

跨时钟域电路设计

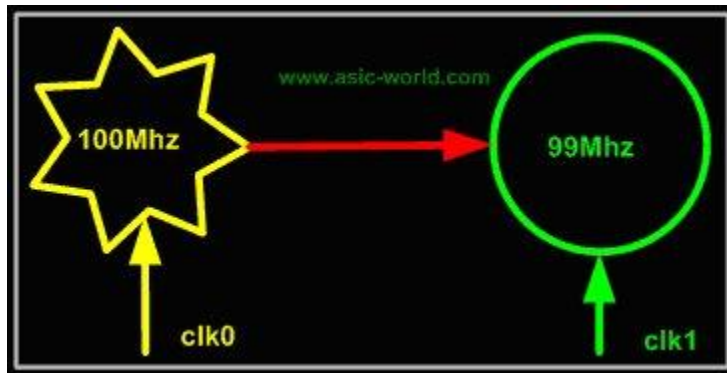
2014年8月
钟杨源

主要内容

- 1.单一信号跨时钟域处理
- 2.总线数据的跨时钟域处理
- 3.数据流的跨时钟域处理
- 4.相关时钟系统数据同步

跨时钟域问题 (Clock Domain Crossing)

- 跨时钟域有以下两种情况：
- (1)两个不同的频率时钟域进行数据交换；



- (2)两个时钟域工作频率相同，但是相位不相同；



跨时钟域问题可能出现的问题和危害

- 1.亚稳态问题
- 2.数据采样错误
- 3.时序收敛

一、单一信号跨时钟域处理

- 慢时钟域→快时钟域

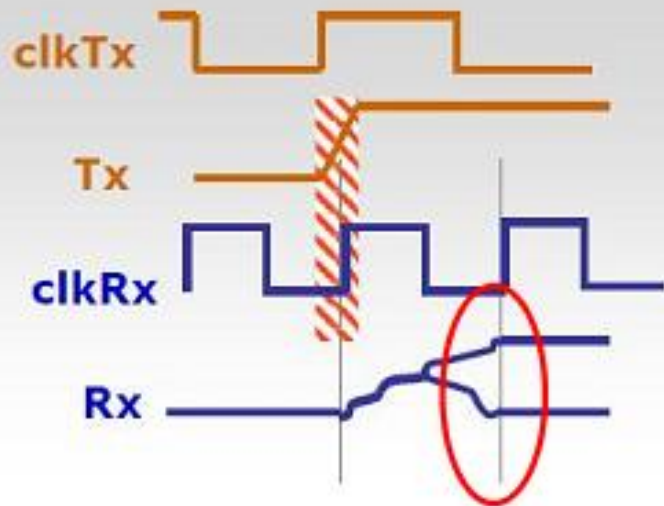
双寄存器同步电路或脉冲边沿采样电路

- 快时钟域→慢时钟域

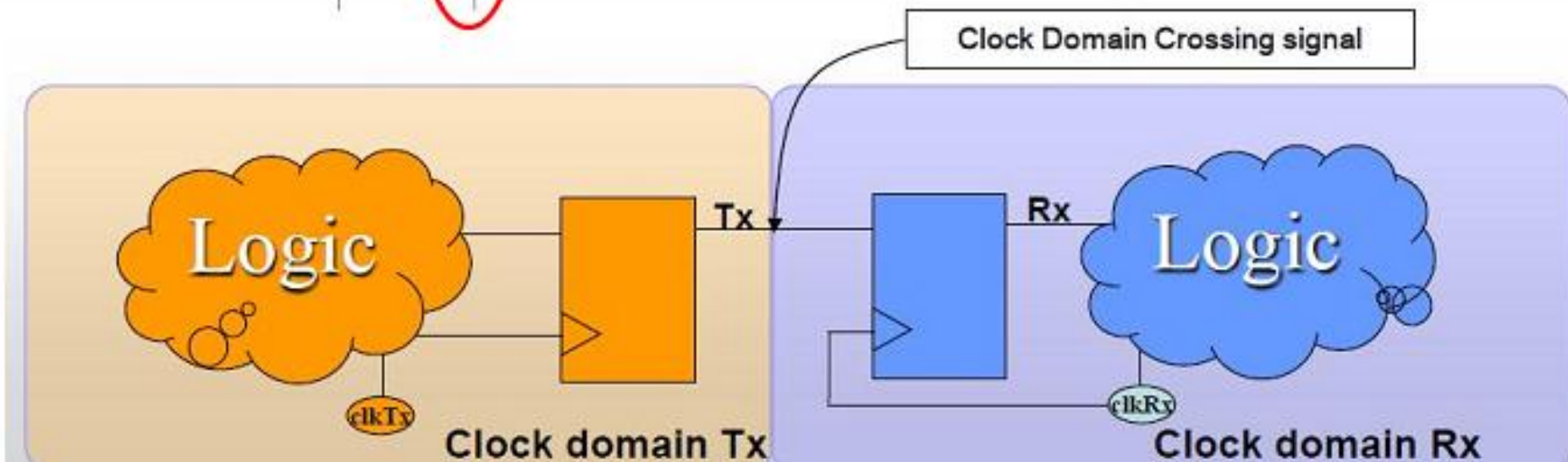
拉伸脉宽

脉冲同步器（结绳法）

亚稳态的产生



- 当建立/保持时间违反时，寄存器进入亚稳态，会在0和1之间反复振荡，直到稳定在一个未知的状态。如果让这个振荡传导到后级逻辑，会导致电路工作混乱。



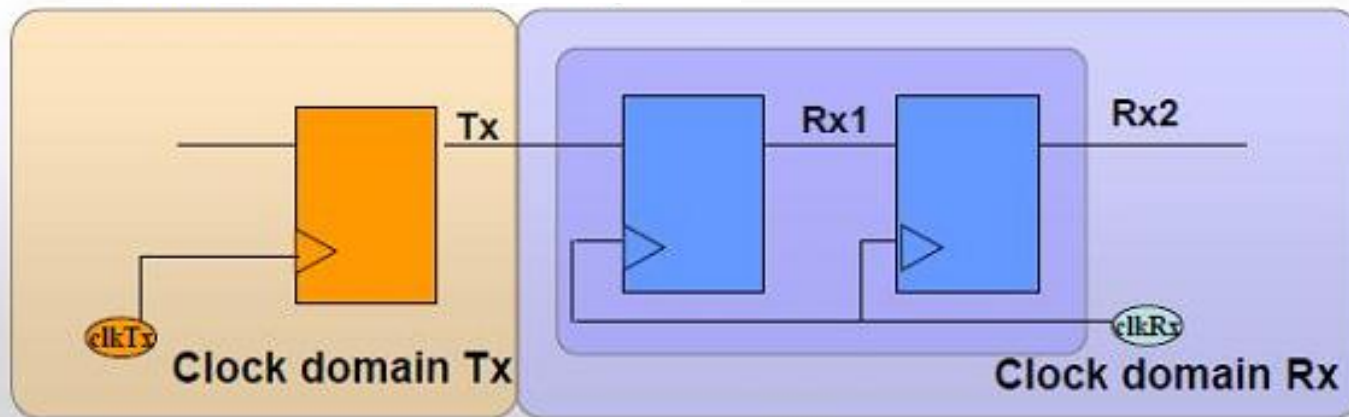
平均失效时间

触发器进入亚稳态的几率可以用触发器采样的平均失效时间间隔 MTBF(Mean Time Between Failures)描述。

若MTBF很大，就认为该设计在实际工作中能够正常运行。

$$\text{MTBF} = \frac{\Sigma (\text{Up Time})}{\text{Number of Failures}} \approx \frac{e^{T/\tau}}{f_d \times f_c \times T_w}$$

