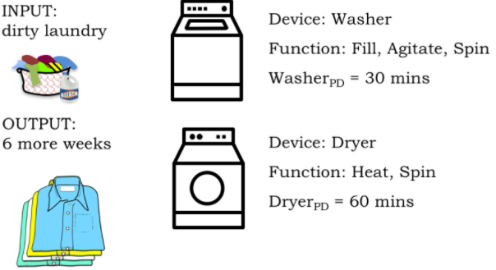
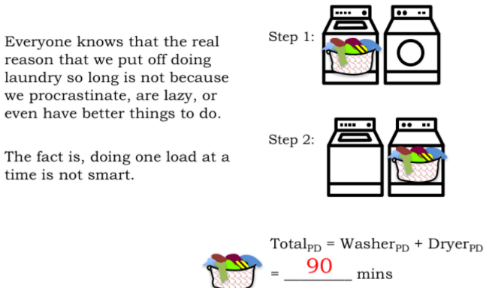
# 课件

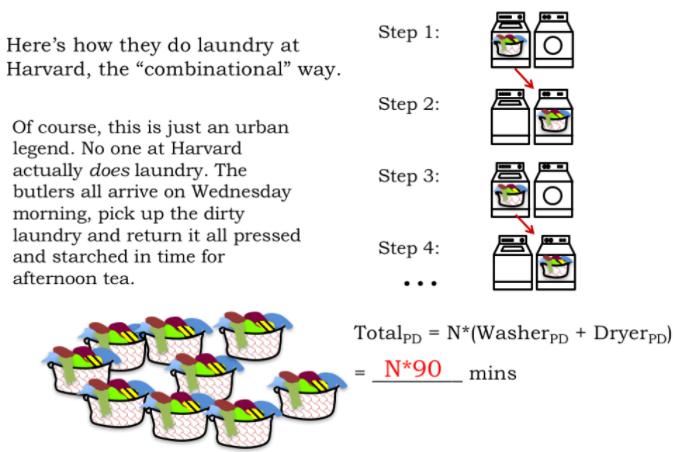
## 忘记电路，让我们解决一个实际问题



## 一次一件

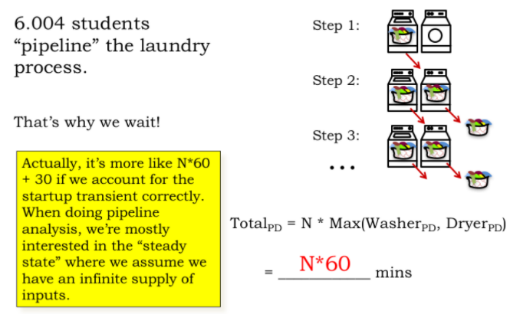


## 洗N件衣服



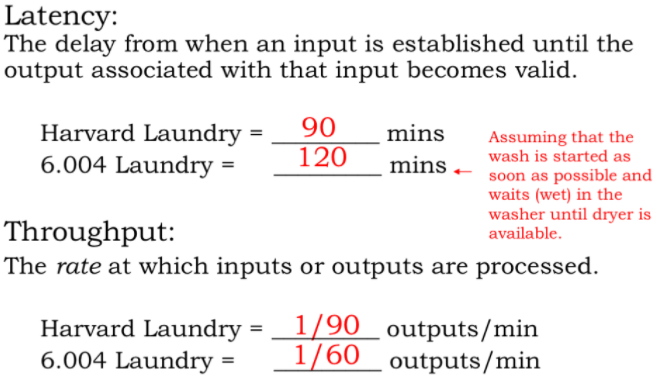
若衣服先进washer、再进dryer，会出现，衣服进入dryer后，washer空着的情况，可以让它继续处理

