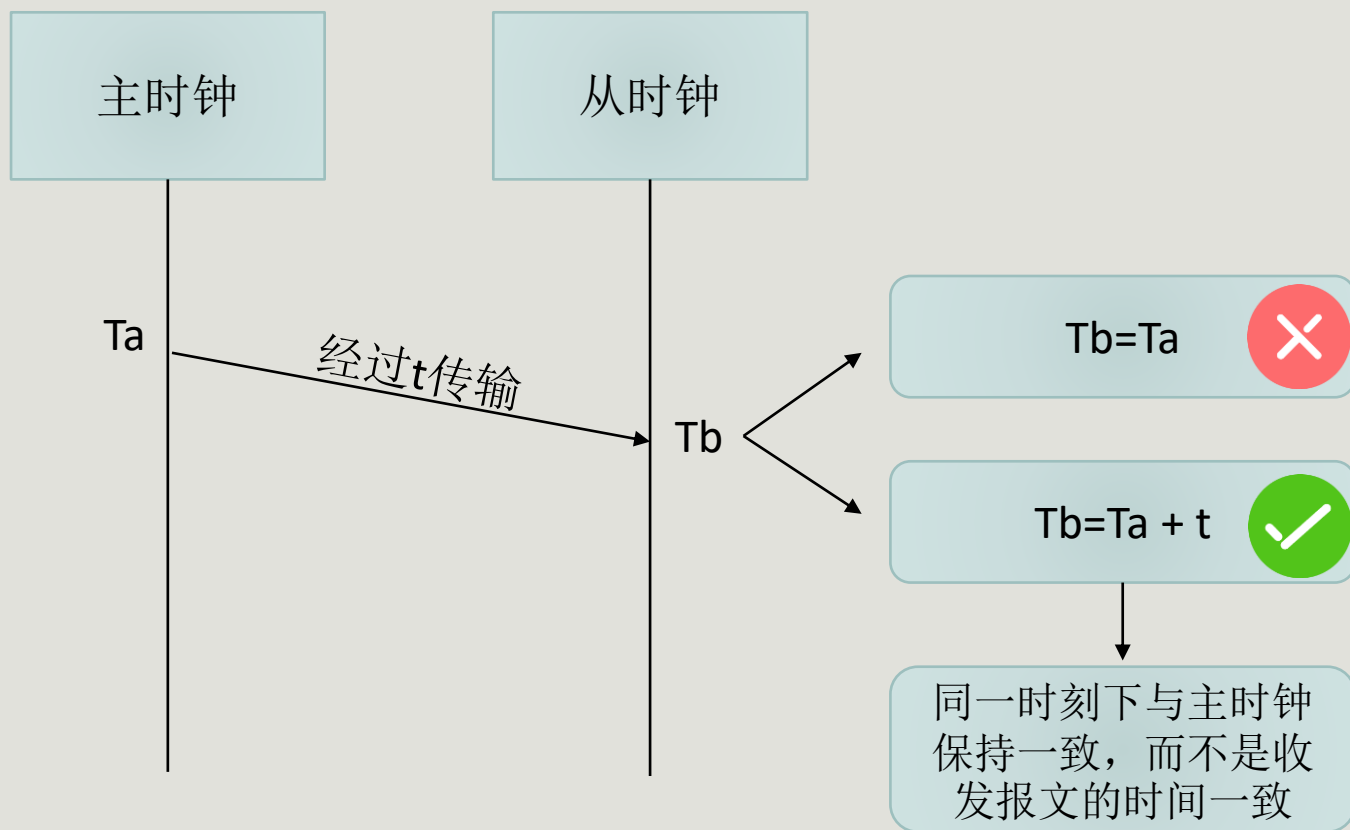


TSN时钟同步——gPTP

gPTP做了什么工作

系统内所有时钟同步
(以主时钟为基准)



gPTP系统组成



gPTP实现同步的流程

主时钟的选择

相邻节点之间的时钟同步 { 报文类型
时钟偏差
频率误差
传输延迟

总体同步

主时钟的选择

BMCA 最佳主时钟算法（也确定了整个系统的时间同步生成树）

- 网络启动，所有gPTP节点发送自身的**Announce**消息给到对端节点
- 接收到**Announce**消息的节点将比对以确定更适合的主时钟候选：（有更好的自己就停止发送）
- 最终经过各节点计算比较，将仅有一个节点的**Announce**消息得到保留，该节点被选为主时钟

