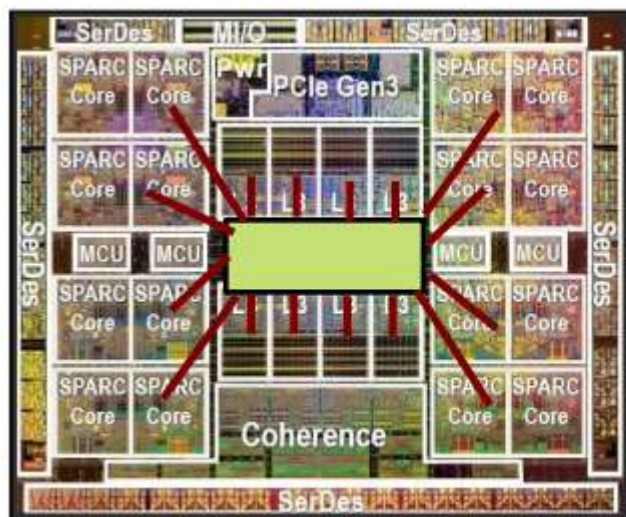
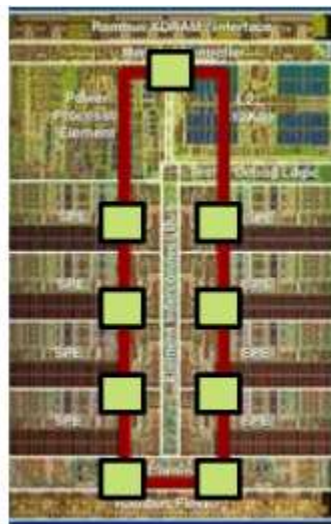


