To 1-Step or Not to 1-Step

IEEE 802.1 Meeting

March 2015 – Berlin, Germany

Don Pannell

Marvell Semiconductor

dpannell@marvell.com

The Problem

- IEEE 1588 supports 1-Step PTP as well as 2-Step
  - 2-Step is “assumed” to be processed in software
  - 1-Step is hardware “assisted” (but there still is software)
    - As IEEE 802.1AS does not support transparent clocks
  - These two modes of operation (1-Step and 2-Step) do not interoperate with each other

- As port counts increase on bridges, the harder it is for them to keep up with all the PTP requirements
  - Hardware “acceleration” or “assistance” is desired if not needed

Some History

- Years ago, the AVB standards committee needed a low cost interoperating PTP solution
  - As IEEE 1588 was targeted at “engineered” networks it did not “plug-and-play” – a feature AVB needed
  - Thus IEEE 802.1AS was developed as an IEEE 1588 profile that does interoperate and is “plug-and-play”
  - 2-Step PTP was chosen for IEEE 802.1AS

- The TSN standards committee is now expanding AVB to support “engineered” networks
  - So should IEEE 802.1AS be expanded to add 1-Step?

Some More History

- 1-Step on receive was proposed to be added to IEEE 802.1AS (the ASbt project at the time)
  - It was found in Jan 2013 that it was not going to be compatible with any existing protocols – so yet another new protocol/frame formats would have to be developed – something few would likely use
- So in Jan 2014, the group dropped the 1-Step on receive goal as an user of TSN could use 1-Step PTP from IEEE 1588 – a standard that already existed – no need to invent a new format
  - All the TSN standards have been modified to support *any* PTP. They no longer require use of IEEE 802.1AS

Its Back….?

- In Jan 2015, support for 1-Step in IEEE 802.1AS was brought up again
- From that presentation:
  - There is a *perception* that two-step processing (sync/follow-up):
    - is less accurate than one-step
    - uses excessive processing in an intermediate system
    - reduces responsiveness because of longer residence times
- While these may be *perceptions*, these are not reasons to create yet another protocol variant!
  - We need to show that these issues are not real!

Accuracy

- As the Jan 2015 presentation stated, IEEE 802.1AS has been shown to be just as accurate in real world systems as IEEE 1588 1-Step systems
  - See: One-Step for 802.1AS-REV http://www.ieee802.org/1/files/public/docs2015/asbt-mjt-one-step-0115-v01.pdf
- So Accuracy is not a reason

Processing

- Processing was shown to be a potential problem in the Jan 2015 presentation, but it doesn't have to be
- Before the ASbt project started, a proposal for hardware accelerating 2-Step was discussed as a solution for the "software" 2-Step performance problem in large bridges
  - This would interoperate with "software only" IEEE 802.1AS implementations
  - It would interoperate so well in fact, that there was no changes needed to the standard at all to support these hardware accelerated or hardware "assisted" implementations

Responsiveness

- Responsiveness was shown in the Jan 2015 presentation to be caused by the longer residency times due to the delays in software getting the Follow up frames out of the bridge
- Hardware accelerating 2-Step would solve this problem/concern to in the same way 1-Step does
  - There is no difference here

Adding 1-Step Sync to AS-REV

- The changes to AS-REV to add 1-Step for Sync only are understood
- But these changes have not addressed the modifications required to be backwards compatible on a port-by-port basis to 802.1AS-2011 (i.e., 2-Step devices)
  - This includes new auto-negotiation protocols
  - As well as state machine changes
- Being backwards compatible to AVB Gen 1 is a clear requirement of any new TSN standards

Hardware 2-Step - Pros

- Same CPU overhead as 1-Step
  - CPU sends out a Sync frame and that’s it – its done
  - Follow up frames are automatically generated
- Interoperates with previous/standard 2-Step implementations
  - The receiving end cannot tell the difference!
  - Except that the Follow up frames show up quicker
- No changes to the standard are needed
  - There is no risk in breaking something that already works
- Lower port latencies can be supported

1-Step Port Latency Problem

- 1-Step was a great idea for 100 Mbit/sec links
- But as port data speeds increase it gets harder and harder to process/update a frame while it is Tx’ing
  - At 100 Mbit/sec it takes 5120 ns to Tx 64 bytes
  - At Gigabit it takes 512 ns to Tx 64 bytes
  - At 10 Gig it takes 51.2 ns to Tx 64 bytes
  - At 100 Gig it takes 5.12 ns to Tx 64 bytes
- There simply is not enough time at 10 Gig speeds and above to timestamp, compute, transform & update a 1-Step frame as it egresses a port
  - Not without adding significant pipeline/latency

1-Step Port Latencies Cont’d

- The added pipeline/latency is needed to “predict’ when the frame will egress & do the work and frame modifications ahead of time
- This added latency not only occurs on the time stamped PTP frames, but ALL frames egressing the port end up experiencing this added latency/delay
  - When there has been numerous presentations & standards being adopted in the TSN group on reducing latency, adding latency as a solution is not in the best interests of the group

