

AN-1838 IEEE 1588 Boundary Clock and Transparent Clock Implementation Using the DP83640

ABSTRACT

TIs DP83640 precision PHYTER™ implements time-critical portions of the IEEE 1588 Precision Time Protocol (PTP), allowing high precision IEEE 1588 node implementations. These same features can be used to implement multi-port boundary clock (BC) and transparent clock (TC) devices as well.

Contents

1	Introduction	2
2	Basic Requirements	2
	2.1 PHY Addressing	3
	2.2 Synchronous Ethernet	3
3	IEEE 1588 Clock Control	3
	3.1 Verification of Time Setting	3
4	IEEE 1588 Boundary Clock	4
	4.1 Recommended Timestamp Delivery	5
5	End-to-End Transparent Clock	5
	5.1 Sync Message Handling	5
	5.2 DELAY_REQ Message Handling	6
	5.3 Clock Synchronization and Syntonization	6
6	Peer-to-Peer Transparent Clocks	6
	6.1 Sync Message Handling	6
	6.2 Peer Delay Mechanism	6
	6.3 Peer Delay Response	7
	6.4 Clock Synchronization and Syntonization	7
7	Performance Results	7
8	Conclusions	7
9	References	7

List of Figures

1	Reference Clock and Management Interface Connections	2
2	GPIO Connection for Time Verification	4
3	Synchronization Versus Number of Boundary or Transparent Clocks.....	7

PHYTER is a trademark of Texas Instruments.
 All other trademarks are the property of their respective owners.

1 Introduction

Boundary clocks and transparent clocks are both mechanisms to provide accurate distribution of the PTP protocol across multi-port network components such as bridges, routers, and repeaters. Applications range from large multi-port Ethernet switches to two or three port daisy chain devices. A boundary clock may be slaved to a master on one port and act as master on all other ports. A transparent clock does not act as a master or slave, but instead forwards PTP event messages and provides corrections for the residence time across the bridge. In all cases, each transparent or boundary clock includes a single PTP clock to accurately synchronize devices across the network.

The DP83640 yields a highly accurate solution while reducing the amount of additional circuitry required to implement the clock and timestamp features of an IEEE 1588 capable boundary clock or transparent clock device. Since timestamping is done internal to the physical layer device (PHY), the timestamps are equally deterministic in all modes of operation, providing for the highest accuracy solution.

2 Basic Requirements

In a boundary clock or transparent clock, there should only be one PTP clock that is shared by all ports. Since each DP83640 precision PHYTER includes an independent internal PTP clock, two basic requirements must be met. The first is that all of the PHY devices must have a shared reference clock source. This is easily met by having a single reference clock source which provides each PHY with the same reference clock, usually through some form of clock distribution.

The second requirement is that each of the PTP clocks must be able to be controlled simultaneously. This is necessary to ensure that all PTP clocks may be set to the same time and set to run at the same adjusted rate. This may be accomplished using a broadcast write operation supported by the serial management interface. For a description of the broadcast write operation, see the DP83640 Datasheet.

The serial management interface supports operation up to 25 MHz on MDC. The reference clock connected to X1 on each PHY should be 25 MHz for MII operation or 50 MHz for RMII operation. It is also possible to use a 25 MHz reference for RMII by using RMII master mode (see [AN-1794 Using RMII Master Mode \(SNLA101\)](#)).

In addition to the basic hardware connections for the device, the BC or TC must incorporate a controller to handle operation of the protocol as well as generation, termination, and modification of PTP packets. The nature of the packet handling is dependent on the type of device being implemented. These are discussed in separate sections.

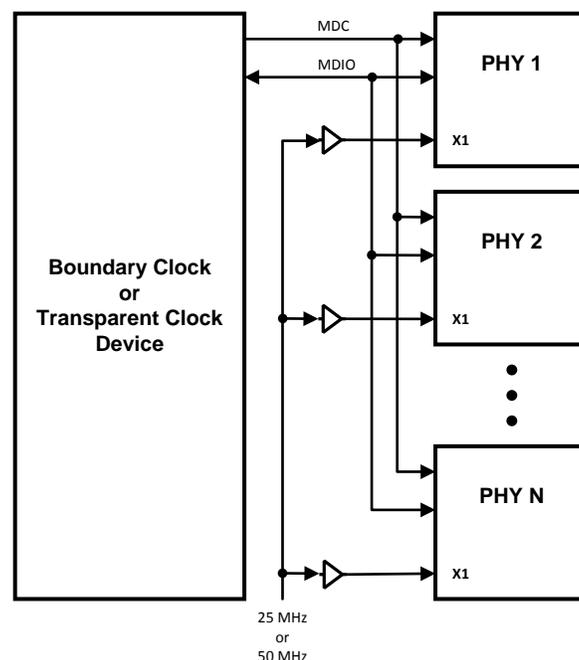


Figure 1. Reference Clock and Management Interface Connections

Since the MDC signal is also used as a clock, a clock distribution network may also be required on MDC to ensure a clean clock signal at each PHY.