# Task : NoC Circuit Design

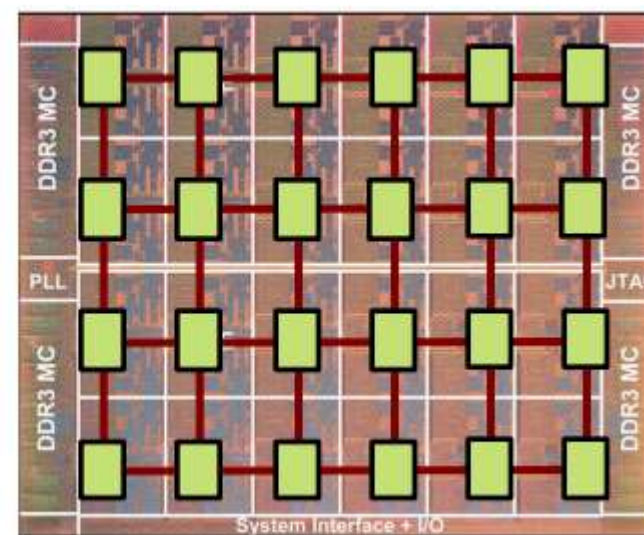
## 1. 片上网络(Network on Chip)简介



SPARC T5(Oracle)  
16核Crossbar互连NoC



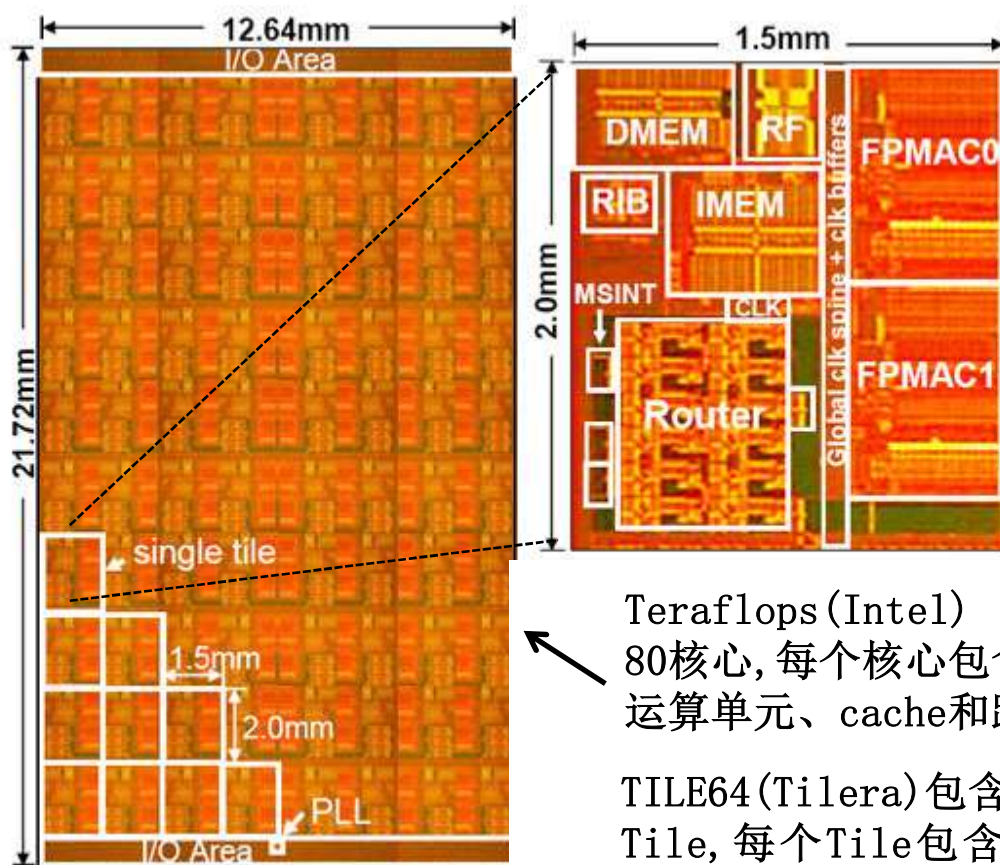
Cell(IBM)  
9核Ring互连NoC



SCC(Intel)  
24核Mesh互连NoC

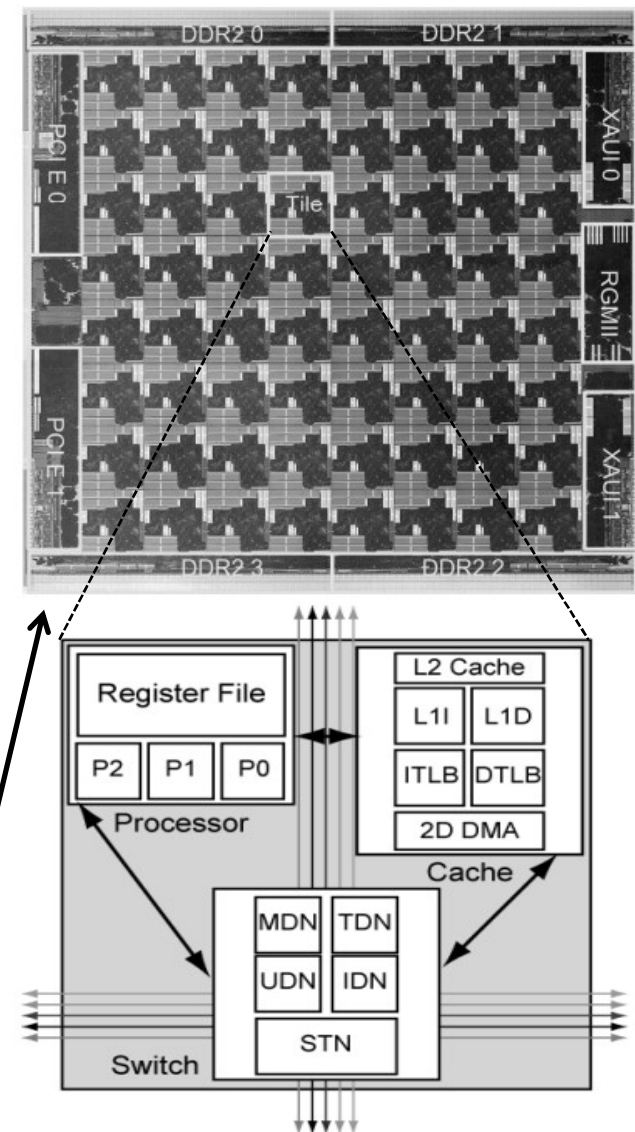
# Task : NoC Circuit Design

## 1. 片上网络(Network on Chip)简介



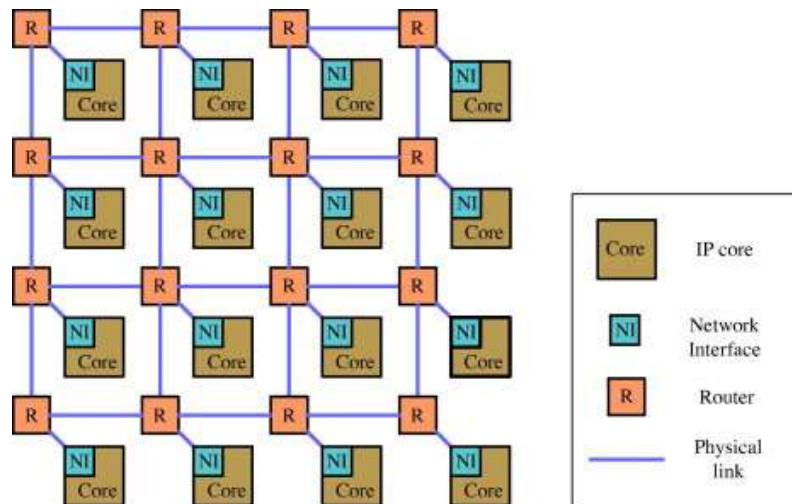
Teraflops(Intel) 包含 80核心, 每个核心包含浮点运算单元、cache和路由器

TILE64(Tilera)包含64个 Tile, 每个Tile包含处理器、cache和路由器



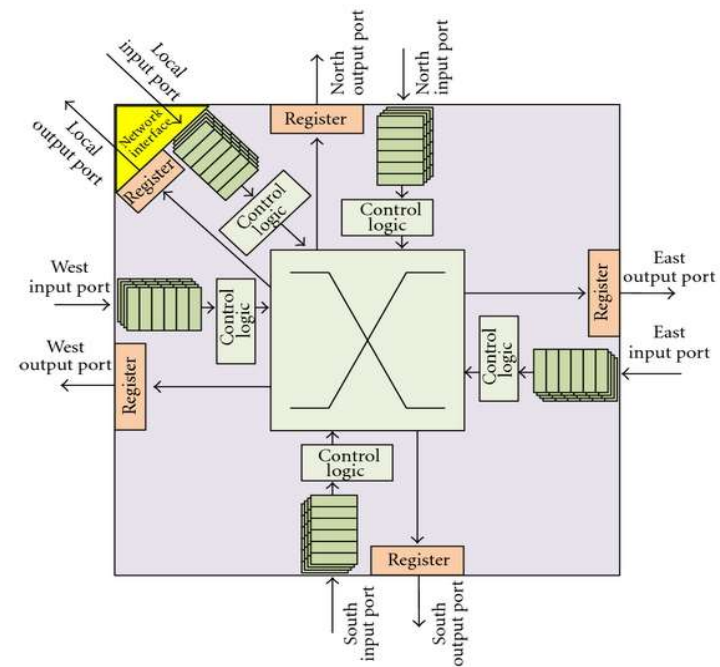
# Task : NoC Circuit Design

## 1. 片上网络(Network on Chip)简介



组成部分:

