MANCHESTER 1824

The University of Manchester

# From Channel Slicing to Spatial Division Multiplexing

## -- the asynchronous router design

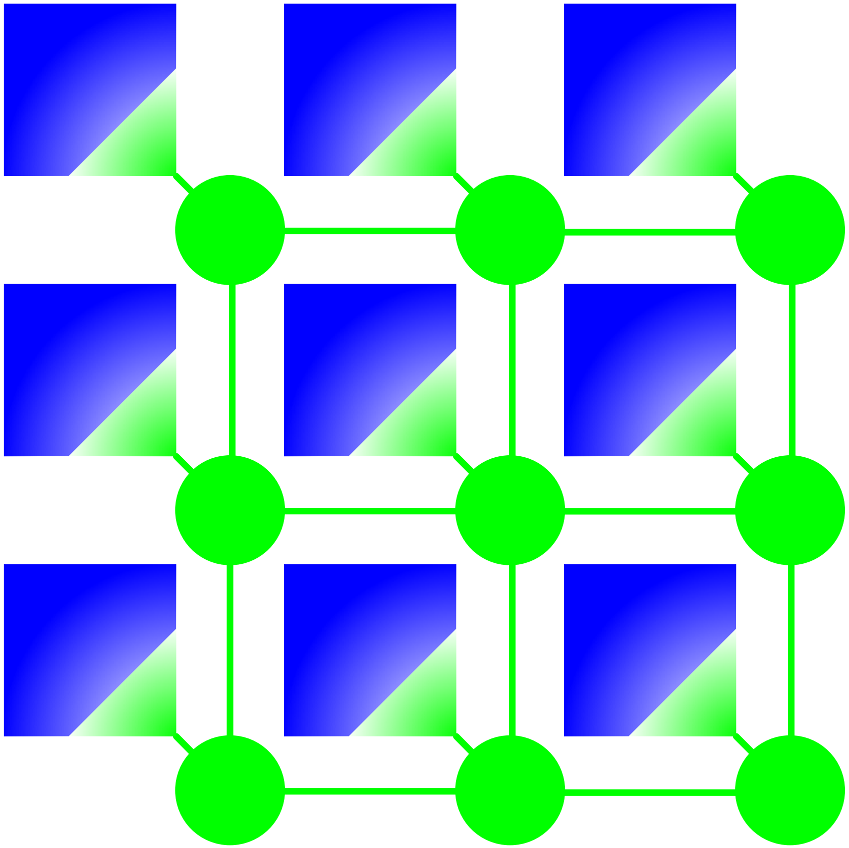
Wei Song

03/12/2009

# Index

- **Channel Slicing**
  - Asynchronous NoCs and routers
  - Channel Slicing
  - A wormhole router design
- Spatial Division Multiplexing (SDM)
  - Motives
  - Switching networks
  - 2-stage Clos network
  - The distributed scheduler
  - Implementation results

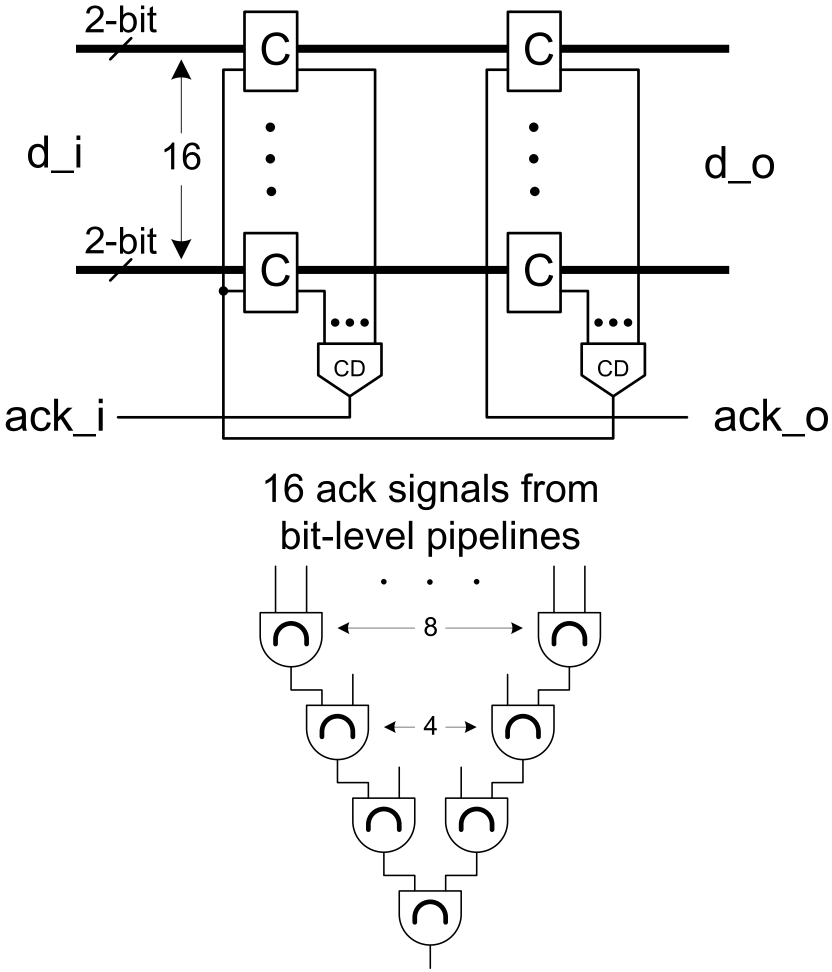
# Asynchronous NoCs

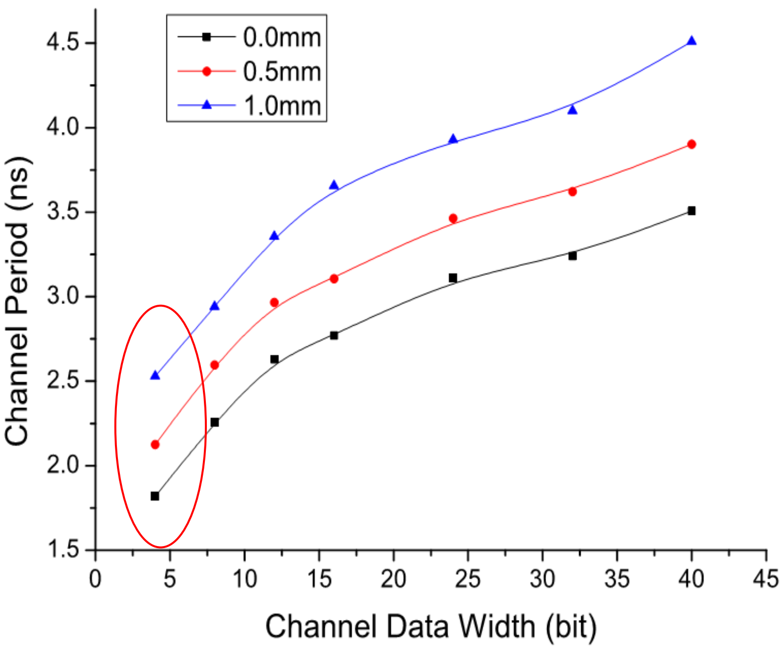


- GALS
- Full async comm fabric
- QDI pipelines

- Low dynamic power
- Tolerance to variation
- Fast prototype

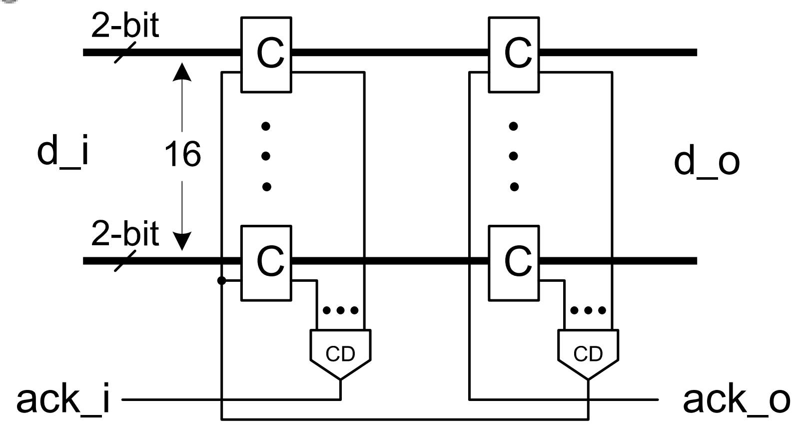
# Synchronised QDI Pipelines



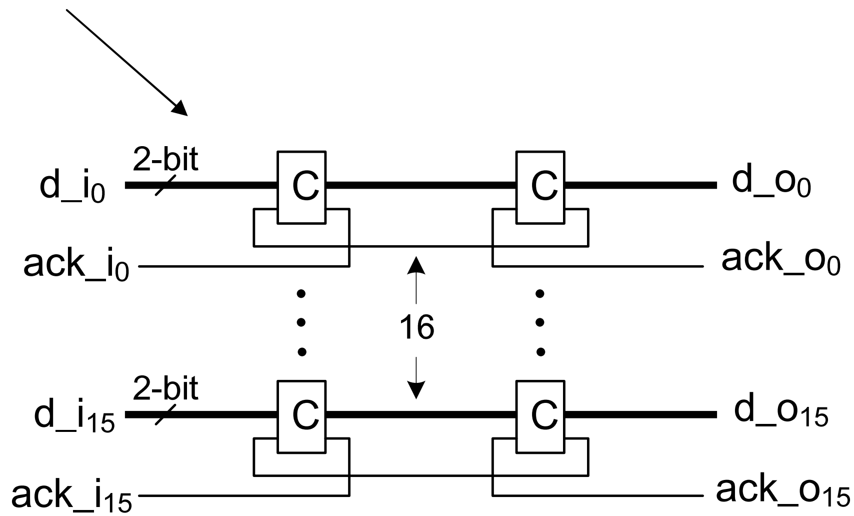




Nangate Cell Lib 65nm 1-of-4

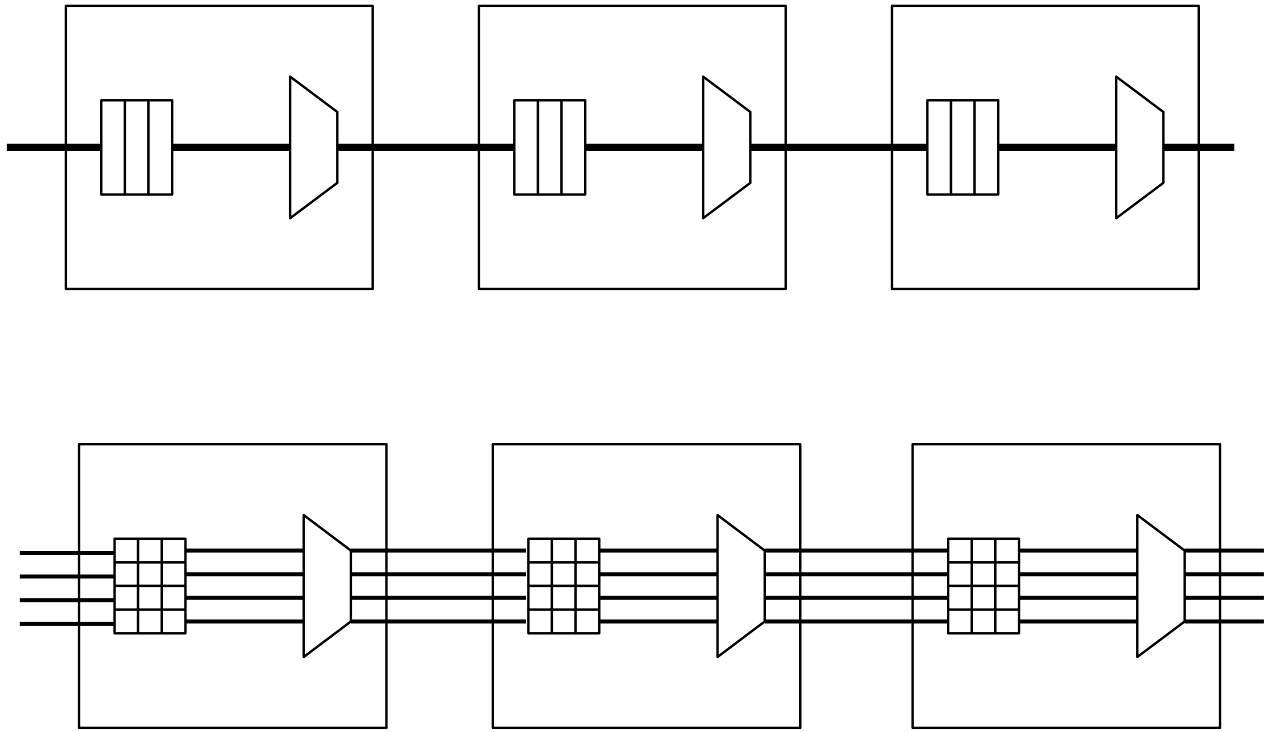
# Channel Slicing (1)



