White Rabbit: a PTP Application for Robust Sub-nanosecond Synchronization

Maciej Lipiński, Tomasz Włostowski, Javier Serrano, Pablo Alvarez
CERN, Geneva
Email: {maciej.lipinski, tomasz.wlostowski, javier.serrano, pablo.alvarez.sanchez}@cern.ch

***Abstract*—This article describes time distribution in a White Rabbit Network. We start by presenting a short overview of the White Rabbit project explaining its requirements to highlight the importance of the timing aspects of the system. We then introduce the technologies used to achieve high clock accuracy, stability and resilience in all the components of the network. In particular, the choice of the IEEE 1588-2008 (PTP) and Synchronous Ethernet standards are explained. In order to accommodate hardware-supported mechanisms to increase PTP synchronization accuracy, we introduce the White Rabbit extension to PTP (WRPTP). The hardware used to support WRPTP is presented. Measured results of WRPTP performance demonstrate sub-nanosecond accuracy over a 5km fiber optic link with a precision below 10ps and a reduced PTP-message exchange rate. Tests of the implementation show full compatibility with existing PTP gear.**

I. Introduction

White Rabbit (WR, [1]) is a project which aims at creating an Ethernet-based network with low-latency, deterministic packet delivery and network-wide, transparent, high-accuracy timing distribution. The White Rabbit Network (WRN) is based on existing standards, namely Ethernet (IEEE 802.3 [2]), Synchronous Ethernet (SyncE [3]) and PTP [4]. It is fully compatible with these standards.

A WRN consists of White Rabbit Nodes (nodes) and White Rabbit Switches (switches) interconnected by fiber or copper links. The focus of this article is on 1000Base-LX [4] single-mode fiber connections only. WR supports integration of nodes and/or switches that are not White Rabbit. A simple WRN is presented in Fig. 1.

A node is considered the source and destination of information sent over the WRN. The information distributed over a WRN includes:

- Timing - frequency and International Atomic Time (TAI).
- Data - Ethernet traffic between nodes.

In order to understand the goals of the WR project – namely determinism, high reliability and accurate synchronization – these terms are explained below.

Determinism is guaranteed by having a worst-case upper bound in frame delivery latency.

Accuracy is a measure of the deviation between the clock of the *grandmaster* node/switch of a WRN and that of any other node. Assuming only fiber interconnections, a WRN is meant to achieve sub-nanosecond accuracy.

Reliability of a WRN refers to robust delivery of data and timing to all the nodes. The timing must allow all the nodes to be synchronized with the required accuracy and the data must always be delivered on time.

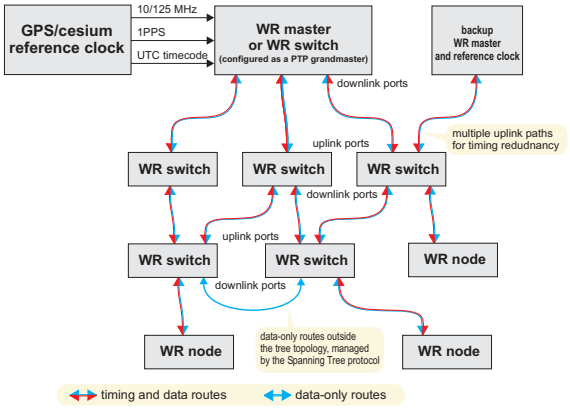

Fig. 1. A White Rabbit Network [5].

All these features are required to create a timing and control system which may replace the General Machine Timing (GMT) [6] at CERN and fulfill a similar role at the Facility for Antiproton and Ion Research (FAIR) in GSI [7]. Such system requires synchronization of up to 2000 nodes with sub-nanosecond accuracy, an upper bound in frame delivery and a very low data loss rate. However, many other applications of White Rabbit are possible. This includes industry, telecommunications and other large distributed systems (e.g. distributed oscilloscopes [8]).

