



先进封装与集成芯片

Advanced Package and Integrated Chips



Lecture 9 : SoC/Chiplet Interconnect
Instructor: Chixiao Chen, Ph. D

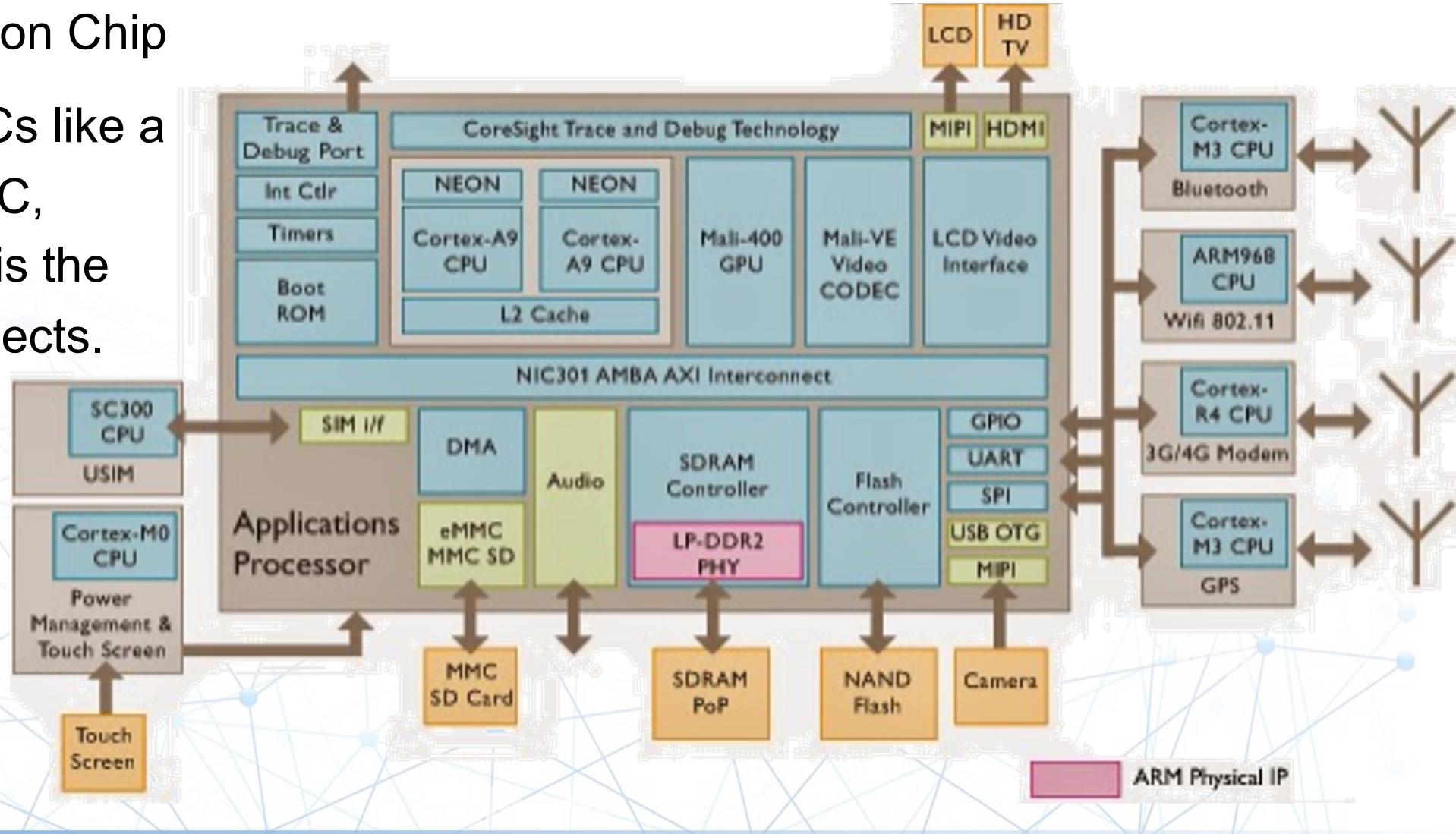
Overview



- Review on SoC Interconnect
 - Bus based on-chip communication
 - Network-on-Chips (NoCs)
- From SoC Peripherals to Chiplet Interconnect
 - Case Studies
 - How advanced packaging affects System Performance

Communication-centric Design for SoC

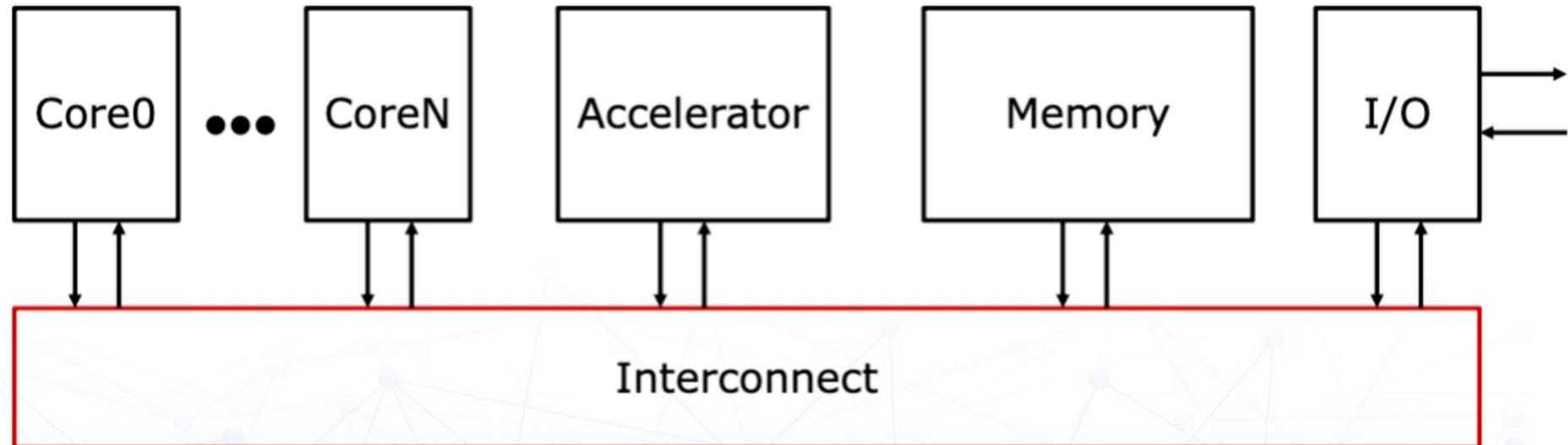
- SoC = System on Chip
- For modern SoCs like a smart phone SoC, communication is the most critical aspects.
- It affects performance, power, area (PPA) & time-to-market.



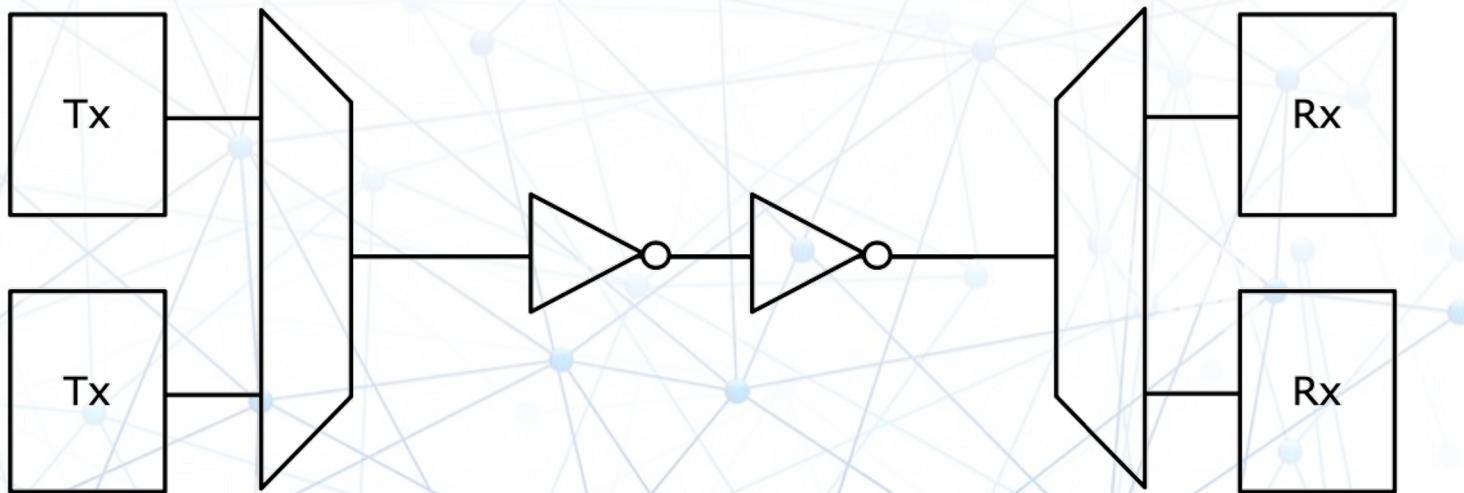
On-Chip Interconnect: Physical and System View