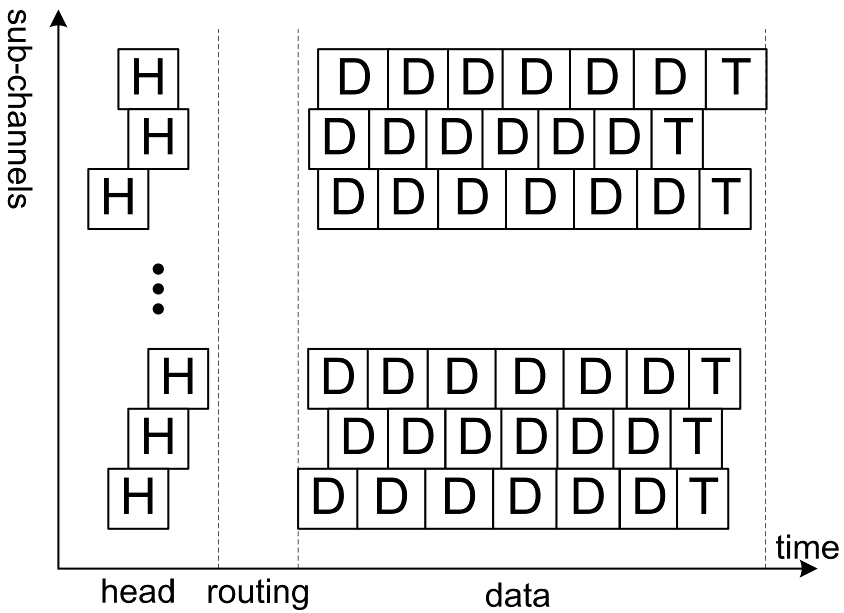

- Remove the C-element tree
- Sub-channels run independently

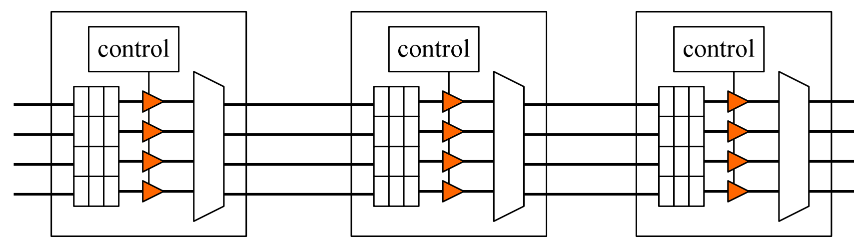


# Channel Slicing (2)

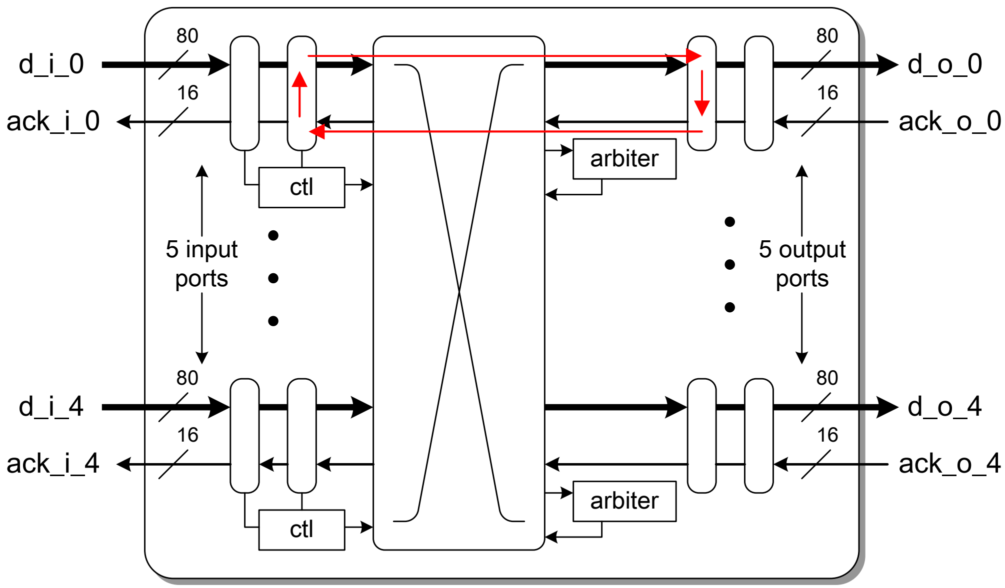


# Channel Slicing (3)

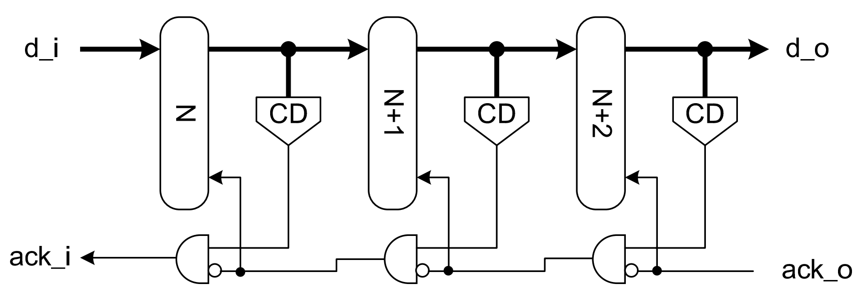


# The Wormhole Router




- Faraday 130 nm
- 5 32-bit ports
- 3 routers:
  - Synchronised
  - Channel Sliced
  - Plus lookahead



# Area Results

TABLE I
AREA OVERHEAD OF CHSLICE AND LH

| Block | ChSlice & LH | ChSlice | No ChSlice/LH |
|---|---|---|---|
| Input Buffers | 6.2K | 5.8K | 4.3K |
| Output Buffers | 4.5K | 4.5K | 4.4K |
| Crossbar | 3.3K | 3.2K | 2.4K |
| Total | 14.5K | 13.9K | 11.3K |

Channel Slicing: 23%
- extra controllers in input buffer
- increased wire count in crossbar

Lookahead: 5.3%
- extra AND gates and C2P elements on critical path

# Speed Results

TABLE II

SPEED IMPROVEMENT OF CHSLICE AND LH

| | ChSlice & LH | ChSlice | No ChSlice/LH |
|---|---|---|---|
| Period | 1.7 ns | 2.2 ns | 2.9 ns |
| Latency | 1.7 ns | 2.1 ns | 2.8 ns |
| Route Overhead | 0.8 ns | 0.8 ns | 0.8 ns |

Synchronised: 345MHz
Channel Slicing: 450MHz
ChSlice+LH: 590MHz

# Compare with Other Routers

| Router | Period | Latency | Tech | Library & Layout | Protocol |
|---|---|---|---|---|---|
| MANGO [12] | 1.26 ns | unknown | 0.12 $\mu$m | unknown | bundled-data |
| ANoC [6] | 4 ns | 2 ns | 0.13 $\mu$m | augmented cell lib | 1-of-4 |
| QNoC [13] | 4.8 ns | 10 ns | 0.18 $\mu$m | standard cell lib | bundled-data |
| ASPIN [8] | 0.88 ns | 1.53 ns | 90 nm | partial customized | dual rail & bundled-data |
| Our Router | 1.7 ns | 1.7 ns | 0.13 $\mu$m | standard cell lib | 1-of-4 & Lookahead |

Asynchronous cell library: constrains the adaptation to other projects

ANoC, ASPIN

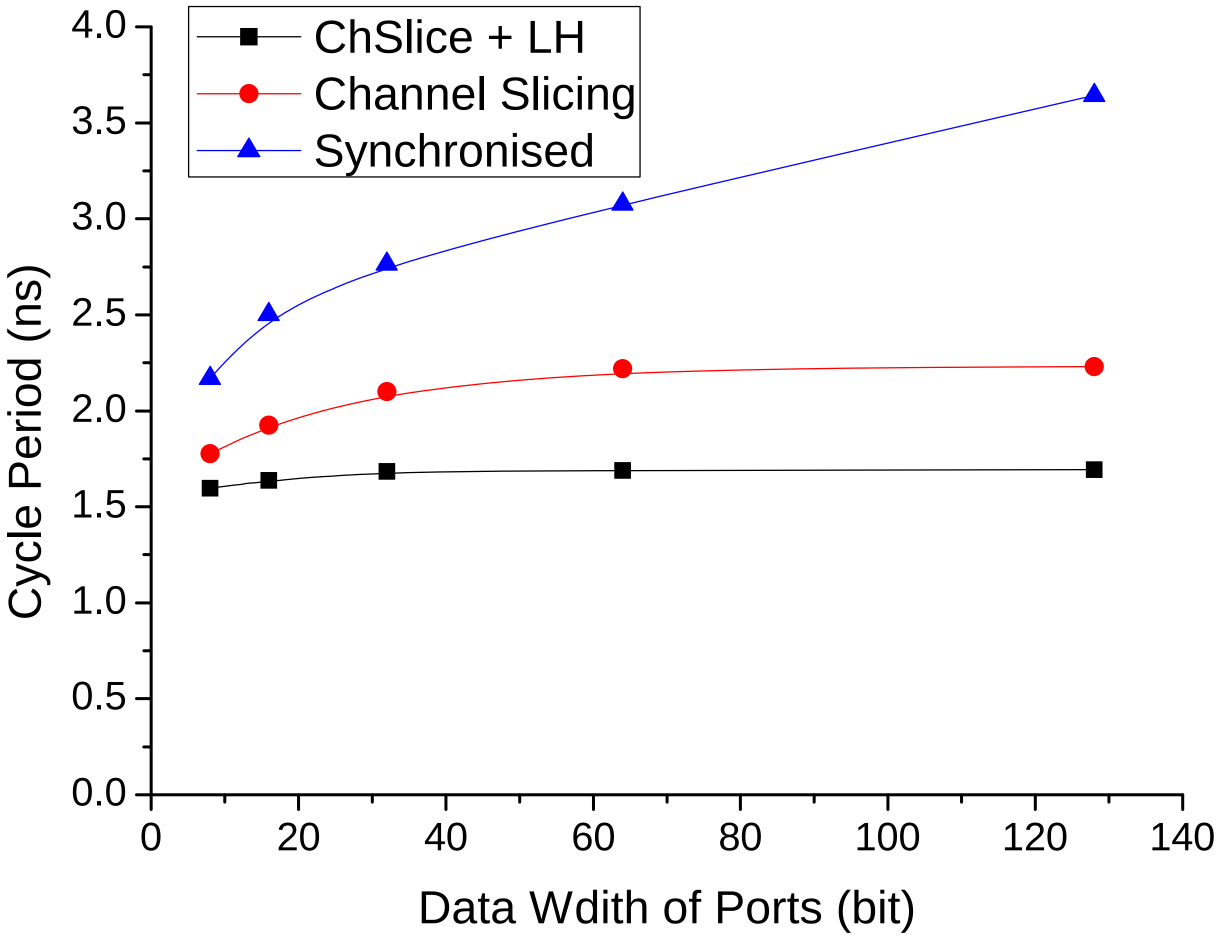
Bundled-data: less tolerant to variation