## 洗N件衣服，6.004的方式



实际用时60N+30，但考虑持续输入时，可以忽略起始的30分钟

## 性能观测

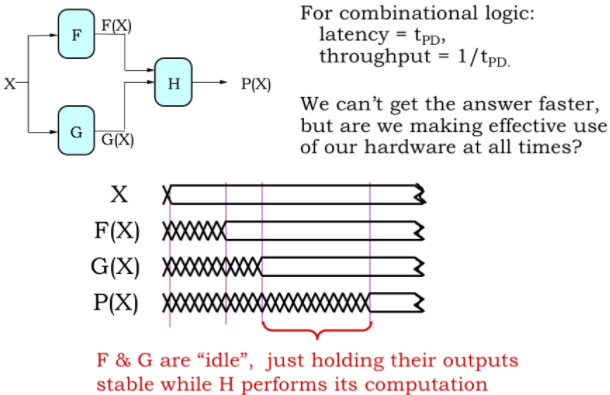


哈佛处理衣服的延迟为90min

6.004处理衣服的延迟为120min（除第一次装载之外）

吞吐量来看，哈佛90min一个，6.004 60min一个

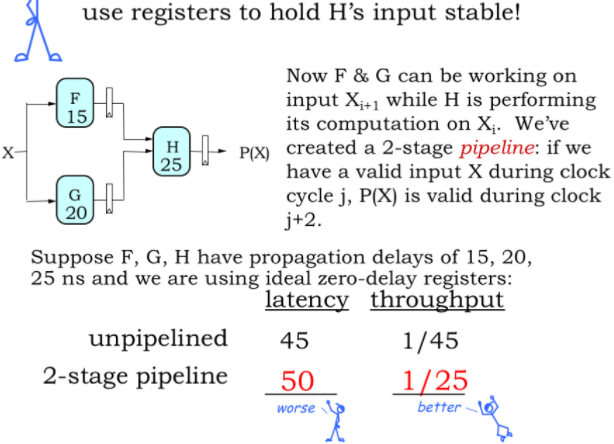
## 回到电路



组合电路，必须等有输出后，才能进行下次输入

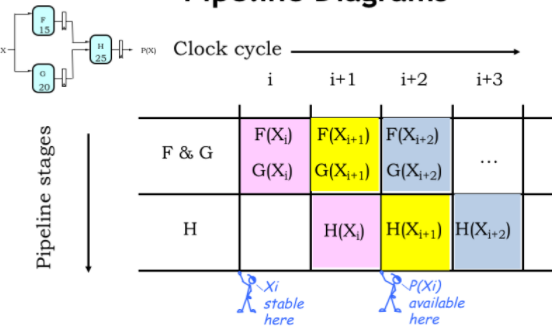
比如，H处理期间，F和G是空闲的。若将电路分为多个阶段，可以提升吞吐量

## 流水线电路

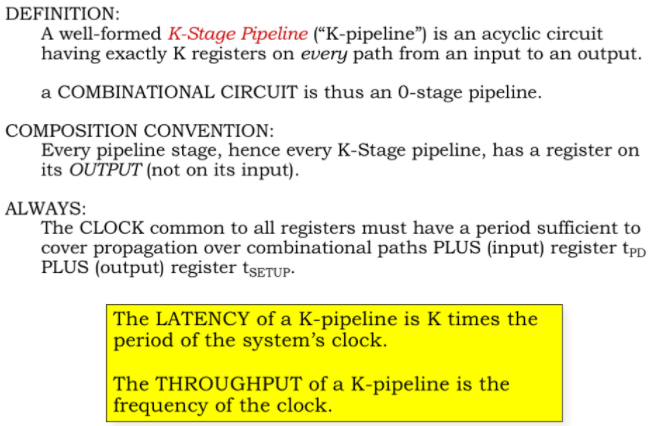


