Implementation and Evaluation of a PTP Transparent Clock Based on White Rabbit Technology

Cesar Prados* ‡
*GSI,Darmstadt Germany
‡Technical University Darmstadt, Darmstadt Germany

***Abstract*—White Rabbit is a new technology that allows to create an Ethernet timing network with low-latency, deterministic packet delivery, sub-nanosecond accuracy and picoseconds precision synchronization. WR is based on the standards Synchronous Ethernet, IEEE 1588-2008 and PTP. Currently the synchronization of the clocks in the network is achieved by using WR Switches, two-step Boundary Clocks (BC) using delay request-respond mechanism. In this paper, the author describes and evaluates the implementation of a WR Transparent Clock (TC) that uses the WR extension of PTP and is endowed with WR hardware.**

I. Introduction

Synchronization is one of the most important feature in control and timing systems. The performance depends on the synchronization accuracy between the networking elements throughout the network, specially in systems where time-critical events are accurately scheduled to be executed in distributed nodes. Such systems are quite common in facilities like particle accelerators like GSI and CERN. WR is the chosen technology for the timing system at GSI [1] and CERN [2]. Besides these two facilities, in the WR project [3], there are others instituted and experiments (LHAASO, KM3NeT etc...) that are taking an active part in development and adoption of the technology, as well as commercial companies (Seven Solutions, Integrasys, Elproma, Creotech, National Instruments etc..).

White Rabbit (WR) provides the technology and techniques to create a reliable and robust data and timing network with low-latency, deterministic packet delivery and synchronization. WR is based on the standards Synchronous Ethernet (SyncE) [4] and Ethernet (IEEE 802.3) [5]. Besides, it extends IEEE 1588 (PTP) [6] for achieving sub nanoseconds accuracy and picoseconds precision.

So far in WR, the synchronization is propagated in a hierarchical topology, from the master clock, WR Master (master), to the slave clocks, WR Slave (slave), using White Rabbit Switches (WRS). A WRS, in terms of PTP, is a two-step boundary clocks using the delay request-respond mechanism [6] for the synchronization. In this paper, the author proposes an implementation of a Transparent Clock (TC) using the existing WR extension to PTP, WRPTP [7] and WR hardware [8], [9], and evaluates the WR TC taking into account the principle features that characterizes the performance of a TC.

II. White Rabbit PTP

A. White Rabbit Synchronization

WR reaches subnanoseconds synchronization with picoseconds jitter, basically, characterizing the asymmetries of the link, and using a clock lookback technique for tracking the phase shift between the master reference clock signal and the looked back clock signal from the slave. In order to gather and realize such as measurements WR extends PTP, WRPTP [10]. The Fig. 1 shows the WRPTP flow of messages and the integration with standard PTP once is stablished.

Bellow, a resume of the steps, measurement and hardware support needed for the WR synchronization between two WR devices. A thoroughly description is presented in [11] and [7].

1) WR Discovery and Synthonization: A WR clock initiates WRPTP with an announcement message in order to discover others WR devices. If a WR device has been discover, the master will initiate the frequency lock procedure. WRPTP uses SyncE to distribute a common frequency throughout the network, the WR Slaves are frequency locked to the WR Master.

2) Asymmetry Calibration: Once the slave is locked to the master, the WR devices initiate to calibrates the asymmetries in the common optical link taking in consideration:

- Fixed delays due to transmission circuitry
- Asymmetry of the optical transceivers and PHYs
- Asymmetry of the propagation delay in the fiber caused by the chromatic dispersion

3) Coarse and Fine Delay Measurement: After the devices are calibrated, a first delay measurement, *Coarse Measurement*, is issued using the delay request-respond mechanism. The next step towards the synchronization requires the measurement of the phase shift between a reference (master) clock signal and the looped-back clock signal from the slave, the *round trip phase shift*, $phase_{mm}$. In figure 3, the clock signal is looped back and the phase shift is measured using a phase detector, DDMTD [12]. With the $phase_{mm}$, the delay round trip, $delay_{mm}$ is calculated. The $phase_s$ is the phase shift of the clock adjustment derived form the $offset_ms$. Using the $phase_{mm}$ and $phase_s$ the timestamps on ingress ports can be enhanced, t_{2p} and t_{4p}, using an decision algorithm described in [11]. Only timestamps on ingress ports need to be enhanced,

since they are generated asynchronously to the reference clock domain. The *Fine Delay Measurement*, round trip, is calculated as follows using now the enhanced timestamps :

$$delay_{mm} = (t_{4p} - t_1) - (t_3 - t_{2p}) \quad (1)$$

4) Synchronization: In order to finish the synchronization, the offset between both clocks must be calculated, $offset_{Mmm}$. The round trip can be expressed as function of the delay master to slave, σ_{ms} , slave to master σ_{sm} and the sum of the fixed delays, Δ , obtained during the calibration.

