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 supports an Ethernet timing networks with low-latency, deterministic packet delivery, sub-nanosecond accuracy and picosecond precise synchronization. WR is based on the standards Synchronous Ethernet, IEEE 1588-2008 and PTP. Currently the synchronization of 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 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, especially in systems where time-critical events are accurately scheduled to be executed in distributed nodes. Such systems are quite common in particle accelerators facilities, like the new timing system at GSI [1] and CERN [2]. The chosen technology in both facilities is based on the White Rabbit project. 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) [3] and Ethernet (IEEE 802.3) [4]. Besides, it extends IEEE 1588 (PTP) [5] for achieving sub-nanoseconds accuracy and picoseconds precision. Besides GSI and CERN, in the WR project [6] there are other institutes 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..).

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 BC using the delay request-respond mechanism (DDR) [5] 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].

Among the advantages of the Transparent Clocks (TC) for large timing networks (e.g more than 2000 slaves in GSI), it is especially interesting for the WR project the Peer-to-Peer TCs. They are suitable for timing systems that require high resilience in the event of changes in the topology (e.g. switch failure), since the delay measurement is already available for the synchronization in the new link, in contrast to Boundary Clocks (BC) that have to calculate the delay once the new link is active. Another attractive feature of TCs for a WR timing system is the measurement of the residence time in the network devices which eliminates the error that results from queuing delays.

In this paper, the author describes the WRPTP and WR synchronization (Section II), and how the extension can be adopted by E2E and P2P TC (Section III). Then, the author presents issues of implementing a WR TC and an estimation of the performance.

II. White Rabbit PTP

A. White Rabbit Synchronization

WR reaches sub-nanosecond synchronization with picoseconds jitter by 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 looped back clock signal from the slave. In order to gather and realize such measurements, WR extends PTP, WRPTP [10]. Figure 1 shows the WRPTP flow of messages and the integration with standard PTP once it is established .

Below, 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 Syntonization: A WR clock initiates WRPTP with an announcement message in order to discover other 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 decodes the clock signal from the data stream sent from the WR Master, and locks to it.

2) Asymmetry Calibration: Once the slave is locked to the master, the WR devices initiate to calibrate 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, Digital Dual Mixer Time Difference (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}[1] and t_{4p}, using a 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 :

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

4) Synchronization: In order to finish the synchronization, the offset between both clocks must be calculated, $offset_{mm}$. 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.

$$delay_{mm} = \Delta + \sigma_{ms} + \sigma_{sm} \tag{2}$$

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:

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

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

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

$$offset_{ms} = t_1 - t_{2p} - delay_{ms} \tag{5}$$

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 it adding the residence time to the 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.

[1]The enhanced timestamps are distinguished from the non-enhanced with a p

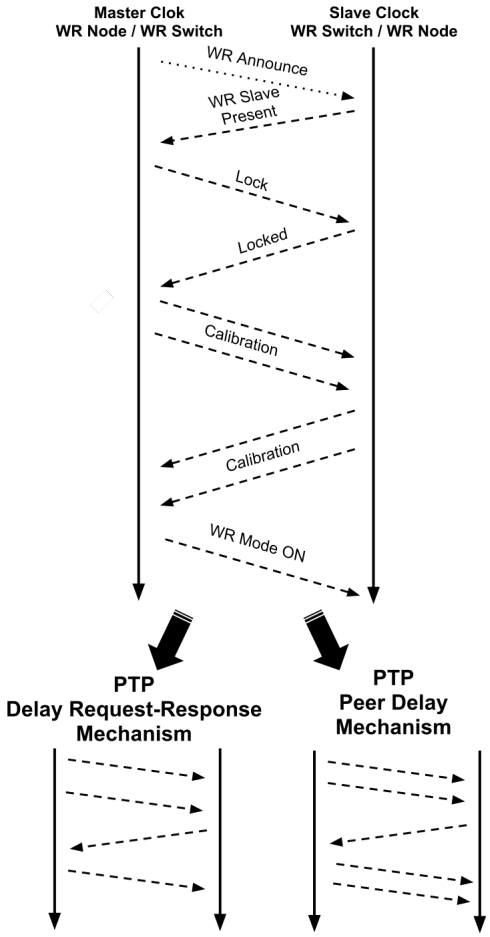

Fig. 1. WR PTP Message Flow and PTP

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

A. End-to-End WR Transparent Clock

As in the standard, End-to-End (E2E) clocks, the WR E2E TC doesn't belong to the master-slave hierarchy and doesn't synchronize to the WR Master. Therefore, a WR E2E accomplishes only the syntonization 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 $phase_{mm}$ or $phase_s$ measurement. Thanks to the syntonization done during the WRPTP there is not errors in the measurement of the residence time.