将F和G放到第一段、H放到第二段，F、G后面的寄存器用于保存F、G的输出，H后面的寄存器保存H的输出

## 流水线图



## 流水线惯例

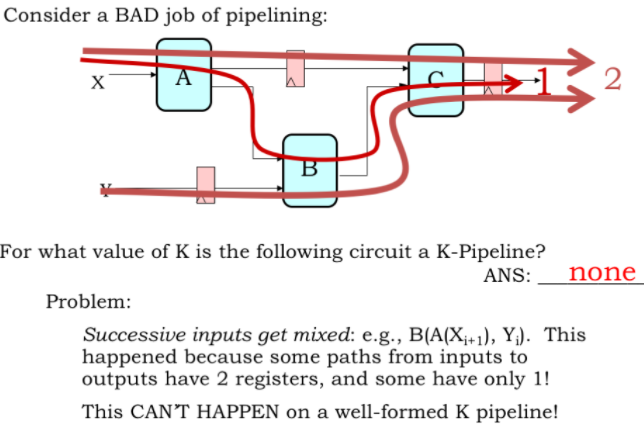


每阶段：输入寄存器tpd+组合电路tpd+输出寄存器tsetup中最大的为x，时钟周期大于等于x

k段流水线的延迟为k\*时钟周期

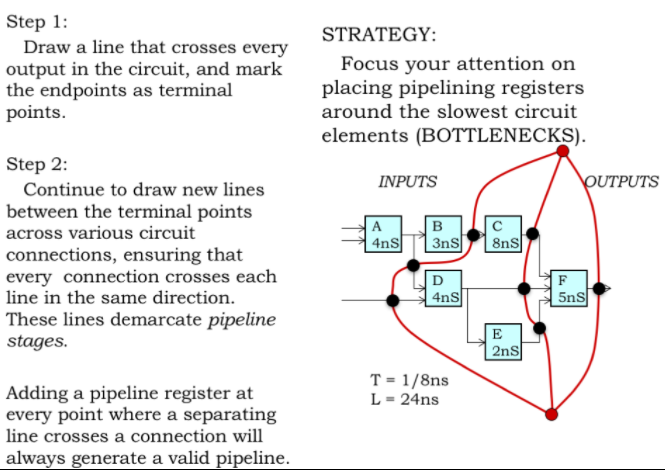
k段流水线的吞吐为1/时钟周期

## 不规范格式的流水线



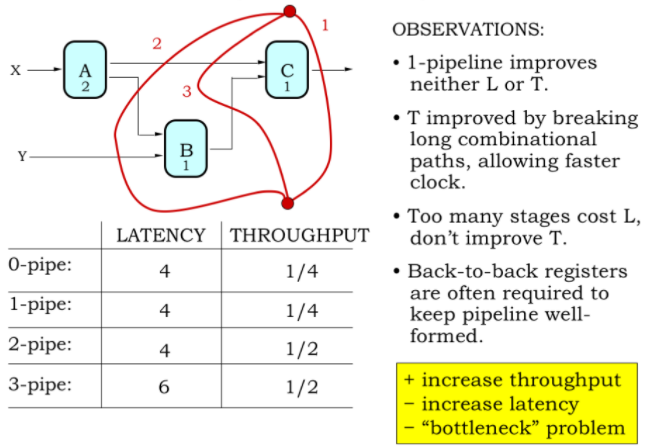
B组合电路为第一个时钟周期的Y和第二个时钟周期的X

## 流水线方法



连接模块输出进行划线，将整个电路切分为多块

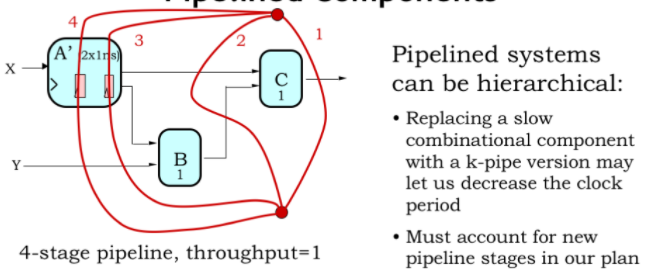