- IP core
  - 计算, 发送和接收数据包
- Router(路由器)
  - 存储和转发数据包
- NI(网络接口)
  - IP core与Router的数据交换



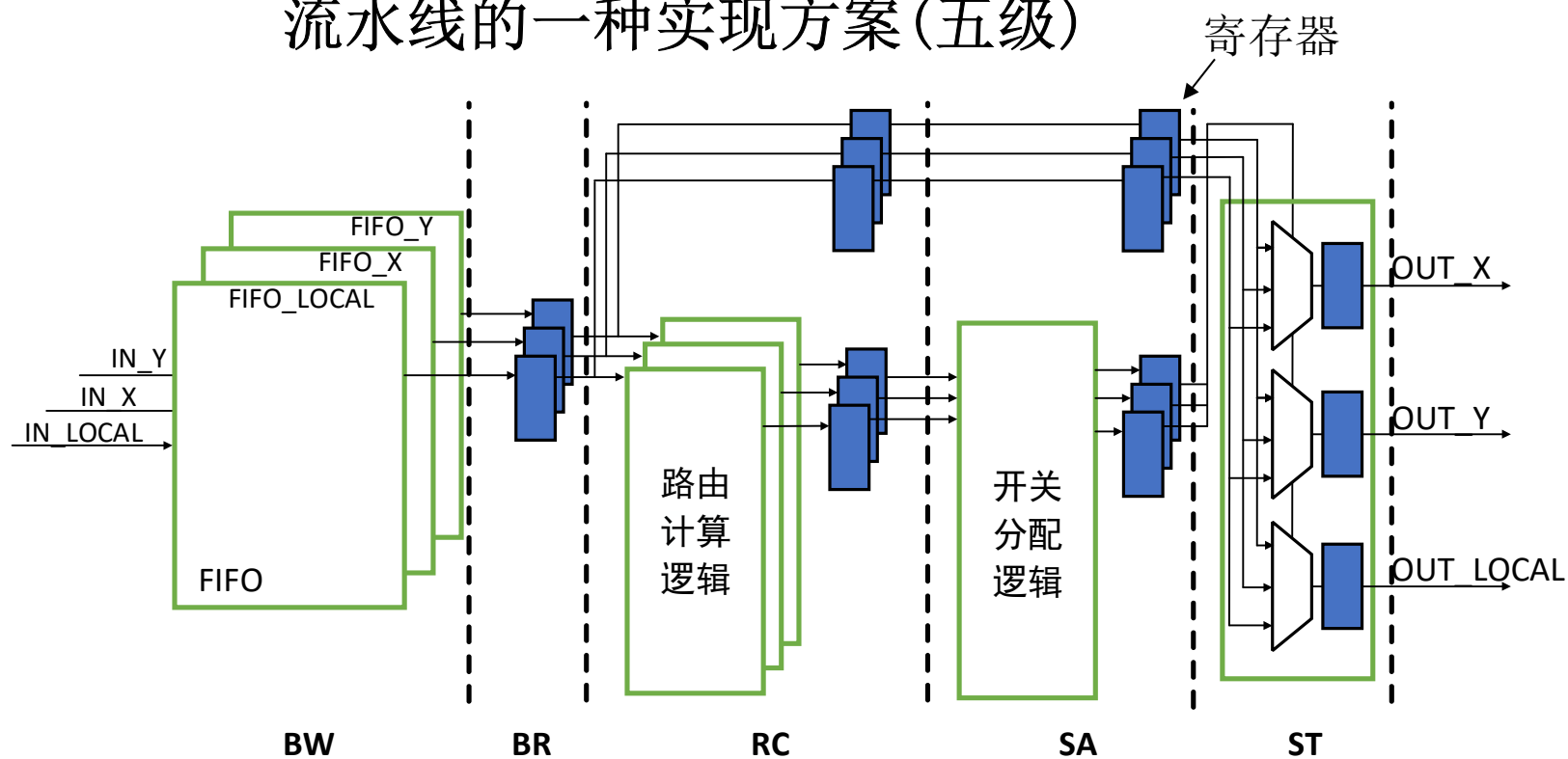
上图为典型的Mesh互连NoC中的路由器结构, 包含5个端口, 端口之间可以互相发送数据。硬件包含端口的缓冲区、控制逻辑和中间的交换开关