This article focuses solely on the PTP-based timing distribution in a WRN. PTP is a packet-based protocol designed to synchronize devices in distributed systems. The accuracy of the PTP synchronization is implementation-dependent. The standard is foreseen for sub-nanosecond accuracies. However, such performance is not achieved in typical PTP implementations for two reasons:

- Limited precision and resolution of PTP timestamps.
- Unknown physical link asymmetry.

Additionally, the quality of PTP-syntonization depends on the exchange rate of PTP messages. The higher the quality of the clock we want to recover, the higher the bandwidth needed for PTP-related traffic.

The precision of timestamps is greatly increased in the existing PTP implementations by hardware-based timestamping. However, such implementations are limited by the resolution

which is specified by the hardware-driving frequency (e.g. 8ns for 125MHz).

The asymmetry is not detectable by PTP; if known, PTP corrects for it to increase synchronization accuracy. Some of the sources of the physical layer asymmetry can be eliminated by proper network configuration, e.g. excluding routers or non-PTP bridges from the network. Others, in particular the inaccuracy caused by physical medium asymmetry (i.e. difference in propagation velocity in two-way communication over fiber) or PCB layout (i.e. connection length between PHY and timestamping hardware) need to be obtained through proper a priori-measurement.

White Rabbit addresses these limitations to achieve sub-nanosecond accuracy of synchronization. It uses SyncE to distribute the common notion of frequency in the entire network over the physical medium. It casts the problem of timestamping into a phase detection measurement. The results of these precise measurements are used both during normal PTP operation and for quantifying physical medium asymmetry during the calibration phase. The improved performance of the synchronization is accomplished without increasing PTP-related traffic (it can be actually decreased) since PTP is only governing the synchronization, while the syntonization is done by SyncE.

A great effort has been made to align WR-specific solutions with the PTP standard and stay fully compatible with PTP gear. Consequently, WR can be seen as an extension to PTP. This extension, called WRPTP, defines its own PTP profile and describes all the WR-specific mechanisms which need to be implemented by a node/switch to enable sub-nanosecond synchronization with another nodes/switches. The compatibility with PTP makes WR more likely to be used in existing PTP-based systems by gradual and/or partial upgrade to WRPTP. It also enables the creation of hybrid, thus cost-effective, systems. WRPTP is presented in section II *White Rabbit PTP*. Section III presents an example implementation of hardware which supports WRPTP. Finally, the measurements of WRPTP performance and conclusions are presented in the last two sections of this article.

In PTP nomenclature, the main component of a WRN, the switch, is a boundary clock (BC) – a clock that has multiple PTP ports. Similarly, the node is an ordinary clock (OC) – a clock that has a single PTP port. Consequently, the WRN can be seen as a set of independently synchronized Master-to-Slave (M-to-S) links with ports of OC or BC on both sides (Fig. 1, red-blue arrows). The ability to synchronize a single link scales into the ability to synchronize the entire network. Therefore, in this article, only a single M-to-S link is considered, where appropriate.

II. White Rabbit extension to PTP (WRPTP)

White Rabbit extends the PTP standard benefiting from PTP's mechanisms, taking advantage of its customization facilities (i.e. PTP profiles and Type-Length-Value, TLV) but also defining implementation-specific functionalities and staying PTP-compatible.

