Lecture 52

NPTEL

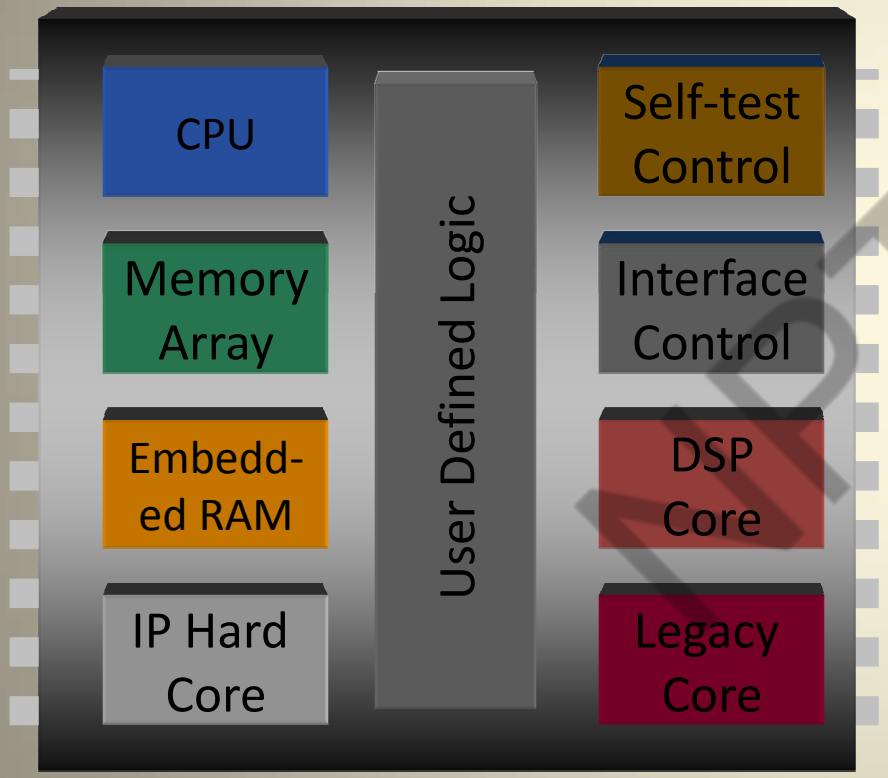
Testing Embedded Cores in NoC

- Reuse of On-Chip Network for Testing
- Test Scheduling
- Test Access Methods and Test Interface
- Efficient Reuse of Network
- Power-Aware and Thermal-Aware Testing

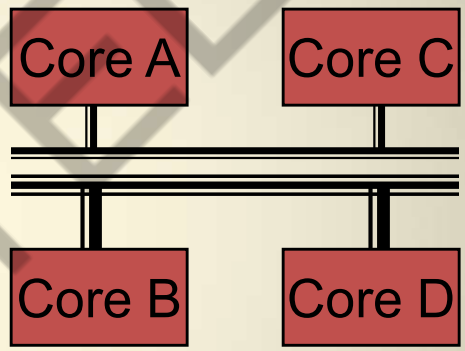
NPTEL

Network-on-Chip

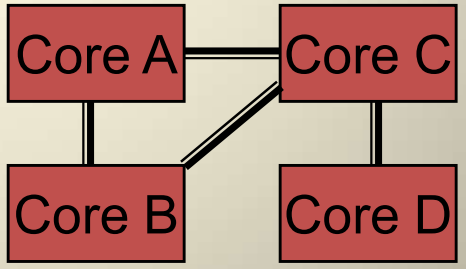
Current Design Methodology: System-on-Chip (SoC)



Interconnection schemes:



Shared bus



Dedicated connection

Need for Network-on-Chip (NOC)

Current Design Methodology: System-on-Chip (SoC)

Design

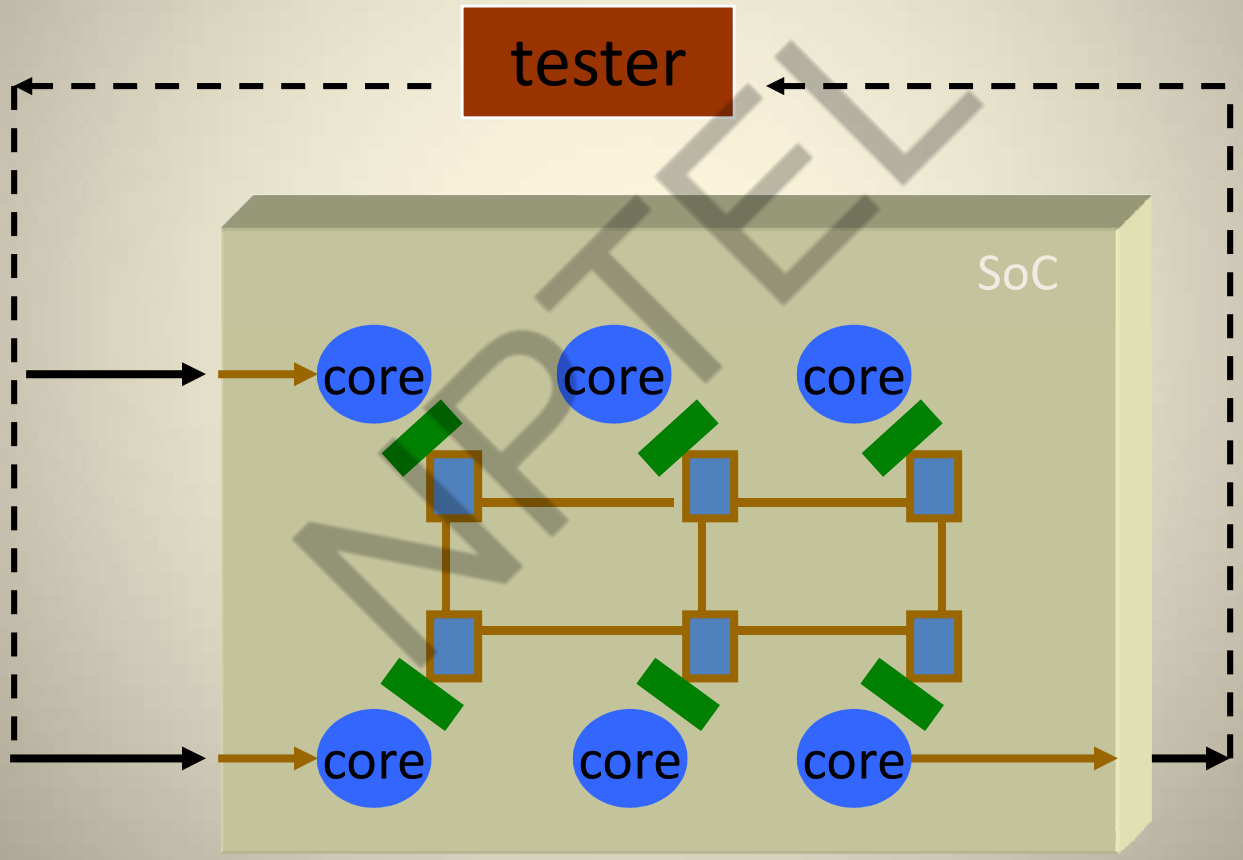
- Communication infrastructure is becoming new bottleneck
  - Wire delay
  - Signal integrity
  - Power dissipation
  - Area vs. speed
- New interconnection schemes needed.