## 流水线例子



将组合电路分割为多块，以提升吞吐量，而延迟不可能比原组合电路低；

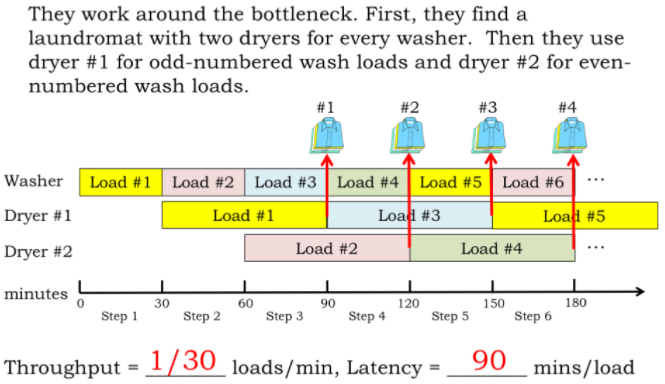
多块电路中，某块的最高延时决定了时钟周期

## 流水线组件

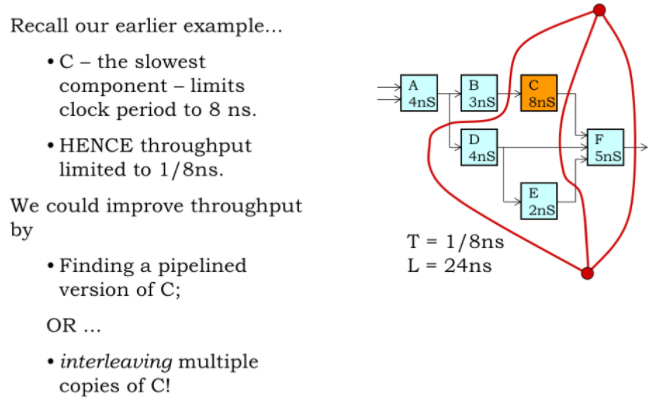


将A组件分割为k-pipe版，整个电路可分割为4块，延迟为4，吞吐为1

## 6.004学生如何洗衣服

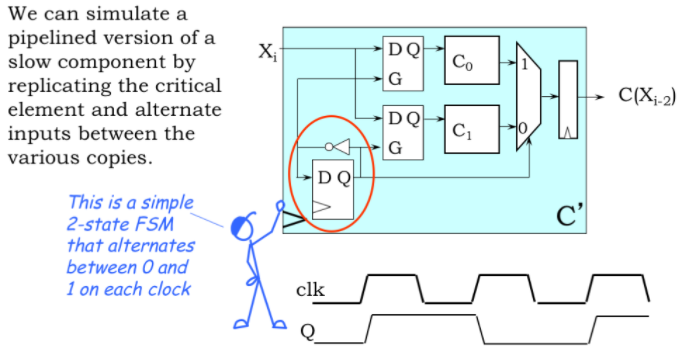


## 回到我们的瓶颈



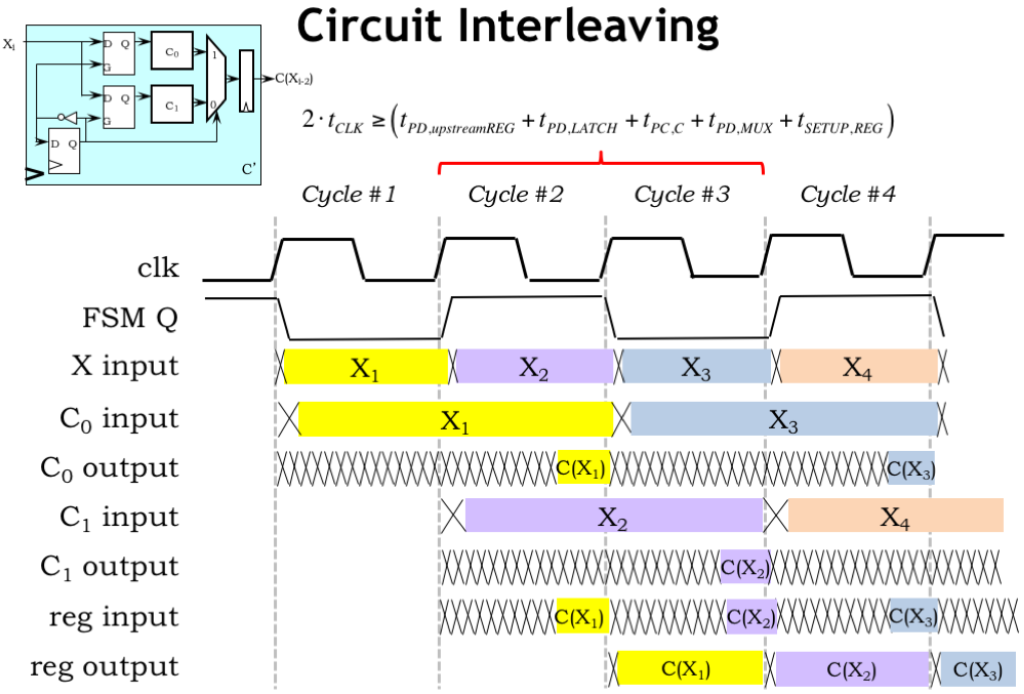
吞吐量受制于C电路的8ns，找到流水线版的C，或交替使用多个C

## 电路交替1



两个C模块，交替输出，从8ns降到4ns

## 电路交替2

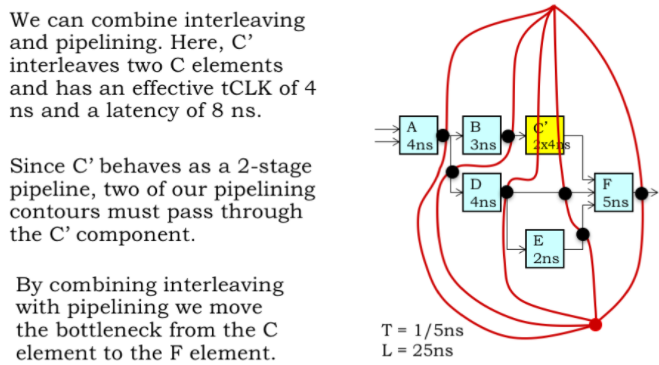