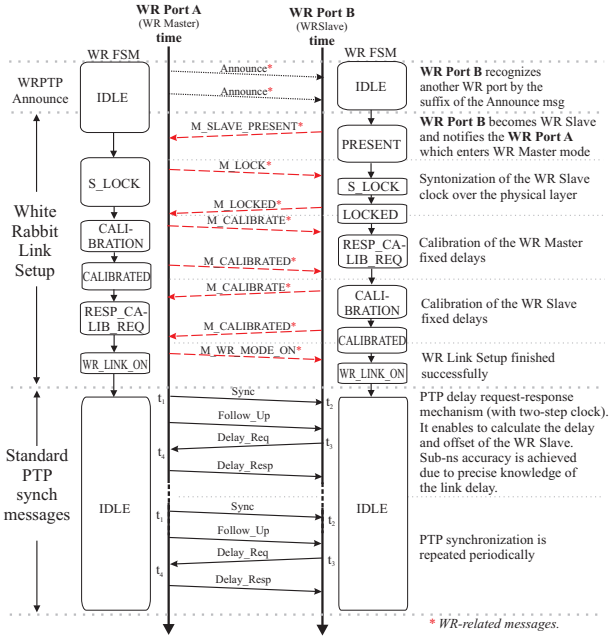

Fig. 2. Simplified overview of the message flow in WRPTP.

WRPTP introduces the *WR Link Setup* (section II-E) process which provides inputs to the *WR Link Delay Model* (section II-A) to obtain accurate delay and offset calculations. The WR Link Setup is a process for establishing the WR link. It includes WR node identification, syntonization, measurement of WR-specific parameters and their distribution over the link. The additional communication during the WR Link Setup is done through extended PTP messaging facilities (section II-C). The WR-specific parameters, which are exchanged and set during this process, are stored in WR-specific data set fields (section II-B).

The flow of events for the standard PTP is extended as depicted in Fig. 2 and described below:

1) **WR Port A** which is in PTP_MASTER state periodically sends WR Announce messages (section II-C).
2) **WR Port B** receives Announce message(s), recognizes the WR Announce message and uses the modified Best Master Clock (mBMC) algorithm (section II-D) to establish its place in the WR network hierarchy.
3) **WR Port B** enters WR Slave mode (based on the conditions in II-E) and starts the WR Link Setup by sending the M_SLAVE_PRESENT message.
4) **WR Port A** enters WR Master mode (based on the conditions in II-E) and sends the M_LOCK message to request the WR Slave to start syntonization.
5) The WR Slave sends the M_LOCKED message as soon as the syntonization process is finished.
6) The WR Master sends the M_CALIBRATE message to request a calibration pattern in order to measure its reception fixed delay.
7) The WR Master sends the M_CALIBRATED message as soon as the calibration is finished.

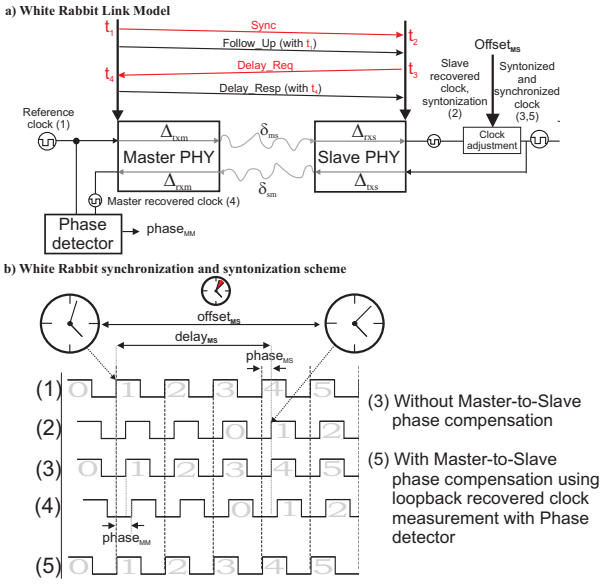

Fig. 3. WR Link Delay Model, synchronization and syntonization scheme.

8) The WR Slave sends the M_CALIBRATE message to request a calibration pattern in order to measure its reception fixed delay.
9) The WR Slave sends the M_CALIBRATED message as soon as the calibration is finished.
10) The WR Master sends the M_WR_MODE_ON message to indicate completion of the WR Link Setup process.
11) PTP two-step clock request-response delay measurement is performed repeatedly (t_1, t_2, t_3, t_4). WR Slave calculates M-to-S delay and clock offset and adjusts its time counters.