Figure 2 shows the PTP exchange of messages from master to slave, going through the WRS transparently, and how the residence time is also calculated taking into account the fixed delays Δ. A two-step WR E2E TC calculates, using (4), the $offset_{ms}$:

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

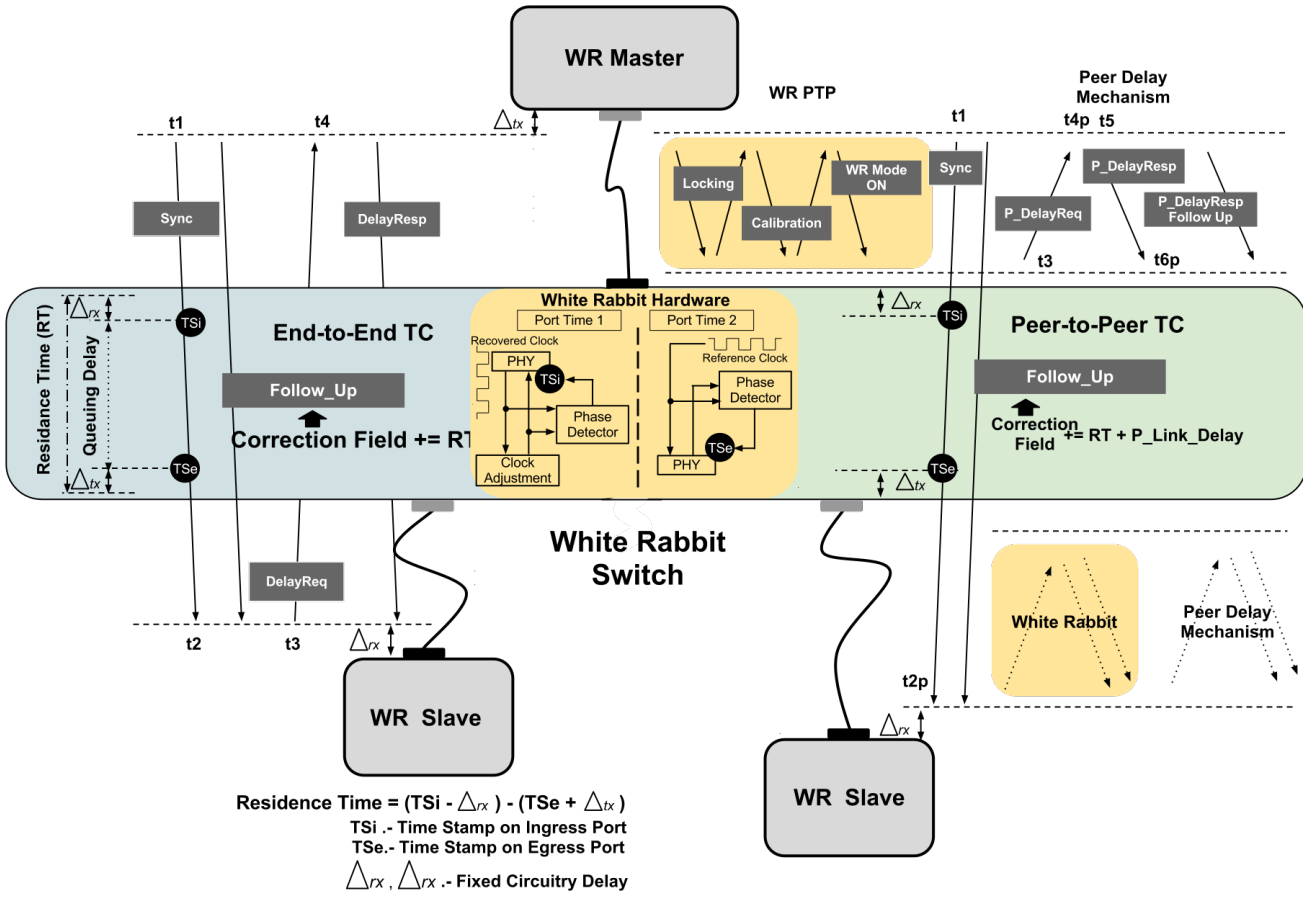

Fig. 2. WR E2E and P2P Transparent Clocks

$$offset_{ms} = t_1 - t_2 - delay_{ms} - CF \tag{7}$$

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 full 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, are done without errors 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 Figure 3 shows, but using t_3, t_{4p}, t_5 and t_{6p} instead. The $offset_{ms}$ is calculated :

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

$$offset_{ms} = t_1 - t_2 - CF - delay\_link^2 \tag{9}$$

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