MANGO, QNoC, ASPIN

Customized design: design complexity

ASPIN

# Data Width Effect



# Index

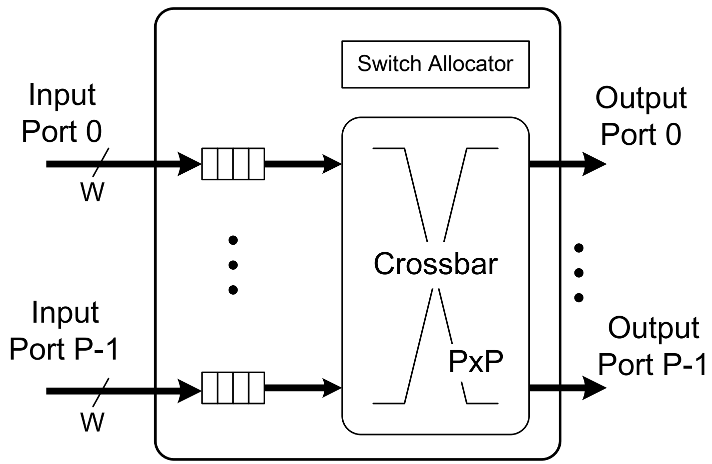
- Channel Slicing
  - Asynchronous NoCs and routers
  - Channel Slicing
  - A wormhole router design
- **Spatial Division Multiplexing (SDM)**
  - Motives
  - Switching networks
  - 2-stage Clos network
  - The distributed scheduler
  - Implementation results

# SDM: Motivation (1)

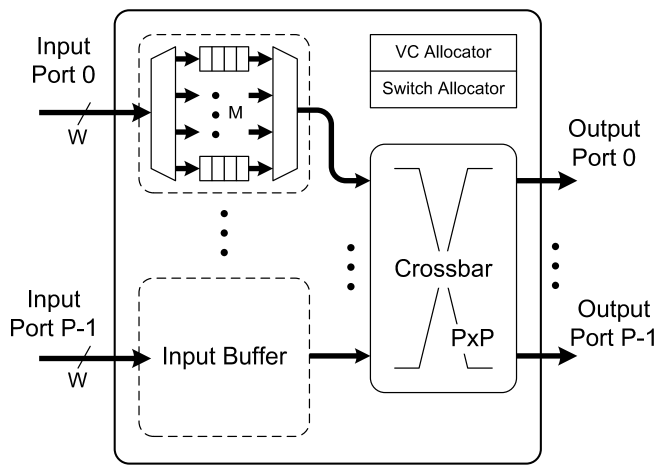
- The problems that the wormhole router cannot handle:
  - QoS, delay and throughput guaranteed services
  - Fault-tolerance
  - Network efficiency

# Motivation (2)

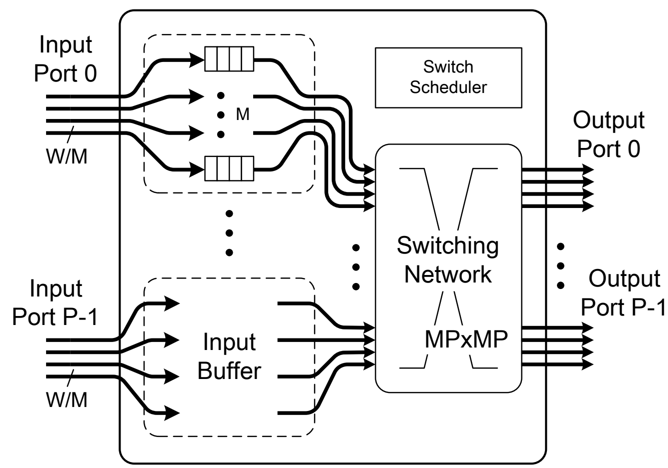
Wormhole




Virtual Channel

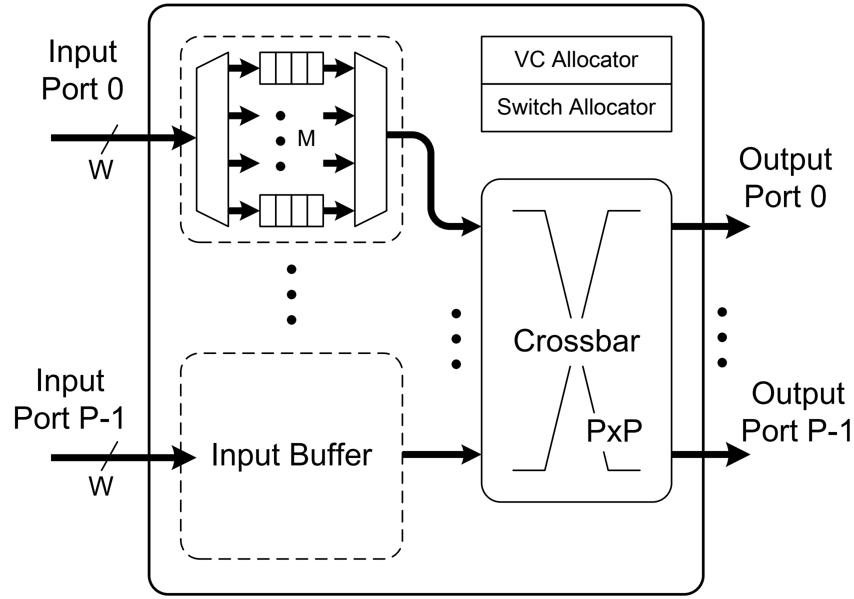



SDM



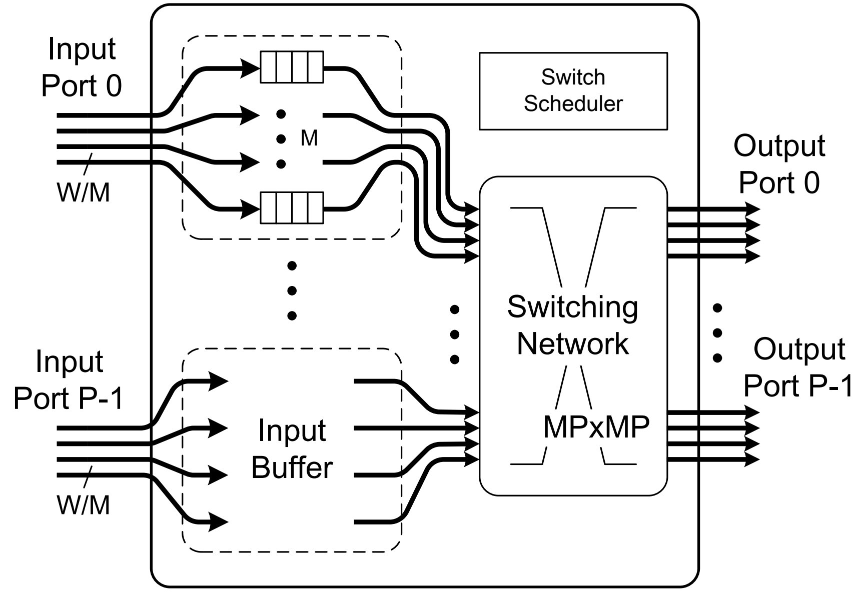
# Motivation (3) – Problems of VC

- Pipelines are synchronised
- Area overhead
- QoS (complicated arbiters)
- TDMA (time slot definition)
- Fault-tolerance (partial faulty link)



# Motivation (4) – Benefits of SDM

- Delay and throughput Guarantee
- Fault-tolerance
- Speed (Channel slicing)
- Area
- Link efficiency
  - interrupts



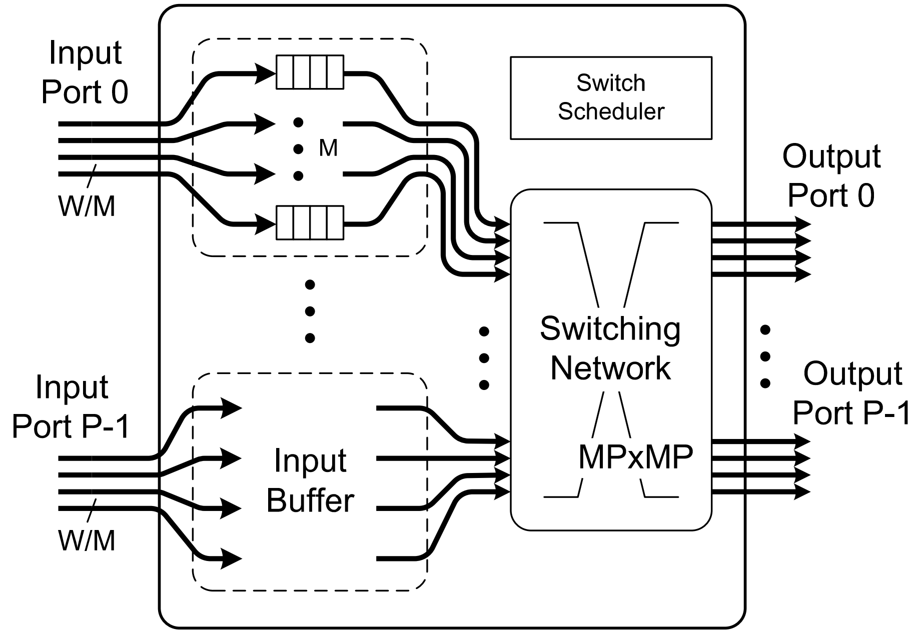
# Motivation (4) – Problems of SDM

- Area overhead

$$C_{CB} = P^2 \times W$$

$$C_{SDM} = M \times P^2 \times W$$

- Scheduling Algorithm
  - Wormhole (*P* to 1)
  - SDM (*MP* to *M*)




$$MP \times MP \times W/M = MP^2 \times W$$

# Index

- Channel Slicing
  - Asynchronous NoCs and routers
  - Channel Slicing
  - A wormhole router design
- Spatial Division Multiplexing (SDM)
  - Motives
  - **Switching networks**
  - 2-stage Clos network
  - The distributed scheduler
  - Implementation results