[Figure 1](#) shows an example of a possible connection for the reference clock and management interface signals.

2.1 PHY Addressing

Since the broadcast write function uses PHY address 0x1F, the boundary or transparent clock device should not have any PHY strapped to address 0x1F.

In addition, PHY address 0 forces a PHY to power-up in an isolate state. If a device is strapped to phy address 0, management accesses are required to exit the isolate state.

All other PHY addresses are available for use without restrictions.

Since only 31 PHY devices may reside on a serial management interface (one address is reserved for broadcast), if more devices are required, additional serial management interfaces must be provided.

2.2 Synchronous Ethernet

In addition to normal Ethernet operation, the DP83640 may be used in a 100 Mb synchronous Ethernet mode. In this mode of operation, all ports may use the recovered clock on one port as the reference clock for all other ports. In a PTP system, this allows the frequency of the grandmaster clock to propagate through the entire network. For more details of the synchronous Ethernet operation, see [AN-1730 DP83640 Synchronous Ethernet Mode: Achieving Sub-Nanosecond Accuracy in PTP Applications \(SNLA100\)](#).

3 IEEE 1588 Clock Control

The basic PTP clock controls are independent of the mode of operation, that is, boundary clock versus transparent clock. In a node device, a slave will update the PTP clock based on the information determined from the PTP message timestamps. This amounts to setting the rate based on a measured rate difference relative to the master. It also includes setting the time and correcting for time offset from the master as measured by the protocol. In a boundary clock or transparent clock, the PTP clock control will occur in a similar manner, with one significant difference. Clock control changes are made to **all** PTP clocks rather than just a single clock. As mentioned previously, this can be done using the broadcast mechanism supported over the serial management interface. Therefore, all clock changes will be made to each PHY PTP clock simultaneously.

3.1 Verification of Time Setting

To ensure the time corrections are made accurately, it may be desirable to verify the PTP clock time values using the GPIO capabilities in the DP83640. This can be done by connecting a signal between a GPIO pin on each PHY. For example, a connection can be made between GPIO9 on each PHY as shown in [Figure 2](#).

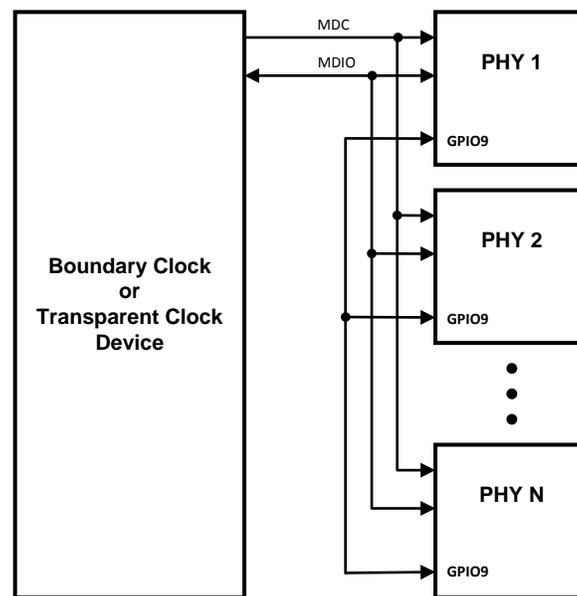


Figure 2. GPIO Connection for Time Verification

The clock times can be verified and corrected using the following algorithm:

1. Enable event timestamp capture on GPIO9 for each PHY.
2. Enable trigger output on a GPIO9 for PHY1.
3. Generate a single trigger pulse, which should be captured on each PHY including PHY1.
4. Read event timestamps from each PHY.
5. Make corrections to all PHYs based on difference with captured timestamp from PHY1.

This procedure should only be required at the initial time setting. All subsequent settings should use a step adjustment or temporary rate adjustment, which should occur at each PHY without introducing any error.

The main reason to verify the initial time setting is due to the synchronization of signals into the PHY. Since the MDC clock may be asynchronous to the reference clock used in the PHY, there may be a small error in the initial PTP clock times due to sampling of the MDC signal by the PHY.

