Qbv计算工具介绍

Table of Contents

[1 背景 3](#_Toc38224703)

[2 工具介绍 4](#_Toc38224704)

[3 工具的使用 5](#_Toc38224705)

# 背景

TSN技术的一个重要子集是Time Aware Shaper / TAS，即802.1Qbv.

Qbv的原理如下

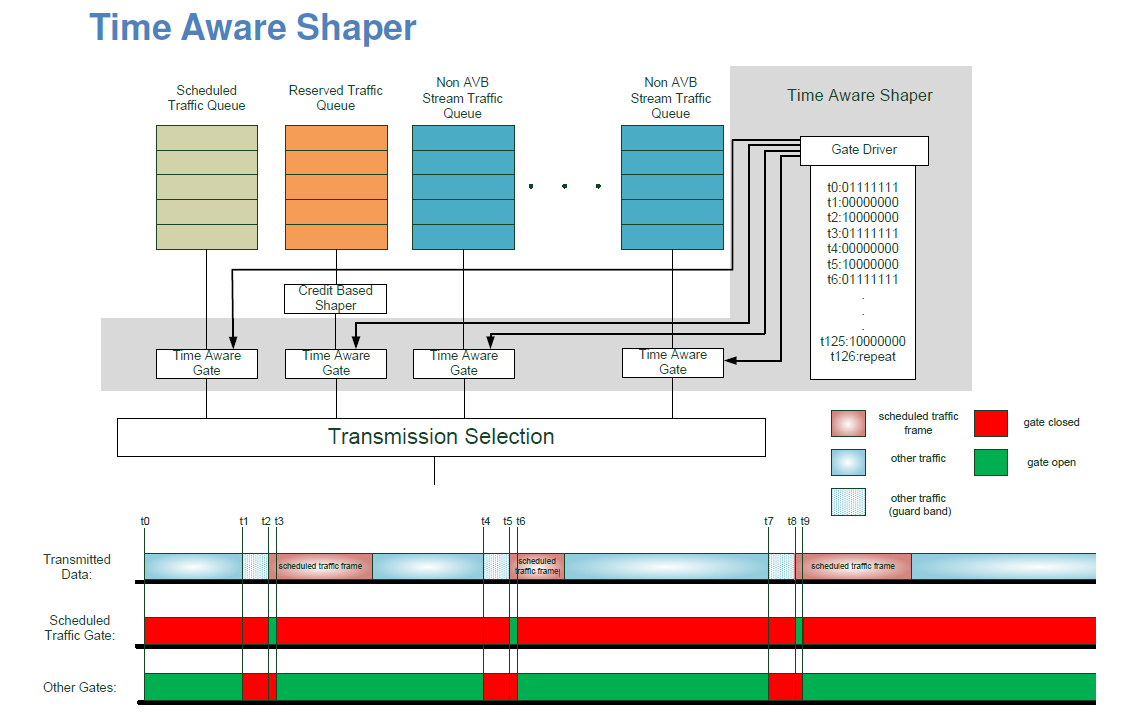


Figure 1

图引自Christian Boiger 的PPT Time Aware Shaper

在Qbv/TAS技术中，每个以太网网口都有如Figure 1上半部分所示的队列结构。整个网口被分成了8个队列（图中只画出来了4个）。另外还存在一个Gate Driver，里面包含了一串数据，来描述在某个时刻哪个队列被打开，哪个队列被关闭。把这样的数据在时间轴上平铺开来，就是Figure 1下半部分中显示出的情形。

在整个TSN系统中，每个以太网网口都需要被配置这样的一份调度表（Gate Control List/GCL，或称schedule table）。同时这些表之间还需要相互协调，以保证关键的报文能迅速地传到目的地。