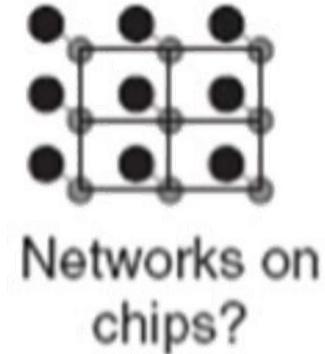
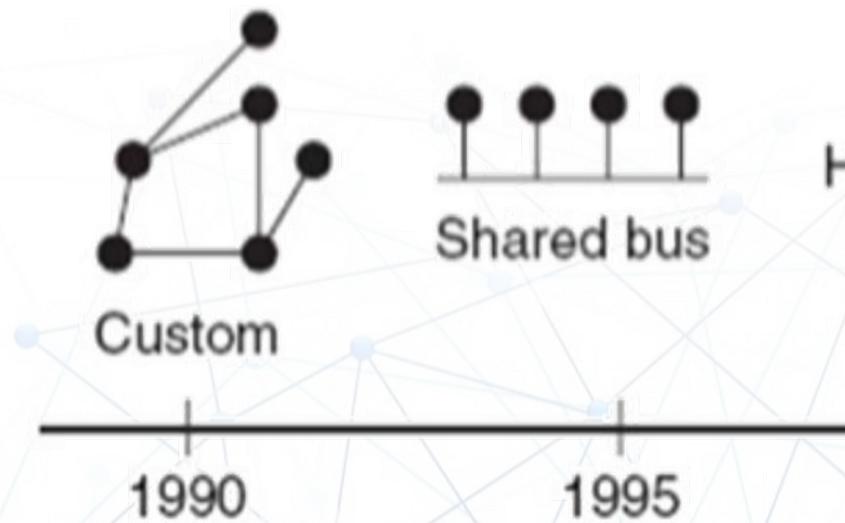
- Interconnect: communication infrastructure connecting all IPs together



- Physical Implementation
Interconnect: to avoid too many wires, we need to multiplex data over a group of shared wires



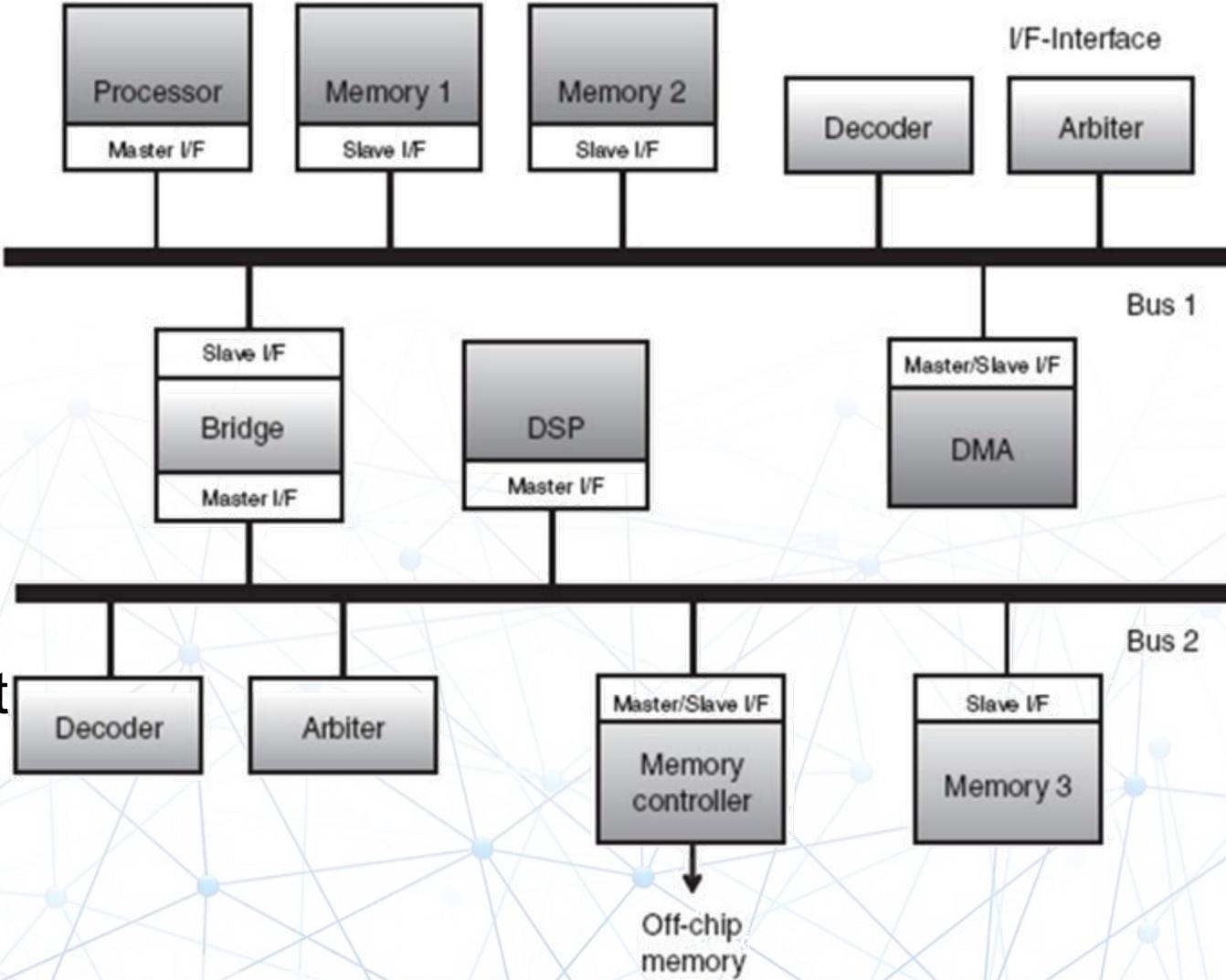
Evolution of on-chip Interconnect



- Bus is the most simplest and widely used SoC Interconnect.
- Bus definition: A collection of signals to which multiple IPs are connected

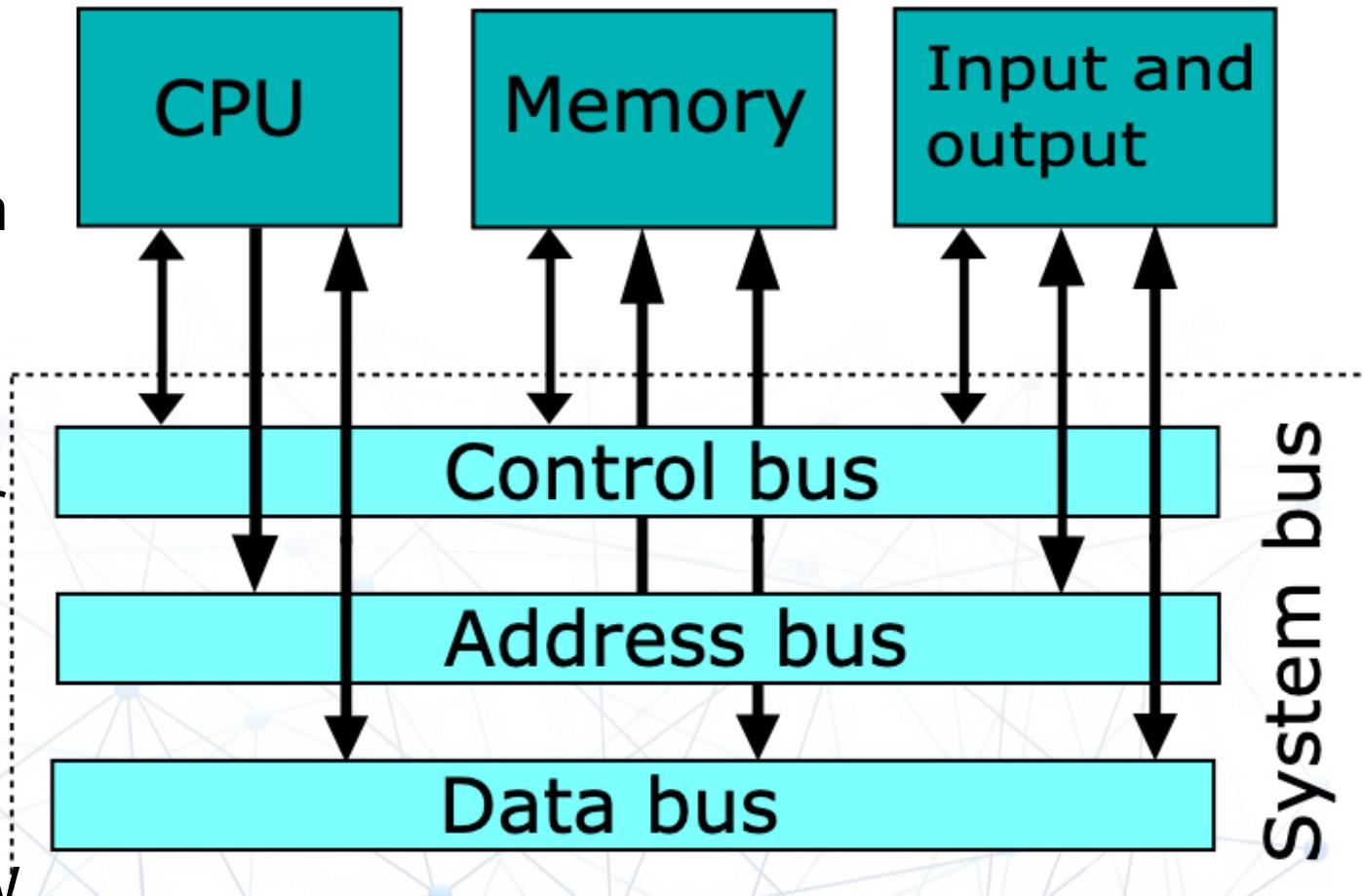
Bus Terminology

- Master: IPs that initiates a read/write data transfer.
- Slave: IPs that only responds to incoming transfer requests.
- Arbiters: Control bus operation by selecting master to grant data transfers.
- Bridge: Connects with different bus, acting as slave one side and master on other side.