- T = 分辨时间（亚稳态转稳定时间）
- τ = 触发器参数
- T_w = 敏感时间窗口参数
- f_d = 输入数据的跳变频率
- f_c = 采样时钟频率



平均失效时间的计算

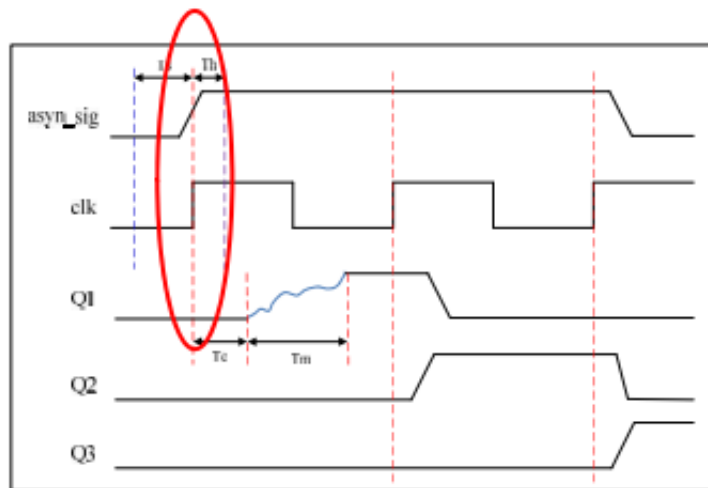
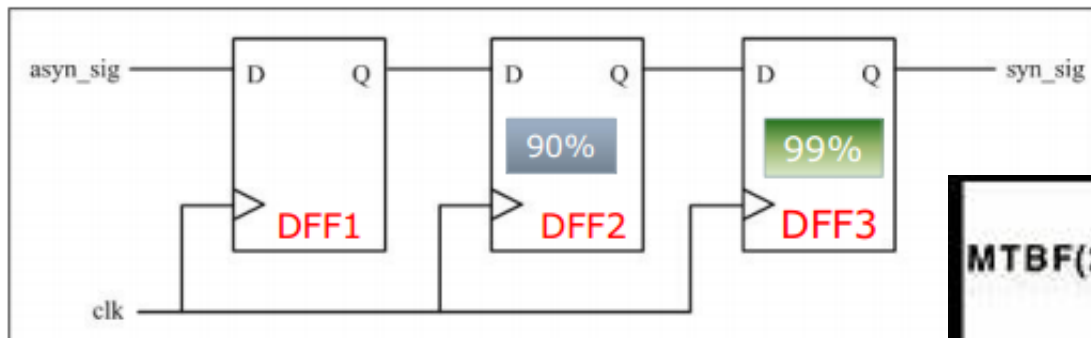
对于一个典型的 $0.25\text{ }\mu\text{m}$ 工艺的ASIC库中的一个触发器，我们取如下的参数：

$T = 2.3\text{ns}$, $\tau = 0.31\text{ns}$, $T_w = 9.6\text{as}$, $f_c = 100\text{MHz}$, $f_d = 10\text{MHz}$, $\text{MTBF} = 2.01\text{ days}$

即触发器每两天便可能出现一次亚稳态，说明一级同步很容易出现问题。

单信号：慢时钟域→快时钟域

- 如果我们仍然使用同样的参数，则第2级触发器的MTBF为 9.57×10^9 (years)。这个值已经足以使系统长期稳定。



$$MTBF(2) = \frac{e^{(T \times \Delta t)}}{f_{in(2)} \times f_{CLK} \times T_0}$$

$$f_{in(2)} = \frac{1}{MTBF(1)}$$

$$= \frac{f_{in(1)} \times f_{CLK} \times T_0}{e^{(T \times 1/f_{CLK})}}$$

$$MTBF(2) = \frac{e^{(T \times t)} \times e^{(T \times 1/f_{CLK})}}{f_{in(1)} \times f_{CLK}^2 \times T_0^2}$$