Announce消息
时间的精确度
频率的稳定性
可追溯性

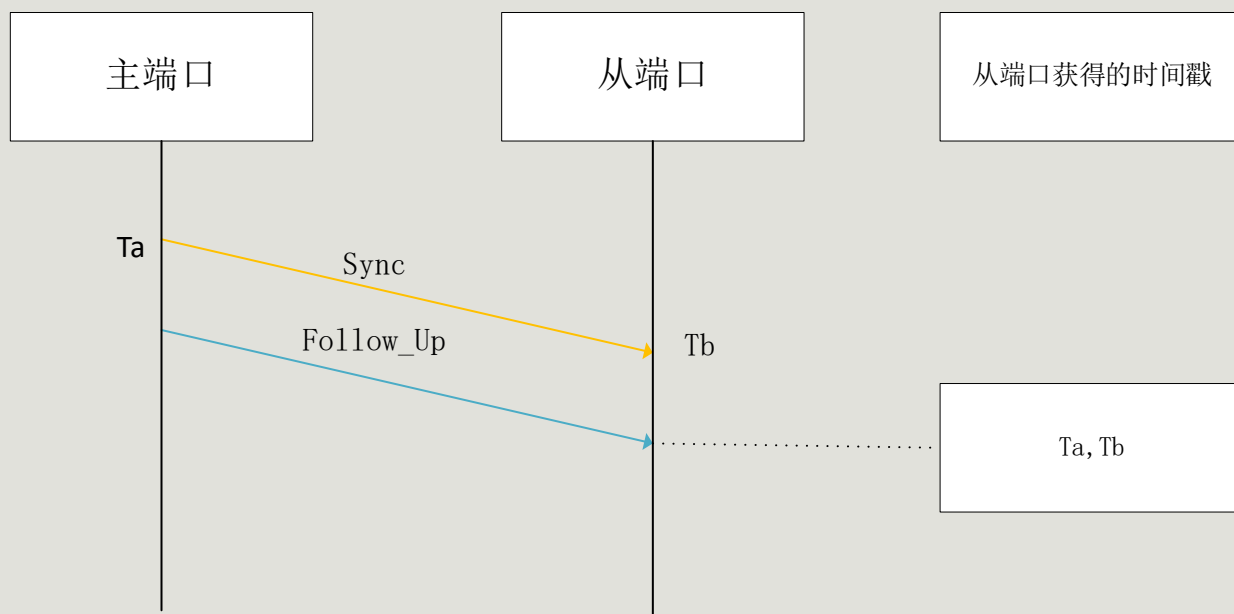
在已知有高精度时钟（**GPS**时钟）的情况下，可以不去计算最佳时钟，通过预设主时钟来实现主时钟的选择

相邻节点的同步——报文类型

一组报文	Sync	事件类	同步报文，周期发送（0.125s一次），触发记录本地时钟
	Follow_UP	一般类	紧随Sync报文之后发送，将Sync报文离开主端口时的时间传递给从时钟
一组报文	Pdelay_Req	事件类	延迟请求报文，周期发送（1s）触发记录本地时钟
	Pdelay_Resp	事件类	延迟响应报文，将Pdelay_Req报文到达的时间戳附在报文中传递，触发记录本地时钟
	Pdelay_Resp_Follow_UP	一般类	延迟响应跟随报文，将Pdelay_Resp报文离开主端口时的时间戳传递给从端口

- 事件类型报文： Sync、 Pdelay_Req、 Pdelay_Resp， 此类报文的接收和发送会触发MAC层对本地时钟进行采样；
- 一般类型报文： Follow_UP、 Pdelay_Resp_Follow_UP， 仅用来携带信息；（不仅仅只是携带时间戳）