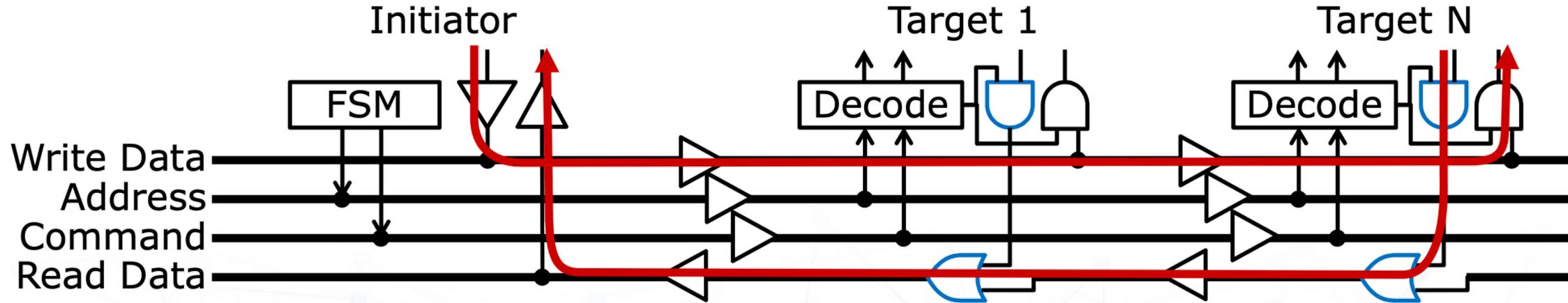


Bus signals

- Address: transfer data's source and destination, uniformed encoded for all on-chip IPs, driven by masters only
- Data: real information sent and received by bus, can be shared or separated for read and write
- Control: includes request and acknowledgements, specify different type of data transfer (R/W, burst, cacheable, byte mask, ...)



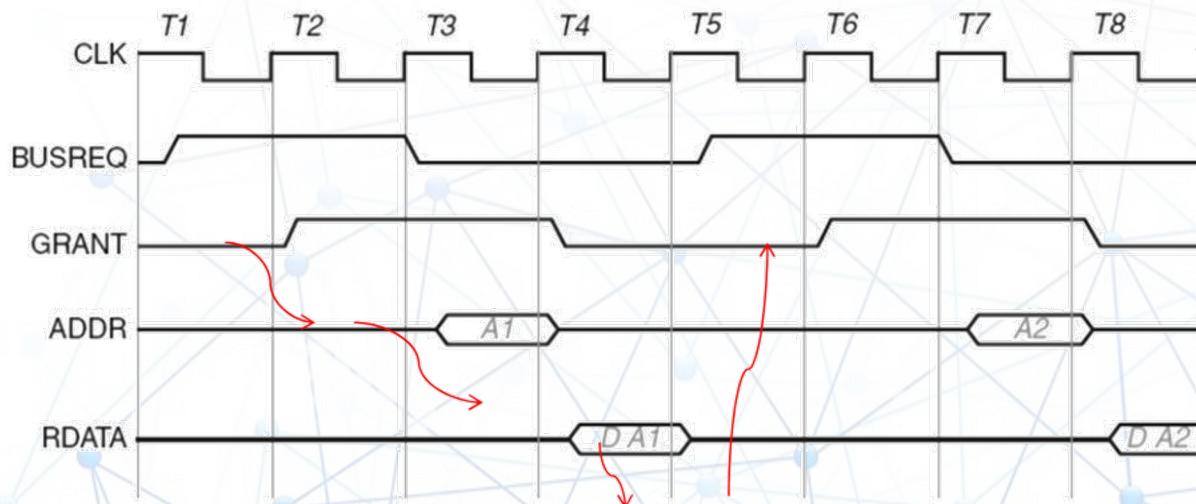
Basic Bus Circuit Implementation (Digital)



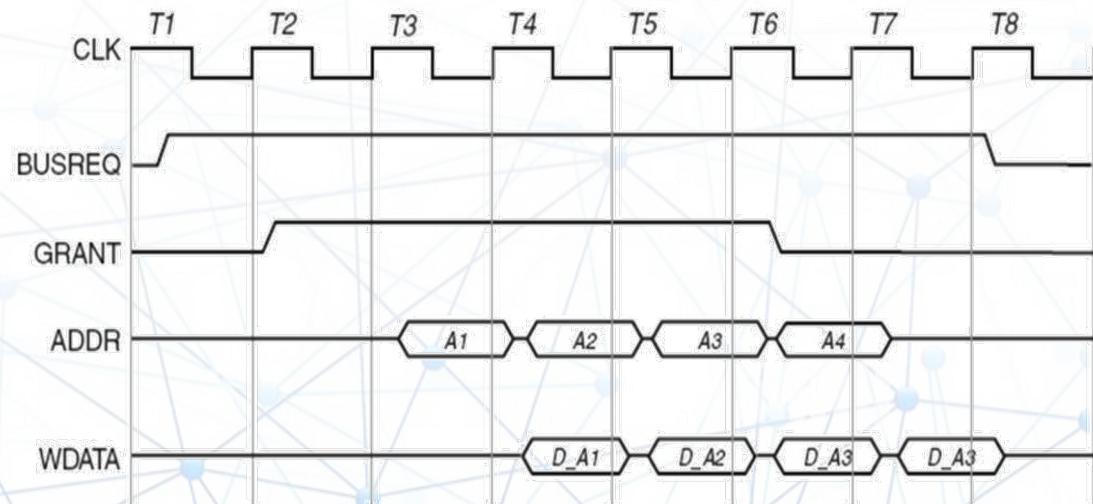
- Historically tri-state drivers (high impedance to disconnect) is not friendly in recent CMOS digital circuit design.
- Current bus implementation adopts differentiate Read & Write Data channels to replace tri-state drivers
- Pipelining stage can be inserted to prevent long-distance transition and latency.

Bus Transfer Modes

- Single data transfer (w/o pipeline)
 - first request to access bus
 - access granted/acknowledged
 - sent address and control signals
 - send/receive data in subsequent cycles



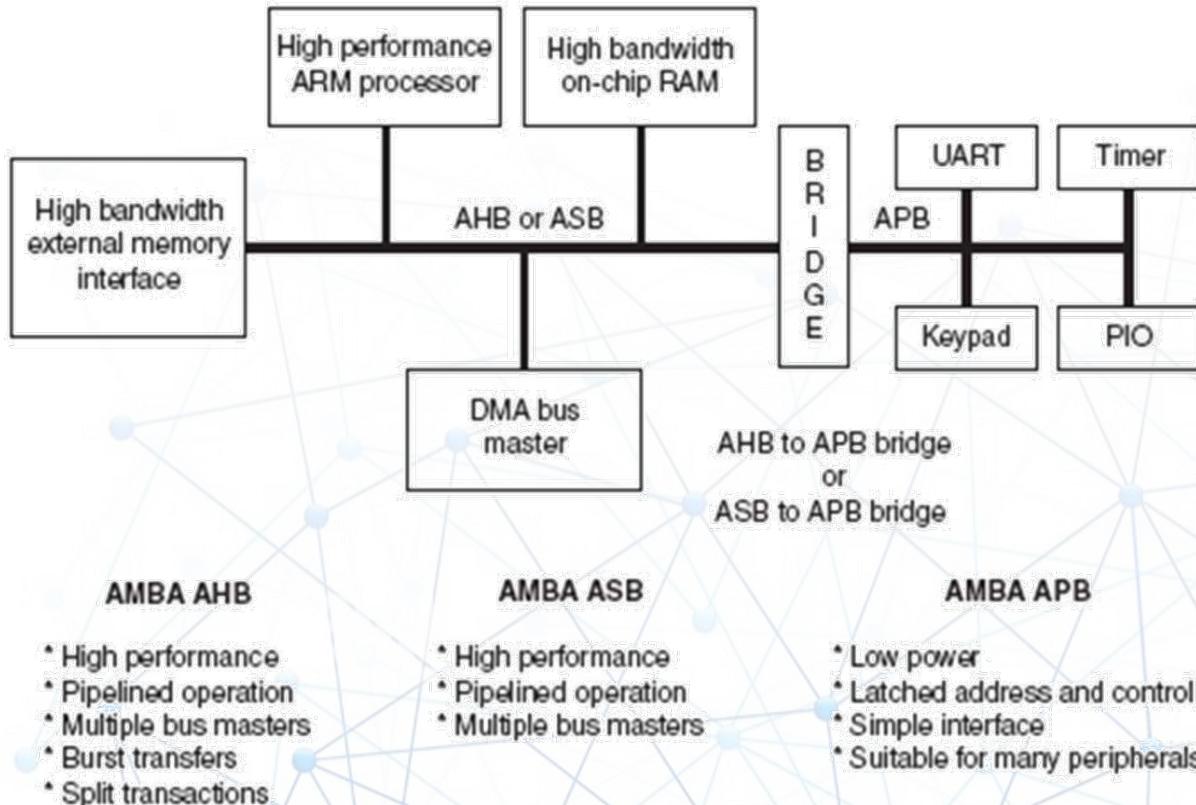
- Burst data transfer (w/i pipeline)
 - send multiple data with only one cycle control (save time for arbitration)
 - Continuous data transfer for recent AI applications



- There should be **a protocol or standard** for bus communication.