A. WR Link Delay Model

Sub-nanosecond synchronization requires a precise knowledge of one-way M-to-S delay ($delay_{ms}$). PTP measures the round-trip delay ($delay_{MM}$). Usually, delay symmetry ($delay_{ms} = delay_{sm}$) is assumed to derive the one-way delay. In White Rabbit, the *WR Link Delay Model* (Fig. 3, (a)) is used to calculate the precise M-to-S delay by including link asymmetry. The delay can be expressed as the sum:

$$delay_{ms} = \Delta_{tx_m} + \delta_{ms} + \Delta_{rx_s} \tag{1}$$

where Δ_{tx_m} is the fixed delay due to the master's transmission circuitry, δ_{ms} is the variable delay of the transmission medium and Δ_{rx_s} is the fixed circuitry reception delay in the slave. Similarly, the S-to-M delay ($delay_{sm}$) can be described using Δ_{tx_s}, δ_{sm} and Δ_{rx_m} respectively. The fixed delays ($\Delta_{rx_s,tx_s,rx_m,tx_m}$) are considered constant for a given connection. They can be measured by nodes/switches at the beginning of the connection, as described in section III-C. For highest precision, WR uses a single fiber for two-way communication – the asymmetry is directly and solely caused by the difference in propagation velocity in both directions. Consequently, the difference between δ_{ms} and δ_{sm} is connected to the wavelength difference for transmitting and receiving the data (e.g. 1550 and 1310 nm). The equation describing the relation between δ_{ms} and δ_{sm} is defined in [9]. For bi-directional fiber, it is governed by the *relative delay coefficient* (α):

$$\alpha = \frac{\delta_{ms}}{\delta_{sm}} - 1 = \frac{n_{1550}}{n_{1310}} - 1 \tag{2}$$

The α coefficient can be calculated using refractive indexes (n_{1550}, n_{1310}) or, preferably, measured for a given fiber type.

Knowing the coefficient, the round-trip delay ($delay_{MM}$) and the fixed delays ($\Delta_{rx_s,tx_s,rx_m,tx_m}$), the precise M-to-S delay (5) can be calculated using (3) and (4). See [9] and [5] for a detailed derivation.

$$\Delta = \Delta_{tx_m} + \Delta_{rx_s} + \Delta_{tx_s} + \Delta_{rx_m} \tag{3}$$

$$delay_{MM} = \Delta + \delta_{ms} + \delta_{sm} \tag{4}$$

$$delay_{ms} = \frac{1+\alpha}{2+\alpha}(delay_{MM} - \Delta) \tag{5}$$

Finally, the value of the offset ($offset_{ms}$) between the master's clock and that of the slave is obtained (6). It is fed into the adjustment algorithm of the slave's clock servo (Fig. 3).

$$offset_{ms} = t_1 - t_{2p} - delay_{ms} \tag{6}$$

The t_{2p} in (6) is the enhanced-precision value of PTP t_2 which is obtained during the *Fine Delay Measurement* described in section III-A. The same process is used to obtain a very precise value of t_4.

B. WR Data Sets

WRPTP adds fields to the data sets defined in the PTP standard and defines a new data set (backupParentDS) to store WR-specific parameters. All new fields, except *primarySlavePortNumber*, are added to the *portDS* data set. The *primarySlavePortNumber* field is added to the *currentDS* data set.

C. WR Messages

WRPTP defines a WR Type-Length-Value (WR TLV) extension to exchange the WR-specific data. In particular, it is used to suffix Announce and create Signaling Messages. WR TLVs are recognized (see TABLE 35 of PTP) by the TLV type (*tlvType=ORGANIZATION_EXTENSION*), CERN's Organizationally Unique Identifier (*OrganizationId* = 0x080030), magic and version numbers. The different types of WR TLVs are distinguished by a WR Message Identifier (wrMessageID), as defined in TABLE I.