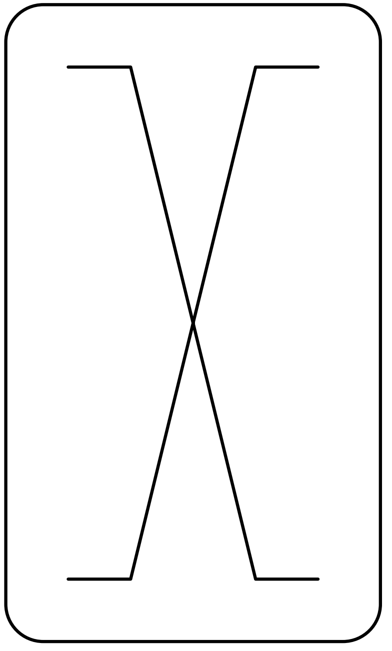
# SDM: Switching Networks

- Strict Non-Blocking (SNB)
  - An input port and an output port is always connectable
- Rearrangeable Non-Blocking (RNB)
  - An input port and an output port is connectable with possible changes on existing connections
- Blocking
  - Not all input ports and output ports are connectable under certain cases

# Crossbar

- SNB

$$C_{CB} = N^2 \times W$$



# Clos Network

SNB/RNB

C(*m*,*n*,*k*)

N = *nk*

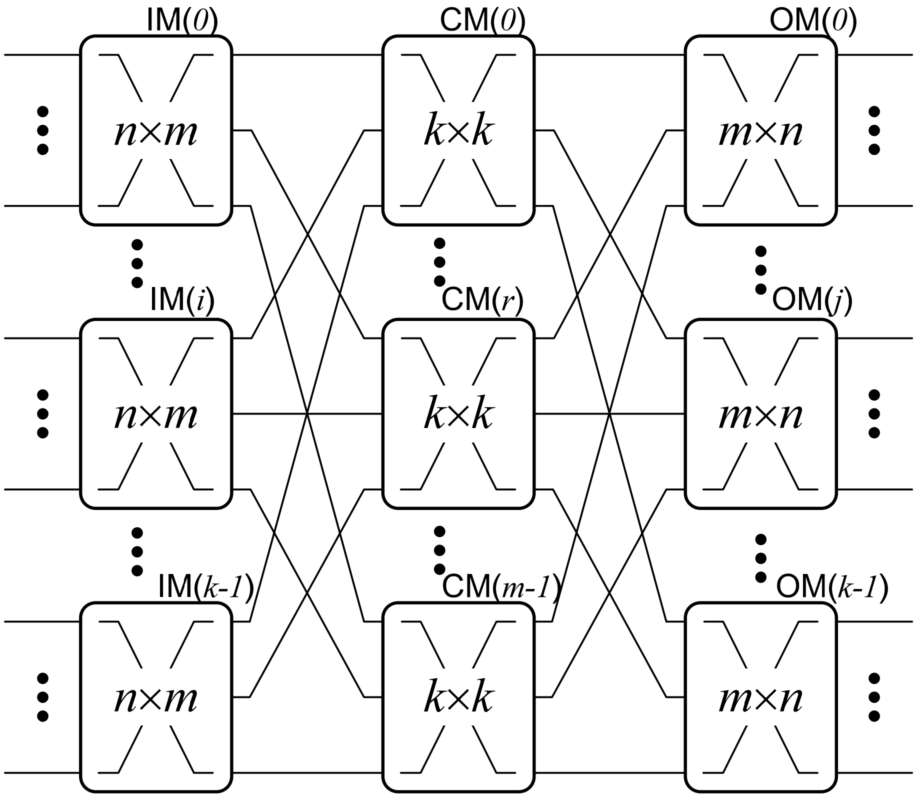
*SNB: m >= 2n-1*

*RNB: m = n*

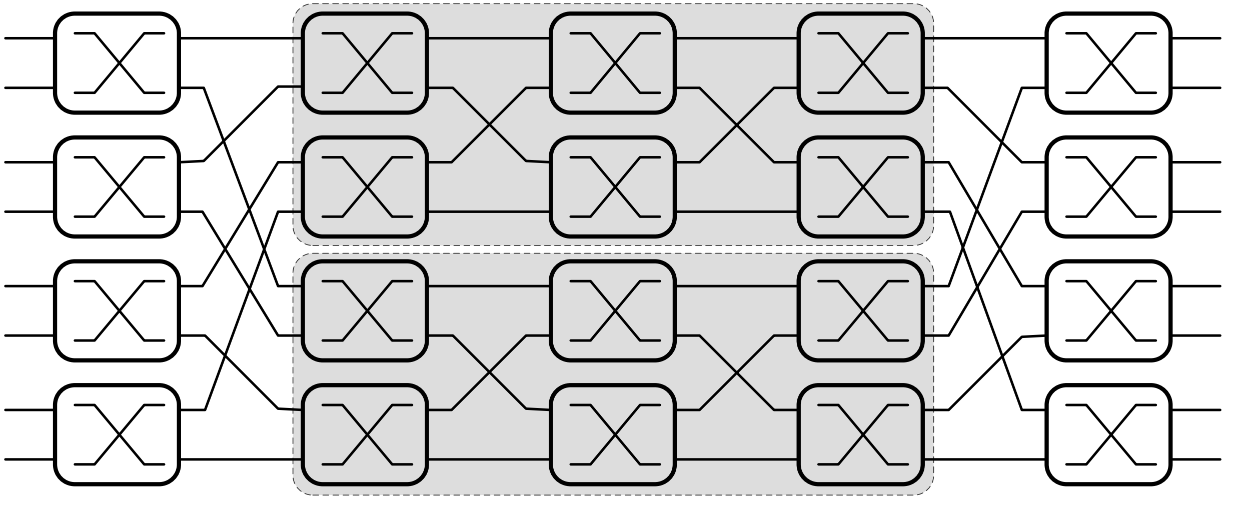
$$k = \sqrt{2N}$$

$$C_{Clos,SNB} \geq \left[2(2N)^{1.5} - 4N\right] \times W$$

$$C_{Clos,RNB} \geq (2N)^{1.5} \times W$$



# Benes Network

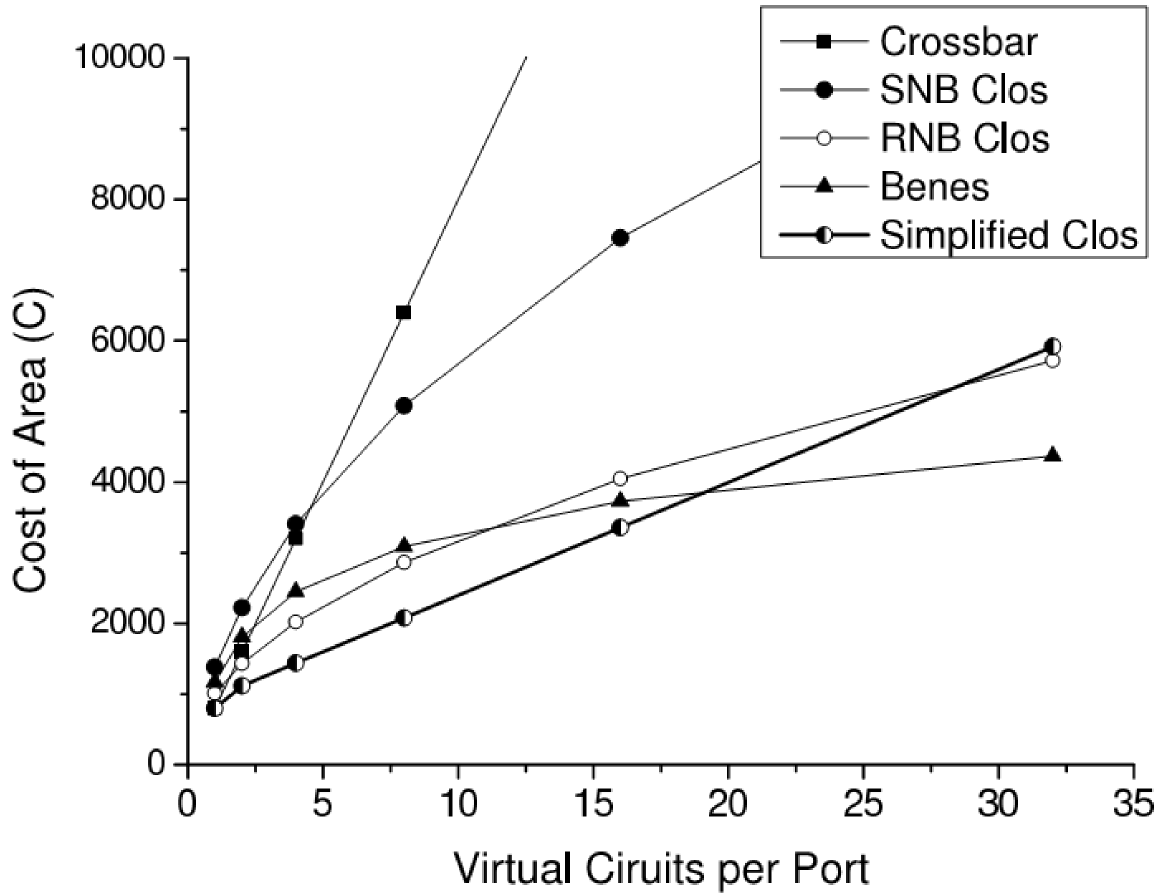


Multi-stage Clos C(2,2,4) + 2C(2,2,2)

SNB

$$C_{Benes} = (4N\log_2 N - 2N) \times W$$

# Area of Switching Networks



# Problems of all Switching Networks

- Crossbar
  - Area ~ $N^2$
  - Easy to schedule
- Clos
  - Area ~ $N^{1.5}$
  - Difficult but possible to schedule by hardware
  - Optimal area is reached when $k = \sqrt{2N}$
- Benes
  - Area ~ $N \log N$
  - Impossible to schedule by hardware (microprocessor)
  - Optimal area is reached when $N=2^n$