AMBA Bus Protocol

- Advanced Microcontroller Bus Architecture, open standard but owned by ARM
- AMBA 2: AHB Advanced High-performance Bus, AMB 3&4: AXI Advanced Extensible Interface



- **AHB Burst**

- Address and Data are locked together (single pipeline stage)
- HREADY controls intervals of address and data



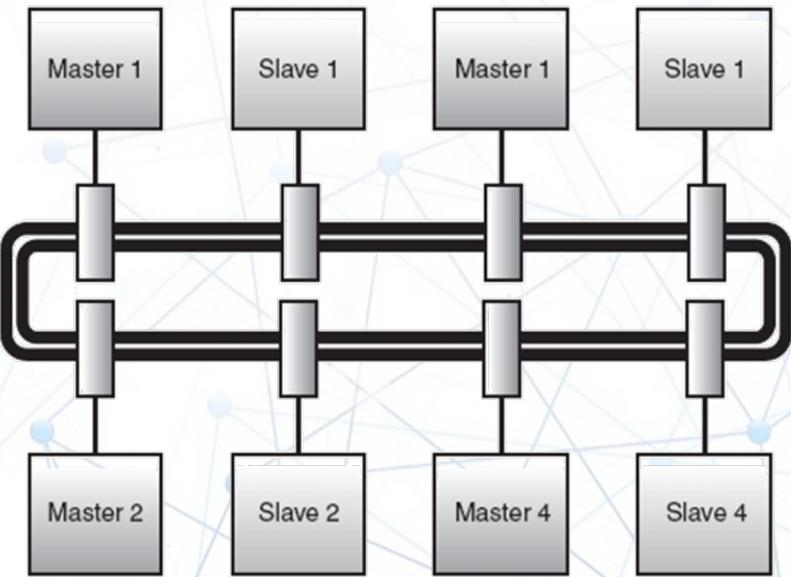
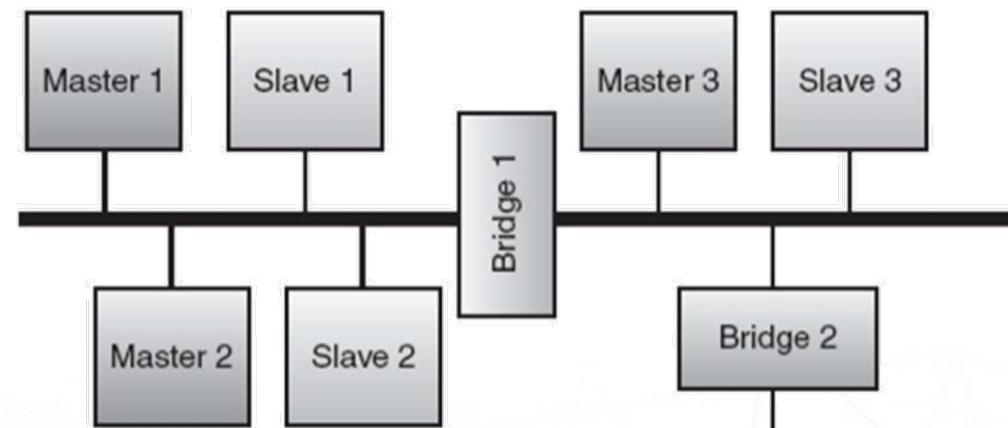
- **AXI Burst**

- One Address for entire burst

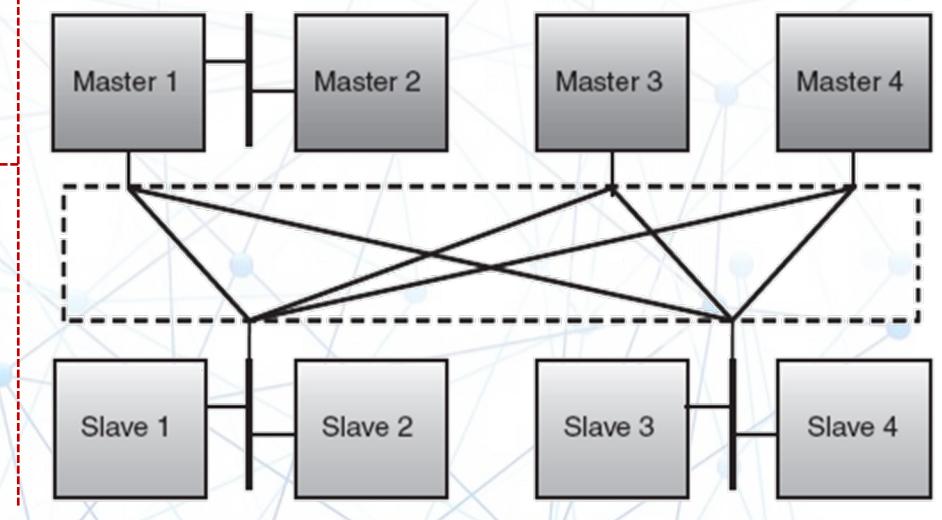
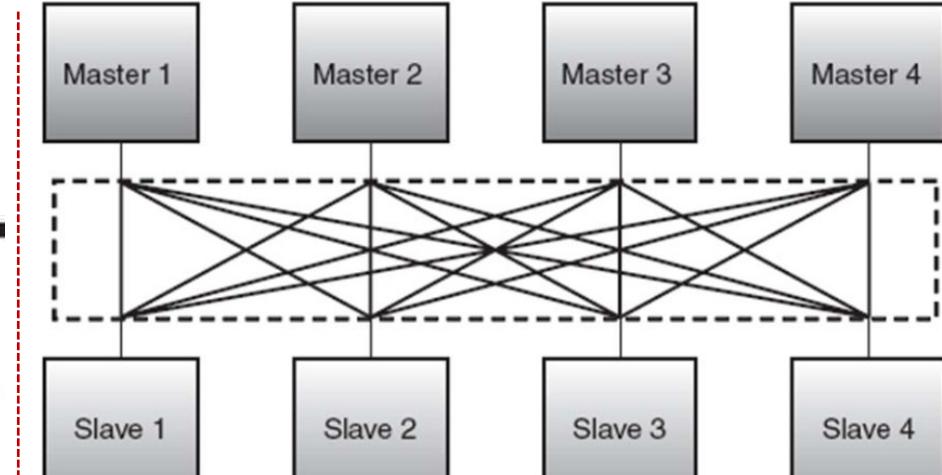