4 IEEE 1588 Boundary Clock

A boundary clock can be used to support both versions 1 and 2 of the IEEE 1588 specification. The boundary clock implements a local PTP clock which can be synchronized to a master on one port and act as a master on other ports. Since a boundary clock is a full PTP clock implementation, both the time and frequency must be simultaneously updated to the local PTP clocks on each PHY. The processing of PTP messages and determination of time and rate changes must be done on a local processor. The basic processing operations required of the boundary clock are:

- Identify and route all inbound PTP messages to the local processor. Terminate all PTP traffic.
- Implementation of the best Master clock algorithm or other mechanism for determining master and slave port assignments.
- Synchronize as a slave on the port connected to the best master.
- Master operation for all other ports.
- Simultaneous control all PHY PTP clocks, including the clock in the Slave port.

4.1 Recommended Timestamp Delivery

Timestamps can be delivered to the controller using several methods. The basic method is to read timestamps using the serial management interface, utilizing the MDC and MDIO pair of signals. However, to minimize information transfer over the serial management interface, it is recommended that the boundary clock use the inband mechanisms of timestamp transfer when possible.

Enable timestamp insertion for receive PTP event messages. This eliminates the need to read the receive timestamp over MDIO. It also eliminates issues with matching timestamps to packets since the timestamp arrives embedded in the packet.

Use one-step operation for sending Sync messages. This eliminates the need to read the transmit timestamp over MDIO and also eliminates the need to send a Follow_Up message.

Use Delay_Req timestamp insertion for sending Delay_Req messages. This eliminates the need to read the transmit timestamp over MDIO as the timestamp will arrive embedded in the Delay_Resp message.

If a port on the boundary clock supports the peer delay mechanism, the transmit timestamps for PDelay_Req and PDelay_Resp messages will probably need to be read using MDIO.

Timestamps could also be delivered using PHY Status Frames if the amount of receive data traffic is not expected to be too significant. For more information on PHY Status Frames, see the *DP83640 Software Development Guide*.

5 End-to-End Transparent Clock

Version 2 of the IEEE 1588 specification introduces transparent clocks as an alternative to implementing boundary clocks for multiport devices such as bridges, switches or routers. The first type, the end-to-end transparent clock, forwards PTP event messages, but modifies the messages for the residence time for the message to propagate from an ingress port to an egress port. Corrections must be made for the propagation of both sync and Delay_Req messages.

5.1 Sync Message Handling

There are two options for handling sync messages: two-step or one-step operation.

5.1.1 Sync Two-Step Operation

For two-step operation, both the ingress and egress timestamps need to be recorded and saved to calculate the residence time. The sync message should be forwarded without modification. If timestamp insertion is used to deliver the received sync timestamp, the reserved timestamp insertion fields should be cleared before forwarding the packet. Upon receipt of an associated Follow_Up message, the residence time should be added to the correctionField of the Follow_Up message. Note that if a PTP message arrives without the two-step flag set, the transparent clock needs to set the two-step flag in the sync message, and generate a Follow_Up message to deliver the residence time in the correctionField.

5.1.2 Sync One-Step Operation

The one-step operation of the DP83640 may be utilized to eliminate the timestamp transfers across the management interface. While the specification calls for adding the residence time to the correctionField, the same results may be obtained by adding the incoming originTimestamp to the correctionField and subtracting the ingress timestamp. Upon transmission, the egress timestamp for the sync message will automatically be inserted into the originTimestamp field. The processor should set:

```
correctionField = correctionField + originTimestamp - sync_ingress_timestamp
originTimestamp = 0
```

Using one-step operation, the PHY will automatically set:

```
originTimestamp = sync_egress_timestamp
```

Since the correctionField only supports a 16-bit seconds field, this only works if the local PTP clock for the TC is synchronized within 215 seconds of the master clock. Before forwarding the first sync message, the transparent clock should first set the PTP clock time and then modify the ingress time-stamp based on the adjusted time.

5.2 DELAY_REQ Message Handling

For Delay_Req messages, the DP83640 does not support a mechanism to add the residence time to the correctionField of the Delay_Req message. Instead, both the ingress and egress timestamps must be recorded for the Delay_Req message. These timestamps are used to determine the residence time that will be added to the correctionField of the associated Delay_Resp message returned from the master. As there may be outstanding Delay_Req messages from multiple downstream slaves, the processor may need to store multiple residence time values. Note that the Delay_Req timestamp insertion is not applicable for transparent clock operation as the feature is only for use when there is a single, local source for sending Delay_Req messages.