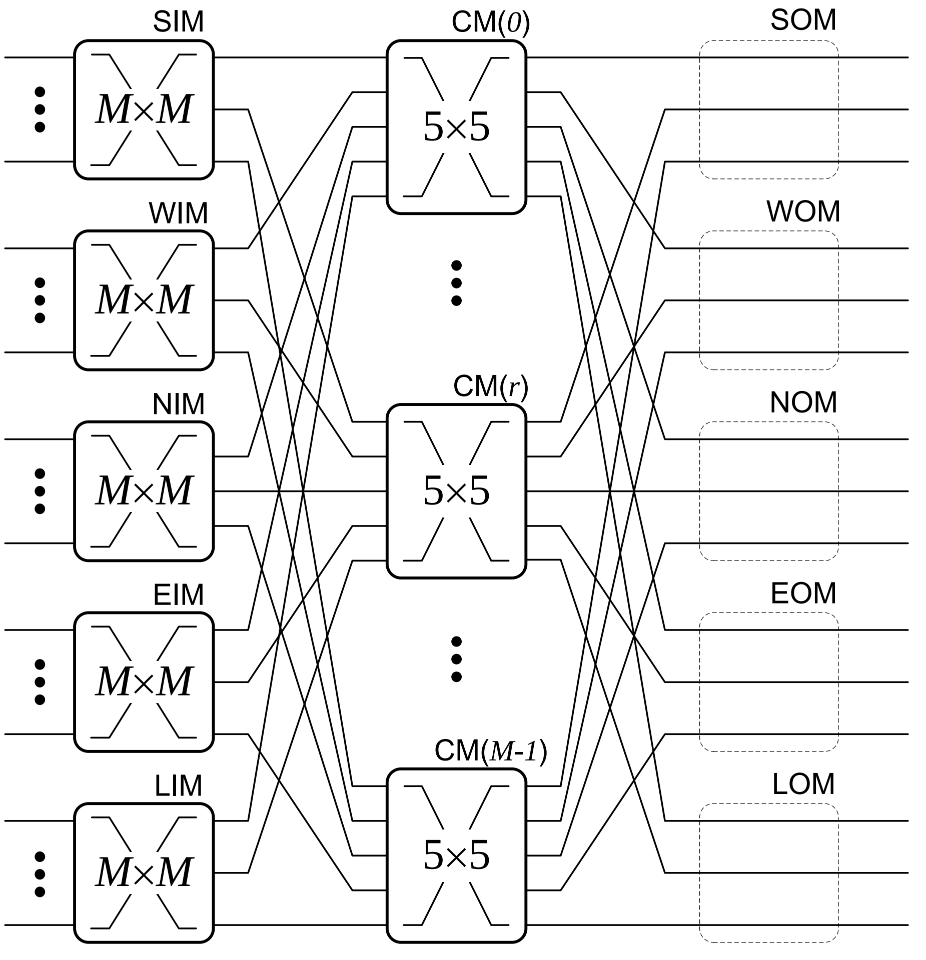
MANCHESTER 1824

The University of Manchester

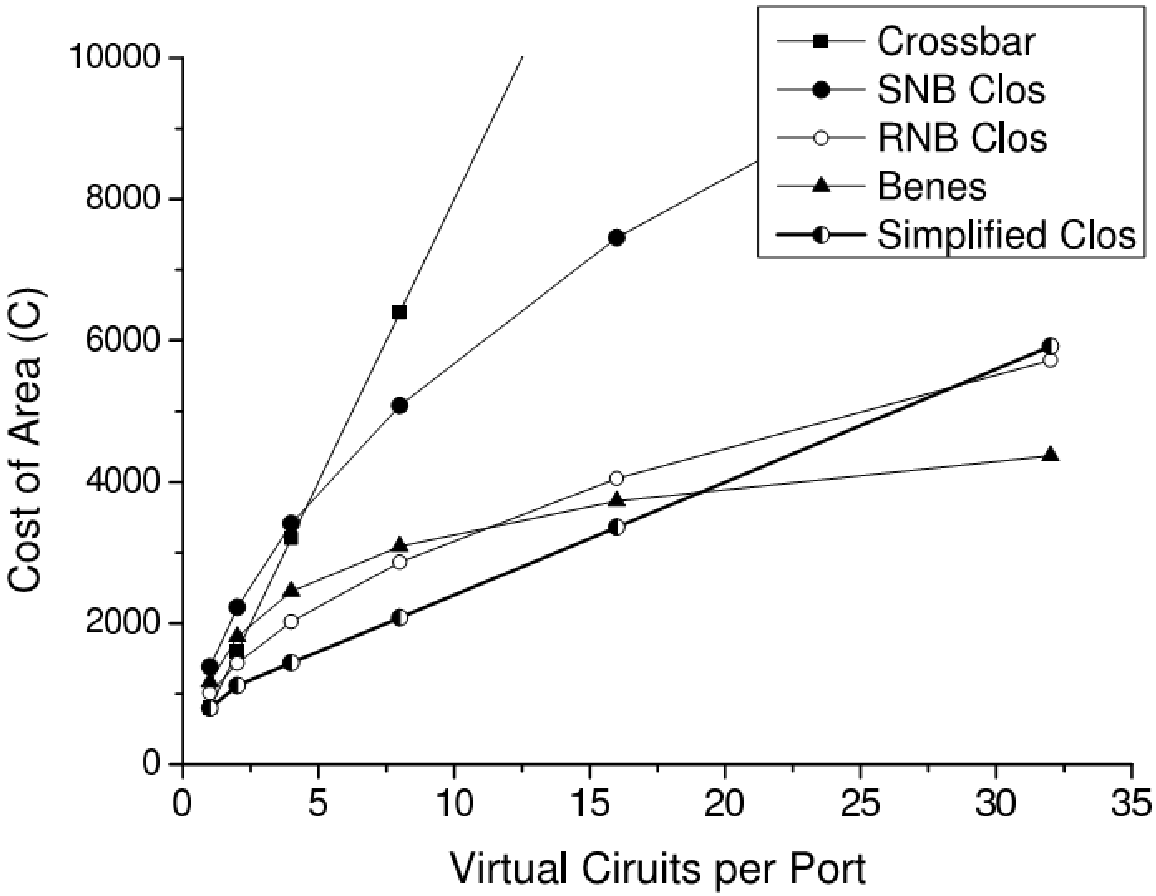
# Index

- Channel Slicing, the wormhole router
- Spatial Division Multiplexing (SDM)
  - Motives
  - Switching networks
  - **2-stage Clos network**
  - The distributed scheduler
  - Implementation results

# SDM: 2-stage Clos Network

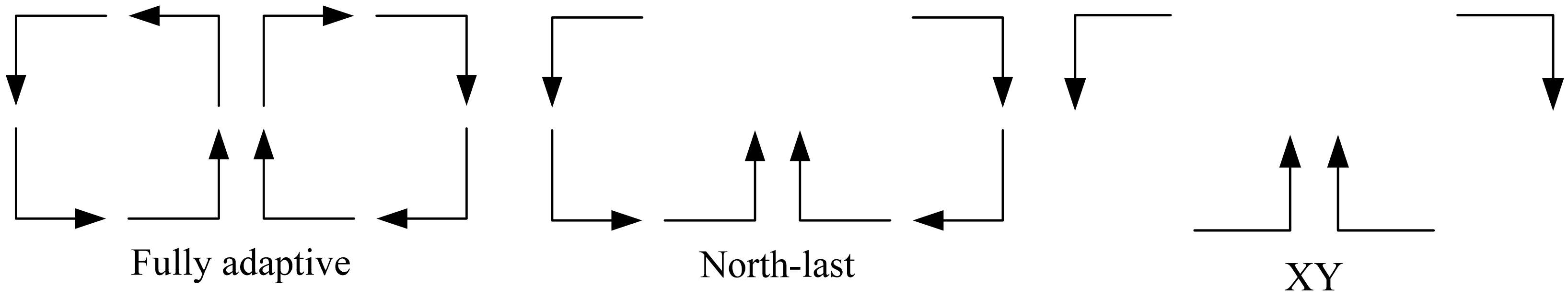


# Area Comparison



# Benefits of the 2-stage Clos Network

- Minimal area when *M* <= 16
- Only have 2-stages, latency is reduced
- Latency bounded
- Scheduling algorithm is also simplified
- The CMs could be further reduced




- ***It is a RNB network. An SNB network requires 3 stages***

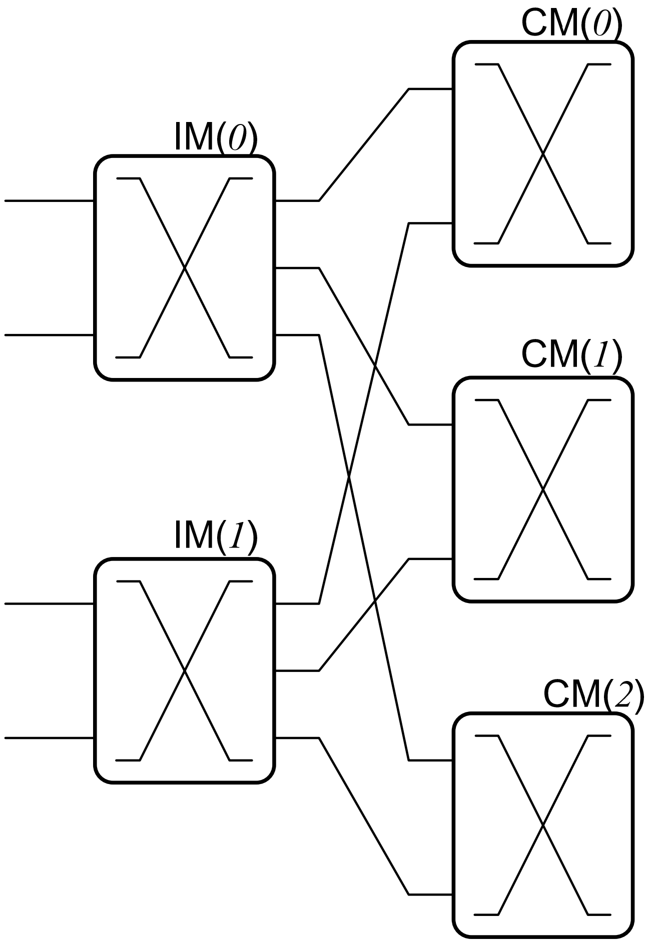
# Index

- Channel Slicing, the wormhole router
- Spatial Division Multiplexing (SDM)
  - Motives
  - Switching networks
  - 2-stage Clos network
  - **The distributed scheduler**
  - Implementation results

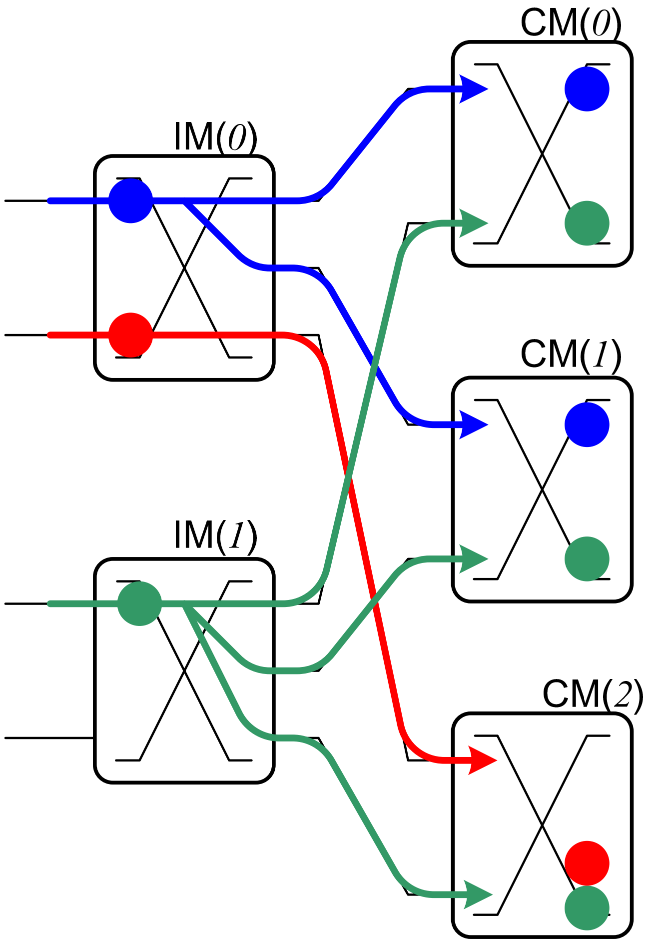
# SDM: Scheduling Algorithms

- *Optimized algorithms*
  - Always reach the optimal configuration that every possible connection is configured
  - Time complexity O($N^2$)
  - Normally software based ( [Leroy 2008] microprocessor, 64 ports, 50us)
- *Heuristic algorithms*
  - Capable of configuring part of the possible connections with less time and area
  - Time complexity O($N$) ~ O(log$N$)
  - Normally hardware implementable, distributed, and scalable

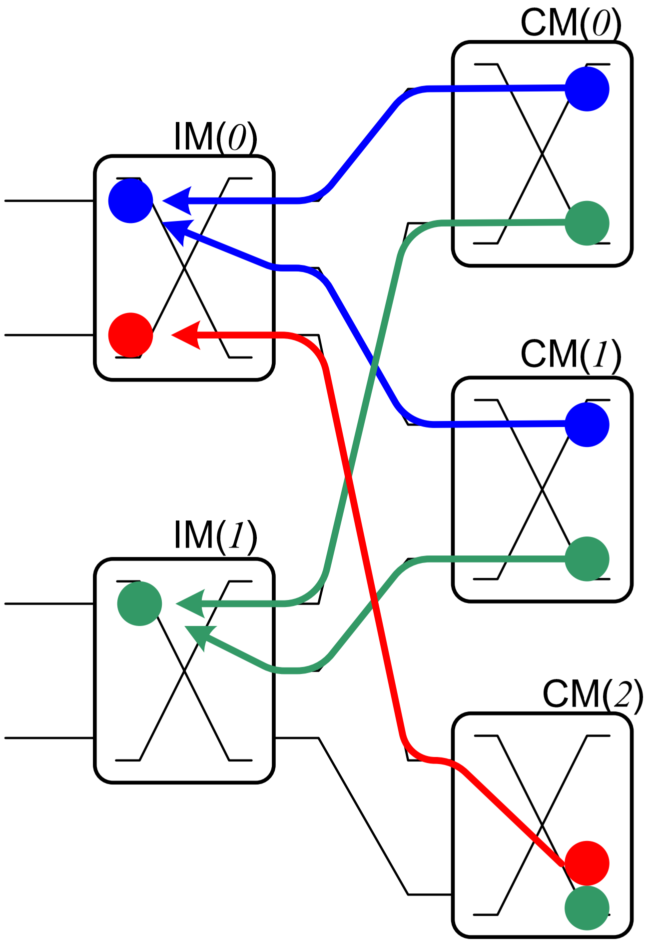
# Synchronous Dispatch Algs.