由此可见调度表的设计是TSN系统设计的核心工作。同时由于参数数量多，因此在实际工作中需要依赖工具来计算整个系统的调度表。

# 工具介绍

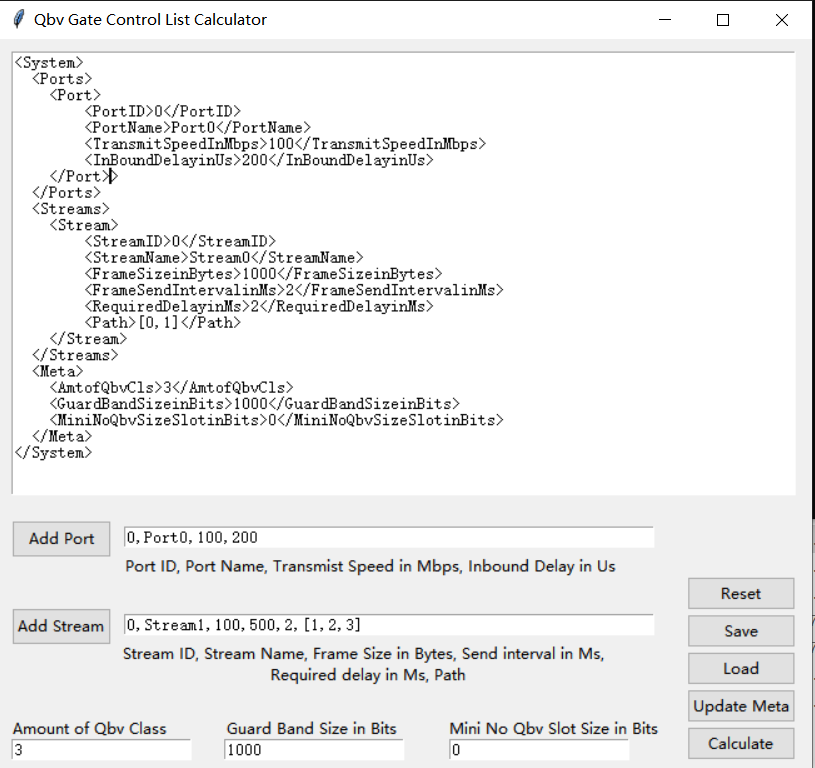


Figure 2

工具提供了一个UI用于描述TSN系统。

为了简化工具开发，工具里没有使用交换机、网线等具体概念，只使用了Port、Stream，Meta参数三类数据。

用户可以直接上上面的窗口中编辑代码，然后点击calculate按键开始计算。上面大窗口中的描述代码是xml规范的。

用户也可以使用Add Port 和Add Stream按钮半自动地添加代码。例如，用户可以在Add Port按钮右侧的输入框内输入端口的相关参数。输入完成后按AddPort按钮，就会自动地把一段描述这个新端口的代码加入进去。

端口描述的格式参考输入框下方的描述。

Add Stream同理

除了定义端口和Stream外，还需要输入一些零碎的Meta参数，在窗口的最下方。输入完后点击UpdateMeta按钮即可把相关的参数填入代码框。

输入完相关的端口、Stream和Meta信息后，即可点击Calculate键开始计算。计算完成后会自动调用excel进行图形化显示。

填入好数据后，可以点击Save把数据保存在工具文件目录下/Accessory/QbvSystemDescription.xml文件中。点击Load可以重新加载回来。没有提供额外的文件操作功能。也可以手动将写好的文件自行保存在其他文件中以备下一次使用。

点击Reset会导致所有数据被重置，得到一个只有框架的描述文件。

工具第一次打开后就有一个已经描述好的小系统，可以直接运行查看效果。

# 工具的使用

以Figure 3所示简单例子，这个小系统中包括了Node0, Node3两个收发节点、和P0,P1,P4,P3四个端口和Stream0这一条数据流。

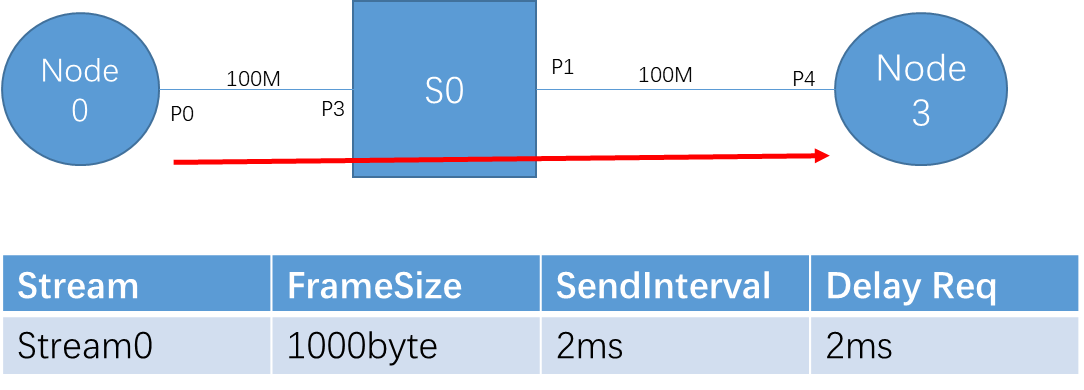


Figure 3

Qbv的计算过程只需要考虑发送端口，因此我们只需要输入P0，P1两个端口。如Figure 4中，其实最后只有Port0和Port1参与了计算。

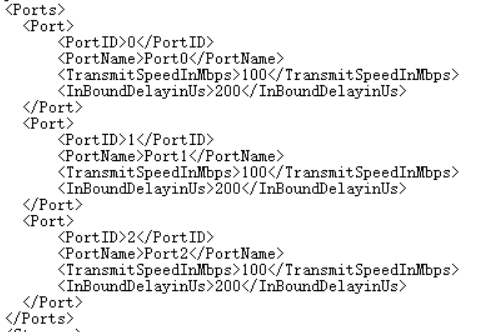


Figure 4

Port参数中可能需要额外提到的就是InBoundDelayinUs。这个Delay值包含了导线传输延迟，从Ingress口经过查表、校验、policy检查等动作后被放到交换机的Egress口对列之前的时间。通常这个值的范围在几十微秒到百微秒的范围，典型值可以查阅相关交换机的手册。

Stream的设置中需要额外说明的是Path部分。

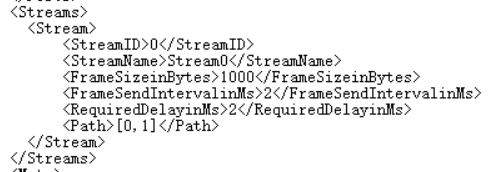


Figure 5

Path描述了这个数据流要流经的端口，如图Figure 5所示，这个Stream会流经Port0和Port1。其余参数应该都容易理解。