注1:本设计中将把IP core替换为traffic generator, 只负责发送和接收数据包, 不负责计算  
注2:本设计中略去了网络接口

# Task : NoC Circuit Design

## 2. 路由器的流水线结构

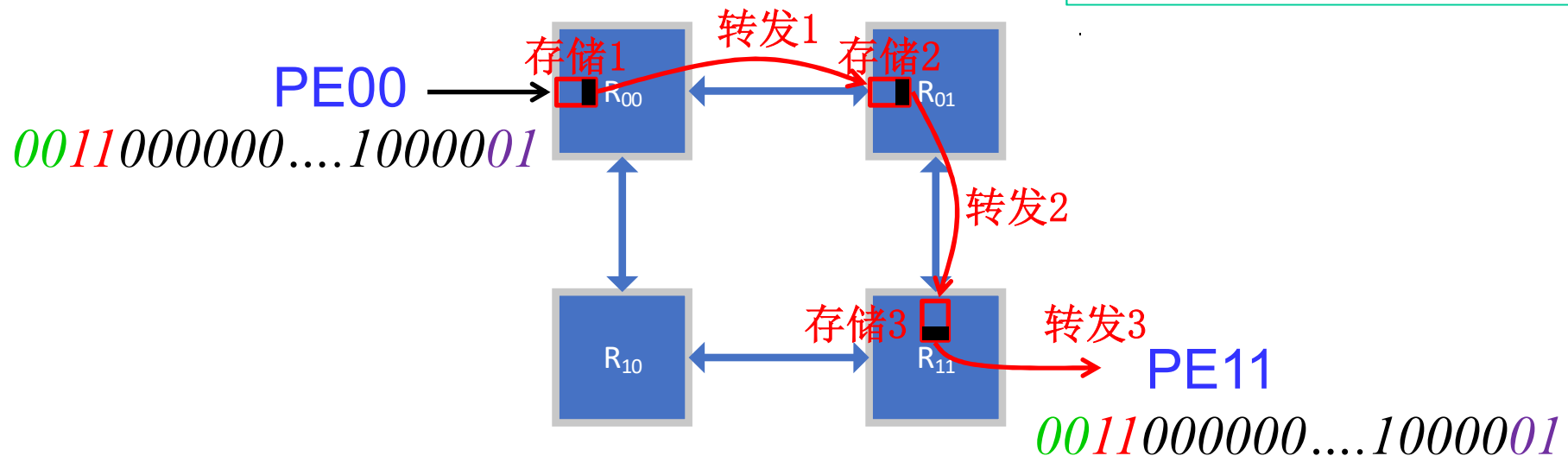
流水线的一种实现方案(五级)



# Task : NoC Circuit Design

## 3. 路由器功能简介(以2x2Mesh为例)

### 存储-转发模型



数据包格式			
4 bits (2/2)		26 bits	2 bits
Src	Dst	Data	Type
Type { 01: Regular packet 00: Retransmission request 11: Retransfer packet			