$$delay_{mm} = \Delta + \sigma_{ms} + \sigma_{sm} \quad (2)$$

The ratio between single delays is proportional to the asymmetry of the speed of the different wavelengths due to the chromatic dispersion in the fiber optic link:

$$(\alpha - 1) = \frac{\sigma_{ms}}{\sigma_{sm}} \quad (3)$$

Combining equations (1), (2) and (3), the delay master to slave and $offset_{MS}$:

$$delay_{ms} = \frac{1+\alpha}{2+\alpha}(delay_{mm} - \Delta) + \Delta_{txm} + \Delta_{rxs} \quad (4)$$

$$offset_{ms} = t_1 - t_{2p} - delay_{ms} \quad (5)$$

After the initial WRPTP synchronization, a DDR or Peer Delay mechanism produces the timestamps, t_1, t_{2p}, t_3 and t_{4p} (in case Peer Delay, also t_5 and t_{6p}) and the DDMTD tracks changes in the $phase_{mm}$ over time.

III. WR Transparent Clocks

PTP Transparent Clock (TC) modifies PTP messages as they pass through them adding the residence time to an accumulative Correction Field (CF) in the PTP messages. Thus, the delay introduced by the network is measured and can be subtracted in the slave clock, which improves distribution accuracy.

In II the WRPTP and WR synchronization steps has been described for master, slave and boundary clock. In this chapter the author presents how a standard TC becomes a WR TC.

A. End-to-End WR Transparent Clock

As the standard End-to-End (E2E) clocks, the WR E2E TC doesn't belong to the master-slave hierarchy and does not synchronized to the WR Master. Therefore, a WR E2E accomplishes only the synthonization and the calibration, the WR Mode is on, but there is no enhancement of the time stamps since the $delay_{ms}$ is calculated end to end, between master and slave, and there is not $phame_{mm}$ or $phase_s$ measurement.

Thanks to the synthonization done during the WRPTP there is not errors in the measurement of the residence time and in the CF.

The figure 2 shows the PTP exchange of messages from master to slave, going transparently and how the residence

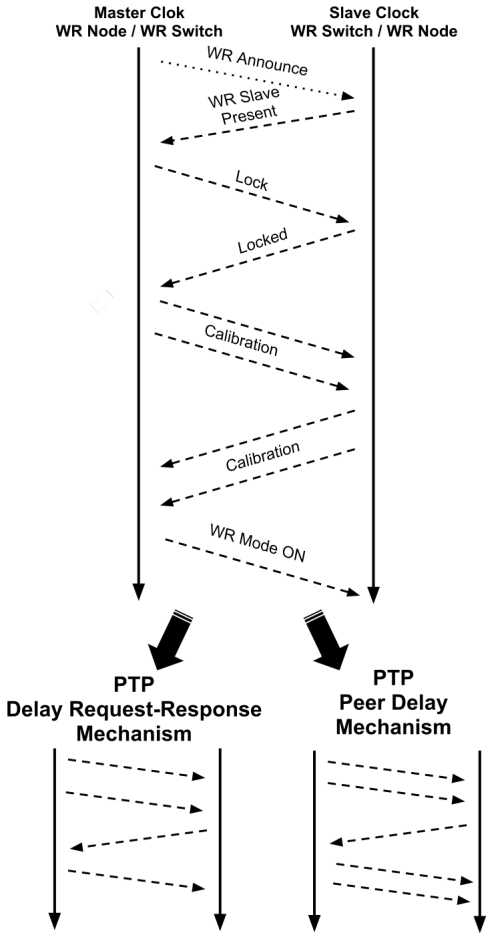

Fig. 1. WR PTP Message Flow and PTP

time is also calculated taking into account the fixed delays Δ. A two-step WR E2E TC calculates, using (4), the $offset_{ms}$:

$$CF+ = (TS_{ingess\_port} - \Delta_{rx}) - (TS_{egress\_port} + \Delta_{tx}) \quad (6)$$

$$offset_{ms} = t_1 - t_2 - delay_{ms} - CF \quad (7)$$

B. Peer-to-Peer WR Transparent Clock

As the standard Peer-to-Peer (P2P) clocks, the WR P2P TC measures residence time of Sync messages, and the link delay in both directions. Since the link delay is done between adjacent clocks using the peer delay mechanism, the fully WR synchronization process can be issued between the two ports. The figure 2 shows how the WR P2P initiates on after sides of the TC.