虽然输入到输出的延迟是两个时钟周期，但吞吐却是每个时钟周期1个

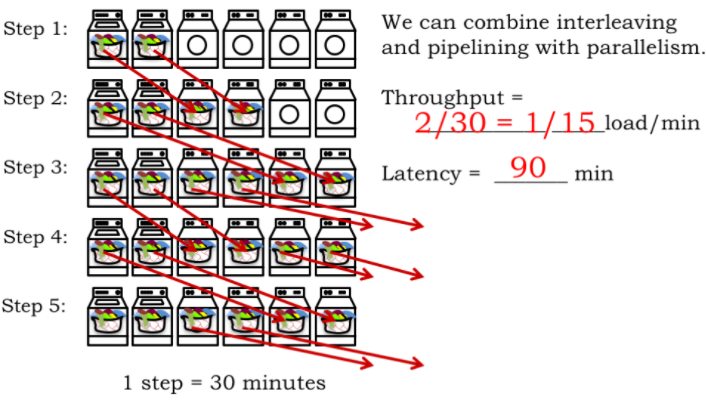
## 电路交替3

在流水线中，N路交替可以等价为N状态流水线

## 组合技术



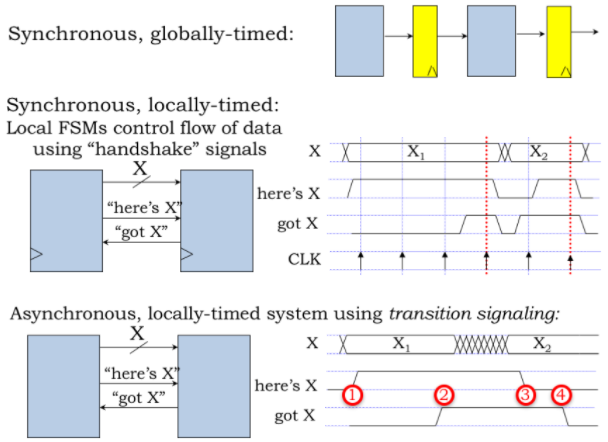
## 添加一点并行



两组：1个washer+2个dryer，可以让吞吐达到15分钟1件衣服

通过多个交替可以降低时钟周期，但并非无限的，终究受限于寄存器延迟

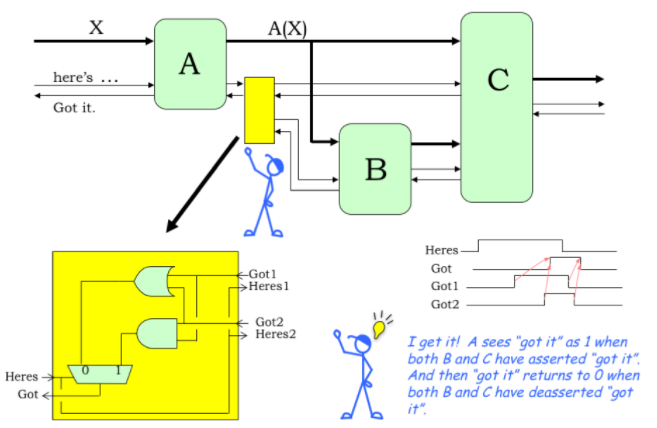
## 可选的控制结构



之前设计的是同步、全局时钟的。

某个步骤利用局部时钟，加快处理。

## 自计时例子

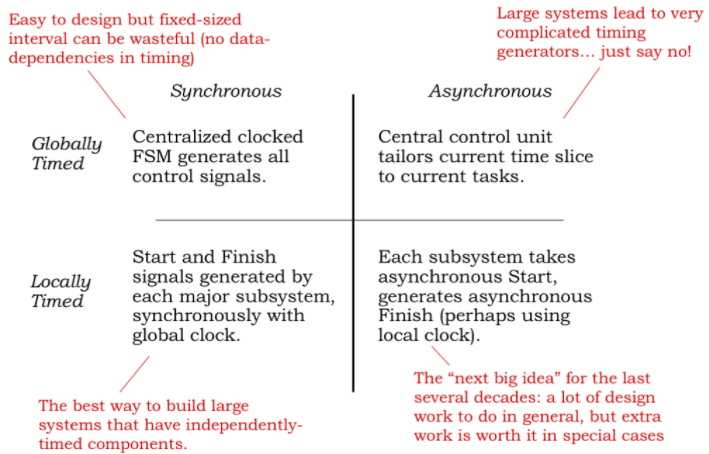