Test

- Test of SoC has been well understood
  - TAM, wrapper
  - Test scheduling
  - IEEE 1500
- Test needs dedicated hardware
- Hardware for mission-mode communication can not be reused for testing

NPTEL

NOC-based System



NPTEL

NOC-based System

Possible next-generation SoC paradigm: Network-on-Chip (NoC)

Design

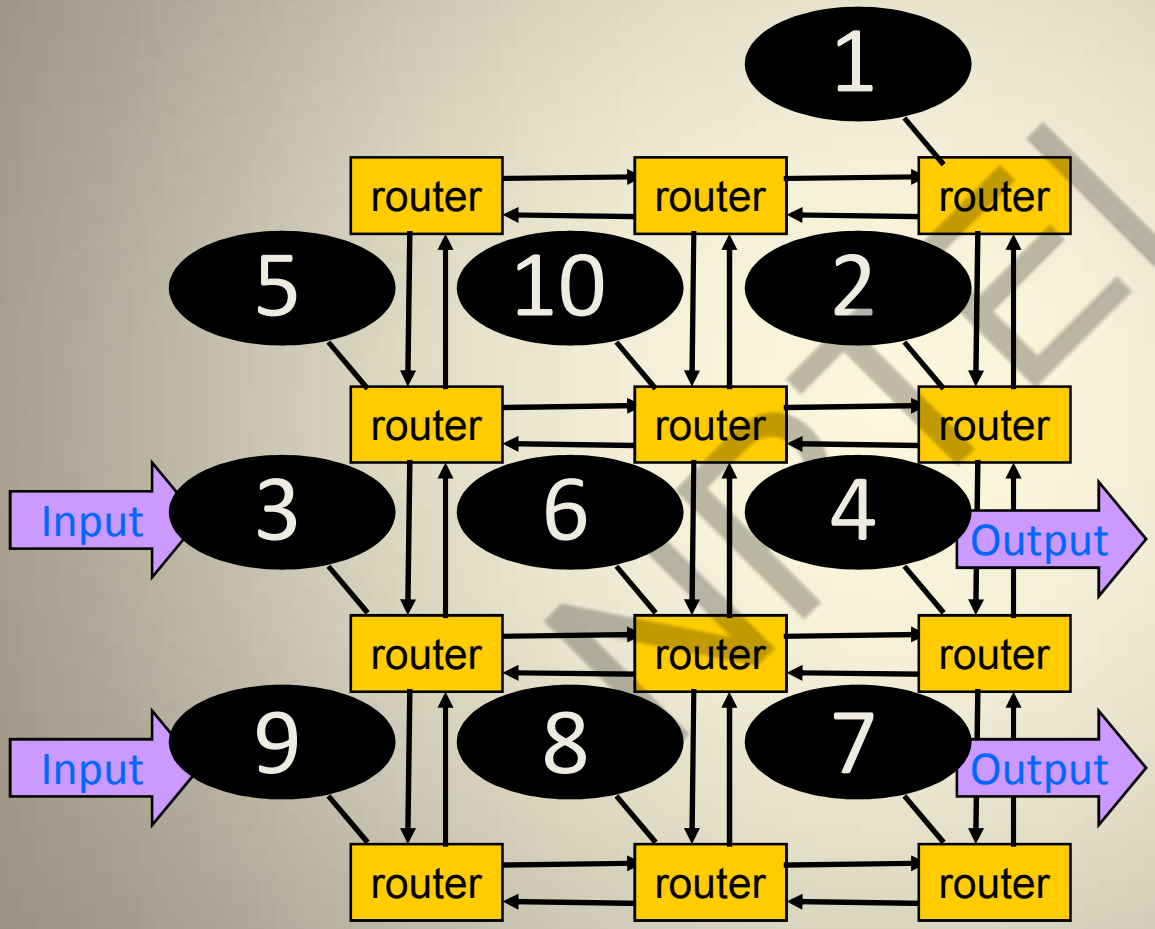
- High performance
  - High bandwidth
  - Low signal delay
- Reasonable overhead
- Suitable for large number of cores
- Network design is versatile
- Methodology of next generation VLSI design

Test

- Test of NoC has not received much attention
  - Core testing
  - Router and interconnection testing
  - Test wrapper design
  - Test scheduling
- No need for dedicated TAMs
- Network can be reused for testing

NPTEL

NoC-based System

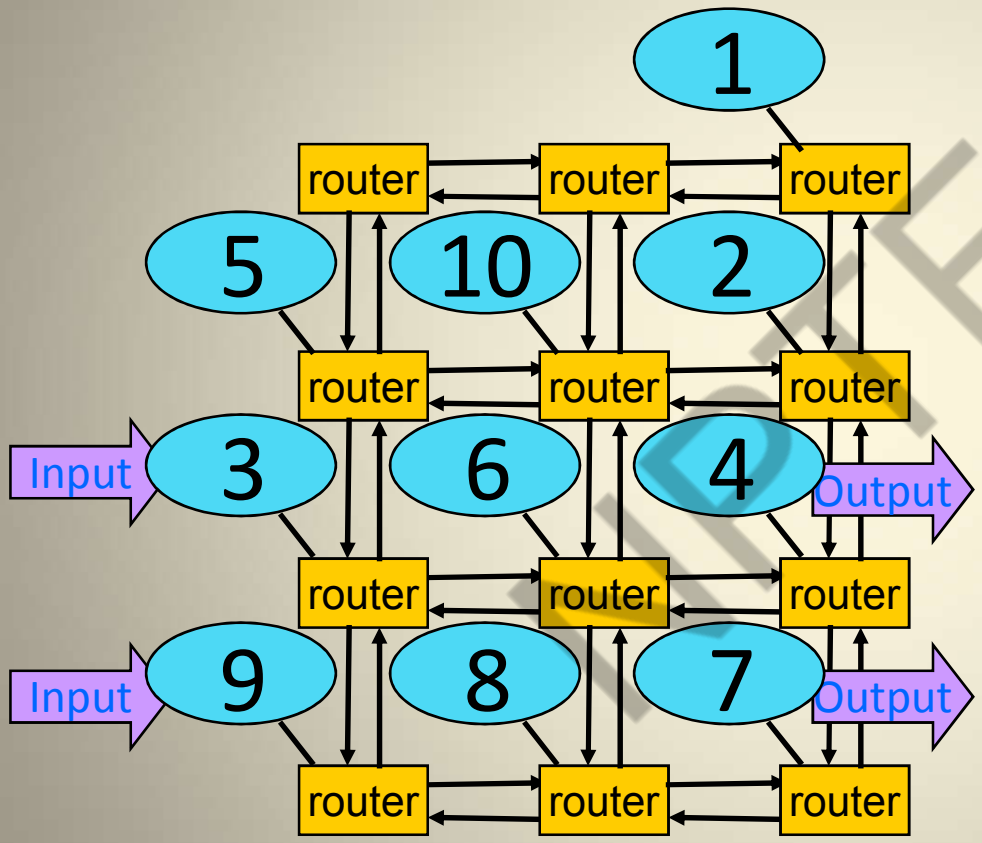


d695 from ITC'02 benchmark

- Packet-switching
- Bidirectional channel
- 2-D mesh, XY routing

- Channels, routers used as TAM
- Input/output ports associated with cores
- Ports, channels are assigned a time tag

Test Scheduling Using Dedicated Routing Path: Non-preemptive



- Each core is associated with a routing path
- All resources are reserved until test completed
- Test pipeline maintained
- No complex logic
- Similar to a circuit switching
- Efficiently assign I/Os and channels to core

Lecture 53

NPTEL

Test Scheduling: Problem Formulation

How to assign I/Os and channels to each core for testing such that the overall test time is minimized?

In an NoC system using dedicated routing path, given N_C cores, N_I inputs, N_O outputs, routing algorithm and the network topology, determine an assignment of cores to input/output pairs and a schedule such that the total test time is minimized.

- Equivalent to the resource-constrained multi-processor scheduling problem
- If the number of input/output pairs ≥ 2, NP-complete

NPTEL

Test Scheduling: Optimal Solution Using ILP

- Problem can be solved exactly using an ILP model
- Large number of none-zero constraints
- CPU time is prohibitive

- Can be simplified using enumeration
- Enumerate the assignment of cores to I/O pairs
- Number of constraints reduced
- A few seconds for small instances with smaller number of I/Os

- For large instances, or larger number of I/Os, CPU time is still prohibitively high
- Not suitable for large systems

NPTEL

Test Scheduling: Heuristic Algorithm

- Sort cores and I/O pairs in decreasing order of testing time
- Permute cores and I/O pairs
- Assign cores with higher priority to free I/O pairs
- Check resource conflicts using time tag: I/Os, channels, cores
- Complexity: $O(N_C^M)$
- CPU time: a few minutes for all benchmarks