总结

- 问题1：为何直接对异步信号进行采样会不稳定？
- 问题2：要使用几级寄存器进行同步最合适？
- 问题3：同步寄存器电路可以防止亚稳态产生吗？

总结

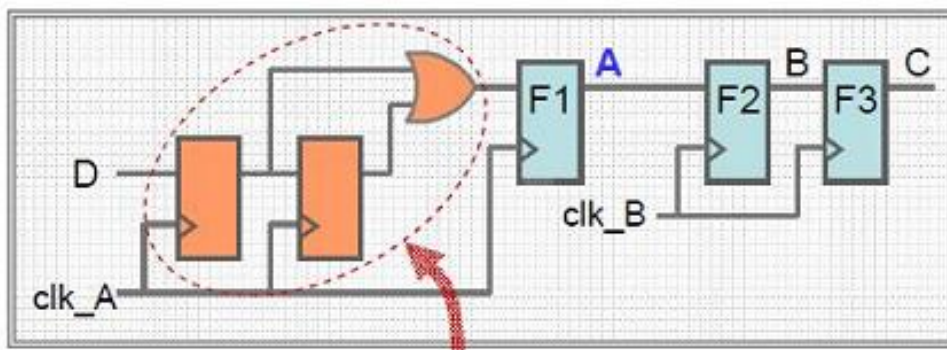
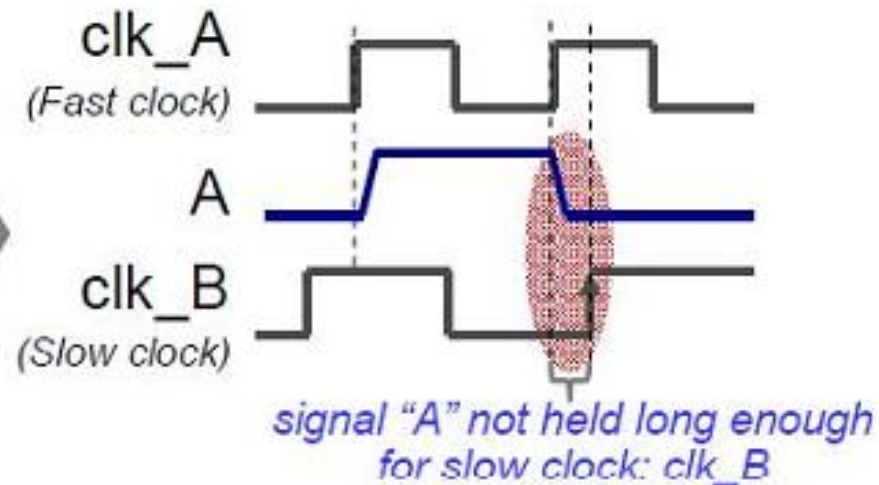
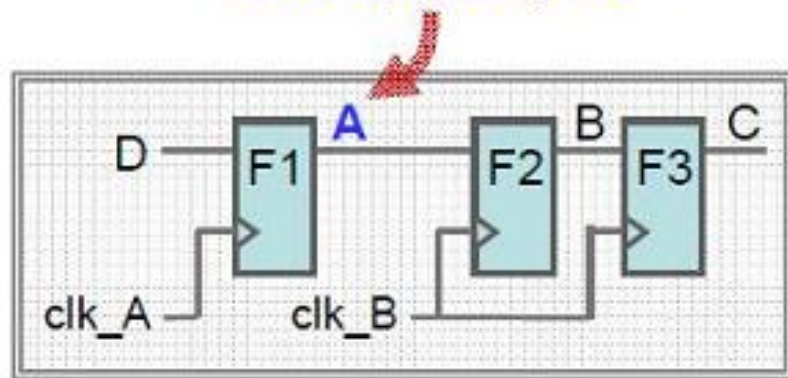
- 问题1：为何直接对异步信号进行采样会不稳定？
- 答：采样时会产生亚稳态
- 问题2：要使用几级寄存器进行同步最合适？
- 答：2级同步器的平均失效时间已经足够大，
3级同步影响速度
- 问题3：同步寄存器电路可以防止亚稳态产生吗？
- 答：亚稳态在异步电路中是不可避免的，同步电路的作用只是为了不让亚稳态传播。

单信号：快时钟域→慢时钟域

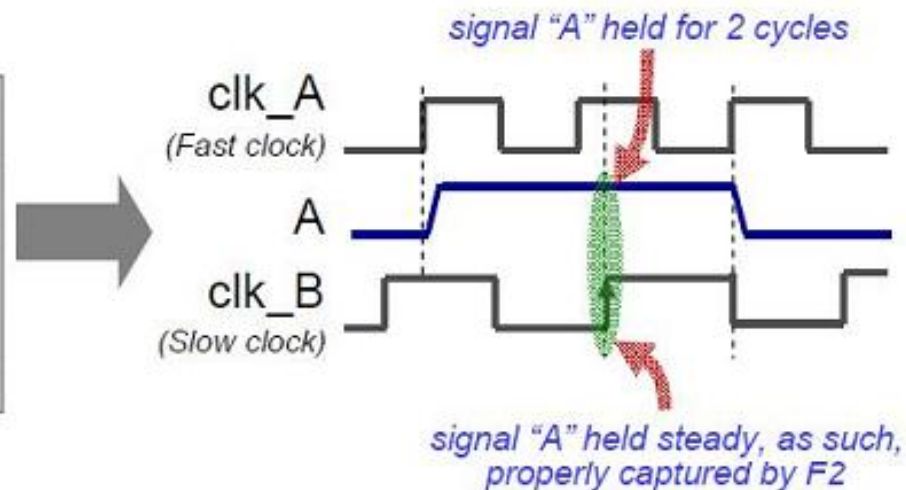
- 当快时钟域的信号跳变太快，导致目标时钟域无法采样时，应如何处理？
- 1.拉伸快时钟域信号的脉冲宽度使其足以被正确采样
- 2.结绳法（脉冲同步器）

① 脉冲拉伸法

Will signal "A" be held long enough to be captured by F2?