The measurement of the residence time, like in the E2E should suffer no error since the ports are syntonized to its master clock, but also the measurement of the link delays, are as precise as the WR project claims [10] for the Boundary Clocks. The link delay is calculate like in (4) and the fig 3 shows, but using t_3, t_{4p}, t_5 and t_{6p} instead. The $offset_{ms}$ is calculated :

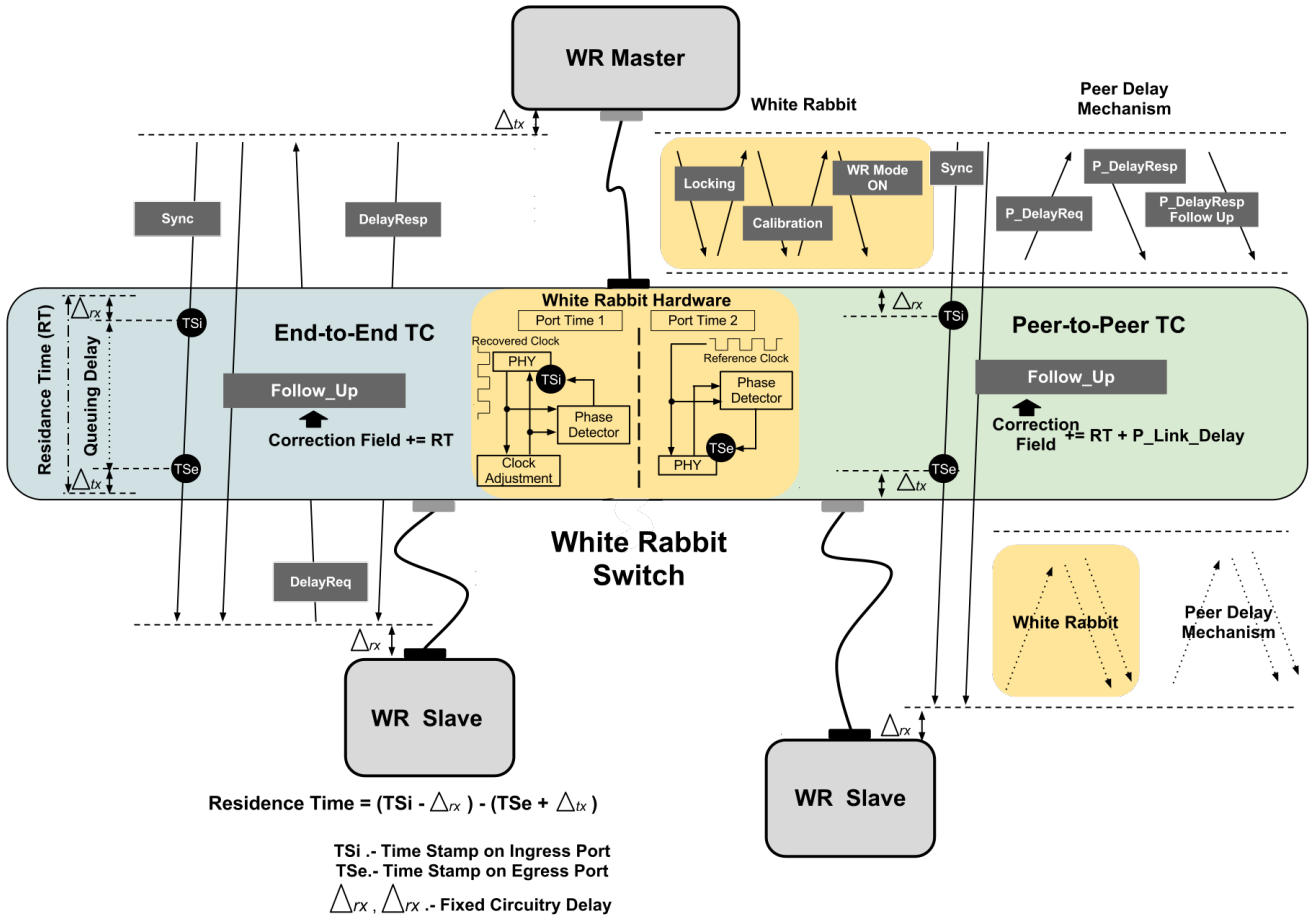

Fig. 2. WR E2E and P2P Transparent Clocks

$$CF+ = residance\_time + delay\_link \tag{8}$$

$$offset_{ms} = t_1 - t_2 - CF - delay\_link^{1} \tag{9}$$

IV. Implementation Issues and Performance Estimation

The performance of a TC is commonly related to the [13], [14] following features:

- Timestamps accuracy
- Correction factor stability
- Maximum update rate
- Rapid reconfiguration after topology changes

It applies to a WR TC as well.