NPTEL

Test Access Method and Test Interface

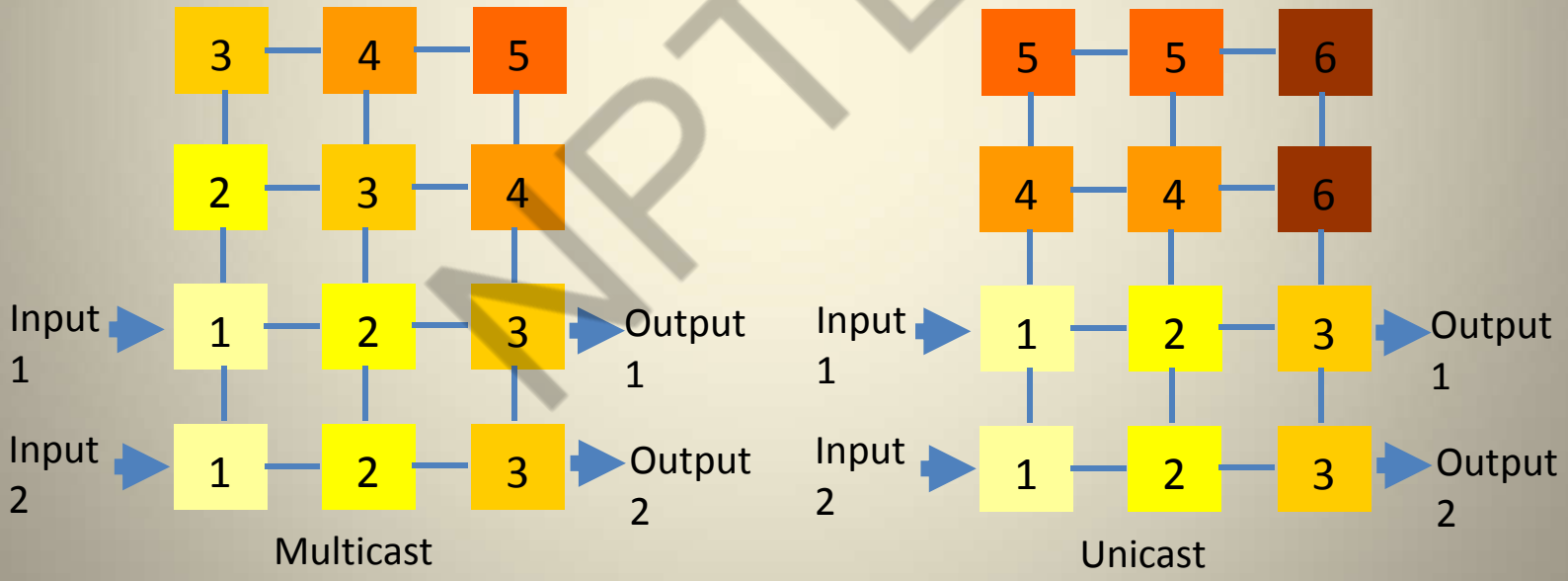
Problems targeted:

- Test access scheme for testing routers at NoC level
- Possible hardware overhead
- Efficient test scheduling that can handle both routers and embedded functional cores

NPTEL

Test Access Method

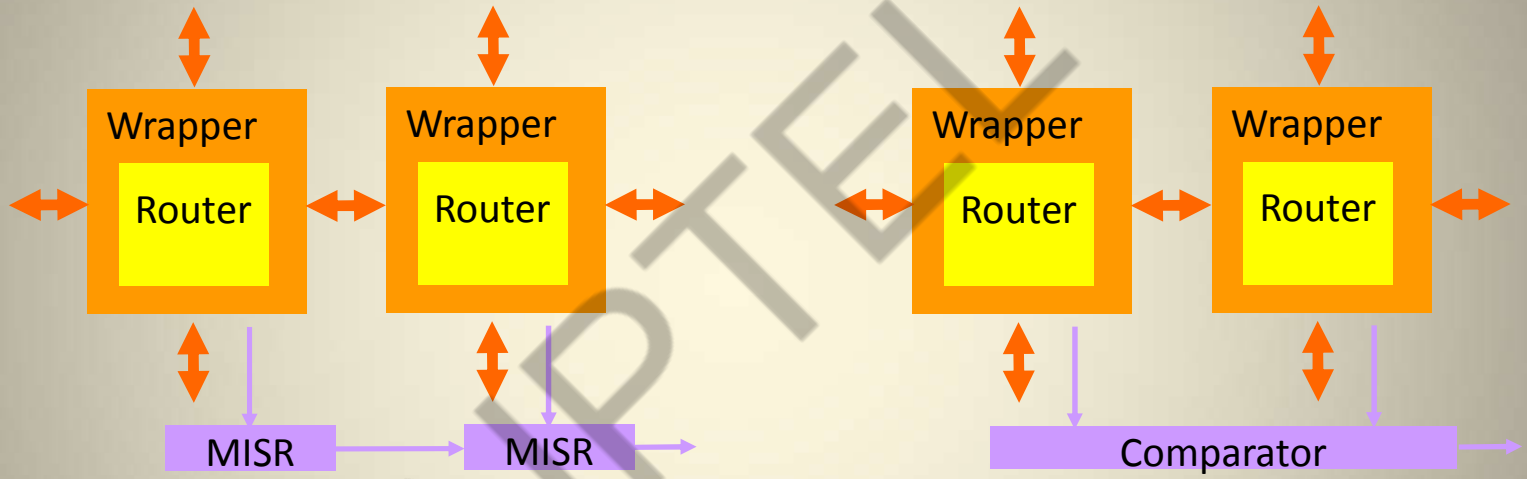
Reuse the network resources for test access
Test data and test control delivered in packet
Responses can be processed by ATE or on-chip



NPTEL

Test Responses

Can be handled on-chip



NPTEL

Minimum overhead
Probability of aliasing

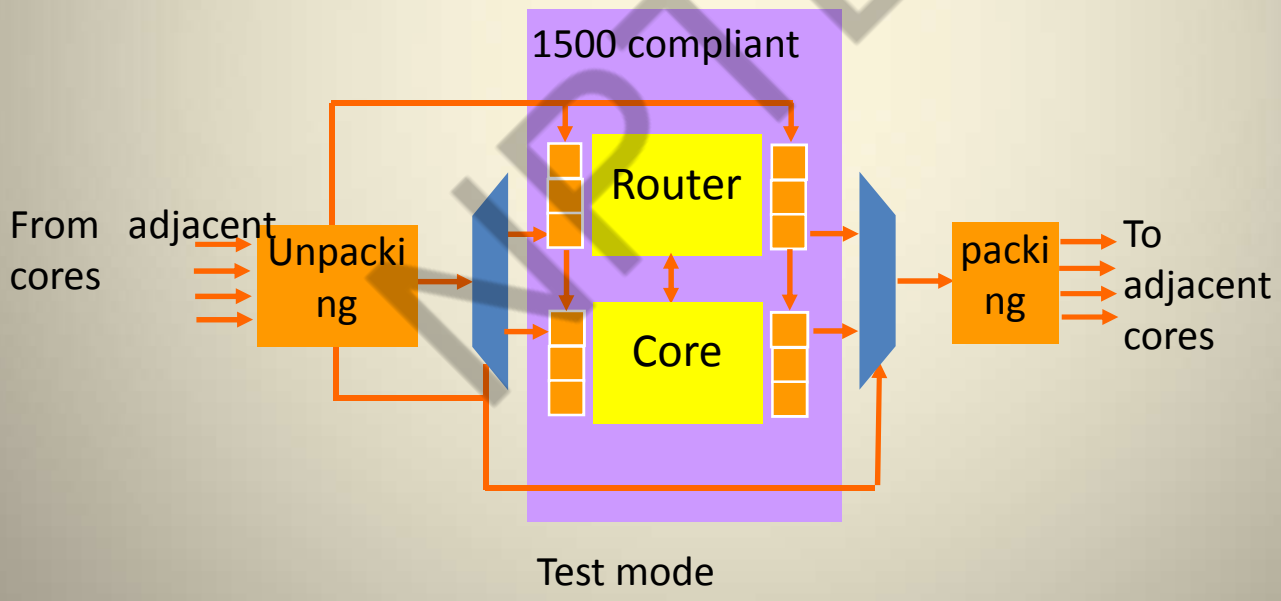
Similar to prior work
Faster, with aliasing