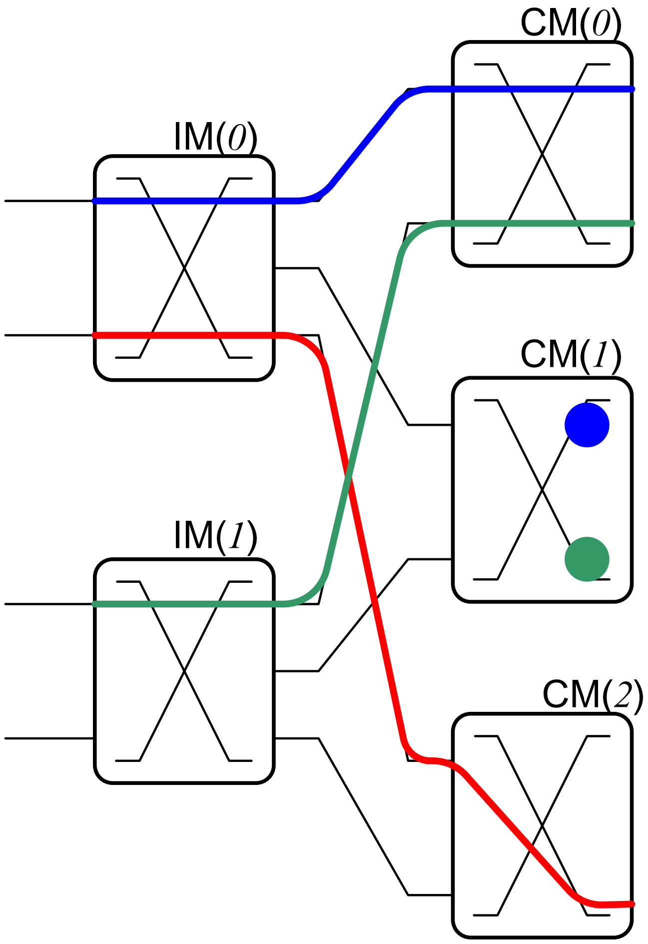
# Synchronous Dispatch Algs.



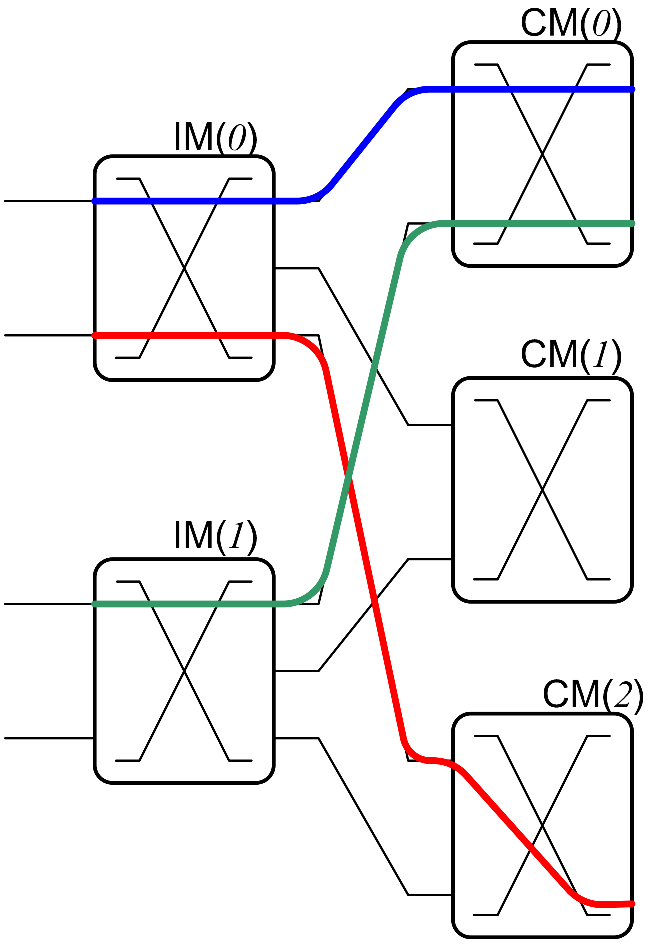
# Synchronous Dispatch Algs.



# Synchronous Dispatch Algs.



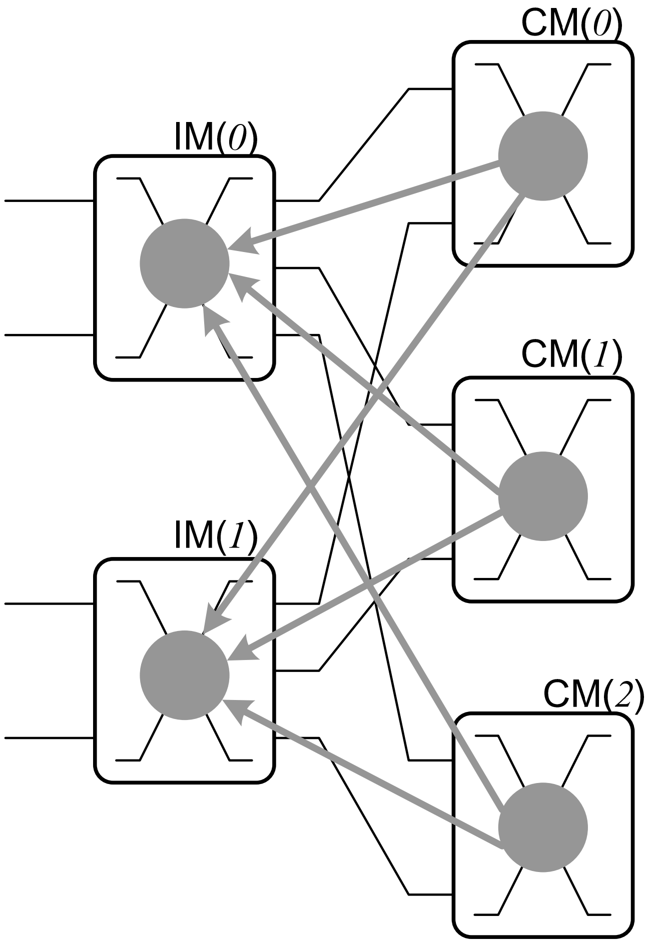
# Synchronous Dispatch Algs.



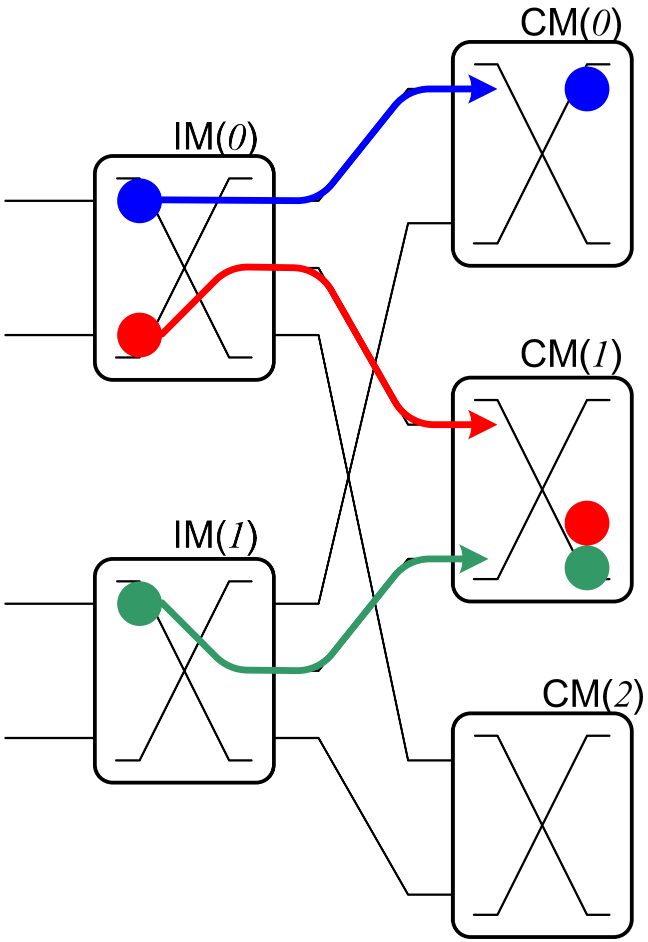
# Problems. Of Sync Algs.

- Iterations are synchronised.
- The requests from IMs are blind and greedy.
- CMs are blind and greedy too.
- Multiple requests are sent out by IMs

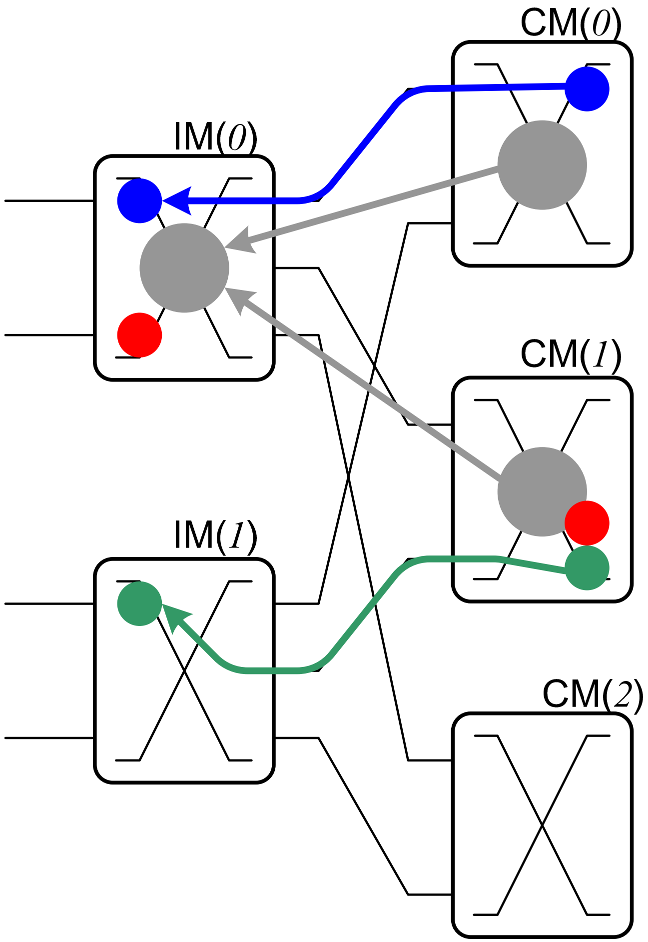
# Asynchronous Scheduling Alg.