WRS has been specially deceived to be compliant to most common Ethernet standards (e.g. VLAN tags and Quality of Service) and fulfil demanding requirements in terms of upper-bound delivery, latency and fault tolerance. This features can be exploited to achieve remarkable performance as a TC.

[1]This link delay correspond to the las link to the slave

A. Time Stamping Accuracy

In the chapter II, the author resumes how WR accomplish sub-nanoseconds synchronization and picoseconds jitter. It is based on the characterization the asymmetries of the link a priory, and a clock lookback technique. By doing this WR is able to enhance the timestamps on ingress ports.

The figure 3 shows that the Peer Delay timestamps between two adjacent nodes can be enhanced since the $pahse_{MM\_1}$ and $phase_{S\_1}$ to the reference master clock signal are known. But it is not the case for t_2, this timestamp can be only enhanced using $phase_{S\_2}$ since the reference clock signal used in this link, is a synchronized clock to reference master clock signal (the same applies for t_4, in a E2E TC). The error introduced if the t_2 is enhanced using $phase_{S\_2}$ and not $phase_{S\_1}$ is:

x

x

x

x Must be calculated!!! x

x

x

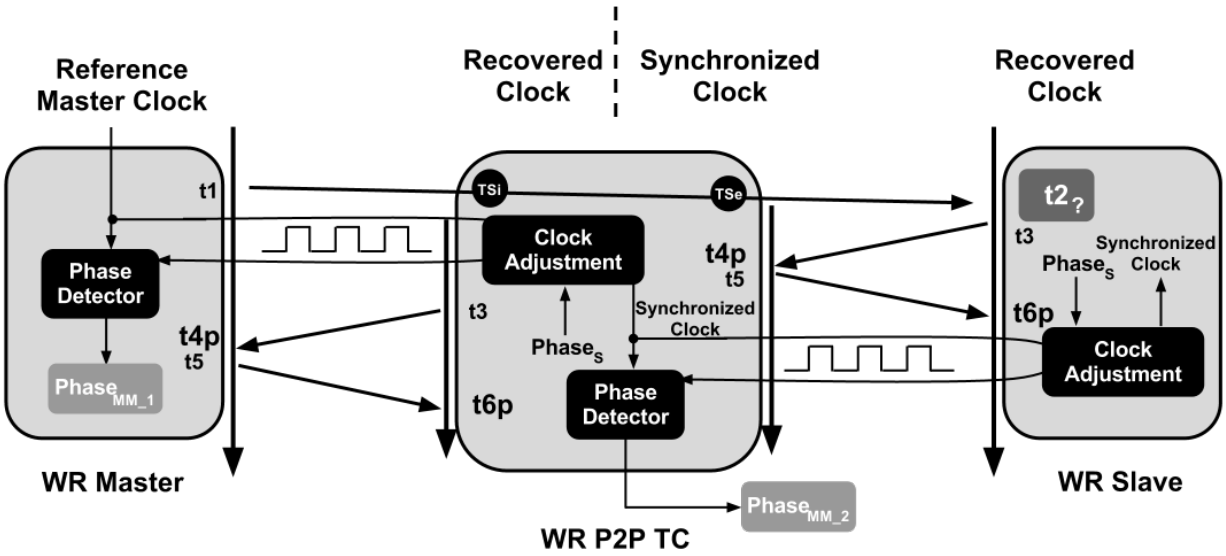

Fig. 3. WR synchronization and P2P TC

B. Correction Factor Stability and Maximum Update Rate

Both, correction factor and update rate, are features greatly influence by the latency in the TC. As a result synchronization could suffer a degradation of the performance. An increment of the residence time in a TC normally is associated to increment of the traffic load handled by this TC. The authors of [13] present the parameter *Correction Factor Error* (CFE). It expresses the difference of the latency measured by a test equipment and the updated CF. The authors test TCs from different vendors under high multicast load and compare the CFE result. The outcome of the test clarifies that switches with hardware treatment of the Sync messages maintain the CFE value low while the latency increases. Regarding the WR TC, this is option that the WRS enjoys already, hardware implementation of the timestamping unit, as well as hardware process forwarding of the Sync.