Lecture 54

NPTEL

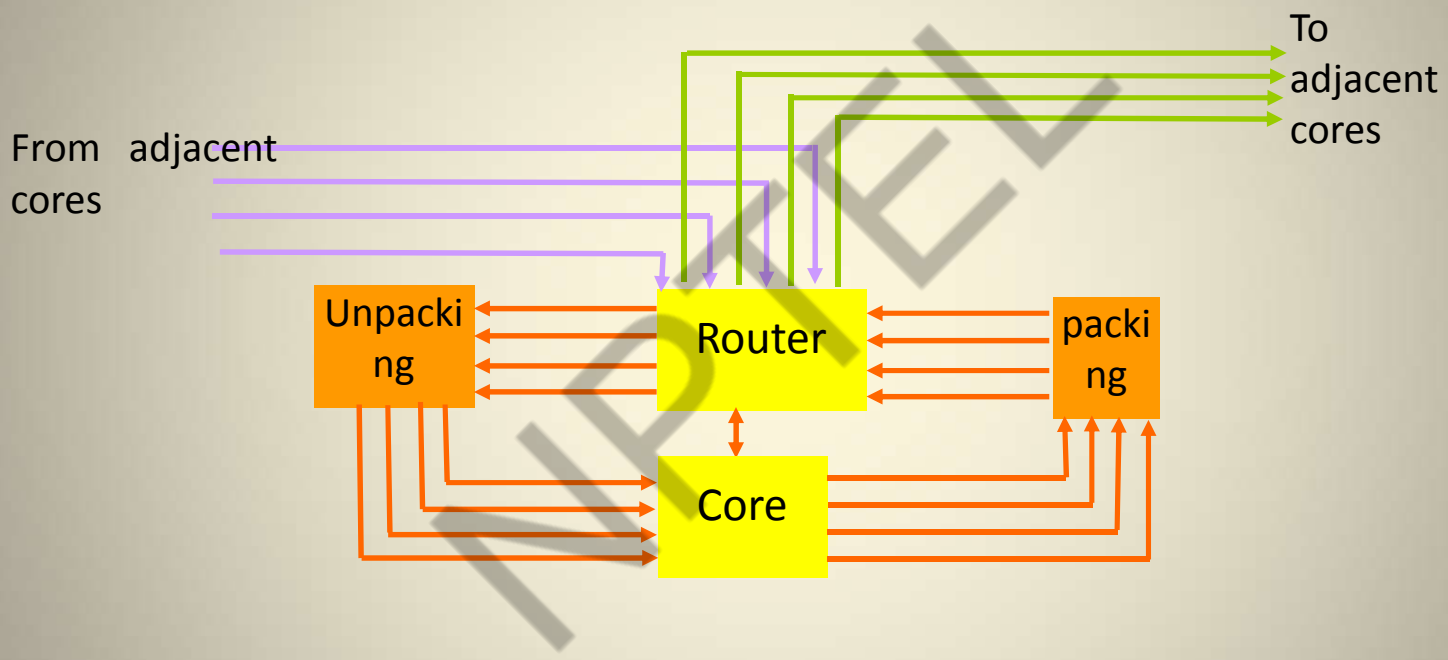
Test Wrapper

On top of the 1500 compliant wrapper
Can wrap both router and core
Packing/unpacking mechanism reused from mission mode



NPTEL

Test Wrapper



Mission mode

NPTEL

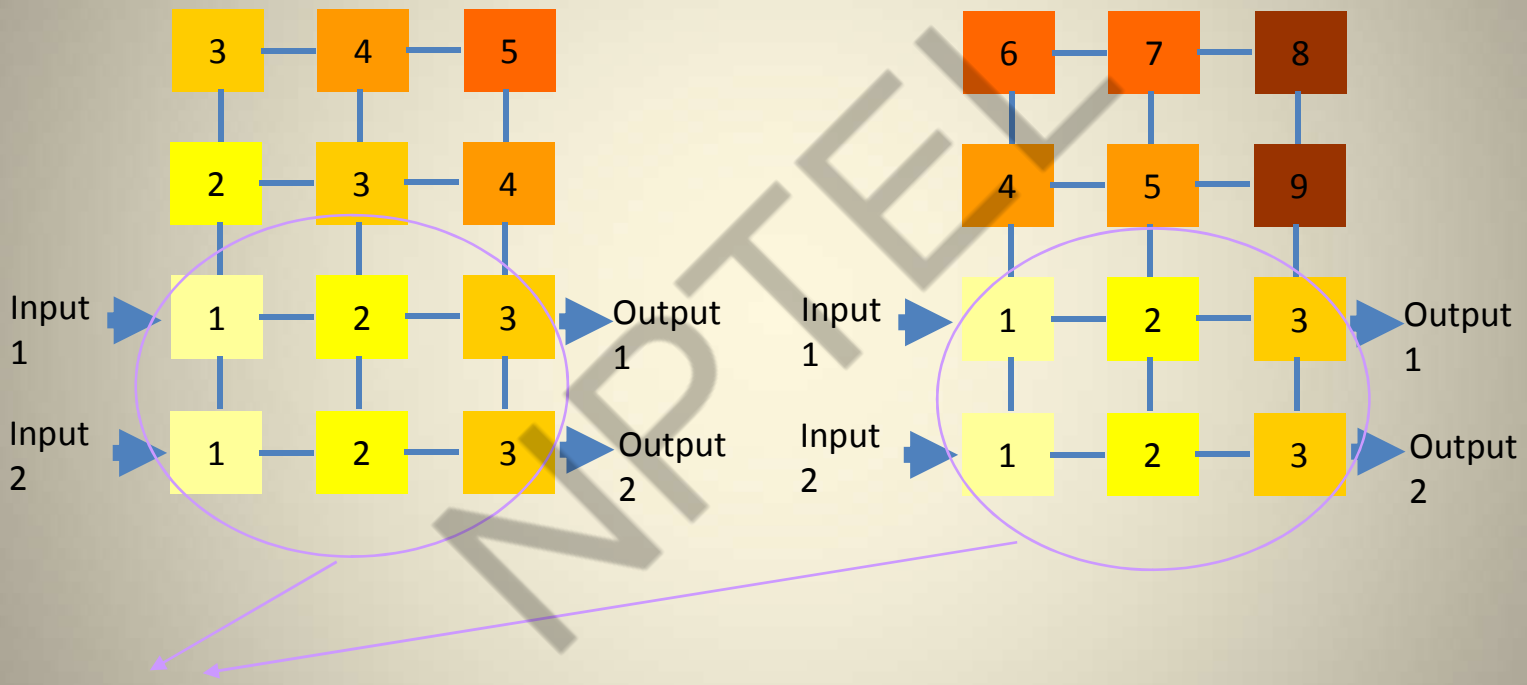
Integrated Test Scheduling

Based on network reuse and dedicated routing path

- Permute cores in the order of test time
- Permute all input/output pairs
- For each permutation
- Find free I/O pair
- Check for resource conflicts
- schedule a core
- Routers on a path should be all tested before functional cores on that path to be tested
- Routers can be tested concurrently with cores
- At least one I/O pair should be used for router testing at any time

NPTEL

Integrated Test Scheduling



NPTEL

After these routers are tested, one of the two I/O pairs can be used for core testing

Efficient Channel Width Utilization

Fixed channel width, not fully utilized

Cores	# of packets	Channel width = 16		Channel width = 32	
		flits/packet	test cycles	flits/packet	test cycles
1	24	2	38	1	25
2	146	13	1029	7	588
3	150	32	2507	32	2507
4	210	54	5829	54	5829
5	220	109	12192	55	6206
6	468	50	11978	41	9869
7	190	43	4219	34	3359
8	194	46	4605	46	4605
9	24	128	1659	64	836
10	136	109	7568	55	3836

