System Block Diagram Explanation

This diagram illustrates the complete integration architecture of the SERV core into the **Chipyard SoC environment:**

SERVING (Top Block) encapsulates:

SERVILE: A minimal RV32I bit-serial RISC-V CPU that executes one bit per cycle.

MUX - ARBITER - INTERFACE: Internal logic to arbitrate and route Wishbone transactions.

Serving RAM: A unified memory that holds instruction memory, data memory, and register file.

Protocol Conversion Bridge:

Translates Wishbone ⇔ AXI protocols.

Enables communication between SERV's internal Wishbone interface and the standard AXI bus used in SoCs.

Adapters Layer:

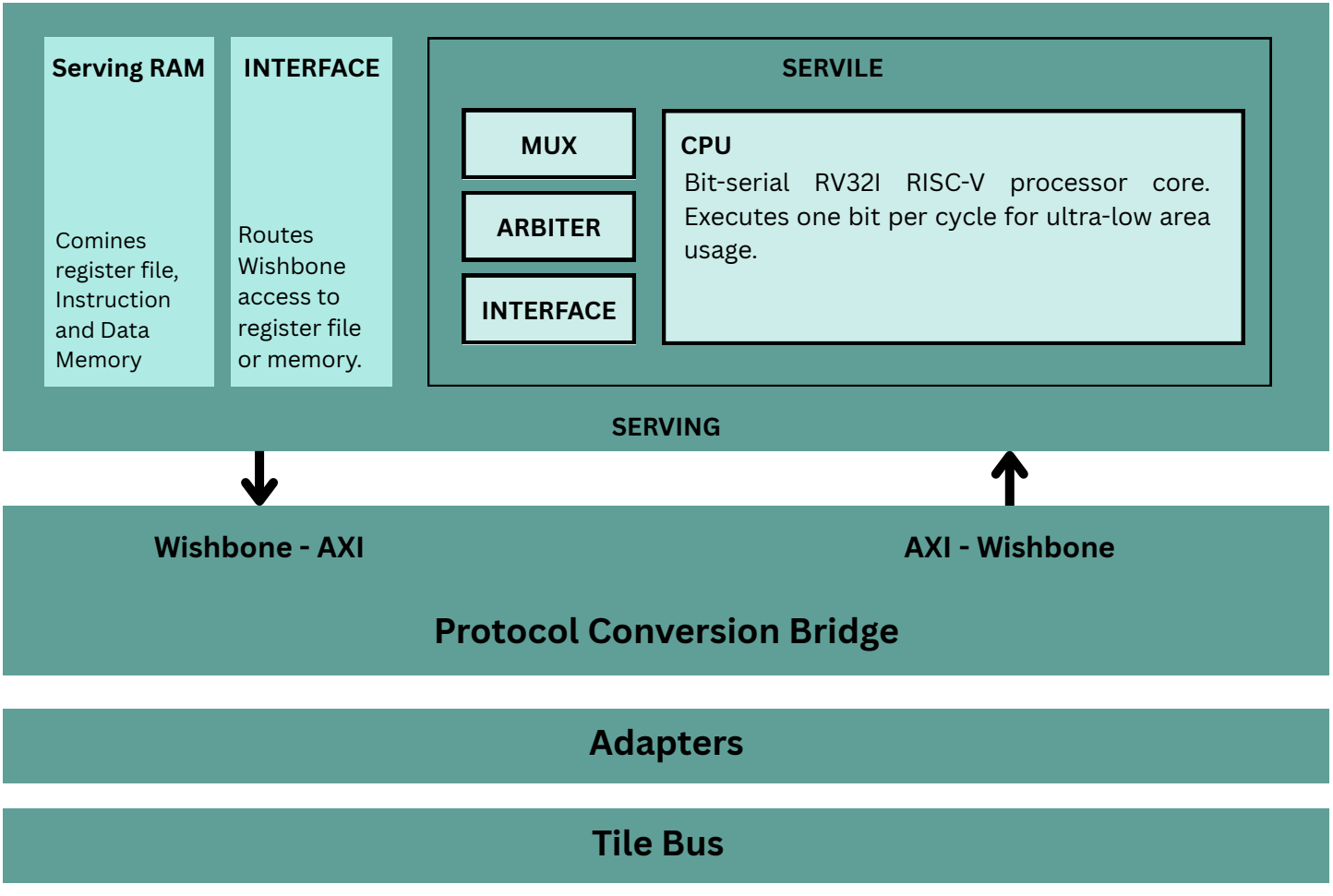
Converts AXI4 ⇔ TileLink using Chisel-based adapters (AXI4ToTL, Fragmenter, etc.).

Ensures compatibility between SERV's AXI output and Chipyard's TileLink-based system bus.

Tile Bus:

The final connection to the rest of the SoC fabric, allowing the SERV tile to perform memory accesses and participate in system-level communication.

This modular design supports a clean, layered integration of a Verilog-only CPU core into a highly configurable Chisel-based SoC framework.


Serving RAM
Comines register file, Instruction and Data Memory
INTERFACE
Routes Wishbone access to register file or memory.
SERVILE
MUX
ARBITER
INTERFACE
CPU
Bit-serial RV32I RISC-V processor core. Executes one bit per cycle for ultra-low area usage.
SERVING
Wishbone - AXI
AXI - Wishbone
Protocol Conversion Bridge
Adapters
Tile Bus