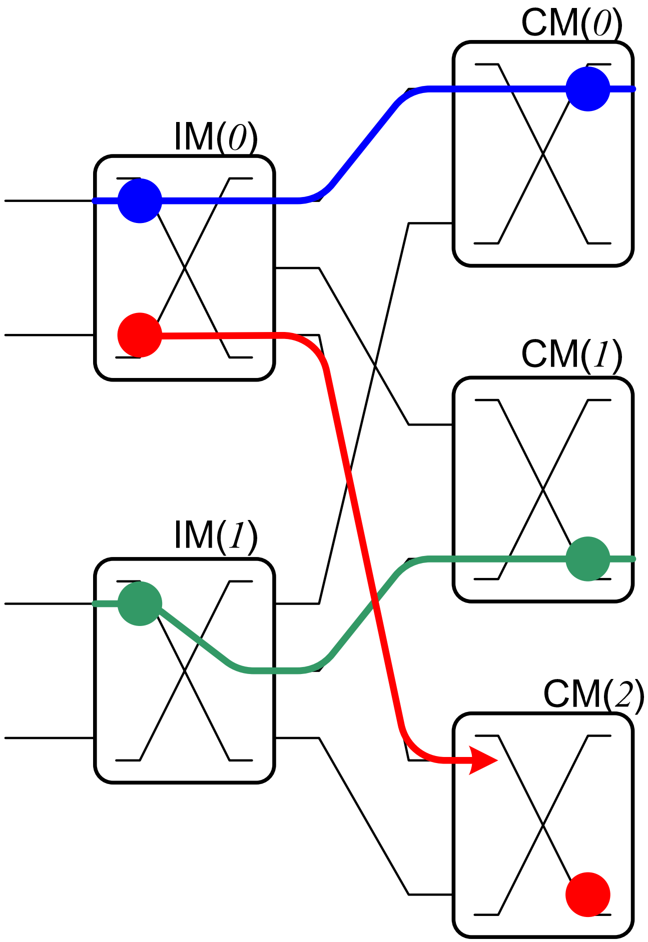
# Asynchronous Scheduling Alg.



# Asynchronous Scheduling Alg.



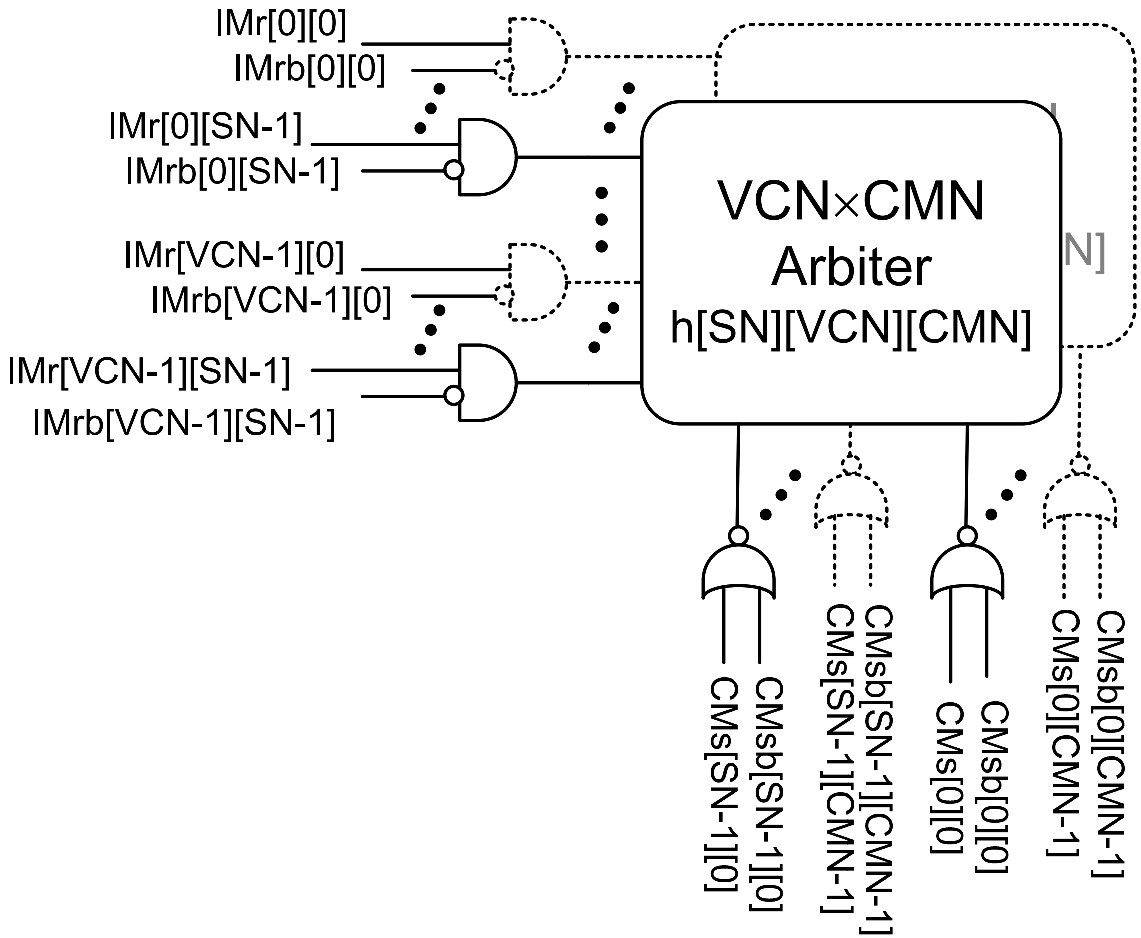
# Asynchronous Scheduling Alg.



# Asynchronous Scheduling Alg.

- IM scheduler and CM schedulers are independent
- The scheduling algorithm can support arbitrary number of CMs
- Less transition rate than synchronous schedulers

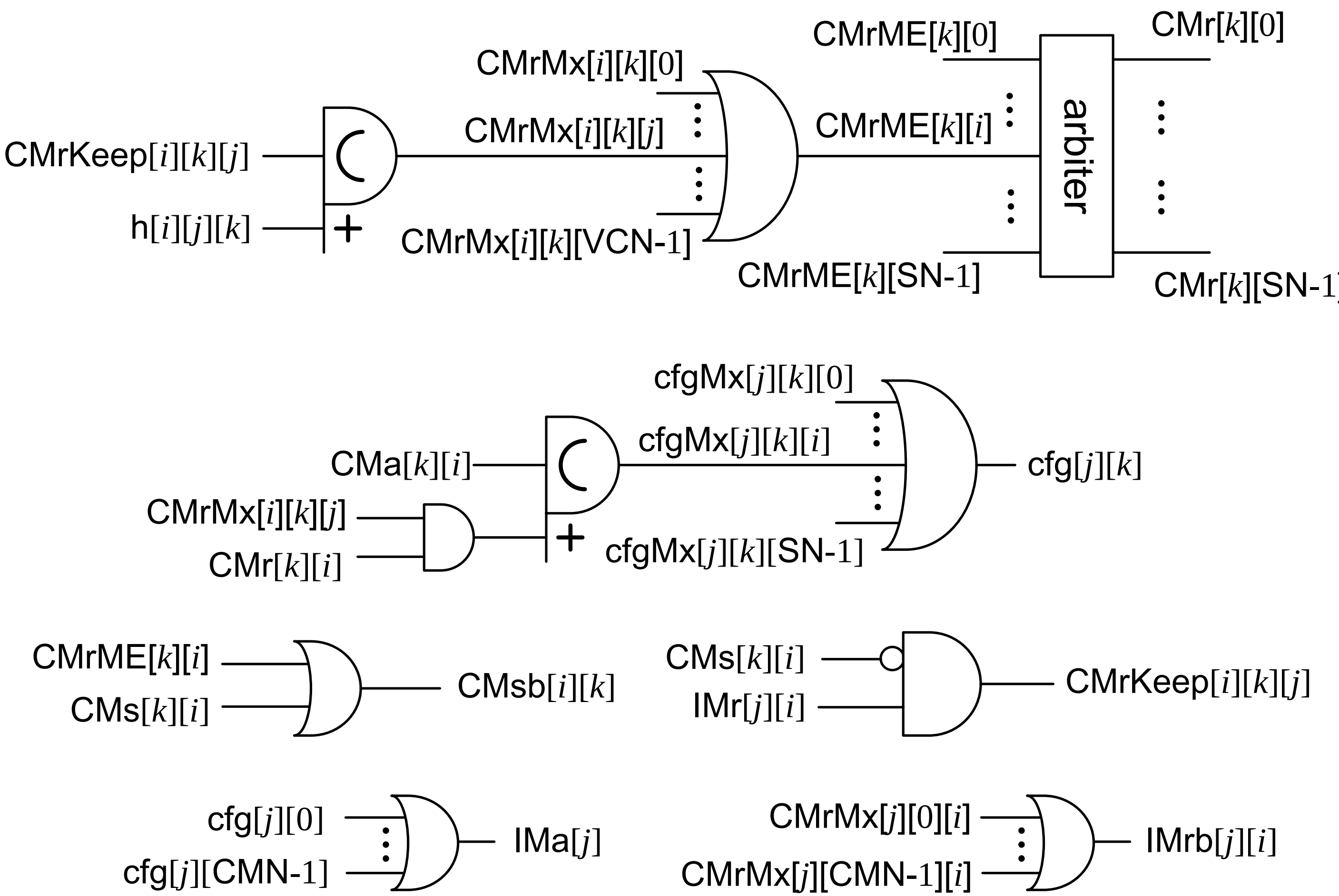
# IM scheduler (1)