NPTEL

Utilization of Idle Channel Width

Variable on-chip test clocks

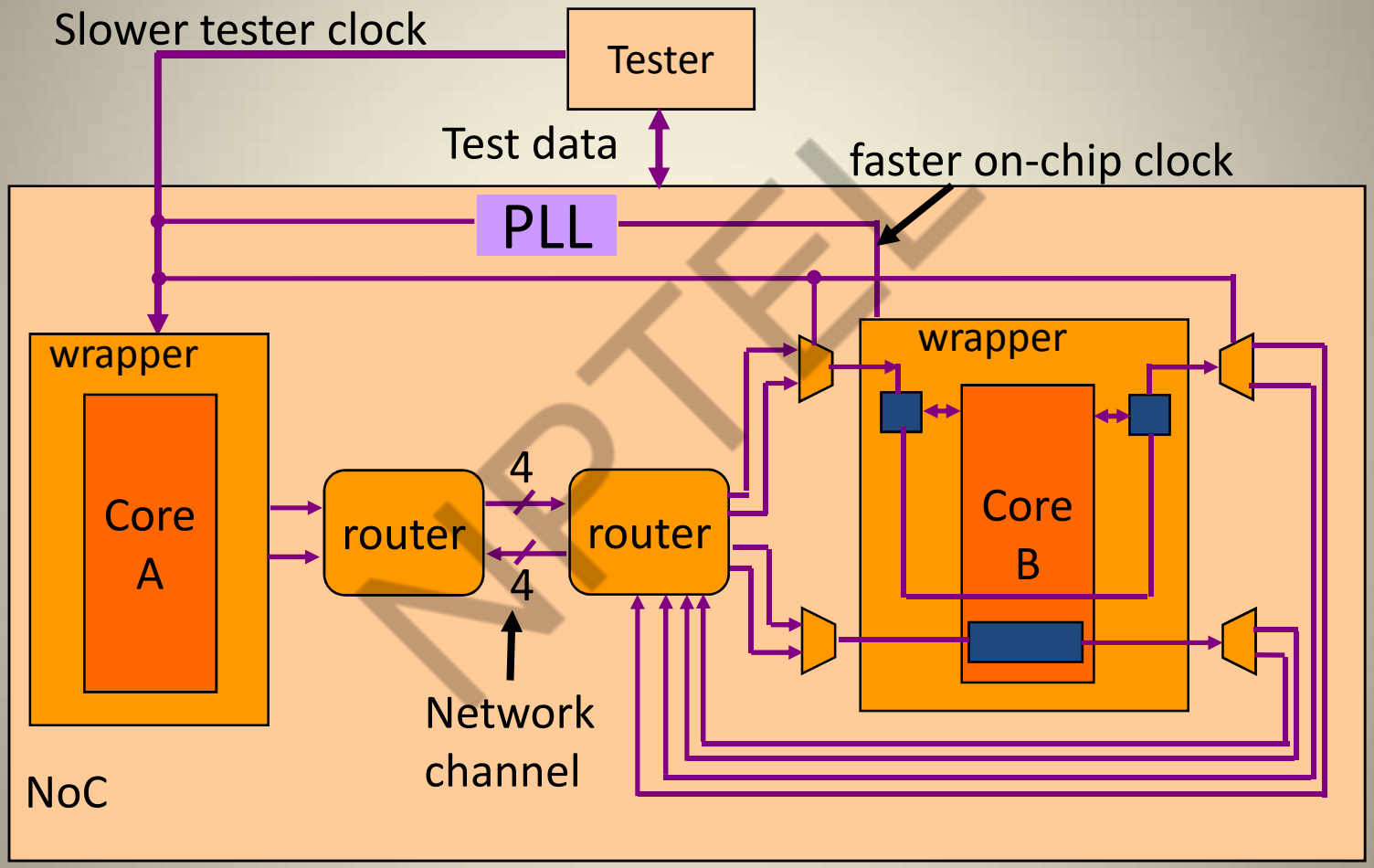
- Use faster wrapper test clocks on cores with idle channel width
- Channel width w, wrapper scan chain w', n flits can be transported in parallel to core in one clock

$$n = \left\lfloor \frac{w}{w'} \right\rfloor$$

- Additional cores can be selected to further reduce test time

NPTEL

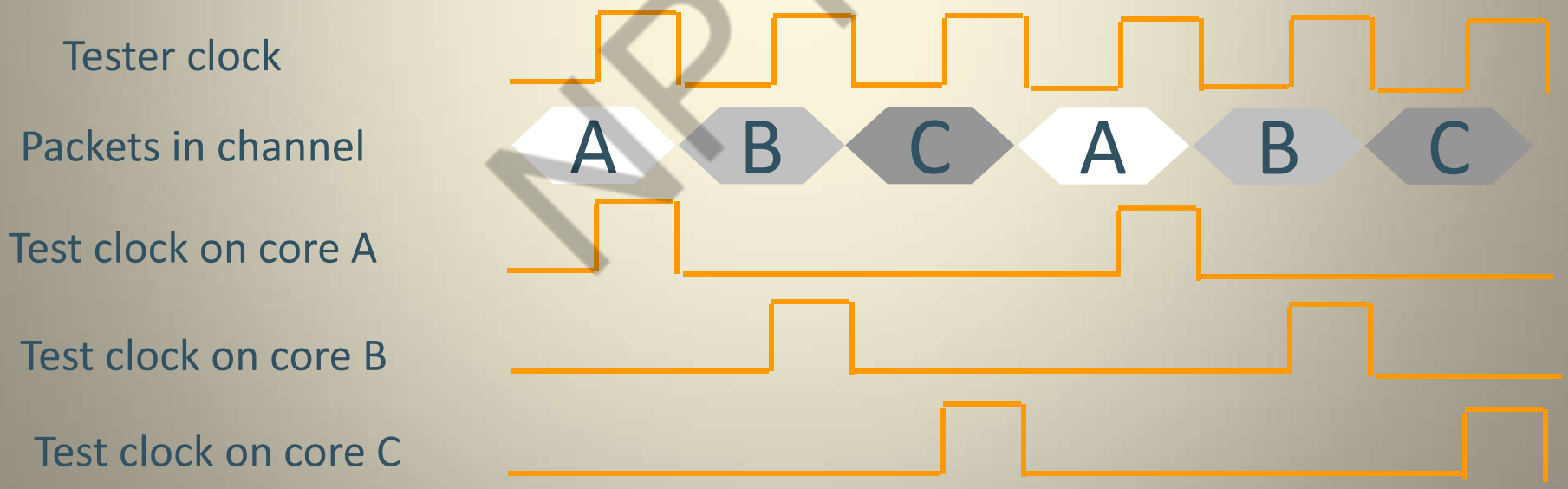
Utilization of Idle Channel Width



Channel Width Utilization Under Power Constraints

Variable on-chip test clocks

- Use slower wrapper test clocks on cores with high power dissipation
- No change on wrapper design
- Physical channel is viewed as n virtual channels



NPTEL

Power-Aware Test Scheduling

Variable on-chip test clocks in NoC-based system

- N cores, tester clock f_T
- Faster on-chip clocks $2f_T$, $3f_T$, …
- Slower on-chip clocks $f_T/2$, $f_T/3$, …
- Determine a clock for each core, such that
  - No network resource conflicts
  - System test application time is minimized
  - Power constraints are not violated

NPTEL

Power-Aware Test Scheduling

- Each core associated with a set of on-chip clocks {…$3f_T$, $2f_T$, f_T, $f_T/2$, $f_T/3$, …}
- Each clock corresponds to a power P(i,j), and the corresponding test time T(i,j)
- Selection of clock for each core controlled by a priority calculated from $\Delta P/\Delta T$
- More than one cores use slower clocks to utilize virtual channels
- Use dedicated routing path
- Power constraints are evaluated

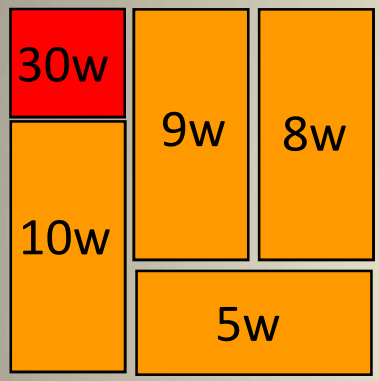
NPTEL

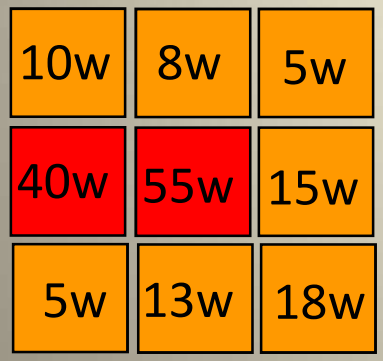
Lecture 55

NPTEL

Thermal-Aware Test Scheduling

High power density causes hot spots





- Existence of hot spots may increase test time because of thermal unbalance
- Layout redesign is impossible
- Layout not optimized for test
- Higher power generation
- Larger thermal variation
- Removal of hot spots can lead to thermal balance and reduced test time