Bus Topologies

- Hierarchical shared bus for different clock SoCs

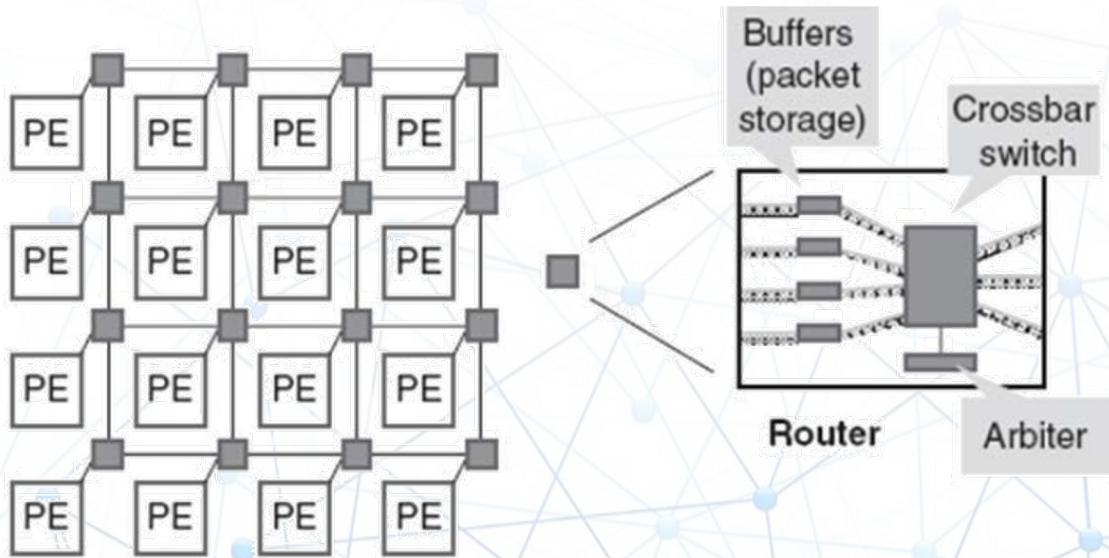


- Fully/partially Crossbar matrix
- Ring bus



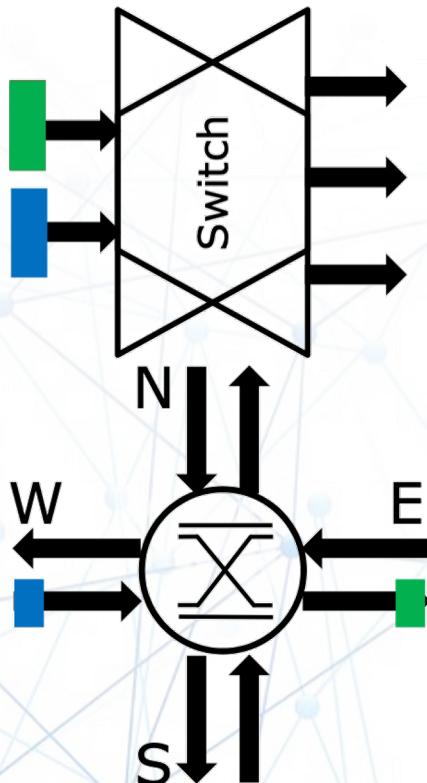
Networks-on-Chips (NoCs)

- Network-on-chips is a packet switch based on-chip interconnection schemes designed by a layered methodology. “route-packets, not wires”
- NoCs use packets to route data from the source to destination PE via a network fabric that consists of routers, as well as links.



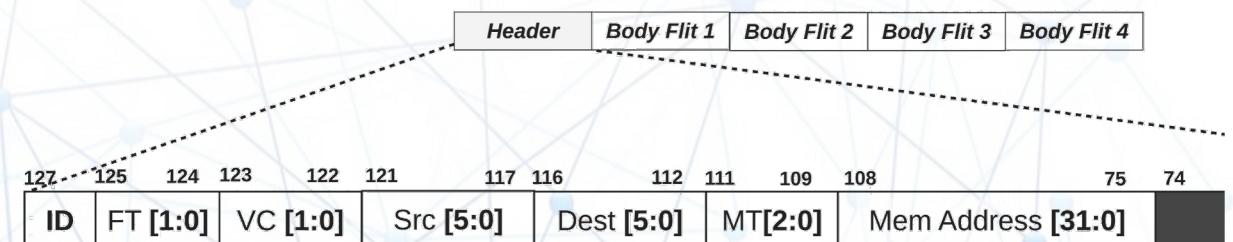
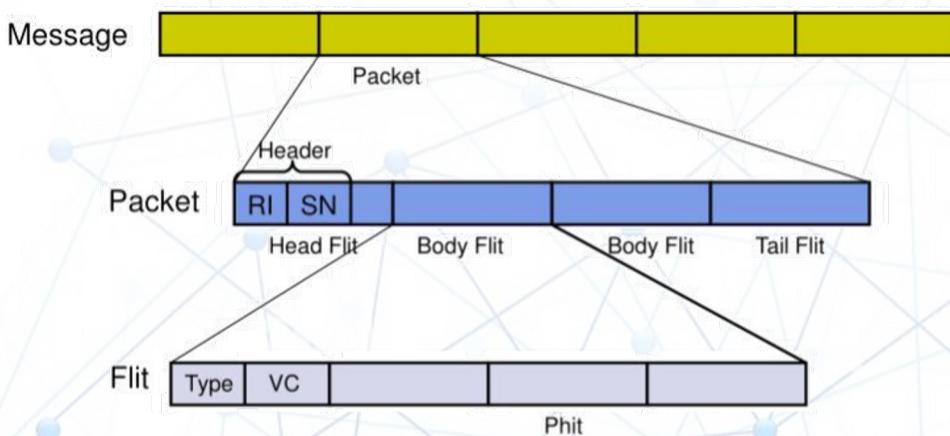
Switch:
little or no queuing,
ports not directional

Router:
often more queuing,
directional ports



Link layer: Packet and Flit

- Provides flow control between network devices, managing link channels to prevent deadlock situations and ensure reliable data transmission
- Phit (Physical control digit) is a unit of data that is transferred on a link.
- Flit (flow control digit) is unit of switching.



Messages, Packets, Flits and Phits are handled in different layers of the network protocol



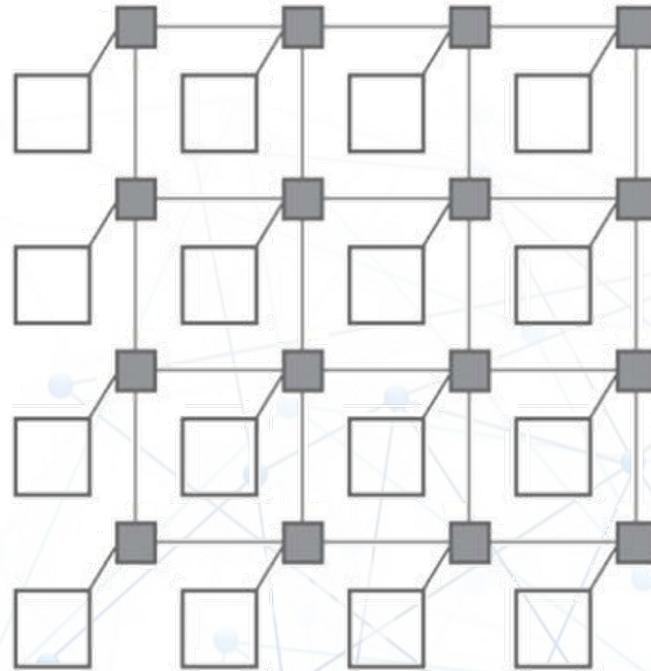
Typical Components of Flit Headers

- **Flit Type (FT)**: Indicates whether the flit is a header, body, or tail. This distinction is vital for routers to process the flit correctly.
- **Virtual Channel Identifier (VCID)**: Assigns the flit to a specific virtual channel, enabling multiple logical channels over a single physical link.
- **Packet Identifier (PID)**: Uniquely identifies the packet, allowing routers to associate all flits belonging to the same packet.
- **Packet Length (PL)**: Denotes the total number of flits in the packet, helping routers manage buffer space and flow control.
- **Priority (PR)**: Determines the urgency of the packet, influencing its precedence in router arbitration.

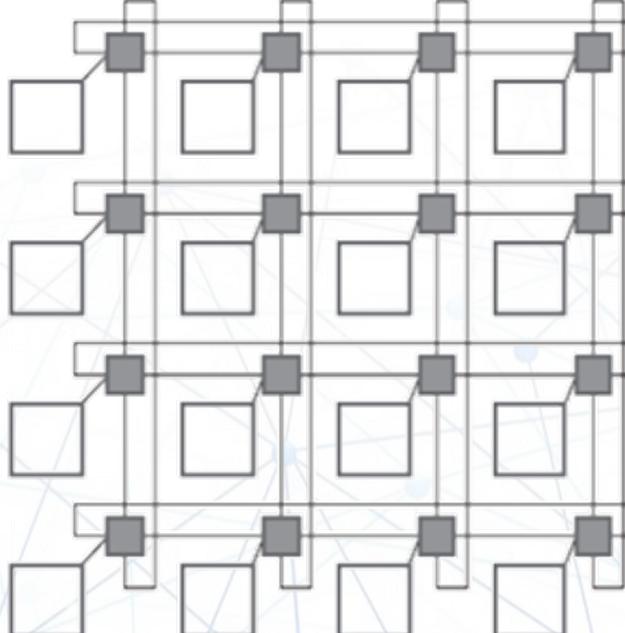
NoC Topologies



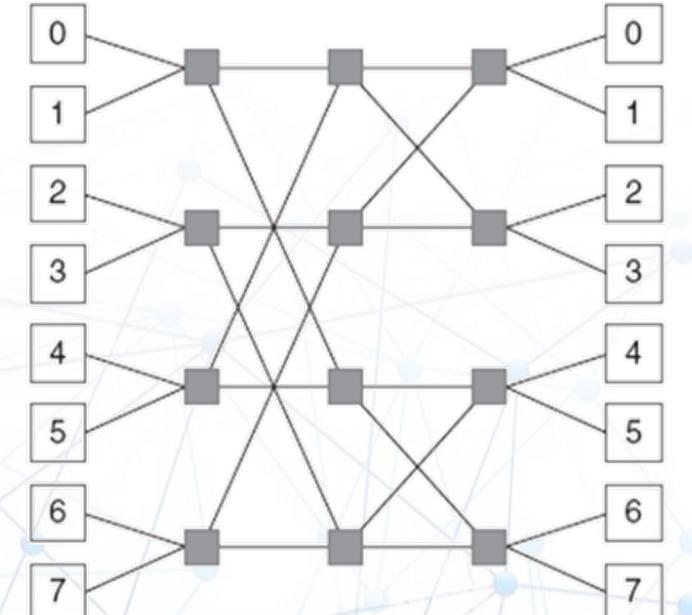
- NoC is kind of an advanced bus, which is more friendly on scalable architectures.
- Many NoC Topologies is used: 2D mesh, Torus, butterfly, fat-tree,...