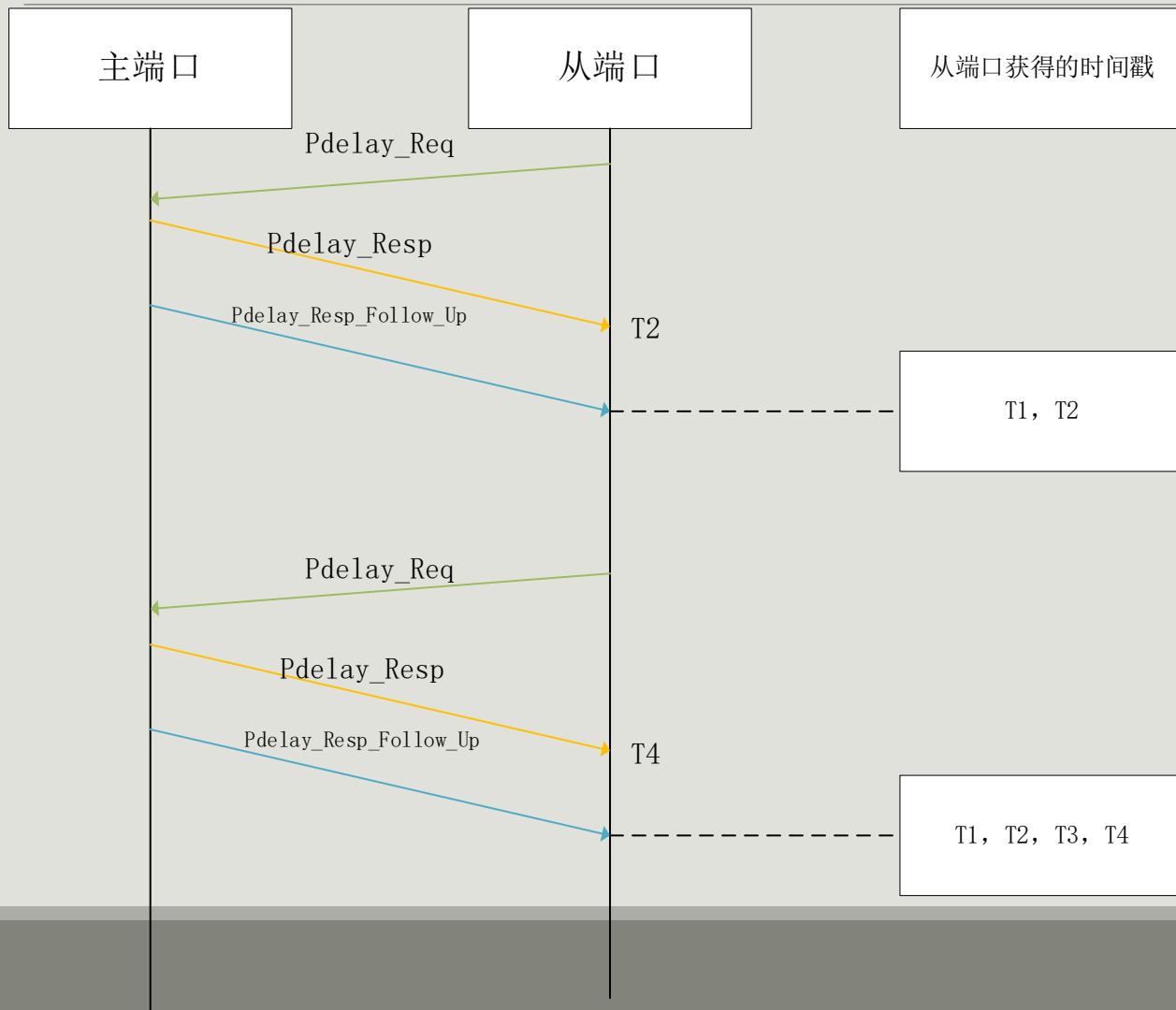
相邻节点的同步——时钟偏差



$$\text{时钟偏差} = T_b - T_a$$

通过时间偏差，调整本地时钟与主时钟一致，以实现时钟同步，但这样计算的时钟偏差，没有考虑传输过程中出现的一些延迟