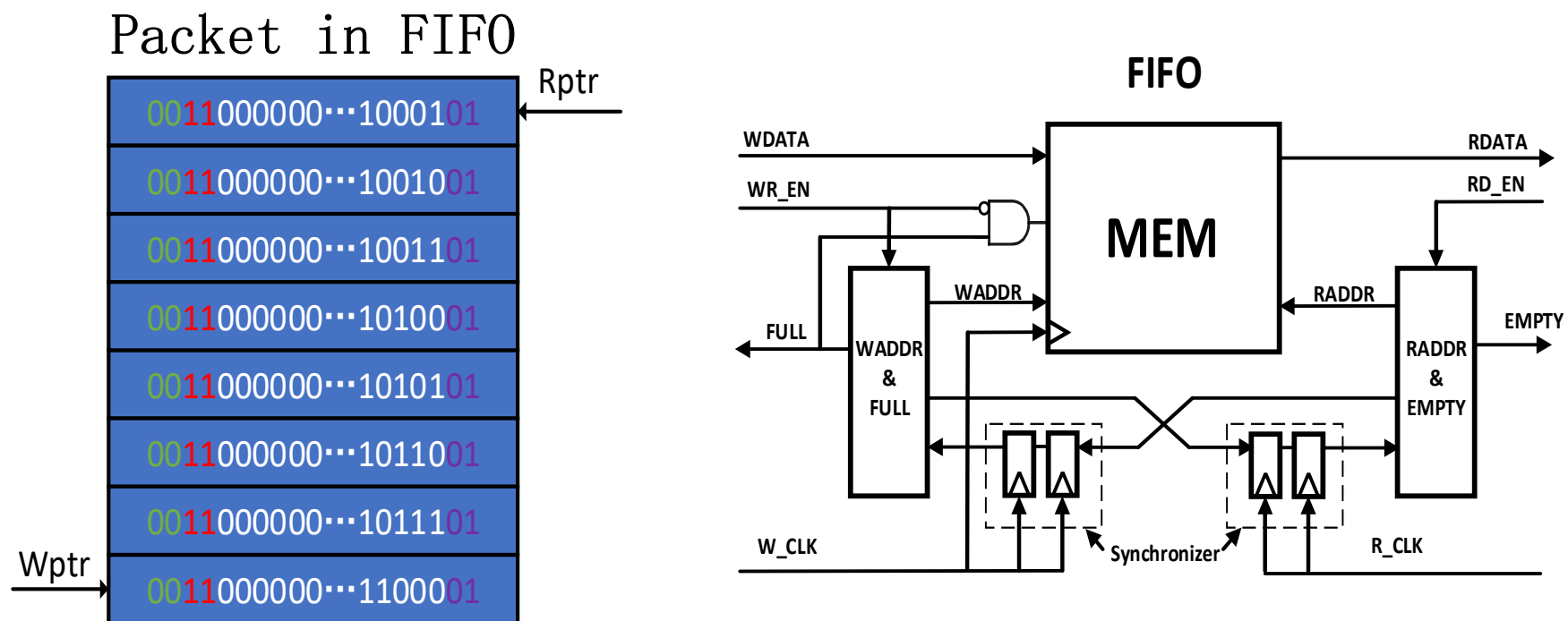
□ 路由器的缓冲区



# Task : NoC Circuit Design

## 3. 路由器功能简介(以2x2Mesh为例)

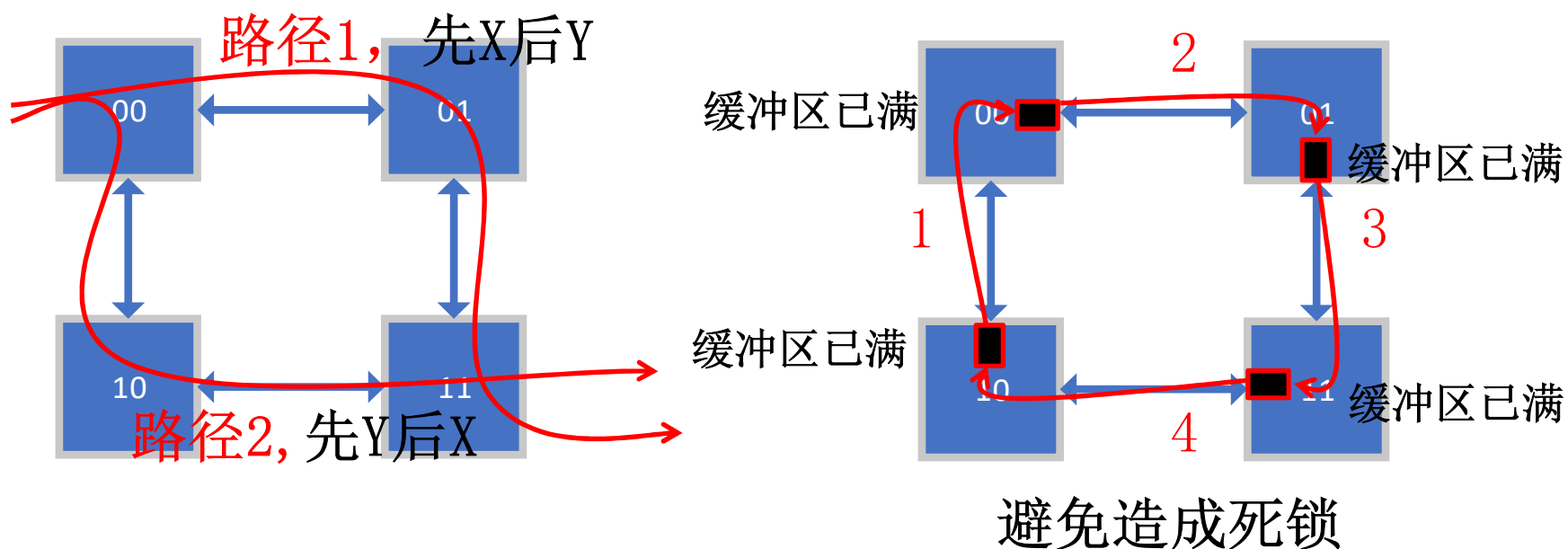
### 缓冲区的实现方法-FIFO



# Task : NoC Circuit Design

## 3. 路由器功能简介(以2x2Mesh为例)

路由 (Routing)



