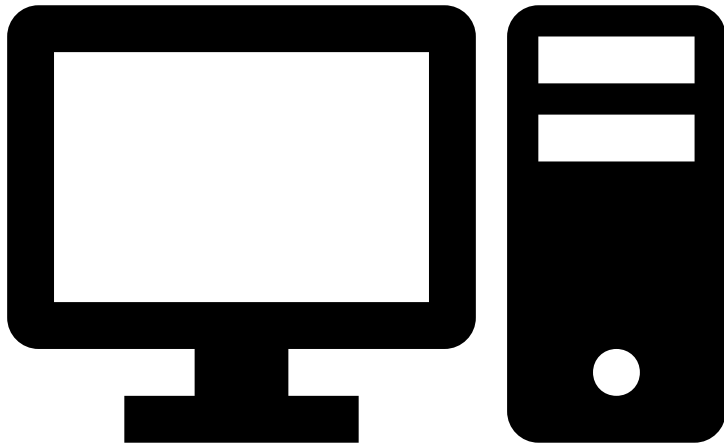
The RISC-V cores in more detail....



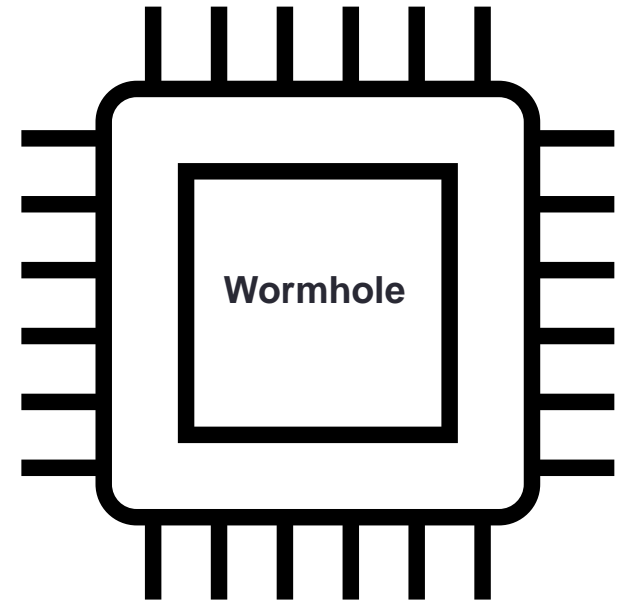
- Each core has some associated SRAM as a local memory for data (and instructions for NC)
- More details on the Tensix instructions and the compute later on....

Programmer's perspective

Host



PCIe accelerator

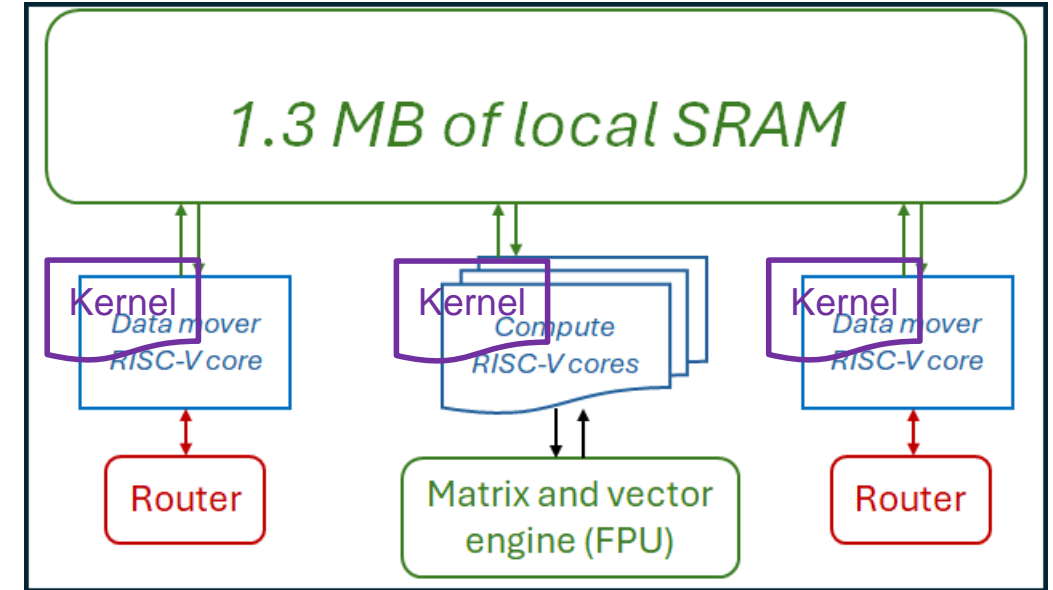


- Input data to DDR
- CB configuration
- Kernels

- Results from DDR

Programmer's perspective

- Host code is written by the programmer
- Three kernels are written by the programmer:
 - Data movement in core
 - Compute cores
 - Data movement out core



- As we will discuss later, these can be replicated across Tensix units or individual kernels allocated on a unit by unit basis
- In the host code, each kernel's path and name is provided
 - When the host code is launched then each kernel is first compiled and launched

TT-Metalium SDK