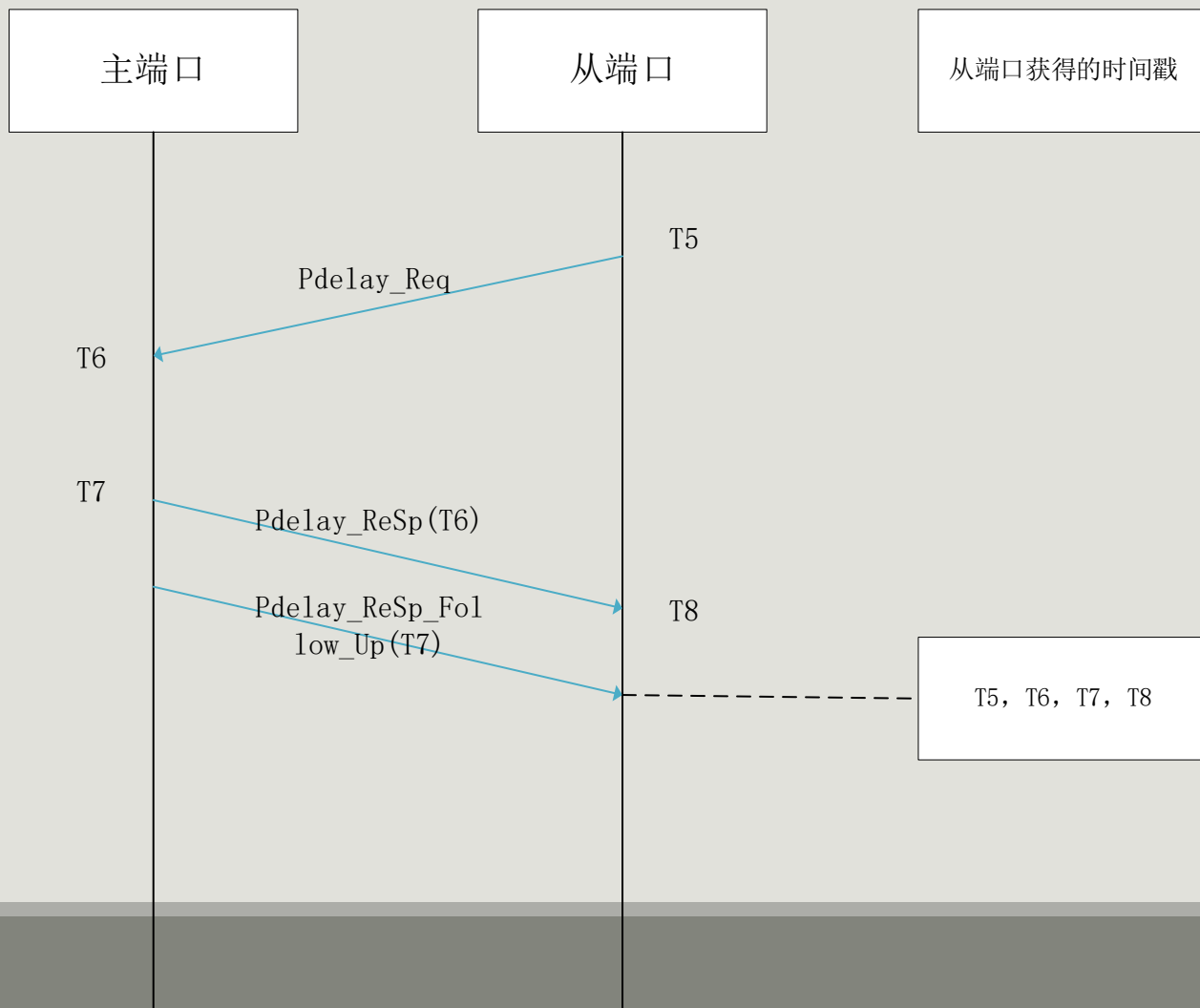
相邻节点的同步——时钟频率误差R



一个钟走了10s，另一个钟也走了10s，这叫频率同步，如果两个时钟的频率不同，那么随着时间的推移，它们之间的时间差异将会逐渐增大。因此计算传输延迟之前要考虑频率误差