# Task : NoC Circuit Design

---

## 3. 路由器功能简介(以2x2Mesh为例)

### 先Y后X路由(路径2)的一种实现方法

#### 步骤1

路由器地址 00

数据包目的地址 11

00≠11, 数据包目前处在目的地址的不同行不同列, 向Y方向发送

数据包到达10路由器

#### 步骤2

路由器地址 10

数据包目的地址 11

10≠11, 数据包目前处在目的地址的同行不同列, 向X方向发送

数据包到达11路由器

#### 步骤3

路由器地址 11

数据包目的地址 11

11=11, 数据包目前处在目的地址的同行同列, 向LOCAL方向发送

数据包到达目的地址



# Task : NoC Circuit Design

---

## 3. 路由器功能简介(以2x2Mesh为例)

### 开关分配(Switch Allocation)

缓冲区	转发端口	仲裁结果
X	Y	成功
Y	LOCAL	成功
LOCAL	X	成功

无转发端口冲突

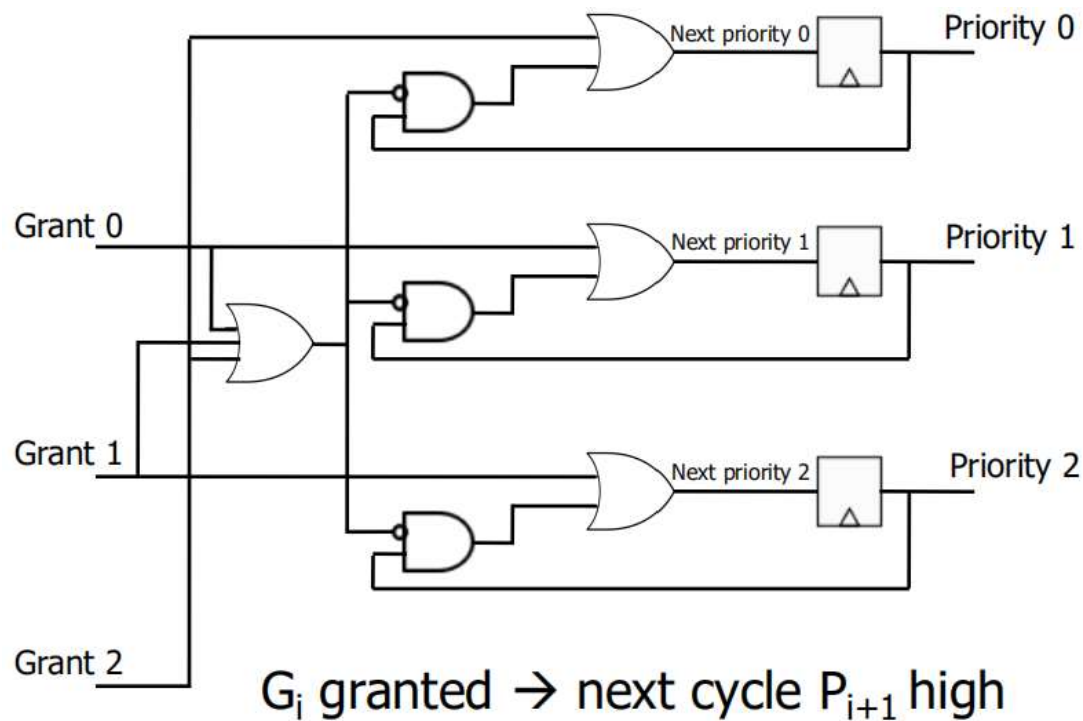
缓冲区	转发端口	仲裁结果
X	LOCAL	成功
Y	LOCAL	失败
LOCAL	X	成功