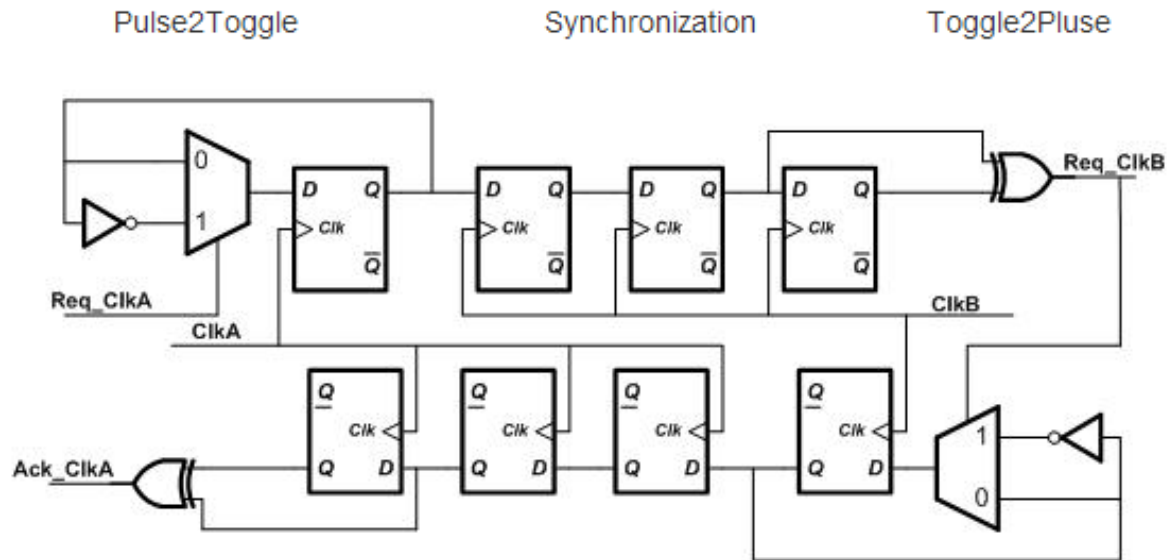
脉冲拉伸同步电路



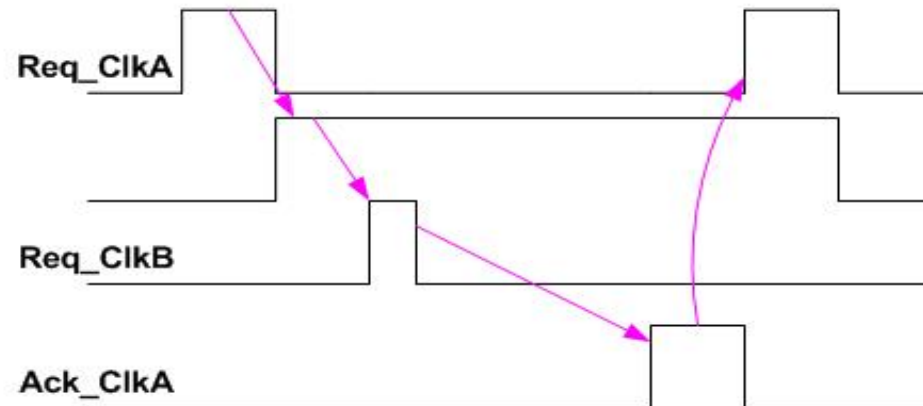
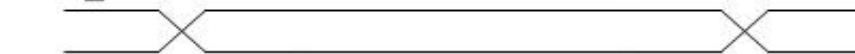
② 结绳法

- 信号从快时钟域向慢时钟域过渡时，慢时钟将可能无法对变化太快的信号实现正确采样。
- “结绳法”原理：将快时钟信号的脉冲周期延长，等到慢时钟同步采样后再“解绳”还原为原来的脉冲周期宽度。

单信号：快时钟域→慢时钟域 结绳同步电路



Data_ClkA



Synopsys DW库内的推荐电路

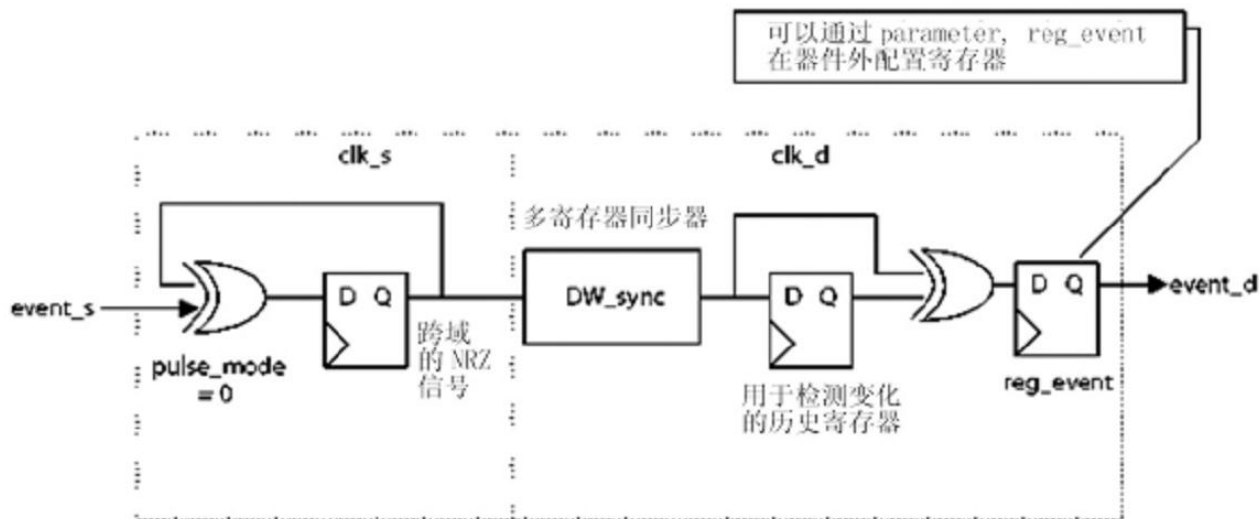


图 9 DWpulse_sync 双时钟脉冲同步器

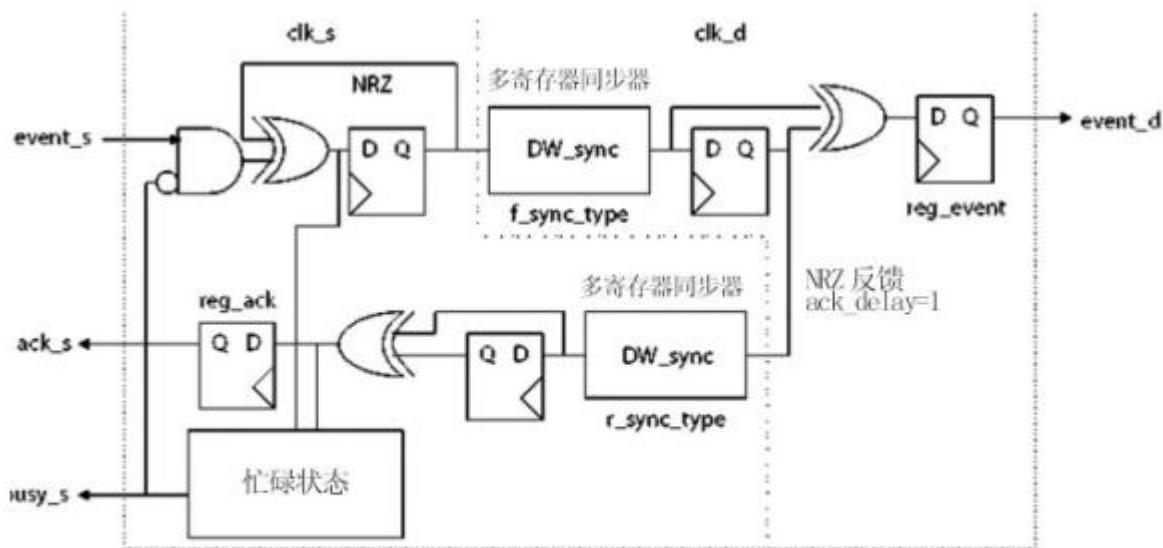


图 10 加了确认的 DWpulseack_sync 脉冲同步器

脉冲同步器使用规则

- 1. 总线数据不能对每个信号单独使用同步器，会导致数据无法同时有效。
- 2. 输入的信号必须是单周期脉宽，否则无法成功进行“结绳”操作
- 3. 输入脉冲之间的最小间隔必须等于两个同步器时钟周期。如果输入脉冲相邻太近，则同步器就不能检测到每个脉冲。