$$R = \frac{T_3 - T_1}{T_4 - T_2}$$

相邻节点的同步——时钟传输延迟 P_{delay}



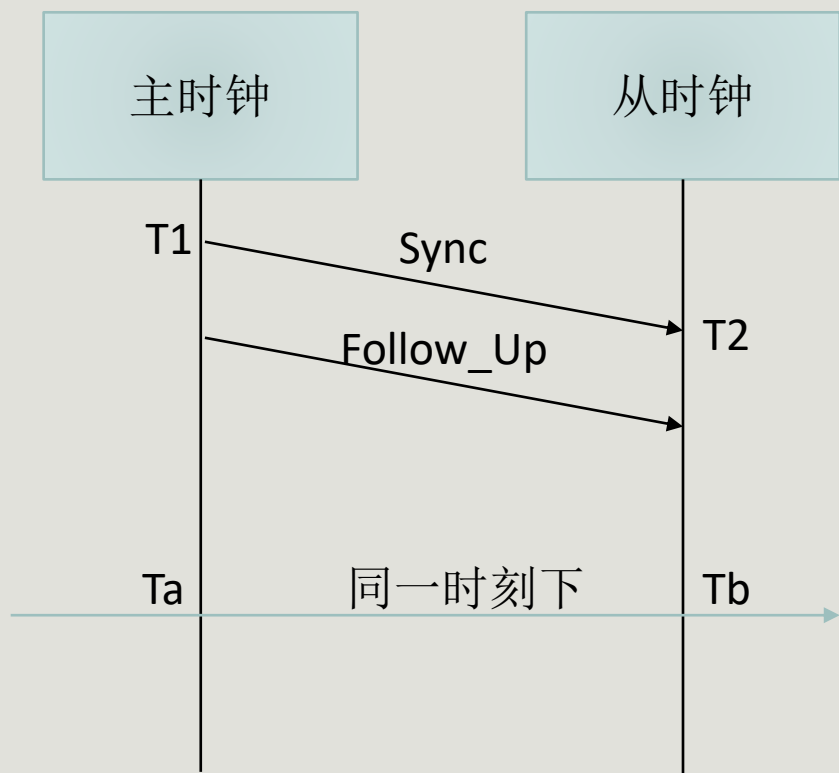
通过双向对称测量来计算传输延迟

$$P_{\text{delay}} = \frac{(T_8 - T_7) + (T_6 - T_5)}{2}$$

考虑频率误差

$$\begin{aligned} P_{\text{delay}} &= \frac{(T_8 * R - T_7) + (T_6 - T_5 * R)}{2} \\ &= \frac{(T_8 - T_5) * R + (T_7 - T_6)}{2} \end{aligned}$$