TABLE I
WHITE RABBIT MESSAGE ID VALUES

Message name	wrMessageId	Sent in message type
M_SLAVE_PRESENT	0x1000	Signaling
M_LOCK	0x1001	Signaling
M_LOCKED	0x1002	Signaling
M_CALIBRATE	0x1003	Signaling
M_CALIBRATED	0x1004	Signaling
M_WR_MODE_ON	0x1005	Signaling
M_ANN_SUFIX	0x2000	Announce

D. Modified BMC

The standard method of handling topology and grandmaster redundancy in a BC-based PTP network is not sufficient for the WRN. The BMC allows only a single port (slave) of a BC to be synchronized to a single grandmaster. A time source failure requires re-synchronization and might introduce fluctuations in the notion of time.

The modified BMC (mBMC) allows for more than one *best* clock in a single domain, enabling the creation of a logic topology with multiple roots. A BC running the mBMC can have more than one port in the PTP_SLAVE state (slave ports). This means that timing information is exchanged between a BC and more than one source of time (i.e. OC or BC). At any time any of these sources can be used to perform synchronization, including a weighted average from all slave ports as mentioned in [10].

The modification applies to the State Decision Algorithm (SDA): the BMC_SLAVE recommended state is enforced instead of the BMC_PASSIVE state for the clocks with clockClass greater than 127 [4]. A port which becomes a slave as a result of the modification, is considered a Secondary Slave. A slave port resulting from the unchanged part of the SDA, is considered a Primary Slave and its number is stored in the *primarySlavePortNumber*.

The best qualified Announce messages (E_{rbest}) from all Secondary Slave ports are compared using the Data Comparison Algorithm (DCA) to determine the "second best master" and the lower order masters. The results are stored in the *backupParentDS* (section II-B).

The provided information about Primary and Secondary Slaves is used by the WR hardware to support a seamless switch-over in case of failure of the best master (section III-B). Such a solution enables robust synchronization with no deterioration of its quality while switching between the redundant components.

E. WR Link Setup

The process of establishing a White Rabbit link between two WR ports is called *WR Link Setup*. It involves the recognition of two compatible WR ports, their WR modes assignment (WR Master or WR Slave), syntonization over the physical layer, measurement of the fixed delays ($\Delta_{rx_s,tx_s,rx_m,tx_m}$) and exchange of their values across the link. The WR Link Setup is controlled by the White Rabbit state machine (WR FSM) which is executed in the PTP_UNCALIBRATED state of the PTP FSM on the WR Slave, and in the PTP_MASTER state on the WR Master. The WR FSM is depicted in Fig. 4 and its usual execution on the WR Master and the WR Slave during the WR Link Setup is presented in Fig. 2.

A WR port can become a WR Master, Slave or enter non-WR mode depending on its place in the network hierarchy (mBMC's outcome) and WR-specific parameters. The following conditions need to be fulfilled by a WR port to enter the WR Slave mode and start the WR Link Setup by entering the PRESENT state of the WR FSM (Fig. 2 & Fig. 4: *D_WR_SETUP_REQ*):

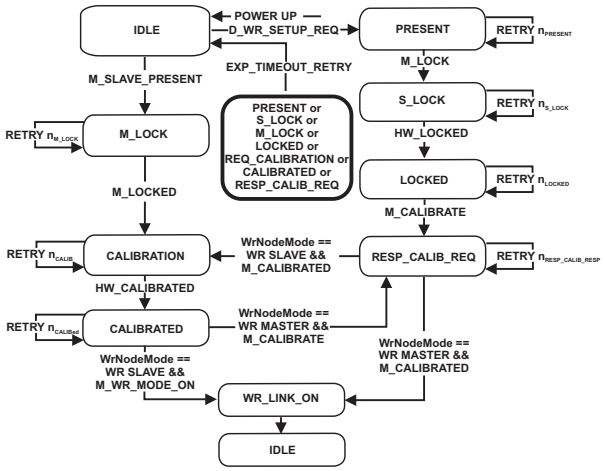

Fig. 4. WR state machine [9].