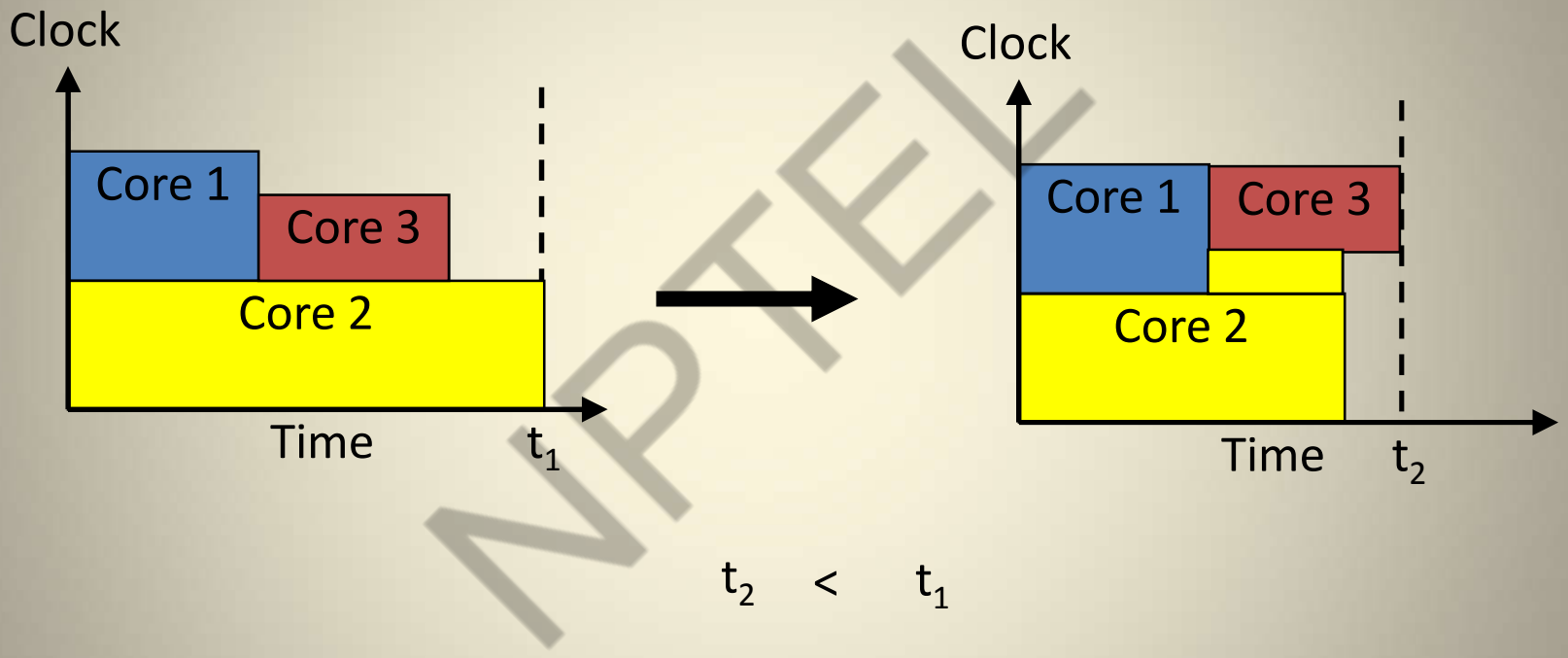
NPTEL

Variable Clocking in Test Session

- Still rely on using multiple variable clocking for thermal management
- Clock assigned to each core can be varied during test application
- A more flexible scheme
- More efficient thermal management
- Extra test control

NPTEL

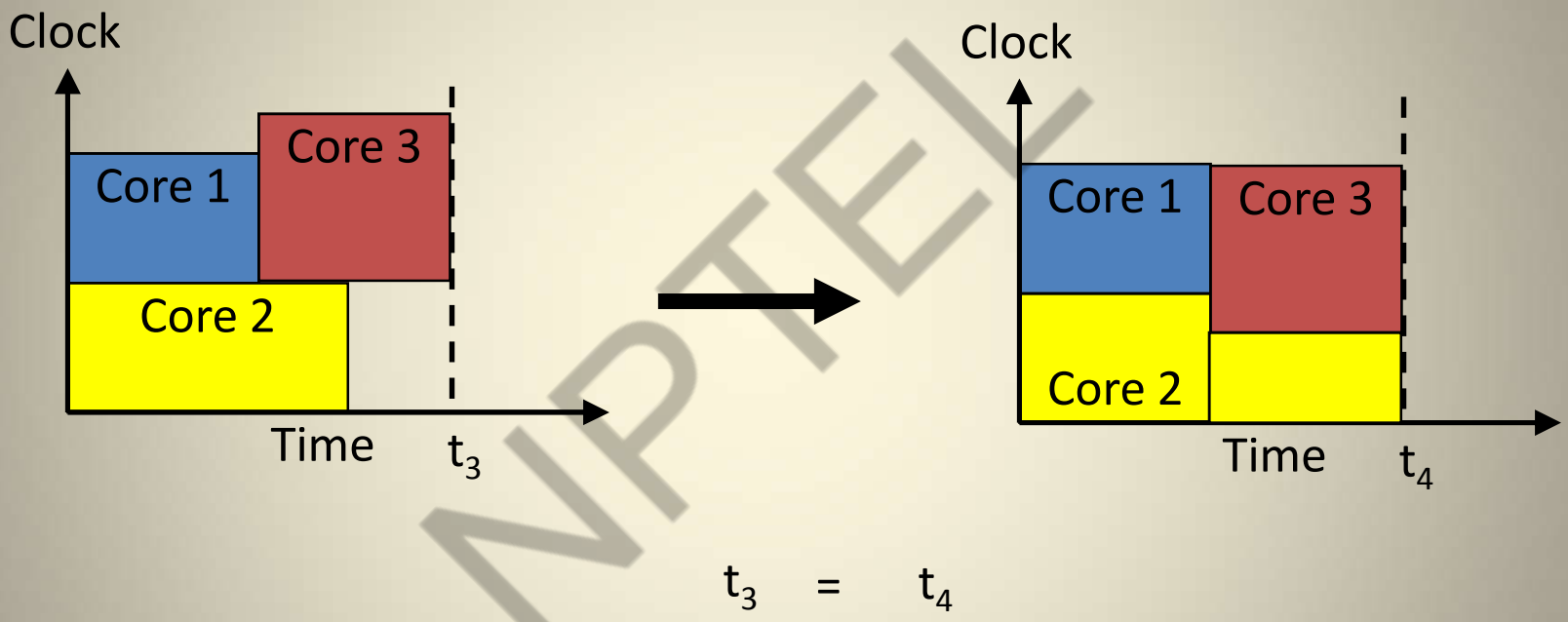
Variable Clocking in Test Session



NPTEL

Thermal safe constraints are not violated
Test time reduced

Variable Clocking in Test Session

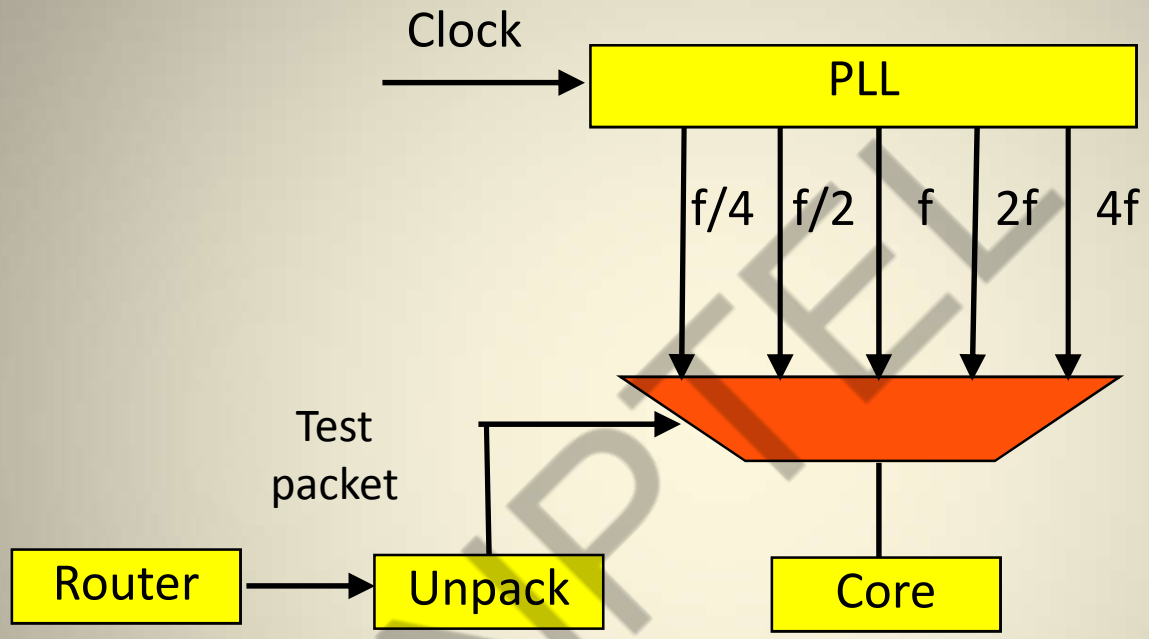


$t_3 = t_4$

NPTEL

Thermal safe constraints guaranteed
Test time not compromised

Clock Selection



Unpack reused
Test control can be carried in packet
Clock varies only when the test of a core finished or started