阶段1：上游有输出时，输出赋1。阶段2：下游拿到上游输出并消费完后，输出赋1。

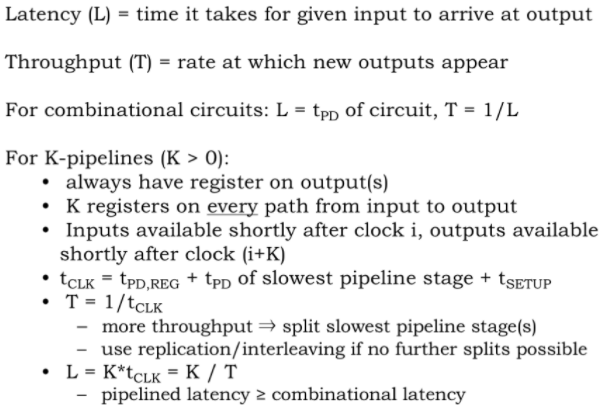
阶段3：上游得知下游已经消费完后，输出赋0。阶段4：下游得知上游赋0后，输出赋0

## 自计时例子2

## 控制结构分类



## 总结



# 习题

## 问题1

A：12、1/12

B：12、1/6

C：18、1/6

D：16、1/4

## 问题2

A：15、1/3

B：16、1/2

## 问题3

A：180、1/180

BC：180、1/60

## 问题4

A：24、1/24

BC：24、1/4

## 问题5

A：22、1/22

B：27、1/9

C：

6ns：invalid

7ns：valid、28、1/7、0、1-2

## 问题6

1/3

## 问题7

A：21、1/7

B：25、1/5