2D Mesh: equivalent link length



Torus : Ring connection for nodes
at the edges, long end-around link



Butterfly

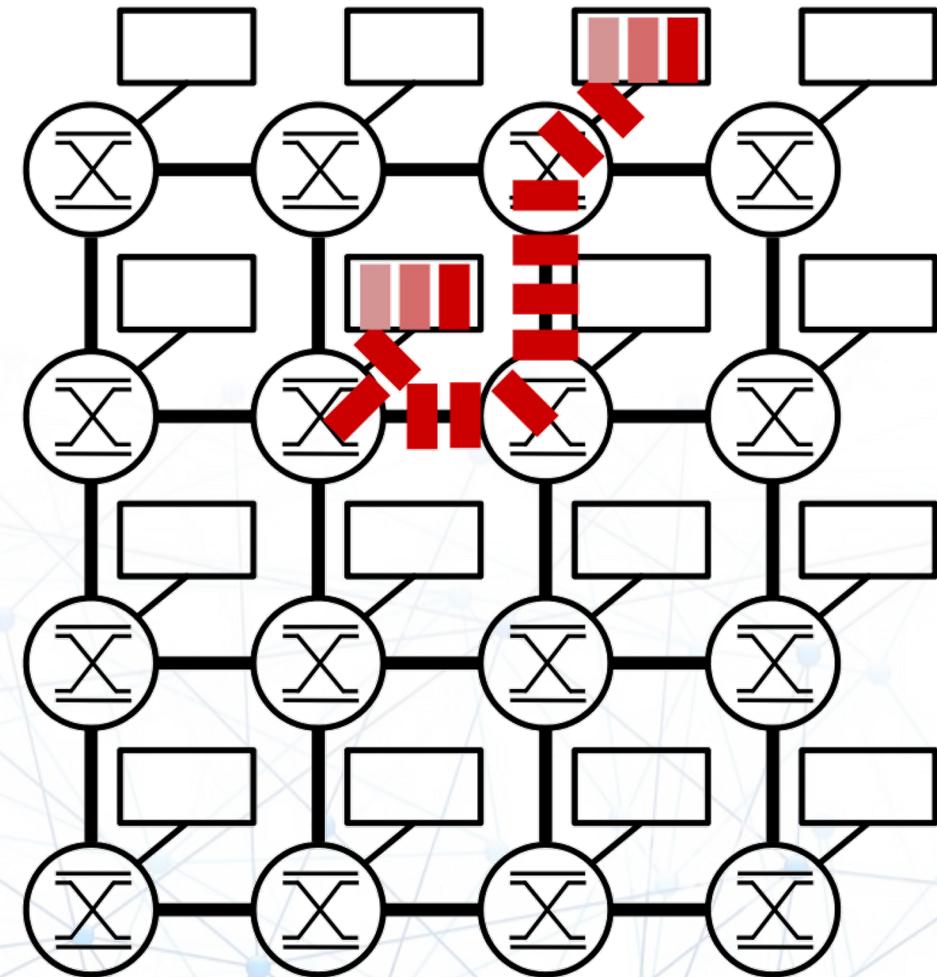
Routing: Packet-Switched Based Interconnect



- Data grouped in packets
- 1 packet : 1 or more data words
 - One word is a "Flit" (Flow-control unit)
 - e.g.: 1st flit = base address & command
2nd flit & next = data burst



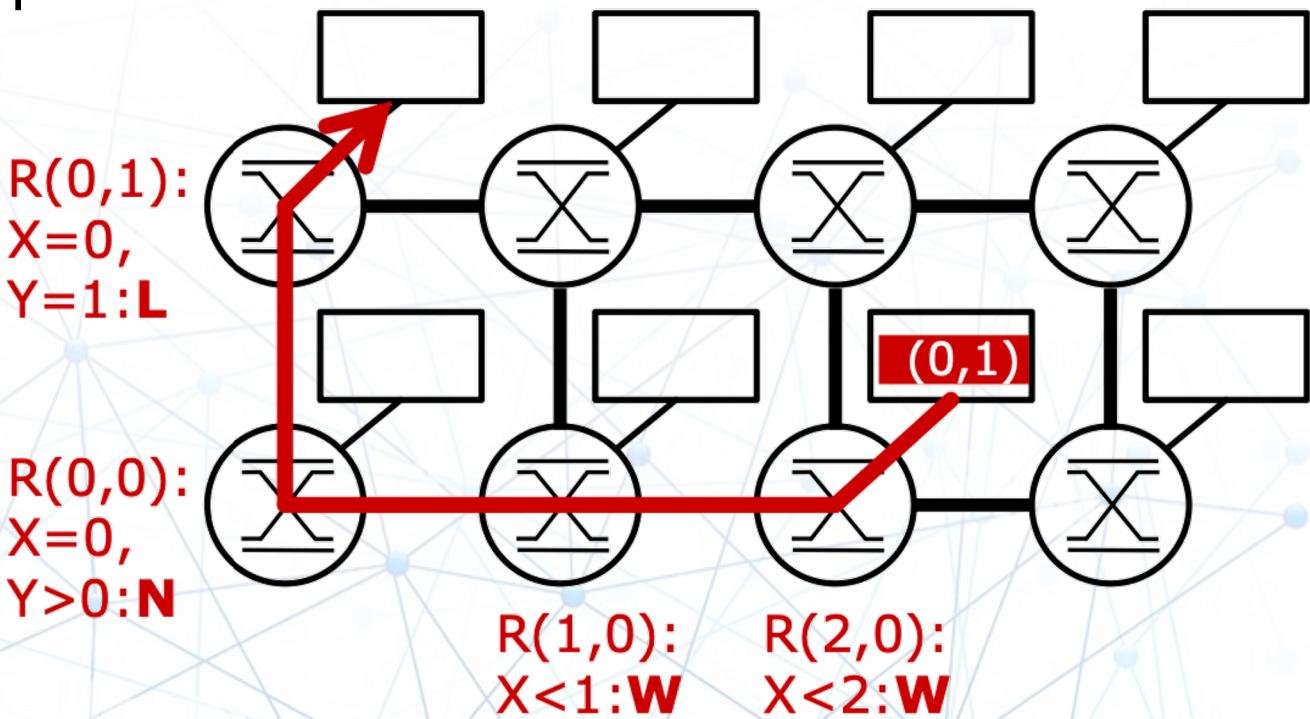
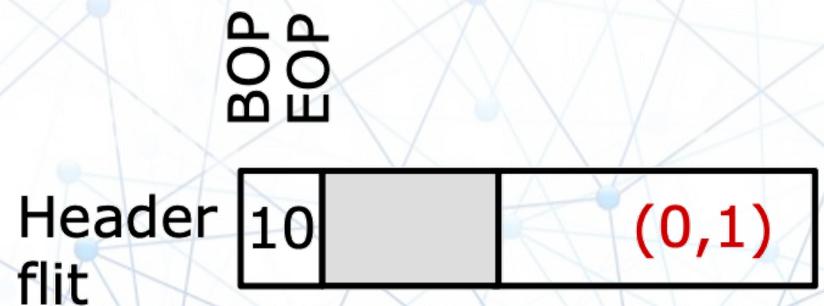
- Each packet contains routing information in the header flit
- Packet routing is atomic
 - No flit interleaving with other packets
 - Can span multiple blocks