TABLE II
WRPTP PROFILE.

profleName	White Rabbit
profileVersion	1.0
profileIdentifier	08-00-03-00-01-00
organizationName	European Organization for Nuclear Research (CERN)
sourceIdentification	http://www.ohwr.org/projects/white-rabbit

- the port is not in the PTP_SLAVE state **AND**
- the port is WR Slave-enabled **AND**
- the mBMC's Recommended State is BMC_SLAVE **AND**
- the parent port is WR Master-enabled **AND**
- at least one of the ports on the link is not in active WR mode.

Similarly, the following conditions need to be fulfilled by a WR port to become the WR Master and start the WR Link Setup by entering the S_LOCK state of the WR FSM (Fig. 2):

- the port is in the PTP_MASTER state **AND**
- the port is WR Master-enabled **AND**
- the M_SLAVE_PRESENT message has been received.

On successful completion of the *WR Link Setup* the WR mode (wrMode) of a port is validated by setting wrModeOn = TRUE.

F. WRPTP Profile

The White Rabbit PTP profile defines the profile's identification (TABLE II). It specifies the values of some of the parameters (e.g. priority1, logSyncInterval) and the options to be used. It indicates the delay request-response mechanism as the only one used by the WRPTP. It also specifies the mBMC to be used and the WR TLV to be supported.

III. WR Hardware Support

SyncE is responsible for clock syntonization in the WRN. In the SyncE scheme, the reference clock (125MHz) is used to encode the outgoing data stream. The same clock is retrieved on the other side of the physical link using the *Clock Recovery System* (CRS, section III-B). Having the same frequency

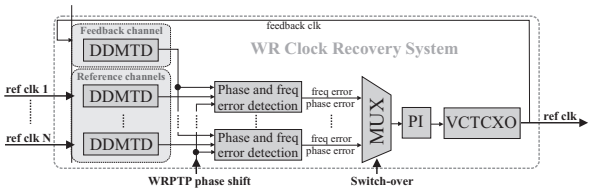

Fig. 5. White Rabbit Clock Recovery System.

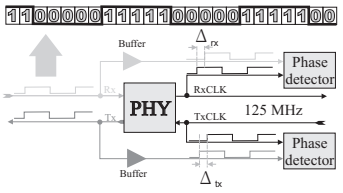

Fig. 6. Rx/Tx latency measurement [9].

allows us to use phase detector technologies as a means of evaluating delays. WR implements Digital Dual Mixer Time Difference (DDMTD) [11] phase detection. The measurement of phase is used for increasing the precision of timestamps beyond the resolution allowed by the 125MHz clock. This process is described in the next section. The DDMTD phase detection is also used to obtain *the transmit/receive (Tx/Rx) latencies* (section III-C) and compare frequencies in the CRS.

A. Fine Delay Measurement

We call *Fine Delay Measurement* the process during which the DDMTD-detected round-trip phase shift ($phase_{MM}$ in Fig. 3) is used to enhance timestamp precision and to calculate the precise round-trip delay ($delay_{MM}$). During the fine delay process only the reception timestamp (t_2, t_4) measurements need to be improved as they are transmitted and timestamped in different clock domains.

A basic timestamp (to be enhanced) is obtained by the detection of the Start-of-Frame Delimiter (SFD) in the Physical Coding Sublayer (PCS). In order to acquire a precision-improved timestamp, we need to eliminate the possible $\pm$ 1 LSB error (8ns) due to jitter of clock signals and clock-domain crossing. Therefore, the Time-Stamping Unit (TSU) produces timestamps on both the rising and falling edges of the clock, and later in the process one of them is chosen. The process involves three steps (see [5] for details):

1) rising/falling edge timestamp choice (t_f or t_r),
2) calculating the picosecond part, checking its sign and adding a clock period if necessary,
3) extending timestamps with the picosecond part (t_{2p}, t_{4p}).