问题

- 1.信号从快时钟—>慢时钟与慢时钟—>快时钟在考虑问题时有什么差别？
- 2.为什么不能直接用慢时钟采样快时钟域的数据？
- 3.快时钟—>慢时钟域同步的核心思想应该是什么？

问题

- 1.信号从快时钟→慢时钟与慢时钟→快时钟有什么差别？
- 慢时钟→快时钟只要考虑亚稳态问题，
- 快时钟→慢时钟的情况还要考虑采样速率的问题
- 2.为什么不能直接用慢时钟采样快时钟域的数据？
- 根据奈奎斯特采样定理，当采样频率低于信号最高频率的2倍时，是无法完整正确采样的。
- 3.快时钟→慢时钟域同步的核心思想应该是什么？
- 握手机制

主要内容

- 1.单一信号跨时钟域处理
- 2.总线数据的跨时钟域处理
- 3.数据流的跨时钟域处理
- 4.相关时钟系统数据同步

二、总线数据同步

- 跨时钟域对总线进行采样时，数据总线上有可能会出现瞬时虚假数值。例如，一条8位总线上有3个位发生了改变，若信号到达时间不一致，就会有看到8个可能出现的数值。
- 1.慢时钟域→快时钟域
 - （1）数据编码采用格雷码
 - （2）结绳法处理有效信号
- 2.慢时钟域→快时钟域
 - 带回复的结绳法

总线信号：慢时钟域→快时钟域

处理方法1：采用格雷码

- 格雷码又称独热码，相邻值只会有一位跳变
- 在任意时刻，总线上的数据只有一位变化，就不会出现虚假数据状态
- 缺点：只有在数据在相邻数值间跳变的情况才有用，不适用大多数情况

十 进 制 数	二 进 制 码	格 雷 码
0	0 0 0 0	0 0 0 0
1	0 0 0 1	0 0 0 1
2	0 0 1 0	0 0 1 1
3	0 0 1 1	0 0 1 0
4	0 1 0 0	0 1 1 0
5	0 1 0 1	0 1 1 1
6	0 1 1 0	0 1 0 1
7	0 1 1 1	0 1 0 0
8	1 0 0 0	1 1 0 0
9	1 0 0 1	1 1 0 1
10	1 0 1 0	1 1 1 1
11	1 0 1 1	1 1 1 0
12	1 1 0 0	1 0 1 0
13	1 1 0 1	1 0 1 1
14	1 1 1 0	1 0 0 1
15	1 1 1 1	1 0 0 0