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. Apart from being a BC, WRS has been specially designed to fulfil demanding requirements in terms of upper-bound delivery, latency and fault tolerance. Besides, WRS supports standards (e.g. VLAN tags and Quality of Service) that can be used by the WR TC.

A. Time Stamping Accuracy

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

Figure 3 shows that the Peer Delay timestamps between two adjacent nodes can be enhanced since the $phase_{MM\_1}$ and $phase_{S\_1}$ to the reference master clock signal are known.

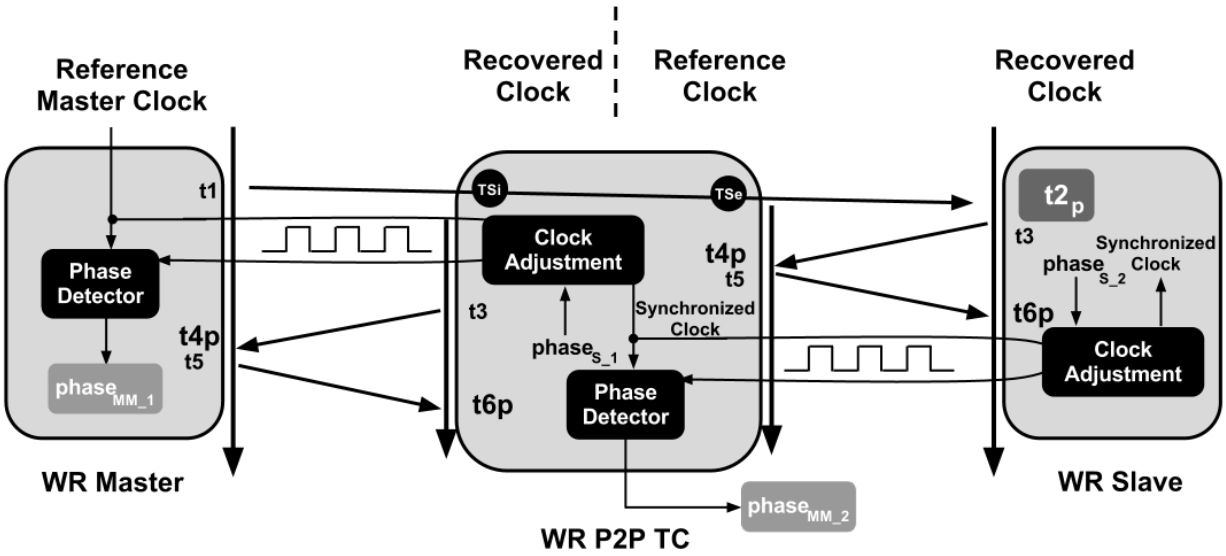

Fig. 3. WR synchronization and P2P TC

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 the reference master clock signal. The calculation of $offset_{ms}$ will be affected by the relative enhancement. In the case of E2E TC all the timestamps on ingress port in the slave, are not enhanced. Further understanding and investigation of enhancement of timestamps across TCs is needed.