转发端口冲突

# Task : NoC Circuit Design

## 3. 路由器功能简介(以2x2Mesh为例)

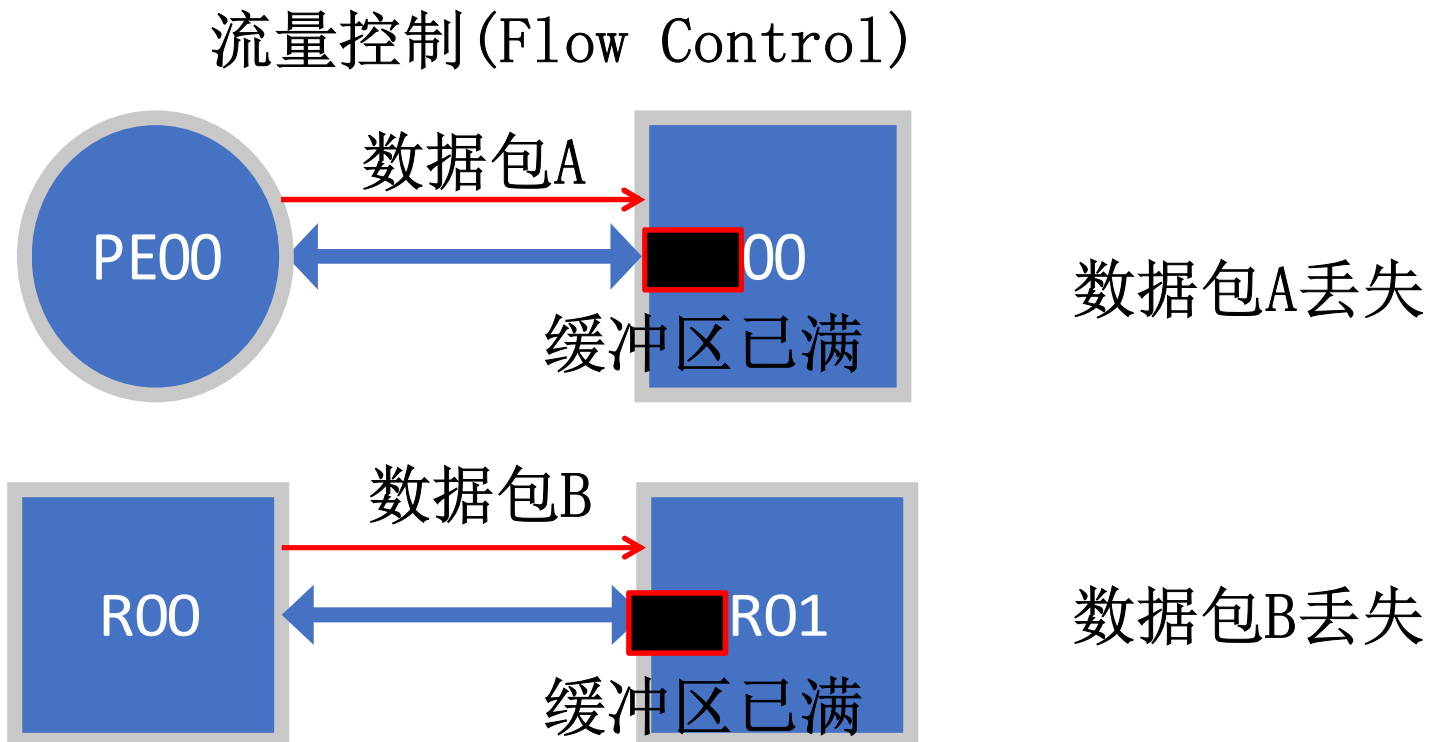
基于轮询的开关分配:避免数据饿死情况



每当一个端口的请求得到相应, 下个端口的优先级变高

# Task : NoC Circuit Design

## 3. 路由器功能简介(以2x2Mesh为例)



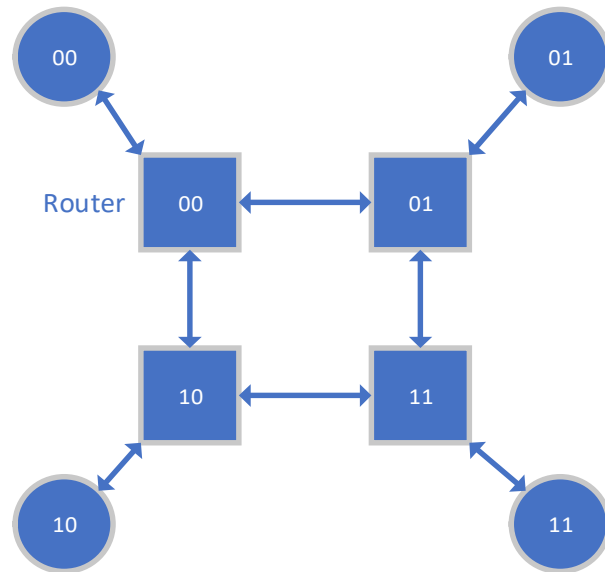
如果不考虑下级缓冲区的占用情况，会造成数据包丢失

# Task : NoC Circuit Design

## 4. 课程设计目标

面向众核处理器的需求，设计满足功能需求的路由器。然后基于所给出的**PE**模块(提供**RTL**代码)，完成**2x2 mesh**结构的**NoC**系统。

TG (Traffic Generator)



- router的流水级数不限;
- router各个部件的设计和实现推荐
  - 缓冲区(同步FIFO);
  - 路由(先X后Y路由);
  - 开关分配(轮询);
  - 开关分配失败解决(流水线暂停);
  - 流量控制(流水线暂停);

# Task : NoC Circuit Design

---

## 5. 具体内容

课程设计目标具体包含以下3个部分的设计内容：

路由器部分：

- ◆ 架构设计
- ◆ **RTL**代码设计
- ◆ 逻辑仿真
- ◆ 逻辑综合
- ◆ 时序分析
- ◆ 物理设计

**TG**部分：

- ◆ 逻辑仿真
- ◆ 逻辑综合
- ◆ 时序分析
- ◆ 物理设计

**NoC**：

- ◆ 代码集成
- ◆ 逻辑仿真
- ◆ 物理设计

# Task : NoC Circuit Design

---

## 6. 评价指标

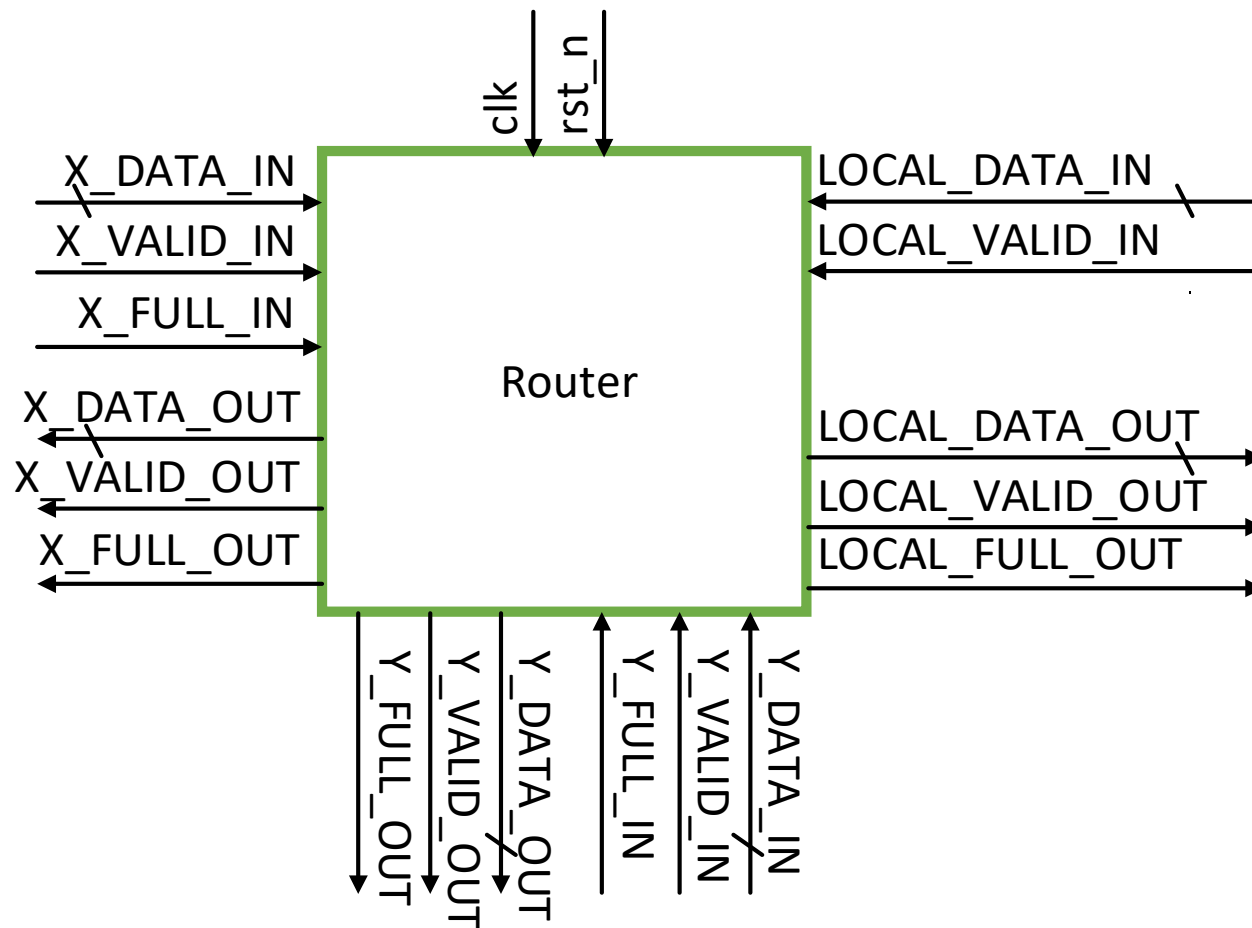
- ◆ 路由器的评价指标：
  - 路由器的最高频率
  - 路由器单位面积、单位频率的功耗
- ◆ NoC的评价指标：
  - NoC的吞吐率
  - NoC的平均数据包传输延时
- ◆ 芯片设计工艺：
  - 55nm 工艺