总线信号：慢时钟域→快时钟域

如果目的时钟要求满足

$$F_{clk_d} \geq F_{clk_s} * (f_sync_type + 1.25)$$

就可以在不到源域1个时钟周期的时间内捕获数据。因此也就不需要回复握手信号。这样可以最大程度地减少时延和或不必要的逻辑电路。

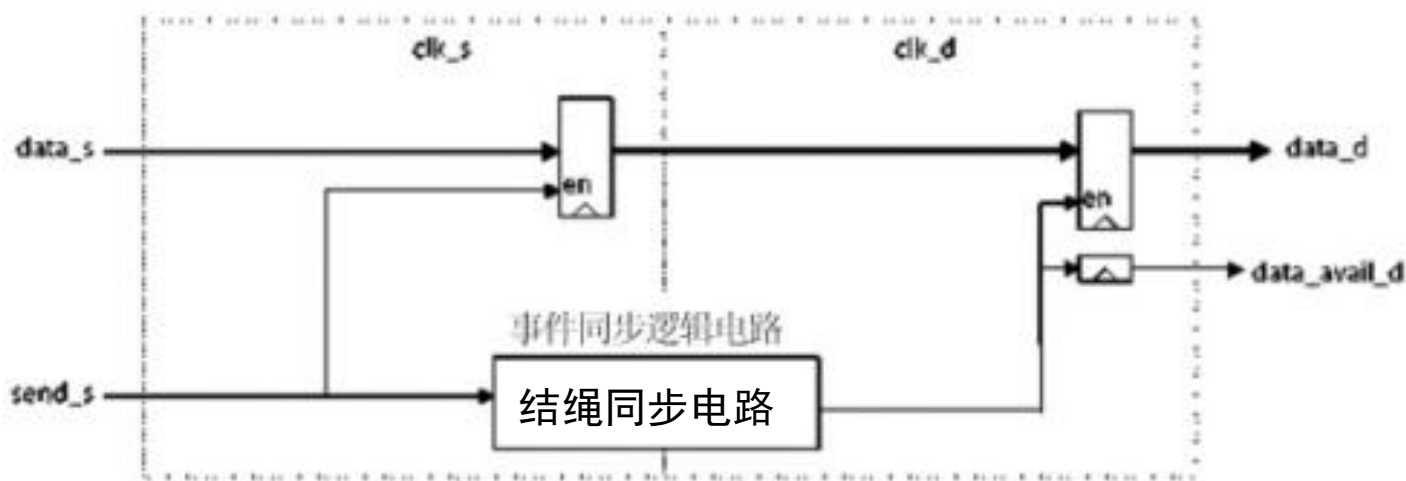


图 13 不带 ac 确认信号的 DW_data_sync_na 总线同步器

总线信号：快时钟域→慢时钟域

握手脉冲同步器

- 升级版结绳法：带握手确认的脉冲同步器
- 数据同步交换顺序为：保持数据，发送请接受，发回已接受，释放数据寄存器

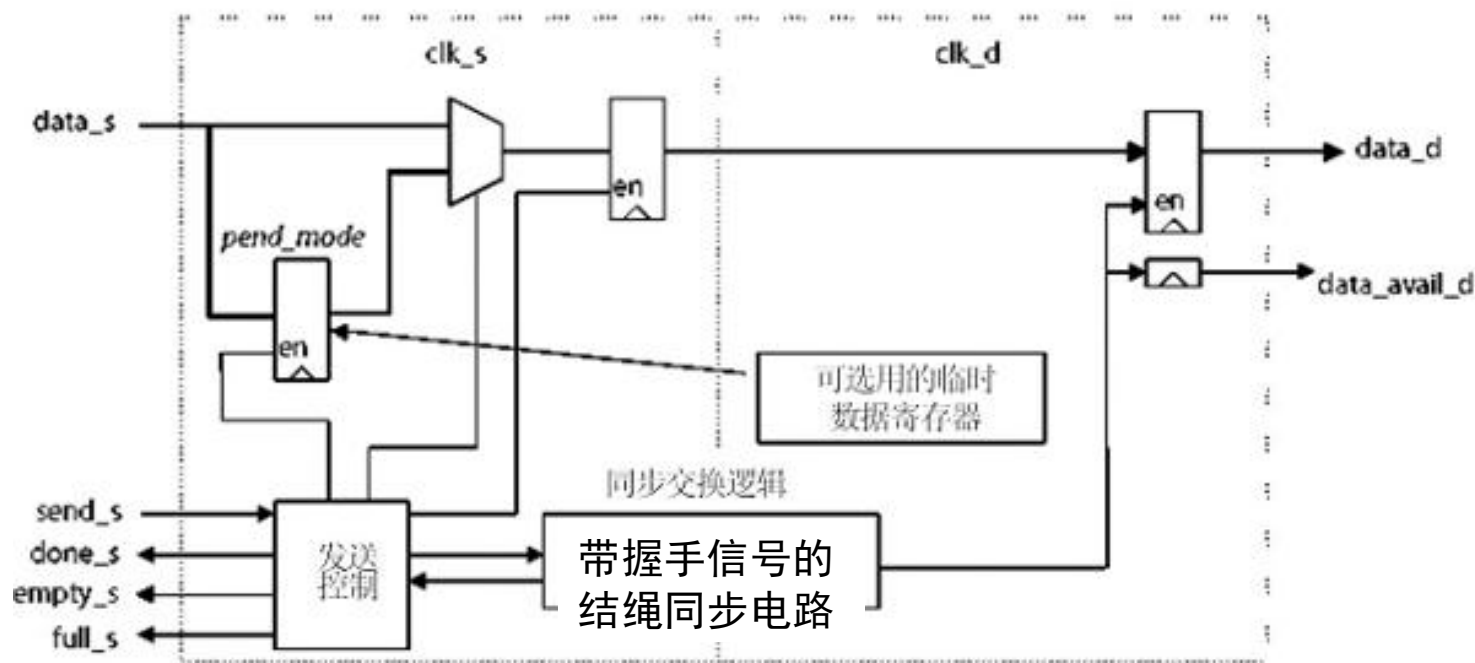


图 12 带确认的 DW_data_sync 总线同步器

主要内容

- 1.单一信号跨时钟域处理
- 2.总线数据的跨时钟域处理
- 3.数据流的跨时钟域处理
- 4.相关时钟系统数据同步

三、数据流同步

- 数据流与数据的不同：
 - 1.数据流大多具有连续性，即背靠背传输；
 - 2.数据流要求信号具有较快的传输速度
- 最好的解决方法：