相邻节点的同步



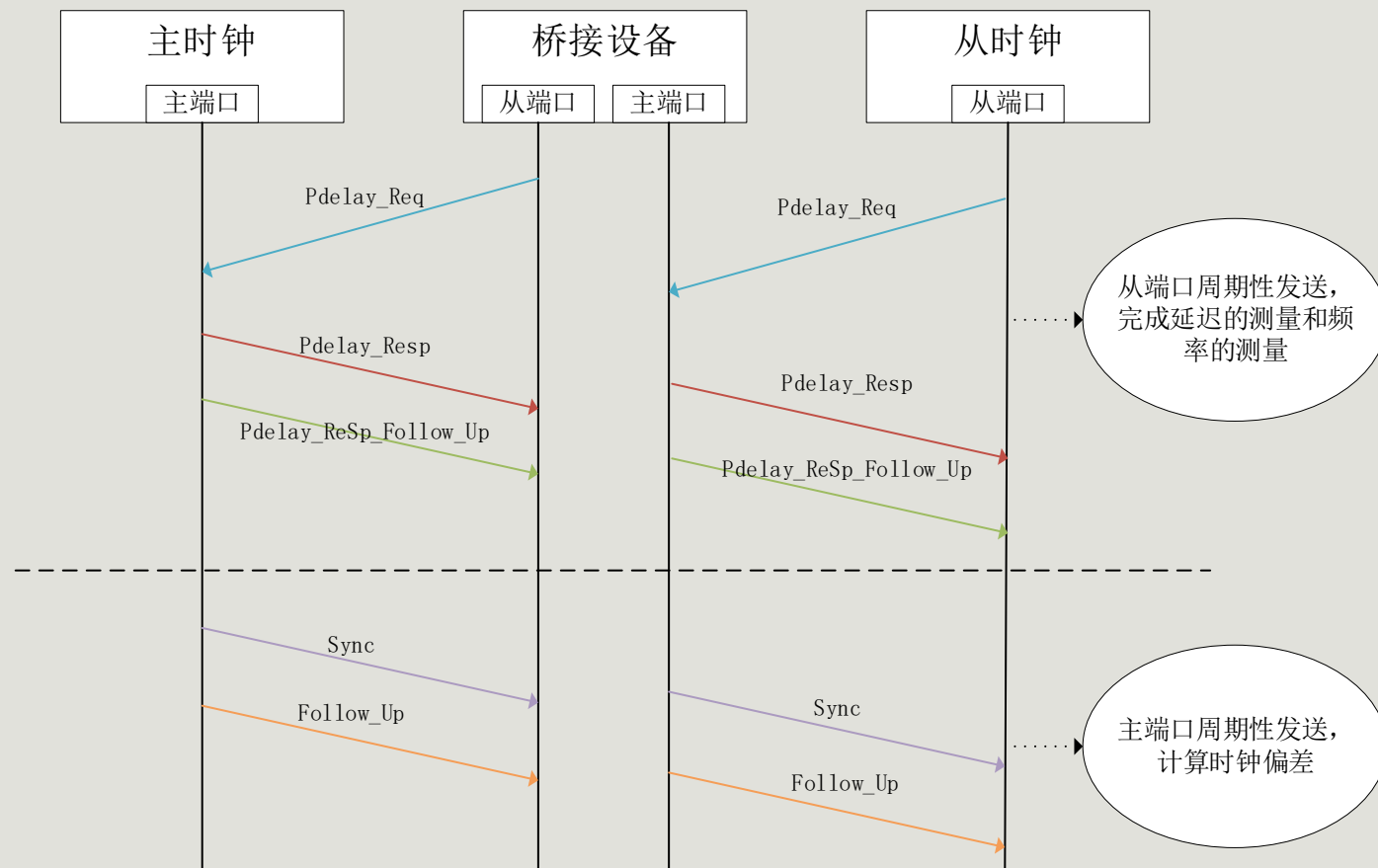
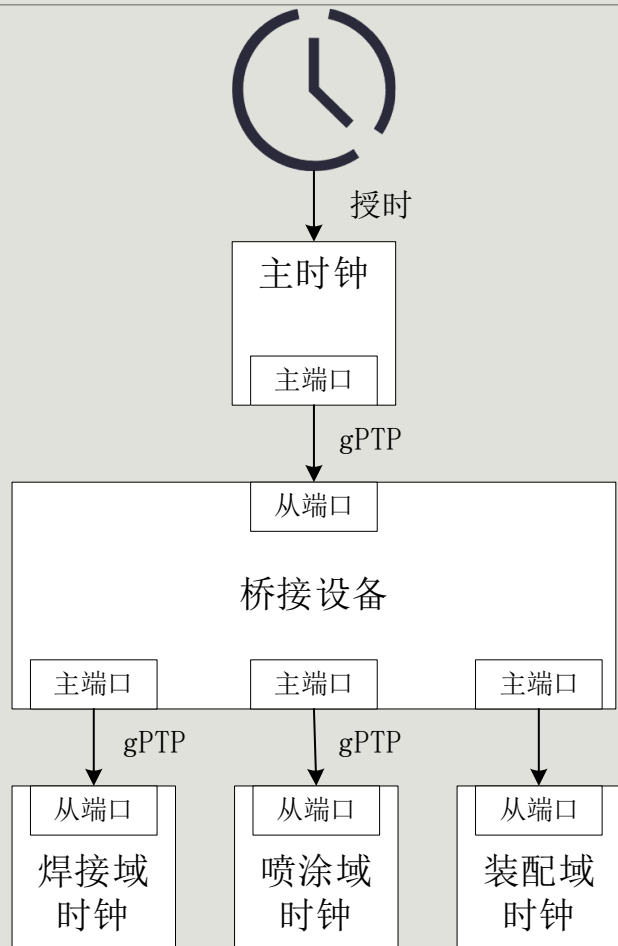
$$R = \frac{T_3 - T_1}{T_4 - T_2}$$