# Task : NoC Circuit Design

---

## 7. 详细设计要求-router的端口定义



# Task : NoC Circuit Design

---

## 7. 详细设计要求-rounter的端口说明

端口	说明
rst_n	复位，低电平有效
clk	时钟信号
LOCAL/X/Y_DATA_IN[39:0]	来自TG/X方向/Y方向的40bit数据包
LOCAL/X/Y_VALID_IN[39:0]	来自TG/X方向/Y方向的数据包有效信号
X/Y_FULL_IN[39:0]	来自X方向/Y方向的缓冲区满信号
LOCAL/X/Y_DATA_OUT[39:0]	送给TG/X方向/Y方向的40bit数据包
LOCAL/X/Y_VALID_OUT[39:0]	送给TG/X方向/Y方向的数据包有效信号
LOCAL/X/Y_FULL_IN[39:0]	送给TG/X方向/Y方向的缓冲区满信号

## Task : NoC Circuit Design

---

### 7. 详细设计要求-数据包的格式

4 bits (2/2)		8 bits	26 bits	2 bits
Src	Dst	Timestamp	Data	Type

Type {  
01: Regular packet  
00: Retransmission request  
11: Retransfer packet

注1. 当Type域为01或11时, Data域为全0

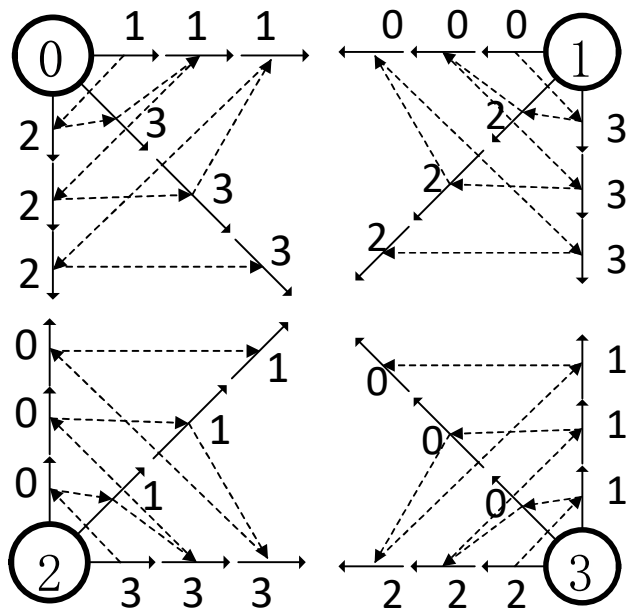
注2. Timestamp域表示数据包产生时的时钟周期数, 用以计算数据包延时

# Task : NoC Circuit Design

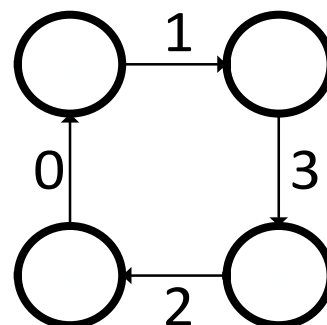
## 7. 详细设计要求-测试模式

### 标准测试用例 (Traffic Pattern)

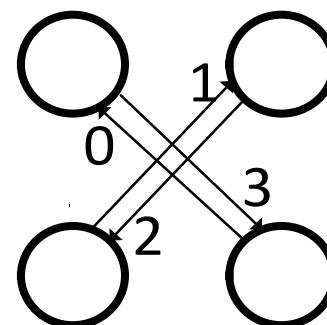
1. 轮询模式



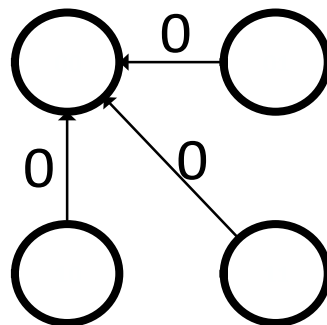
2. 邻居模式



3. 交叉模式



4. 热点模式



5. 并行模式