- 异步FIFO

DESIGNWAVE库中的双时钟FIFO

该模块依赖于双端口 RAM 的使用，源域内的推送逻辑电路负责将数据写入到 RAM，而目的域内的弹出逻辑电路负责将数据从 RAM 读取

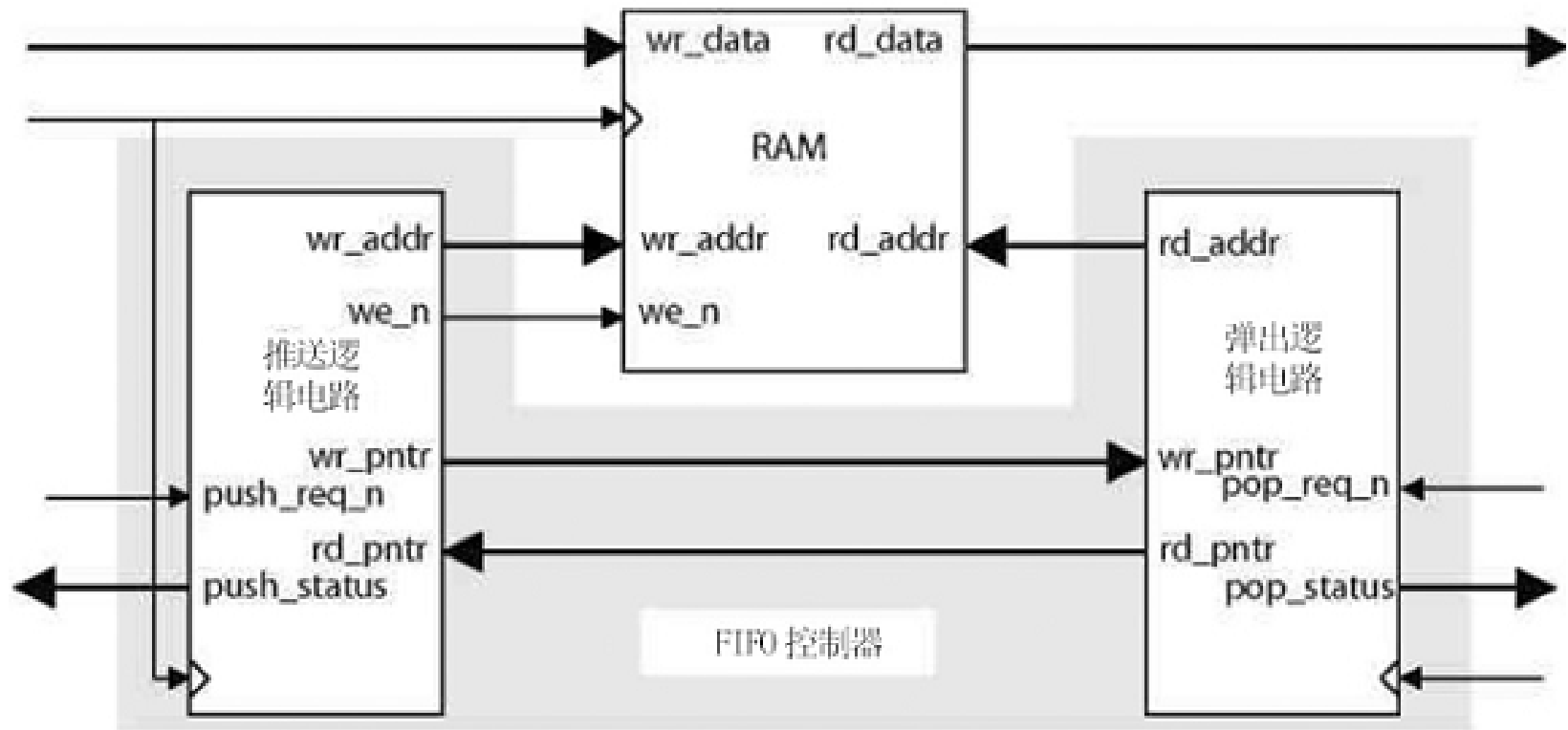


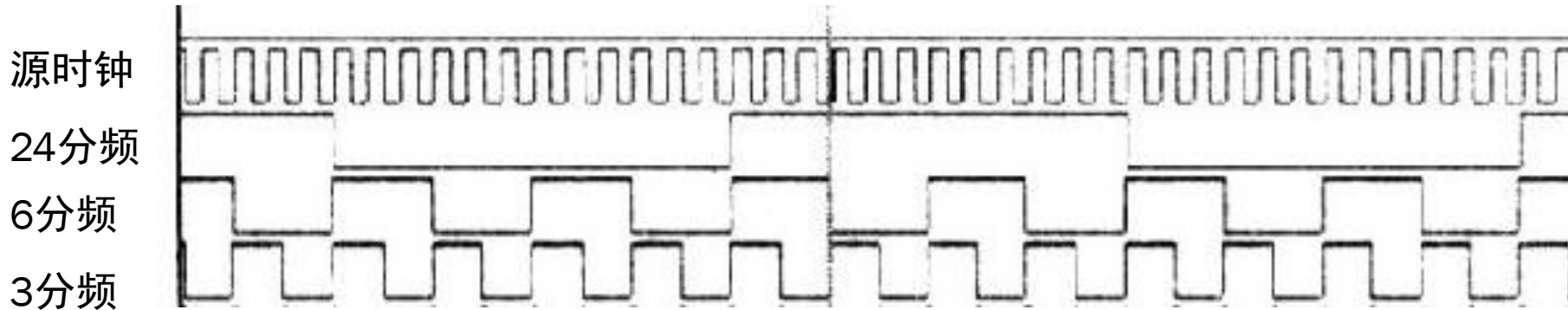
图 15 配合双端口 RAM 的双时钟 FIFO

主要内容

- 1.单一信号跨时钟域处理
- 2.总线数据的跨时钟域处理
- 3.数据流的跨时钟域处理
- 4.相关时钟系统数据同步

相关时钟系统

- 如果两个时钟分别为同一个源时钟的分频或倍频时钟，则称为相关时钟
- 相关时钟的相位是可以通过工具进行对齐的
- 跨越的时钟域为相关时钟，这时可以用更简便的处理方式，简化同步电路



相关时钟域：慢→快时钟域

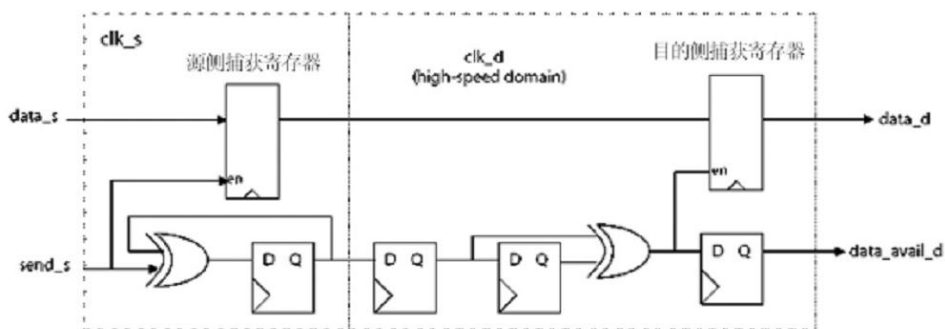


图 25a 大于 2 的时钟比率

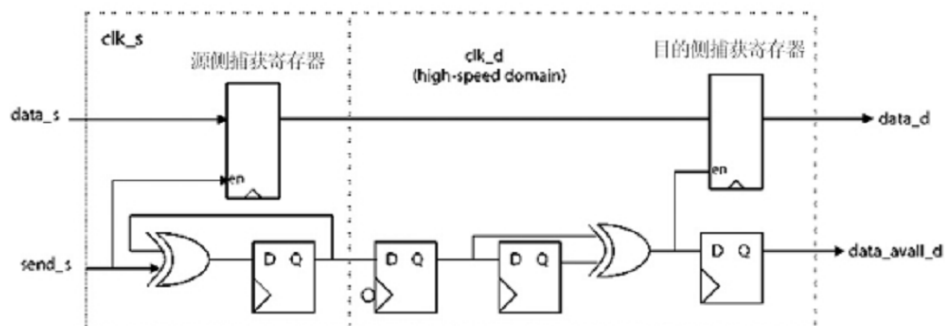
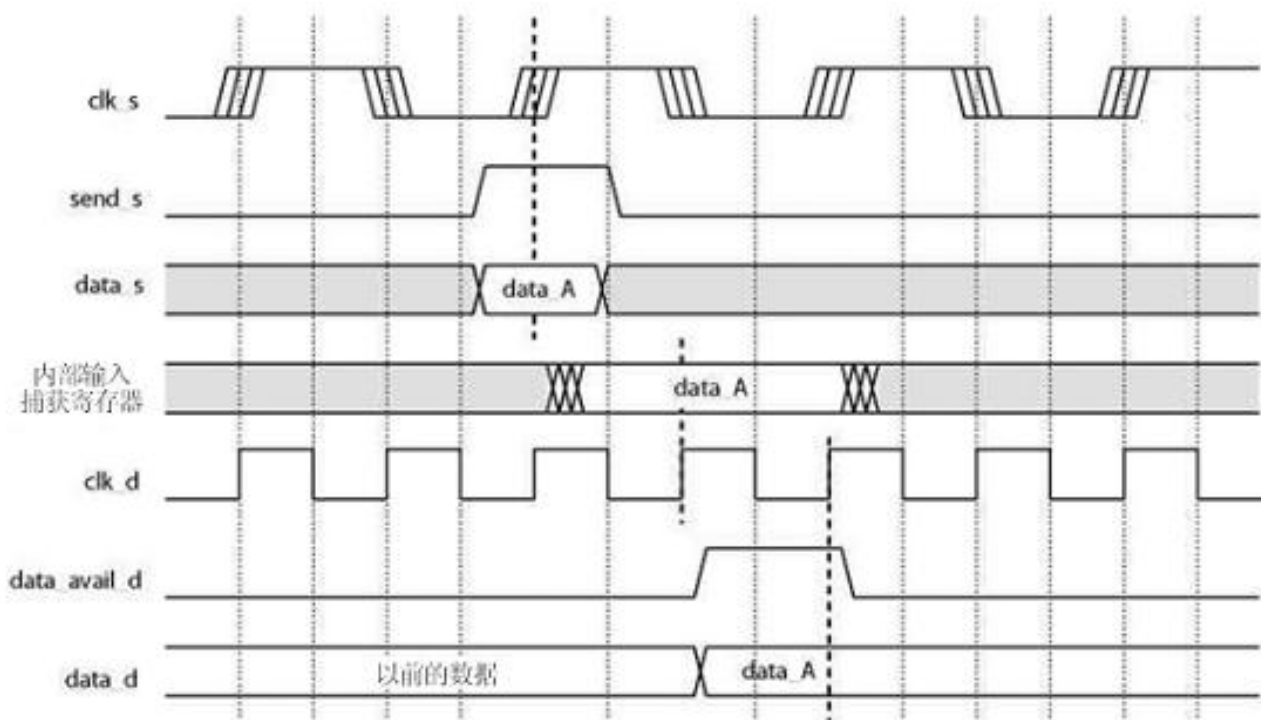