The *Update Rate* (UR) for a two-step clock, is defined as the time between the transmission of a Sync message from the master till the reception of the Follow Up message by the slave. High latencies, 10-30% [13] of the UR, can create instability in the slave. In addition, high latencies makes the TC no suitable for applications demanding high UR. WRS is designed to provide low latency for time critical information (e.g control accelerator information) using Cut-Through forwarding schema and QoS [**?**, vlan]n the output ports it is possible to queue the PTP traffic in the second highest priority. By doing this the UR will be still high.

C. Reconfiguration after Topologies Changes

The application of the timing system defines not only the requirements of accuracy, precision, but also the stability and continuation of the synchronization in the events of(?) failure of TCs. The author doesn't (past/present??) find references of this features for timing system based on PTP in the literature. Nevertheless, the new timing system for GSI and CERN have a demanding requirement regarding the stability of the timing.

TABLE

Both timing systems achieve resilience against network failures using redundant topologies. Lowe layer protocols resolves the topology to a loop-free topologies. As a result, the protocol sets port connected to cyclic paths to block/passive state. In this case only the P2P TC still issues the synchronization between the ports, and in both directions. As a result both clocks have the link delay information. This offers the possibility of having immediately available the link after change in the topology. It means that the reconfiguration time doesn't depend on TC anymore but in the lower layer protocol. WRS has already a proposals for implementing a transparent mechanism [9] for recovering from single points of failure in a redundant network.

V. Conclusions and Future Work

Conclusion–¿¿¿

Currently the WR Switch V3 [9] behaves as Boundary clock. After an exhaustive examination of the WR project, the following features should be added to the existing gateware and software (no modifications in the hardware) for a proof of concept WR TC implementation:

- Peer Delay Mechanism
- WR TC clock behaviour
- Hardware process of Sync and Follow Up messages
- Enhanced Timestamping across TC using WR

The openness of the project [2], plus the amount of work already done, makes the implementation achievable with a reasonable effort.

[2]The WRS is licensed under *CERN Open Hardware Licence* [15] and is publicly available in the Open Hardware Repository.

VI. Acknowledgment

The author would like to recognize the vast work done by the WR Community and specially to, T. Włostowski and M. Lipiski, for their support, design and code of the White Rabbit Switch, on which a I have based this paper.

References

[1] T. Fleck, C. Prados, S. Rauch, and M. Kreider, "FAIR timing system," GSI, Darmstadt, Germany, Tech. Rep., 2009, v1.2.

[2] J. Bau and M. Lipiński, "Discussion On A White Rabbit based CERN Control and Timing Network," http://www.ohwr.org/documents/85, October 2011, v1.1.

[3] "White Rabbit Project. http://www.ohwr.org/projects/white-rabbit.

[4] *Timing characteristics of a synchronous Ethernet equipment slave clock (EEC)*, ITU-T Std. G.8262, 2007.

[5] *IEEE Standard for Information Technology–Telecommunications and Information Exchange Between Systems–Local and Metropolitan Area Networks–Specific Requirements Part 3: Carrier Sense Multiple Access With Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications - Section Three*, IEEE Std. 802.3-2008, 2008.

[6] *IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems*, IEEE Std. 1588-2008, 2008.

[7] E. Cota, M. Lipiński, T. Włostowski, E. Bij, and J. Serrano, "White Rabbit Specification: Draft for Comments," http://www.ohwr.org/documents/21, july 2011, v2.0.

[8] Simple PCIe FMC carrier (SPEC). http://www.ohwr.org/projects/spec.

[9] White Rabbit Switch v3. http://www.sevensols.com/whiterabbitsolution/.

[10] M. Lipiński, T. Wostowski, J. Serrano, and P. Alvarez, "White Rabbit: a PTP application for robust sub-nanosecond synchronization," *Proceedings of ISPCS*, 2011.

[11] T. Włostowski, "Precise time and frequency transfer in a White Rabbit network," Master's thesis, Warsaw University of Technology, may 2011.

[12] P. Moreira, P. Alvarez, J. Serrano, I. Darwezeh, and T. Wlostowski, "Digital Dual Mixer Time Difference for Sub-Nanosecond Time Synchronization in Ethernet," *Frequency Control Symposium (FCS), 2010 IEEE International*, 2010.

[13] J. Burch, K. Green, J. Nakulski, and D. Vook, "Verifying the performance of transparent clocks in ptp systems," *Proceedings of ISPCS*, 2009.

[14] J. Han and D.-K. Jeong, "A Practical Implementation of IEEE 1588-2008 Transparent Clock for Distributed Measurement and Control Systems," *IEEE Transactions on Intromentation and Measurement*, vol. 59, no. 2, 2010.

[15] CERN Open Hardware Licence.