Courtesy by Y. Thonnart, ISSCC 2021 Tutorial 8

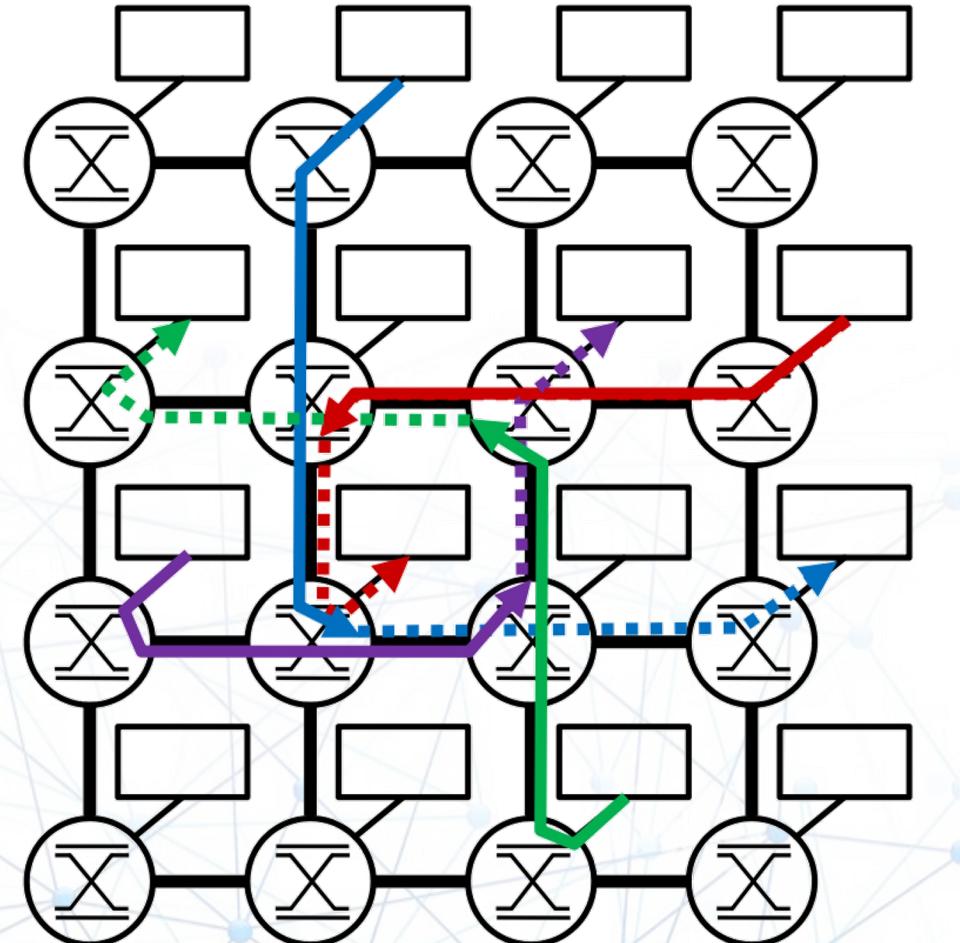
Routing & Packet Format

- In 2D mesh NoC, Coordinates-based Routing is most commonly used
 - Destination coordinates is located in header
 - Comparison to Router coordinates for X-Y routing
- Other methods includes indicate sequence of turning encoded in header flit
 - “East East North Local” ...



Routing: Traffic and deadlock

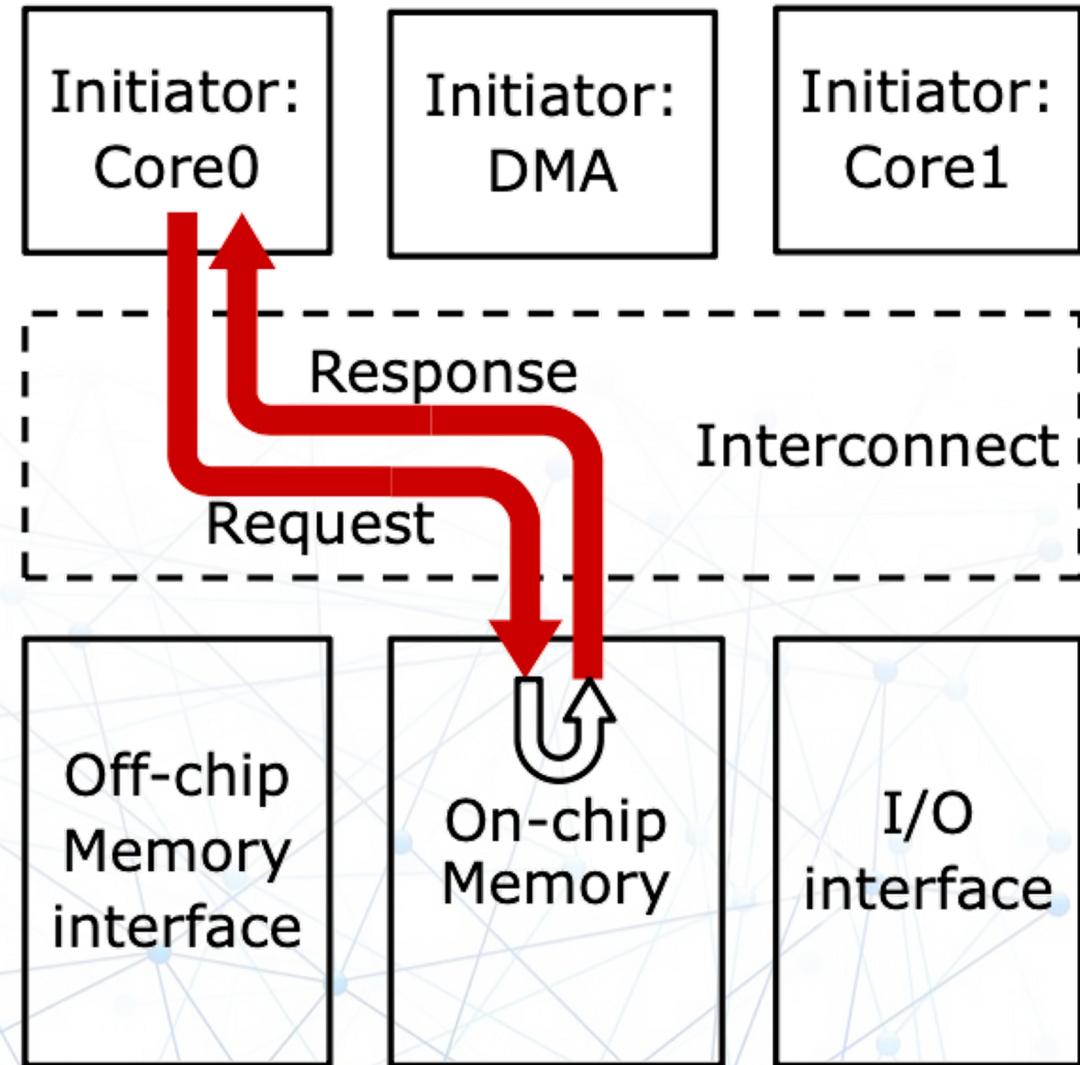
- Queuing behind a stalled packet waiting for an output
 - Potential trail accumulating
- Invalid routing algorithms may create cycles of stalled packets
- Potential deadlock
 - No packet can make progress to destination
- Solved by forbidding some turns
- E.g. X-Y routing: always X first



Courtesy by Y. Thonnart, ISSCC 2021 Tutorial 8

Transaction-based Interconnect

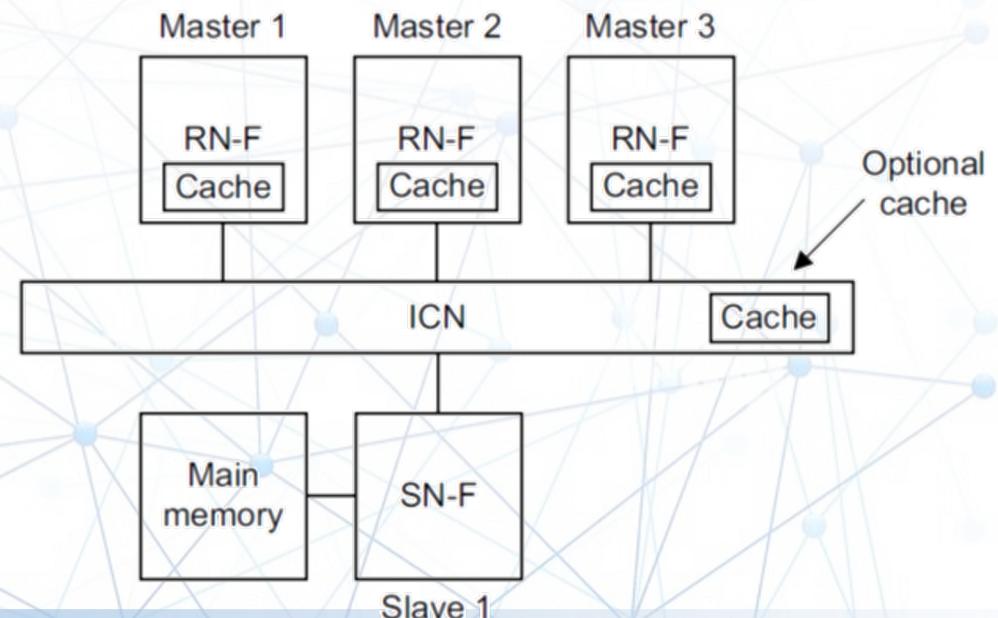
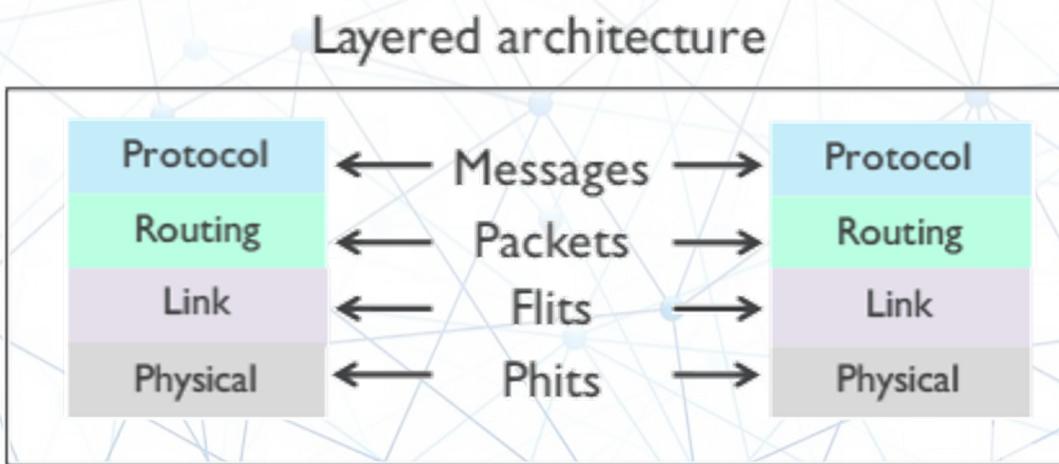
- Memory access is another common type interconnect, which normally use transaction based interconnect.
- Normally memory have specific protocols
- Memory Request/Response have different/independent channels, therefore multiple request are allowed.
- Memory Interconnect issues: coherence in multi-core architecture
- AMBA 5: CHI (Coherent Hub Interface)



AMBA CHI bus Protocol



- Protocol Layer: Manages the generation and processing of requests and responses between protocol nodes. It defines permissible cache state transitions and oversees transaction flows for each request type.
- Network Layer (Routing): Responsible for packetizing protocol messages and determining the routing paths by adding source and destination node identifiers to each packet.