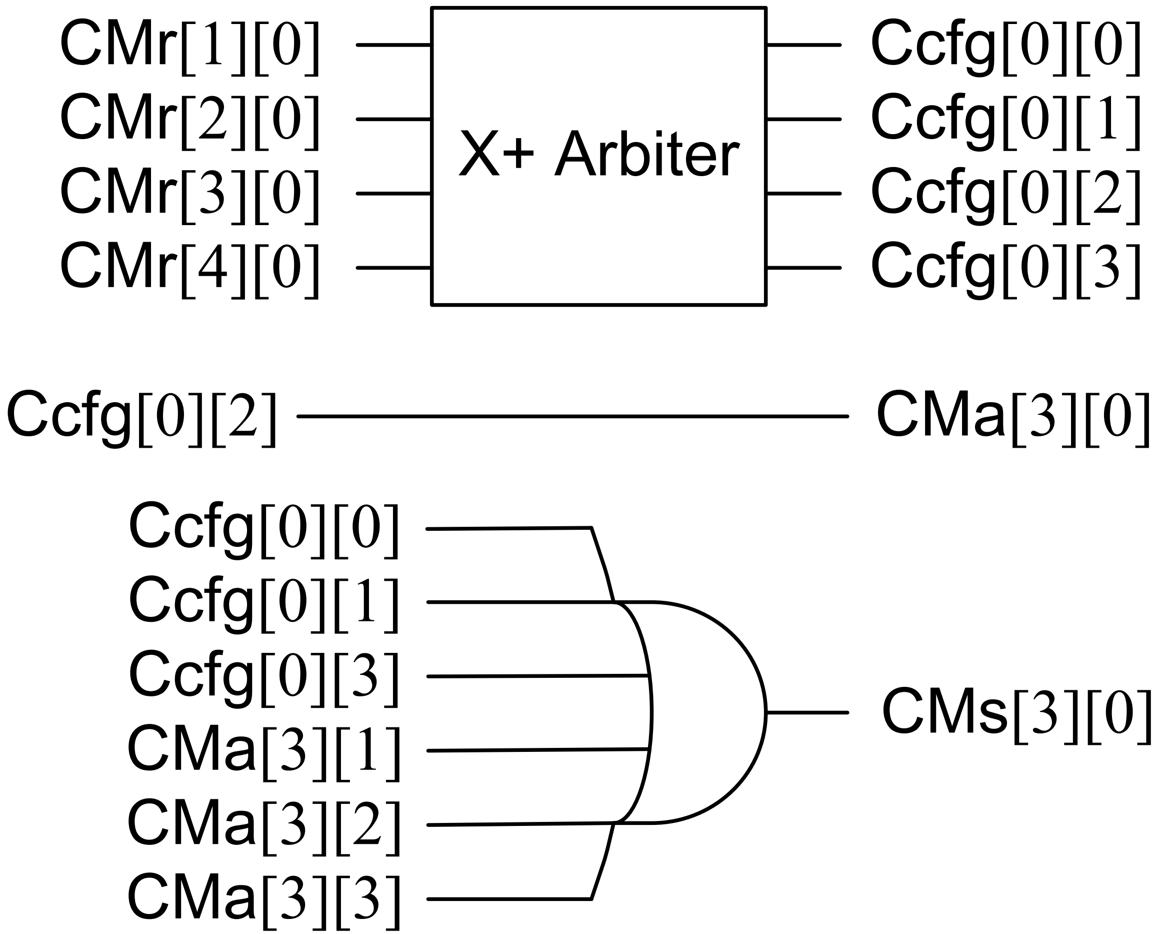
MANCHESTER 1824

The University of Manchester

# IM scheduler (2)



# CM scheduler



# Index

- Channel Slicing, the wormhole router
- Spatial Division Multiplexing (SDM)
  - Motives
  - Switching networks
  - 2-stage Clos network
  - The distributed scheduler
  - **Implementation results**

# SDM: implementation (1)

- Faraday 130nm
- Wormhole, SDM crossbar, and SDM Clos
- 64-bit ports, 4 virtual circuits/port
- Design Compiler synthesized
- System Verilog for testbench
- Switches are back-annotated with latency from synthesis

# SDM: implementation (2)

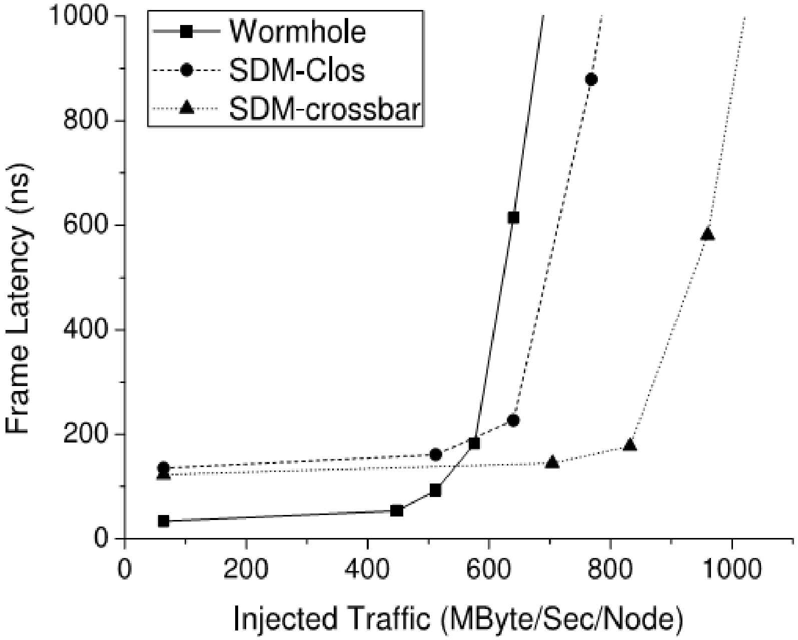
Table 1. Area of Routers

| Block | Wormhole | SDM Crossbar | 2-stage Clos |
|---|---|---|---|
| switching | 6.7K | 28.1K | 16.1K |
| scheduler | 0.4K | 8.6K | 11.0K |
| buffer | 10.1K | 12.0K | 11.7K |
| Total | 17.2K | 48.7K | 38.8K |

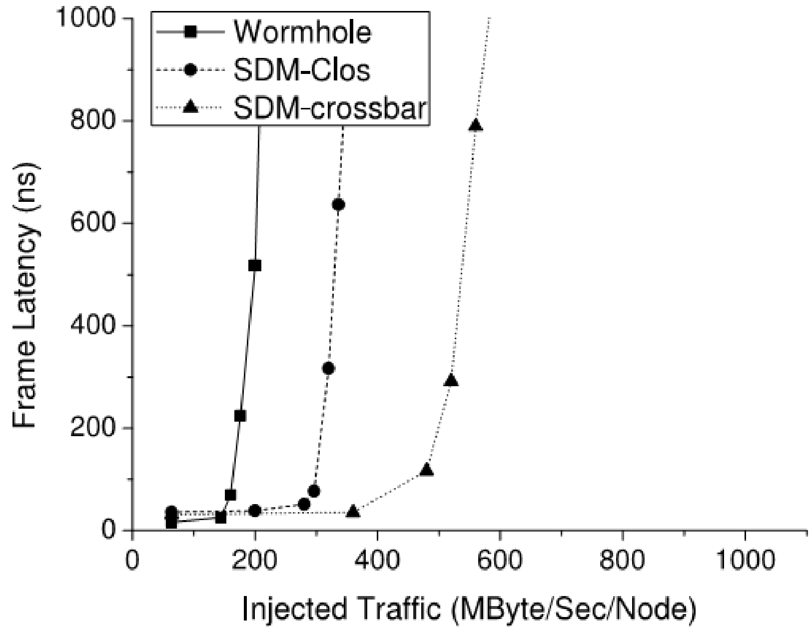
Table 2. Speed of Routers

| | Wormhole | SDM Crossbar | 2-stage Clos |
|---|---|---|---|
| switch delay | 0.23 ns | 0.41 ns | 0.53 ns |
| scheduler | 0.4 ns | 2.2 ns | 3.1 ns |
| router period | 2.4 ns | 3.4 ns | 3.6 ns |
| router delay | 1.2 ns | 1.7 ns | 1.9 ns |

# Network Performance (1)

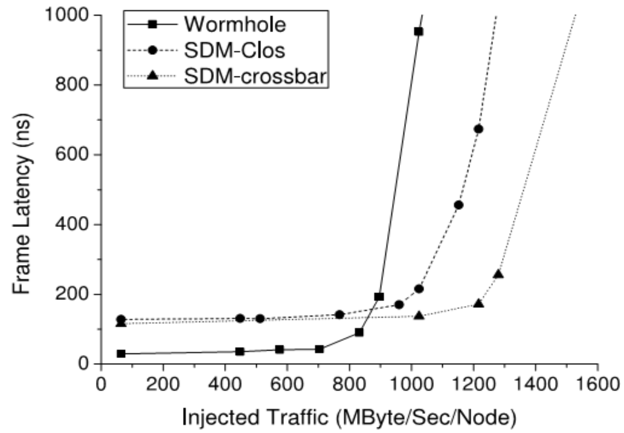



(b) Uniform, $len = 64$

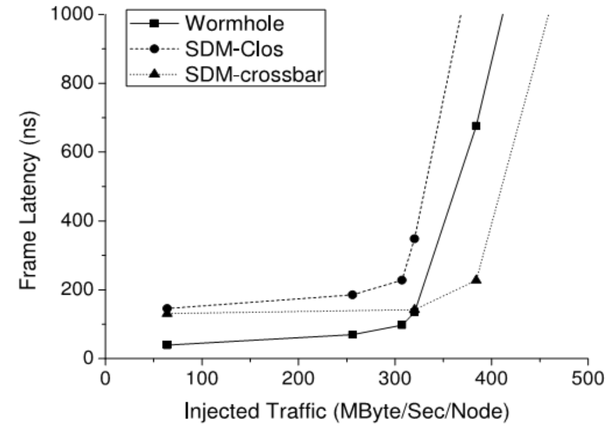



(e) Uniform, $len = 8$

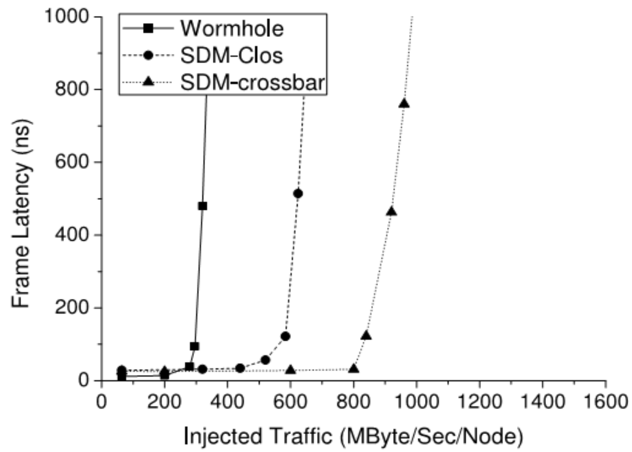
# Network Performance (2)



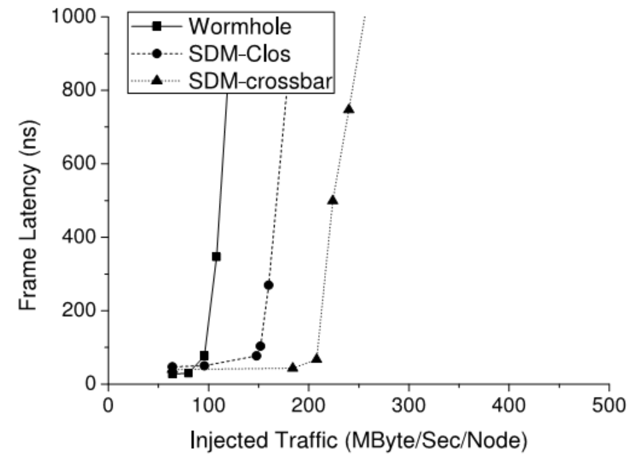

(a) Local ($P \leq 3$), $len = 64$




(c) Non-local ($P \geq 6$), $len = 64$




(d) Local ($P \leq 3$), $len = 8$




(f) Non-local ($P \geq 6$), $len = 8$

# Conclusion of Results

- SDM outperforms Wormhole with short frames and local traffic
- The connection loss from SNB to RNB is significant
- SDM is good at GT traffic, this work is the first step to a QoS router
- How to configure the SDM to settle GT paths is the next problem.

# References

- Channel Slicing:
  - *ASP-DAC* 2010.
  - *UK Async Forum*, 2009.
  - *International Symposium on SOC*, 2009.
- SDM
  - In submission to *ASYNC* 2010.

# Questions?