The modified timestamps are used to calculate the precise round-trip delay (Fig. 3):

$$delay_{MM} = (t_{4p} - t_1) - (t_3 - t_{2p}) \tag{7}$$

B. Robust Clock Recovery System

The White Rabbit implementation of the CRS (WR CRS) is explained in detail in [5], Fig. 5 presents a very simplified overview of the WR CRS. It is designed to accommodate multiple sources of frequency (physical links) with a single source being used (active) at a given time. A seamless switching of the active source is one of the goals of the design to enable network topology redundancy and, as a consequence, offer robust and stable synchronization (see section II-D).

All input reference clocks (*ref clk*) and a feedback clock (*feedback clk*) are fed into DDMTD and degliching units (*DDMTD*). The input clocks are mixed by the DDMTDs with the offset frequency (8) which is derived from the active ref clock. The frequency-mixing results in a low frequency clock signal which maintains the original phase shift.

$$f_{offset}[ns] = 125[MHz] * \frac{2^N}{2^N + \Delta} \tag{8}$$

A prototype implementation using $N = 14$ and $\Delta = 1$ has demonstrated satisfactory performance [5]. Each output of the DDMTD reference channel is compared with the output of the feedback channel in the *phase and frequency error detection* units. The additional input to the units is the *WRPTP phase shift* obtained in the *Fine Delay Measurement*. The phase and frequency error (*phase and freq error*) between the feedback clock and each reference clock, corrected by WRPTP phase shift, is calculated and fed into the multiplexer (*MUX*). During normal operation, the active reference clock (from the Primary Slave port, section II-D) is used to feed the PI controller and produce the output reference signal. However, if the malfunction of the active ref clock is detected, the switch-over process takes place: the MUX is switched to feed the PI with the error data from another reference channel (from the Secondary Slave port). An estimate of the average phase and frequency errors can be provided to the PI controller to improve hold-over performance if all the reference channels fail.

The active clock malfunction takes 3 consecutive invalid symbols ($\sim 24ns$) to detect while the phase and frequency error detection is performed at a low frequency (kHz). Therefore, the continuity of the synchronization is guaranteed during the switch-over.

C. The transmit/receive (Tx/Rx) latencies

The transmit/receive (Tx/Rx) latencies in most PHYs vary for each Phase-Locked Loop/Clock Data Recovery (PLL/CDR) lock cycle, but stay constant once the PHY is locked. This is the case of the PHY used in the WR Switch prototype (TCK1221), therefore an Tx/Rx latency measurement is required. Tx latency is measured by feeding the transmit path with a sequence of RD+ K28.7 code-group (Appendix 36A.2 of [2]). Such signal creates a 125MHz clock on the SerDes I/O. Since the Tx clock frequency is also 125MHz, the DDMTD can be used to measure the phase shift between the SerDes I/O and the Tx clock, effectively measuring Tx latency. By receiving the K28.7 signal from the link partner, the Rx latency can be measured using the same method. This process, depicted in Fig. 6, is performed during the WR Link Setup (section II-E).

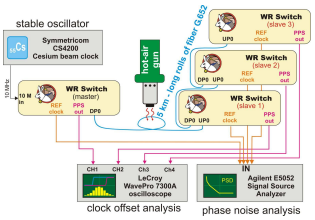

Fig. 7. Test setup.

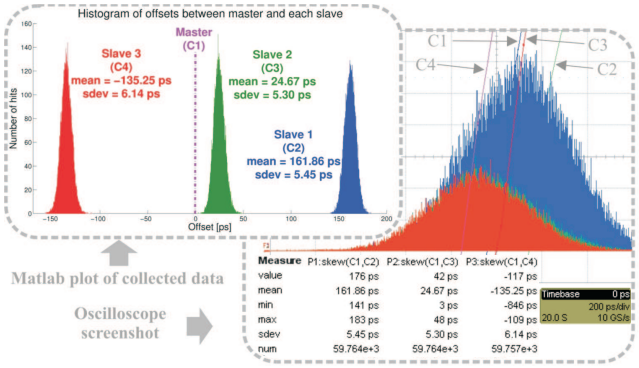

Fig. 8. Synchronization performance.