5.3 Clock Synchronization and Syntonization

For most accurate residence time measurements, the PTP clocks in each PHY should be syntonized with the grand master. Syntonization only requires correction to the frequency and is, therefore, simpler than full synchronization. The processor can use the ingress timestamps from sync messages to determine a rate correction required for the PTP clock. All PHY clocks should be modified as described previously using the broadcast write over the serial management interface. While the local clocks are not required to be set to the same time as the grand master, they should be initially set to the same time as each other.

Alternatively, syntonization may be handled on the processor without adjusting the rate of the PHY clocks. The rate correction may be used to modify the computed residence times inserted into Follow_Up and Delay_Resp messages. This method may not be used with one-step operation. If syntonization is required to multiple PTP domains, then this must be handled on the processor.

If one-step operation is used for sync messages, then the local PTP clocks must also be roughly synchronized to the grand master. The more accurate the synchronization, the closer the originTimestamp will be to the original origin-Timestamp. For this level of synchronization, corrections to the local PTP clock only need to be made periodically and only if the PTP clock time varies outside an acceptable range. For time corrections, a simplified offset from the master may be computed directly from the originTimestamp, correctionField, and sync_ingress_timestamp.

6 Peer-to-Peer Transparent Clocks

Version 2 of the IEEE 1588 specification also defines peer-to-peer transparent clocks, which measures the local link delays using the peer delay mechanism, rather than using the delay request mechanism to measure full path delay. Peer-to-peer transparent clocks must determine the residence time and make corrections to sync messages. In addition, corrections must be included for inbound path delays as measured using the peer delay mechanism. Since the Delay_Req mechanism is not supported on peer-to-peer TCs, no special processing is required for Delay_Req messages.

6.1 Sync Message Handling

Processing of sync messages is essentially the same as in end-to-end TCs. The major difference is that inbound path delay must be added to the correctionField of the sync or associated Follow_Up message. Either of the two-step or one-step operations may be used as described for the end-to-end TC.

6.2 Peer Delay Mechanism

As required by the specification, a peer-to-peer TC should periodically send out PDelay_Req messages on each port to measure the path delay to the peer attached to that port. Outbound PDelay_Req timestamps should be recorded by the PHY and returned through MDIO reads. Inbound PDelay_Resp timestamps may be delivered via timestamp insertion (recommended) or via MDIO reads. Based on the information in the PDelay_Resp, and possibly PDelay_Resp_Follow_Up, the processor should compute the inbound path delay on the port. This value should be added to the correctionField for sync messages arriving on the port.

6.3 Peer Delay Response

In addition to initiating the peer delay mechanism, the peer-to-peer TC must be able to respond to peer delay requests on each port. To do this, each PHY must timestamp inbound PDelay_Req messages and deliver the timestamp to the processor via timestamp insertion or via MDIO reads. The processor should then send a PDelay_Resp message and place the ingress timestamp for the PDelay_Req message into the requestReceiptTimestamp field of the PDelay_Resp message. The egress timestamp for the PDelay_Resp message must be captured by the PHY and returned to the processor through MDIO reads. The PDelay_Resp egress timestamp should then be sent in the responseOriginTimestamp field of a PDelay_Resp_Follow_Up message.

6.4 Clock Synchronization and Syntonization

Clock synchronization and syntonization requirements are the same as for end-to-end transparent clocks. See [Section 5.3](#).

7 Performance Results

National has tested simple boundary clock and transparent clock implementations using an FPGA-based evaluation system to generate and forward PTP messages. [Figure 3](#) shows synchronization results for basic testing across a number of BC or TC devices. As a baseline, back-to-back performance is included that does not include a BC or TC. Test conditions include:

1. 1 second sync interval
2. 25 MHz crystal or standard oscillator
3. Each test length > 1000 seconds

The results (shown in [Figure 3](#)) were obtained using a digital oscilloscope to measure the Master to Slave offset using a pulse-per-second (PPS) from each device.

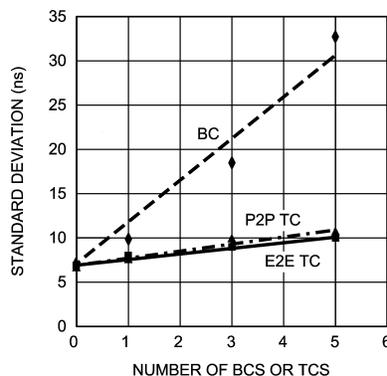


Figure 3. Synchronization Versus Number of Boundary or Transparent Clocks

8 Conclusions

The DP83640 precision PHYTER can be used to implement IEEE 1588 boundary clock and transparent clock devices, providing a highly accurate solution and reducing the amount of additional circuitry required. Timestamps are captured at the most accurate point by implementing the timestamp capabilities in the physical layer. In addition, the DP83640 includes mechanisms to control the independent PTP clocks simultaneously, providing a simple method to ensure the PTP clocks are always synchronized.