AMBA CHI bus Protocol

- The introduction of the Coherent Hub Interface (CHI) in AMBA 5 represents a significant evolution from the AXI protocol.
- CHI offers a packet-based, layered architecture that improves upon AXI's capabilities by providing advanced features such as quality-of-service (QoS) support, efficient flow control

AMBA 4 ACE	AMBA 5 CHI
Dedicated channels for different functions: write (AW, W, B), read (AR, R), snoop (AC, CR, CD)	Generic signals for all functions (TX, RX) with transaction type encoded in the data transfer
Valid/Ready Flow control makes it easy to observe back pressure	Flow control is credit based and more difficult to observe
Easy to understand address map routes transactions to ports	Address map routing is more complex
Internal details of interconnect not required to understand functionality	Internal functionality of interconnect plays a larger role in understanding system behavior

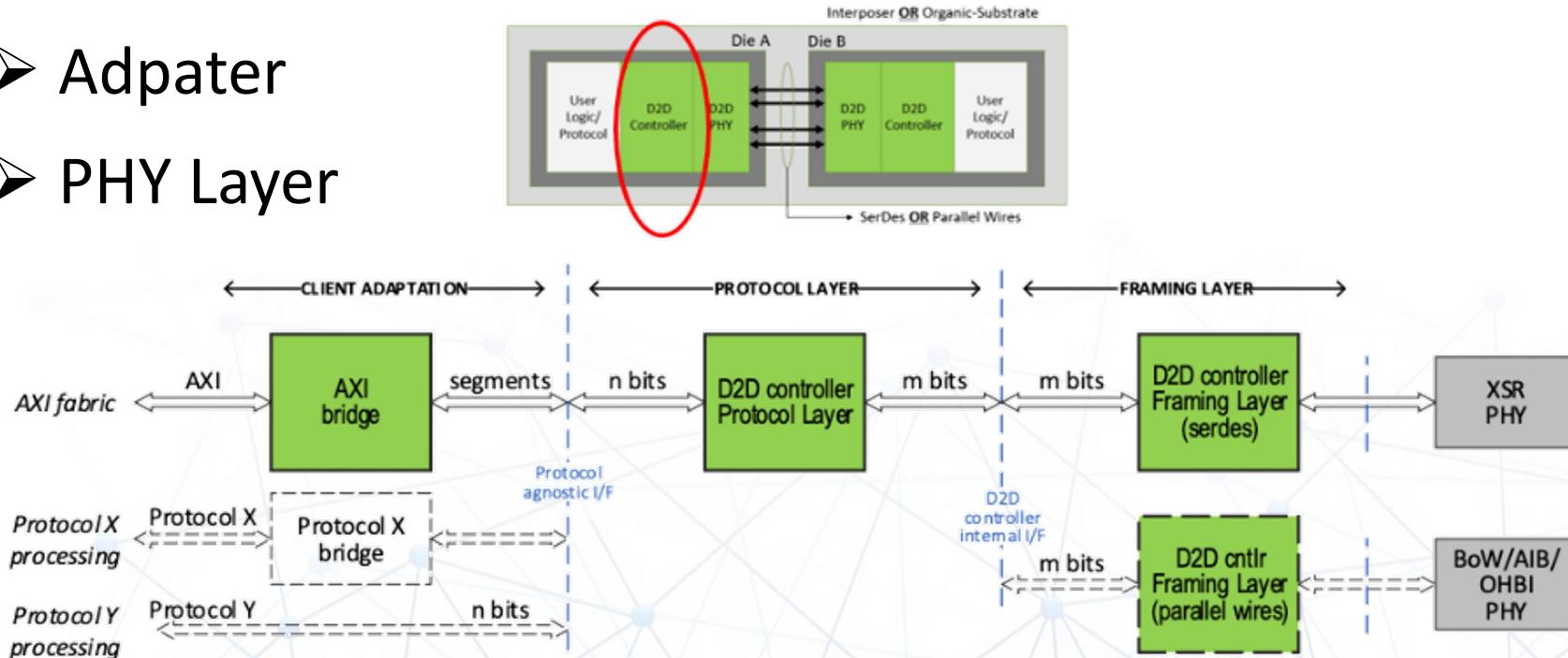
From Bus/NoC to Off-Chip Interconnect

➤ Protocol Layer

D2D Controller Features

➤ Adapter

➤ PHY Layer



Client Adaption

- Adaption of Client signal to Controller Interface
- Support for AXI4, TL, xGMII etc.

Protocol Layer

- End to End error free delivery
- Optional Flow Control
- Optional Re-transmission

Framing Layer

- Function depends on Parallel wires or SerDes
- Lane alignment/De-skew
- Optional FEC Engine

UCle Protocol Stack

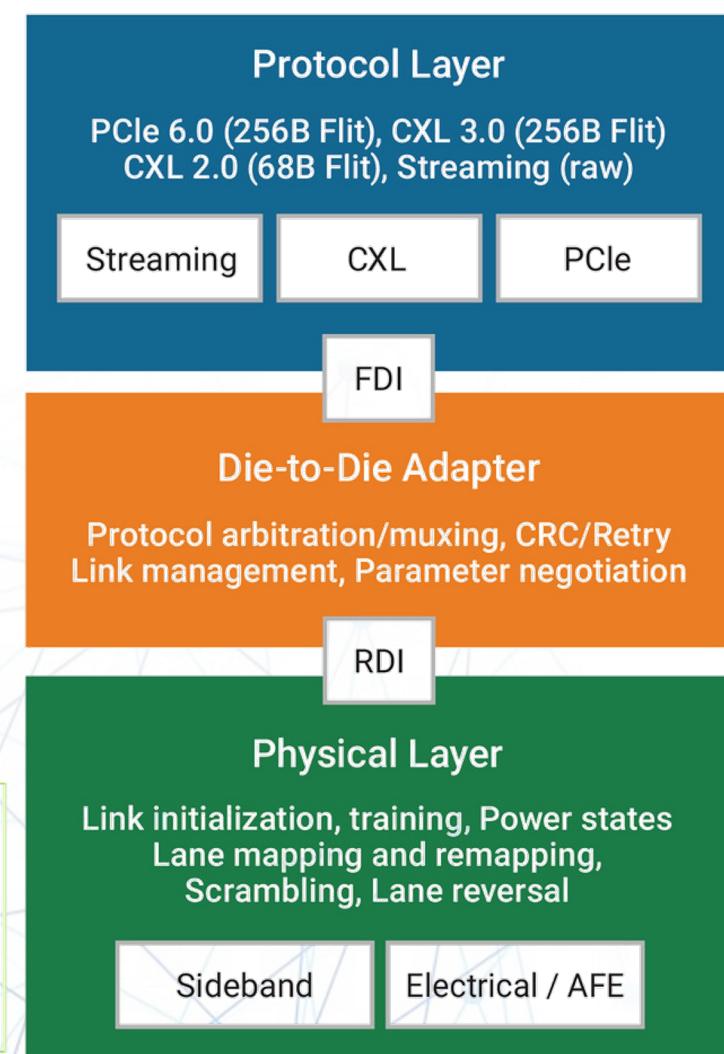


Figure 29. Format 3: Standard 256B Flit Mode for PCIe 6.0

Byte										
0	Flit Chunk 0 64B (from Protocol Layer)									
64	Flit Chunk 1 64B (from Protocol Layer)									
128	Flit Chunk 2 64B (from Protocol Layer)									
192	Flit Chunk 3 44B (from Protocol Layer)	Flit Hdr (Byte 0)	Flit Hdr (Byte 1)	DLP Bytes 2:5	10B Reserved	CRC0 (Byte 0)	CRC0 (Byte 1)	CRC1 (Byte 0)	CRC1 (Byte 1)	

Figure 30. Format 4: Standard 256B Flit Mode for CXL.cachemem

Byte																		
0	Flit Hdr (Byte 0)	Flit Hdr (Byte 1)	Flit Chunk 0 62B (from Protocol Layer)															
64	Flit Chunk 1 64B (from Protocol Layer)																	
128	Flit Chunk 2 64B (from Protocol Layer)																	
192	Flit Chunk 3 50B (from Protocol Layer)				10B Reserved	CRC0 (Byte 0)	CRC0 (Byte 1)	CRC1 (Byte 0)	CRC1 (Byte 1)									

Cyclic Redundancy Check

- The 16-bit CRC is integrated within the 256B Flit structure, ensuring high data integrity during inter-chiplet communication.