$$Pdelay = \frac{(T_8 - T_5) * R + (T_7 - T_6)}{2}$$

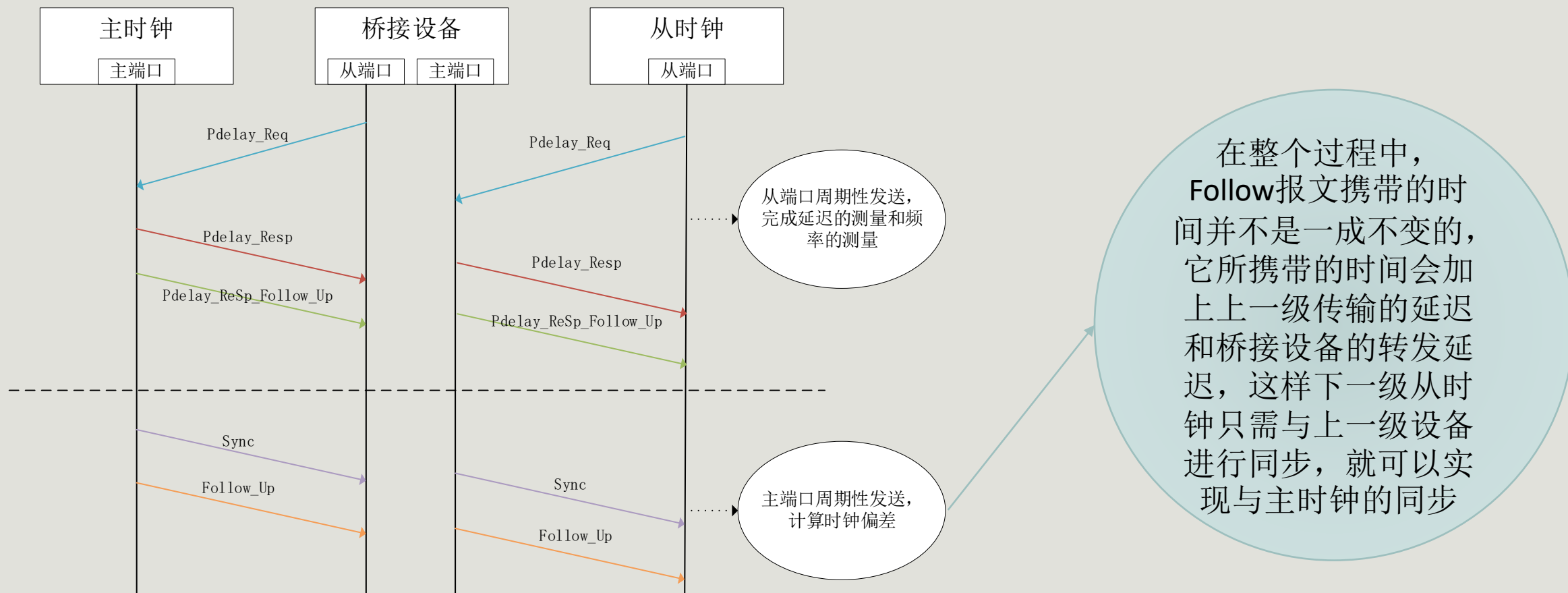
同一时刻下，从时钟可以通过Tb计算出主时钟的时刻Ta

$$Ta = T1 + (Tb - T2) * R + Pdelay$$

gPTP系统的同步



gPTP系统的同步



TSN时钟同步应用——实际场景中的应用

TSN交换机：支持多种TSN协议，主要用来控制流量和整形，不适合单独作为时钟同步的实现设备

gPTP系统控制程序：没有足够的硬件支持