9 References

- *IEEE P1588TM D2.2 Draft Standard for a Precision Clock Synchronization Protocol for Networked Measurement and control Systems* (2007) Institute of Electrical and Electronics Engineers, Inc.
- *DP83640 Software Development Guide*
- *DP83640 Precision PHYTER - IEEE 1588 Precision Time Protocol Transceiver* ([SNOSAY8](#))

-
- *AN-1730 DP83640 Synchronous Ethernet Mode: Achieving Sub-Nanosecond Accuracy in PTP Applications* ([SNLA100](#))
 - *AN-1794 Using RMI Master Mode* ([SNLA101](#))
 - *National Semiconductor Ethernet PHYTER - Software Development Guide*. (2007) National Semiconductor Corporation.

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, enhancements, improvements and other changes to its semiconductor products and services per JESD46, latest issue, and to discontinue any product or service per JESD48, latest issue. Buyers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All semiconductor products (also referred to herein as "components") are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its components to the specifications applicable at the time of sale, in accordance with the warranty in TI's terms and conditions of sale of semiconductor products. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by applicable law, testing of all parameters of each component is not necessarily performed.

TI assumes no liability for applications assistance or the design of Buyers' products. Buyers are responsible for their products and applications using TI components. To minimize the risks associated with Buyers' products and applications, Buyers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right relating to any combination, machine, or process in which TI components or services are used. Information published by TI regarding third-party products or services does not constitute a license to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of significant portions of TI information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

Resale of TI components or services with statements different from or beyond the parameters stated by TI for that component or service voids all express and any implied warranties for the associated TI component or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Buyer acknowledges and agrees that it is solely responsible for compliance with all legal, regulatory and safety-related requirements concerning its products, and any use of TI components in its applications, notwithstanding any applications-related information or support that may be provided by TI. Buyer represents and agrees that it has all the necessary expertise to create and implement safeguards which anticipate dangerous consequences of failures, monitor failures and their consequences, lessen the likelihood of failures that might cause harm and take appropriate remedial actions. Buyer will fully indemnify TI and its representatives against any damages arising out of the use of any TI components in safety-critical applications.

In some cases, TI components may be promoted specifically to facilitate safety-related applications. With such components, TI's goal is to help enable customers to design and create their own end-product solutions that meet applicable functional safety standards and requirements. Nonetheless, such components are subject to these terms.

No TI components are authorized for use in FDA Class III (or similar life-critical medical equipment) unless authorized officers of the parties have executed a special agreement specifically governing such use.

Only those TI components which TI has specifically designated as military grade or "enhanced plastic" are designed and intended for use in military/aerospace applications or environments. Buyer acknowledges and agrees that any military or aerospace use of TI components which have **not** been so designated is solely at the Buyer's risk, and that Buyer is solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI has specifically designated certain components as meeting ISO/TS16949 requirements, mainly for automotive use. In any case of use of non-designated products, TI will not be responsible for any failure to meet ISO/TS16949.

Products

Audio	www.ti.com/audio
Amplifiers	amplifier.ti.com
Data Converters	dataconverter.ti.com
DLP® Products	www.dlp.com
DSP	dsp.ti.com
Clocks and Timers	www.ti.com/clocks
Interface	interface.ti.com
Logic	logic.ti.com
Power Mgmt	power.ti.com
Microcontrollers	microcontroller.ti.com
RFID	www.ti-rfid.com
OMAP Applications Processors	www.ti.com/omap
Wireless Connectivity	www.ti.com/wirelessconnectivity

Applications

Automotive and Transportation	www.ti.com/automotive
Communications and Telecom	www.ti.com/communications
Computers and Peripherals	www.ti.com/computers
Consumer Electronics	www.ti.com/consumer-apps
Energy and Lighting	www.ti.com/energy
Industrial	www.ti.com/industrial
Medical	www.ti.com/medical
Security	www.ti.com/security
Space, Avionics and Defense	www.ti.com/space-avionics-defense
Video and Imaging	www.ti.com/video

TI E2E Community

e2e.ti.com