NPTEL

Problem Formulation

- Test set information of core set C
- N_C cores, N_I inputs, N_O outputs,
- Set of on-chip variable-rate clock CLK
- Set of thermal parameters $P_{thermal}$
- Chip floorplan, and maximum temperature T_{TH}
- Determine: (1) clock variation of each core during test application, (2) test scheduling of cores on I/Os and channels, such that:
  - Test application time is minimized
  - Maximum temperature not over T_{TH}

NPTEL

On-Chip Network Testing

- Testing of interconnect infrastructures
- Testing of routers
- Testing of network interfaces and integrated system testing
- Unless on-chip network of an NoC has been completely tested, it cannot be used to test the embedded cores.

NPTEL

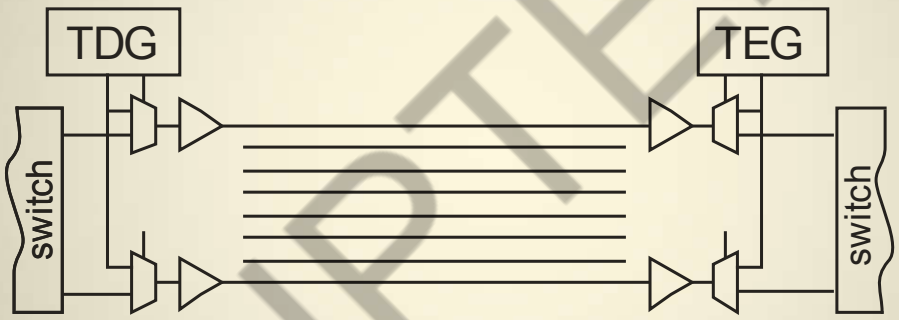
Testing of Interconnect Infrastructures

- Interconnect testing has been discussed in many papers.
- This discussion is mainly based on the well-known maximal aggressor fault (MAF) model.
- Apply identical transitions to all wires except the victim line to create maximal integrity loss in the victim line.
- Contains six crosstalk errors in victim line: rising/falling delay, positive/negative glitch, and rising/falling speed-up.
- For an interconnect structure with N lines, totally 6N faults are to be tested using 6N two-vector test patterns.

NPTEL

Self-Test Structure

- A pair of test data generator (TDG) and test error detector (TED) is inserted to each set of interconnects between two routers (switches).



- This is called *point-to-point* MAF self-test.
- Test patterns are launched before line drivers, and sampled after receiver buffers.
- Highly parallel testing if power consumption is within the power budget.

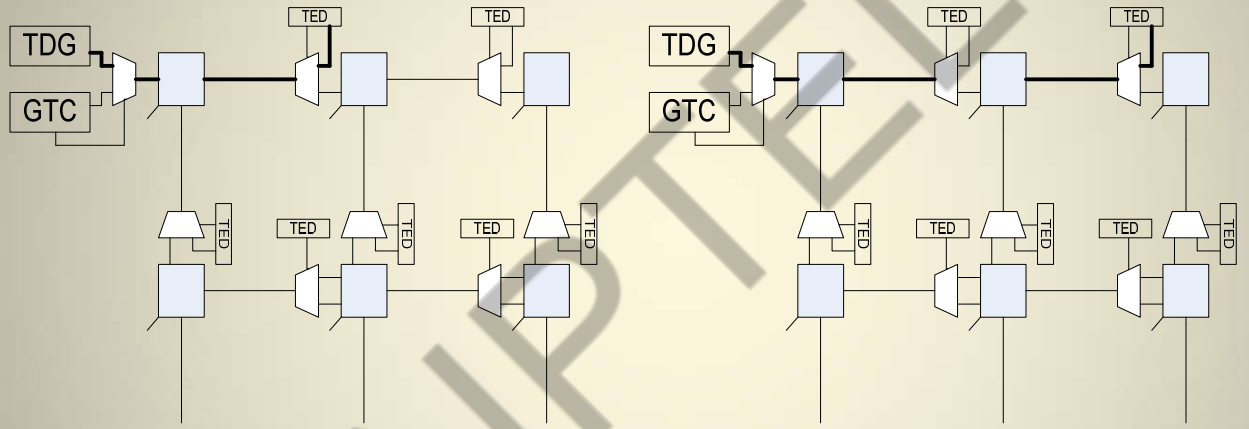
NPTEL

Lecture 56

NPTEL

Test Application by Unicast

- MAF test patterns can be broadcast to all interconnects by test packets with only one TDG.

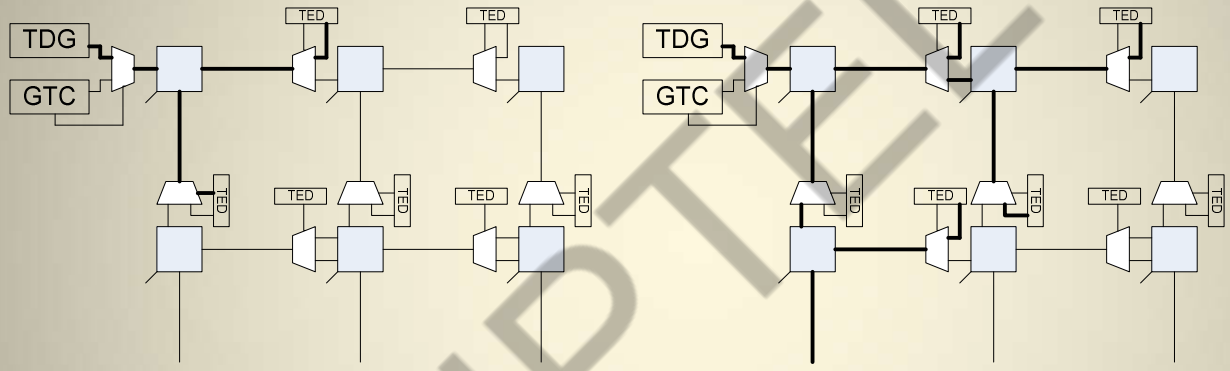


- Only one set of interconnects between a pair of routers can be tested for each test pattern broadcast.
- A global test controller (GTC) and many TEDs are required.

NPTEL

Test Application by Multicast

- Test packets are broadcast to interconnects of different pairs of routers to achieve maximum parallelism.

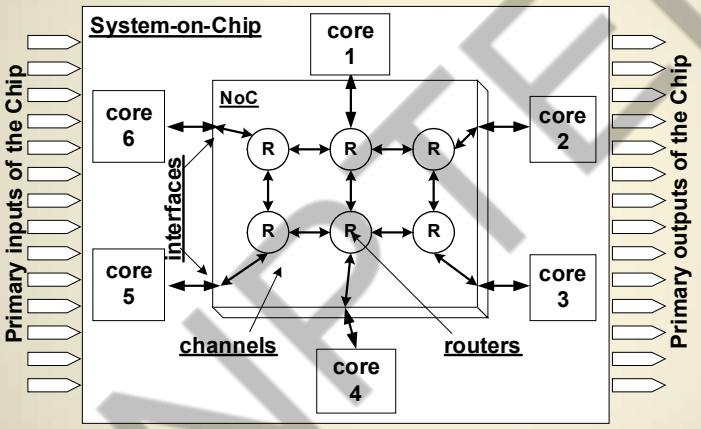


- Multicast is a good compromise between test application time and hardware overhead.
- Point-to-point (unicast) test method has the smallest (largest) test application time but the largest (smallest) hardware overhead.

NPTEL

Testing of Routers

- Routers are used to implement functions of flow control, routing, switching and buffering of packets.