B. Correction Factor Stability and Maximum Update Rate

Both, *correction factor* (CF) and *update rate* (UR), are features greatly influenced by the latency in the TC. As a result, synchronization could suffer a degradation of performance. Long residence time in a TC is associated with an increase of 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 still maintain low the CFE value when the latency increases due to an increment of traffic. WRS enjoys already hardware implementation of the timestamping unit, as well as hardware process forwarding of the messages.

The 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 unsuitable 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 [15] in the output ports.

Thus, PTP traffic can be queued in the second highest priority like Figure 4 shows. The queue scheduling algorithm

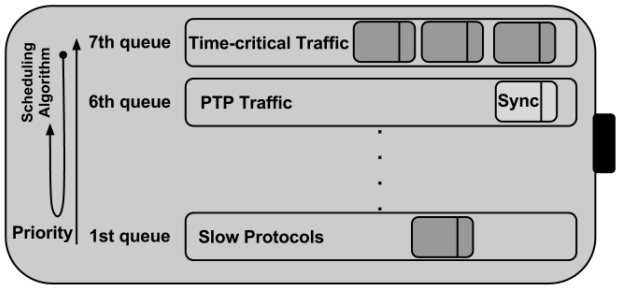

Fig. 4. Queues in Output Port

TABLE I
SYNC TRANSMISSION - FOLLOW UP RECEPTION DELAY ESTIMATION

Number WRS	5
Time-critical Traffic burst period	500 μs
Delay	**Max Value**
Syn to Follow Up Tx	2 ms
White Rabbit Switch	10 ms
Total	52 ms

is in charge to decide from which queue packets are sent. The highest priority, 7th queue, is always emptied before the others, and right after, the algorithm orders to check the PTP queue. If there is a PTP message, it is sent immediately. In this case, the delay in the switch would be affected only by time-critical traffic and not by the rest of the traffic. Table I presents an estimation of the delay between the Sync transmission and the Follow Up message reception in a WR network made of five layer of switches. In this estimation, time-critical traffic is issued in burst of packets every 500 μs. The queue scheduling algorithm grantees that the maximum latency in the WRS for PTP traffic is 10 ms. The UR in this scenario could be around 20 Hz.

TABLE II
GSI AND CERN TIMING SYSTEM REQUIREMENTS

	GSI	CERN
Accuracy	8 ns	1 μs
d_{max} WR Master	2 km	10 km

C. Reconfiguration after Topologies Changes

The use case of the timing system defines not only the requirements of accuracy and precision, but also the stability and resilience of the synchronization in the event of failures of TCs. The author doesn't find references of this features for timing system based on PTP in the literature. Nevertheless, as the Table II shows the new timing system for GSI and CERN have a demanding requirement regarding the synchronization accuracy.

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

V. CONCLUSION AND FUTURE WORK

The implementation and evaluation presented in this paper, indicates that WR TCs can provide a high performance in terms of CFE, UR and reliability.The calculation of the residency time and delay between clocks in a P2P should be as accurate as in WR BC, although the end-to-end synchronization accuracy of the WR P2P and E2E TC needs a model for an enhancement of timestamping across TC in order to achieve the accuracy and precision that WR offers.

Currently the WR Switch V3 [9] behaves as BC. After an exhaustive examination of the WR project, the following features will 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
- Enhanced timestamping across TCs using WR

The openness of the project[3], plus the amount of work already done, makes the implementation achievable with a reasonable effort.

VI. ACKNOWLEDGMENT

The author would like to recognize the vast work done by the WR Community and thank 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. I want also to thank to the timing team at GSI: D.Beck, M.Kreider, S.Rauch and W.Terpstra for the review and feedbacks of this paper.

[3]The WRS is licensed under *CERN Open Hardware Licence* [16] and is publicly available on the Open Hardware Repository.

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] *Timing characteristics of a synchronous Ethernet equipment slave clock (EEC)*, ITU-T Std. G.8262, 2007.
[4] *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.
[5] *IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems*, IEEE Std. 1588-2008, 2008.
[6] "White Rabbit Project. http://www.ohwr.org/projects/white-rabbit.
[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] *802.1Q-2011 - IEEE Standard for Local and metropolitan area networks–Media Access Control (MAC) Bridges and Virtual Bridged Local Area Networks*, Std.
[16] CERN Open Hardware Licence.