IV. Test Results

In order to test the performance of time and frequency transfer, the test setup depicted in Fig. 7 was assembled. The system consists of 4 switches connected with 5km bare fiber rolls in a daisy chain (15 km total). Varying operating conditions were simulated by heating the fiber with a hot air gun.

The frequency transfer performance was evaluated using an Agilent E5052B signal source analyzer. A cesium clock served as a source of 10 MHz reference frequency for the master. The Power Spectral Density (PSD) of the phase noise of the master and each slave REF clocks was determined. The integrated jitter from 10Hz to 40 MHz measured on each slave is presented in Table III. The jitter on a single link is below 2 ps.

The accuracy and precision of synchronization were characterized by measuring the skew between the master and each slave clock signal over a period of 1 hour with a LeCroy oscilloscope (Table III). A histogram of master-slave offsets constructed from the obtained samples is depicted in Fig. 8. The single link skew is well below 1 ns. In this particular case, the skews of the first and second link cancel, therefore the skew between the master and the last slave is below 200 ps. However, if the cancellation did not take place, the accumulated skew would be still below 0.5 ns. The varying conditions introduce only a transient increase of the skew's standard deviation (sdev) of $\sim$ 5 ps. This is a consequence of the purposely low exchange-rate of PTP messages (1 s), compared to the fast heating provided by the hot air gun.

TABLE III
Synchronization and syntonization performance.

Switch	Integrated jitter [ps]	Offset [ps]	
	10Hz-40MHz	mean	sdev
1	1.6637	161.86	5.45
2	2.4887	24.67	5.30
3	2.3025	-135.25	6.14

V. Conclusions

White Rabbit is an open hardware [12] and open software project pushing the frontiers of technology but also introducing new trends in cooperation between public institutions and companies.

The WRN is purposely based on well-established standards to ensure its long lifetime, wide support and commercial feasibility. By blending existing technologies, hardware-supporting and extending them, still staying compatible, exceptional results are achieved. This includes sub-ns accuracy of high precision and robust synchronization, which proved to be the most accurate known PTP implementation (ISPCS, September 2010, USA) and frequency transfer performance of below 2ps integrated jitter using a FPGA-based PLL design.

The WRPTP extension allows not only for increased accuracy, but also offers much lower PTP-message traffic and higher reliability by supporting network redundancy without any loss of performance, thus offering robust synchronization.

Compatibility with existing standards enables hybrid networks, where time- and data-critical end nodes are connected directly to a White Rabbit Network, while less critical devices use standard switches to connect to it.

References

[1] White Rabbit. http://www.ohwr.org/projects/white-rabbit.

[2] *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.

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

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

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

[6] J.Serrano, P.Alvarez, and J. D.Dominguez, "Nanosecond level UTC timng generation and stamping in CERN's LHC," in *Proceedings of ICALEPSC2003*, Gyeongju, Korea, 2003.

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

[8] S. Deghaye, D. Jacquet, I. Kozsar, and J. Serrano, "OASIS: A new system to acquire and display the analog signals for LHC," in *Proceedings of ICALEPCS2003*, Gyeongju, Korea, 2003.

[9] 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.

[10] T. Murakami and Y. Horiuchi, "A Master Redundancy Technique in IEEE 1588 Synchronization with a Link Congestion Estimation," in *Proceedings of ISPCS*, 2010.

[11] J. Serrano, P. Alvarez, M. Cattin, E. G. Cota, P. M. J. H. Lewis, T. Włostowski *et al.*, "The White Rabbit Project," in *Proceedings of ICALEPCS TUC004*, Kobe, Japan, 2009.

[12] M. Giampietro, "Hardware joins the open movement," *CERN Courier*, 2011.