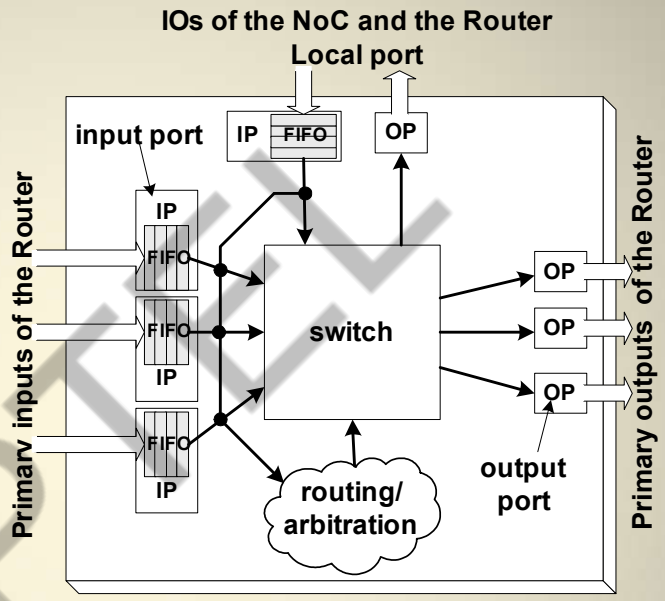


- Router testing can be treated as sequential circuit testing by taking its special property of regularity.
- Test pattern broadcasting can be applied to reduce test time.

NPTEL

Testing A Router

- Testing a router consists of testing the control logic (routing, arbitration, and flow control modules) and first-in first-out (FIFO) buffers.

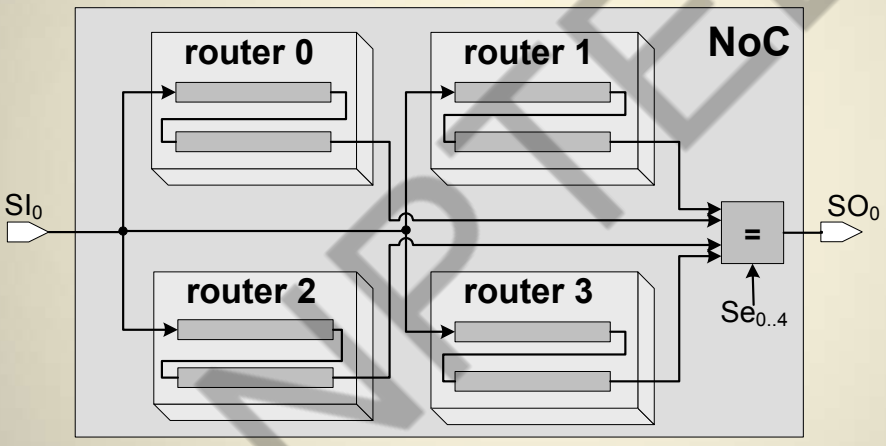


- Control logic can be tested by typical sequential circuit testing methods such as scan testing.
- A smart way to test FIFO is to configure the first register of FIFO as scan register, and others can be tested by the scan register.

NPTEL

Testing All Routers

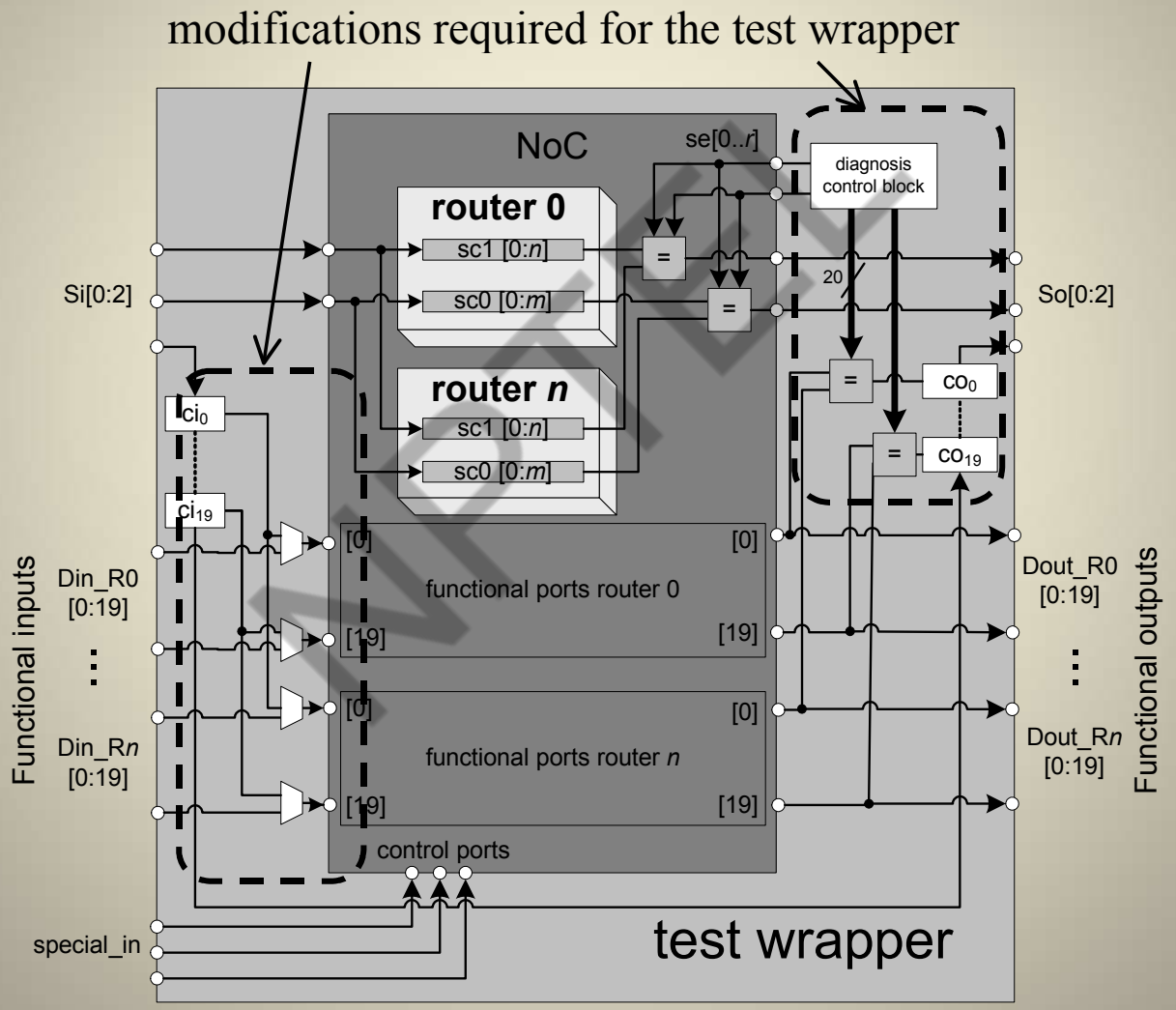
- Since all routers are identical, all can be tested in parallel by test pattern broadcasting.



- Comparator is implemented by XOR gates. It can also support diagnosis.

Router Test wrapper Design and Test

- IEEE-1500 compliant test wrapper is designed to support test pattern broadcasting and test response evaluation.



Router Test Wrapper Design and Test (Contd.)

- For example, all SC1 chains of these routers share the same set of test patterns.
- Similarly, all Din[0] (i.e., Din-R0[0], …, Din-Rn[0]) data inputs of these routers share the same set of test patterns.
- The wrapper also supports test response comparison for scan chains and data outputs.
- Diagnosis control block can activate diagnosis.
- Small hardware overhead (about 8.5%) and small number of test patterns (several hundreds) due to test broadcasting. Small test application time (several thousands test cycles) using multiple, balanced scan chain and test broadcasting. *The method is scalable.*

NPTEL

Concluding Remarks

- State-of-art techniques for SoC testing have been described.
- Modular test techniques for digital, mixed-signal, and hierarchical SoCs must be developed further to keep pace with technology advances.
- Test data bandwidth needs for analog cores are very different from digital cores, and unified top-level testing of mixed-signal SoCs remains a major challenge.
- Research is also needed to develop wrapper design techniques and test planning methods for multi-frequency core testing.
- Revolutionary RF interconnect technology might emerge to address future SoC testing.

NPTEL

Concluding Remarks (Contd.)

- Advances in testing NoC-based systems have been discussed.
- Key point: how to utilize on-chip network as a TAM without compromising fault coverage or test time.
- Research on NoC testing is still premature when compared to industrial needs, and future research and development are needed.
- Wrapper design techniques for SoC testing can be adopted by NoC-based systems.

NPTEL