Hardware 2-Step - Cons

- There is that *perception* problem
  - This is a Marketing/Education issue that we all can help with by educating our customers
  - Hardware acceleration is Hardware acceleration - reducing the CPU overhead equally
  - The end customer gets the performance they were looking for
- The bigger issue is getting Hardware 2-Step to work when PTP is placed into the PHYs
  - PHYs are like wires – they really want to see one byte in for every byte out they transmit down the wire
  - So PHYs creating bytes (Follow up frames) is a problem

Introducing 1.5-Step

- 1.5-Step:
  - Solves the *perception* problem
  - Solves the PHY issue
  - Results in the same CPU overhead as 1-Step
  - Is interoperable with IEEE 802.1AS 2-Step
  - Requires no modifications to the standard
  - And solves the 1-Step latency issue as port speeds increase

Solves the ***perception*** problem

- 1.5-Step solves the *perception* problem
  - Because the number “1” is in the name and everyone “knows” that “1” means hardware acceleration which reduces the PTP bridge software overhead

Solves the PHY problem

- 1.5-Step solves the PHY problem
  - This is what is unique about this solution
  - PHYs can still act as wires as they don’t “generate” or “create” Follow up frame bytes
  - The PHYs transmit one byte out on the wire for every byte they receive
- Instead the CPU builds two frames, the Sync frame and a Dummy Follow up frame and sends them both out the port one right after the other
  - The PHY timestamps the Sync frame and when it “sees” the Dummy Follow up frame it updates the few fields it needs to, converting the frame to a real Follow up

Same CPU Overhead as 1-Step

- 1.5-Step has the same CPU Overhead as 1-Step
  - The CPU builds two frames, the Sync frame and a Dummy Follow up frame and sends them both out at the same time - one right after the other
  - That's it, the CPU is done
  - No waiting for the Sync frame to egress, & then read its timestamp, & then build a Follow up frame & send it out
- OK, the CPU has to build two frames back to back so its overhead is a very small bit more than 1-Step
  - The major software overhead in software 2-Step is the waiting for the Sync to egress & processing its timestamp – neither of which is needed in 1.5-Step

Interoperable with 802.1AS-2011

- 1.5-Step is interoperable with 802.1AS 2-Step
  - For the very small cost of the CPU building and sending out two back-to-back frames in one operation (the Sync frame and a Dummy Follow up frame) the benefit is 100% interoperability with all past and future IEEE 802.1AS-2011 systems
  - This is true regardless if these 802.1AS systems support the hardware acceleration of 1.5 Step or not

Requires no changes to the standard

- 1.5-Step requires no changes to the standard
  - Its 100% interoperability with all past and future IEEE 802.1AS-2011 systems
  - No new frame formats are needed
  - No new negotiation or capabilities need to be transferred between link partners
  - No new state machines
  - No risk of breaking anything in the standard because there are no changes

Improves the 1-Step Latency Issue

- 1.5-Step improves the 1-Step latency issue
  - There is at least one minimum frame time reduction in egress pipeline latency with 1.5-Step
  - This is due to the CPU generated Dummy Follow up frame – the frame that gets modified – is at least 84 byte times behind the start of the Sync frame
    - 84 = 64 frame bytes + 12 IFG bytes + 8 Preamble bytes
  - The added latency does not go to zero, but it is less than the added latency required to support 1-Step

The Grand Unification Issue

- The Jan 2015 presentation also stated:
  - not providing one-step in 802.1AS is a potential obstacle in the “grand unification” of 1588 and 802.1AS
- Jan 2013 work showed that adding 1-Step to 802.1AS would create yet another variation of 1-Step
  - This does not help “grand unification” as this makes 802.1AS yet again different from 1588
- When “grand unification” work actually takes place (no sooner than the next versions of each standard) then 1-Step’s addition makes sense
  - For legacy reasons

Summary

- Adding 1-Step to IEEE 802.1AS-REV is yet another protocol/format – We don’t want that
- The group already decided this – in Jan 2014
- 2-Step can be hardware “accelerated” giving the same benefits/performance as 1-Step
- 1.5-Step may be a better name for this as it does differentiate this implementation
- 1.5-Step can work in PHYs
- 1.5-Step reduces the latency of the ports
- 1.5-Step interoperates with 2-step

Proposed Actions

- As agreed in Jan 2014, do not add 1-Step to IEEE 802.1AS-REV
- Continue supporting IEEE 1588 as a PTP solution for all TSN standards
  - If a customer wants 1-Step, use a 1588 solution
- Educate customers on the benefits of 1.5-Step
- When the “grand unification” work really starts, support 1-Step, but only at lower link speeds (like 100 Mbit/sec & Gigabit), for backwards compatibility reasons

Thanks

Questions?