图 25b 等于 2 的时钟比率



相关时钟域：慢→快时钟域

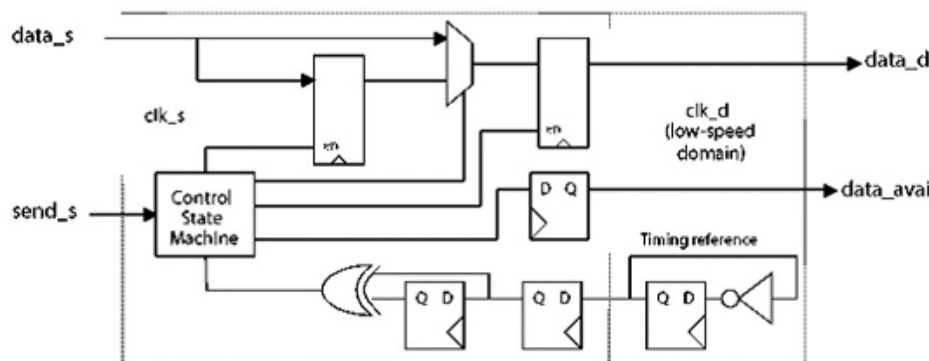


图 27a 时钟比率大于 2

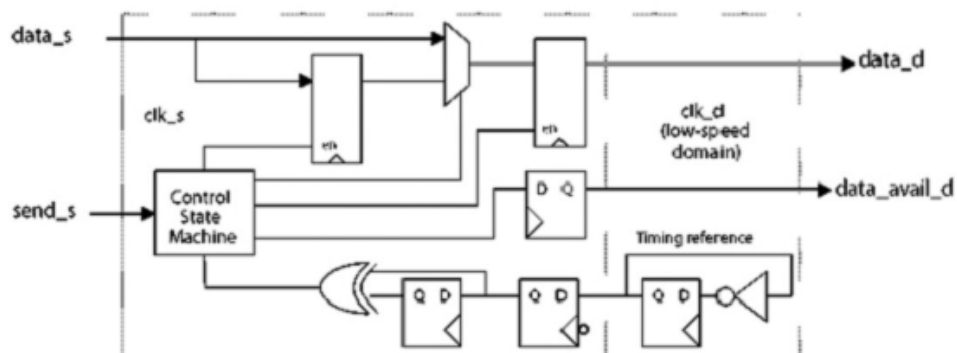
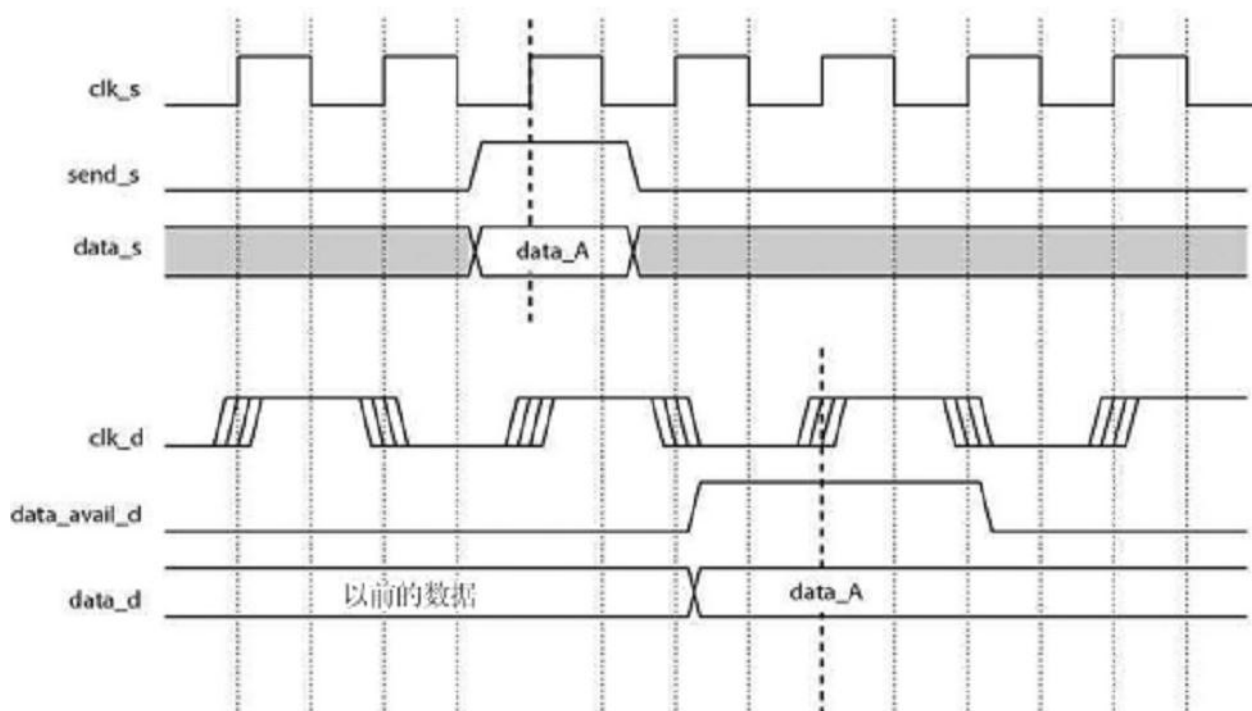


图 27b 等于 2 的时钟比率



CDC电路的实现手段

- 1.RTL设计

直接描述或例化synopsys的相关单元

- 2.综合方式

create_clock与set_dont_touch_network

- 3.工艺映射

有些工艺库有包含“亚稳态强化”触发器

工艺器件映射

- 在DC将两个触发器映射到一个目标库上时，综合工具会选择能够满足时序限制条件的最小面积的单元，而不是对同步最好的触发器
- 因此通常想要通过手动方式或通过脚本方式用已知的“亚稳态强化”触发器来替代这些触发器
- 自DC2009版本开始，可以指定一个或多个“亚稳态强化”触发器用在上述DW库中的CDC组件。
- 方法：通过将相关器件赋给“synlib_preferred_ffs”变量实现特定寄存器的调用

NEXT TIME

- 1.基于AMBA3.0的SOC设计原理
- 2.数字系统全局信号处理
- 3.其他.....?