

International Telecommunication Union

**ITU-T**

TELECOMMUNICATION  
STANDARDIZATION SECTOR  
OF ITU

**Series G**  
**Supplement 65**  
(10/2018)

SERIES G: TRANSMISSION SYSTEMS AND MEDIA,  
DIGITAL SYSTEMS AND NETWORKS

---

**Simulations of transport of time over packet  
networks**

ITU-T G-series Recommendations – Supplement 65

ITU-T



ITU-T G-SERIES RECOMMENDATIONS  
**TRANSMISSION SYSTEMS AND MEDIA, DIGITAL SYSTEMS AND NETWORKS**

INTERNATIONAL TELEPHONE CONNECTIONS AND CIRCUITS	G.100–G.199
GENERAL CHARACTERISTICS COMMON TO ALL ANALOGUE CARRIER-TRANSMISSION SYSTEMS	G.200–G.299
INDIVIDUAL CHARACTERISTICS OF INTERNATIONAL CARRIER TELEPHONE SYSTEMS ON METALLIC LINES	G.300–G.399
GENERAL CHARACTERISTICS OF INTERNATIONAL CARRIER TELEPHONE SYSTEMS ON RADIO-RELAY OR SATELLITE LINKS AND INTERCONNECTION WITH METALLIC LINES	G.400–G.449
COORDINATION OF RADIOTELEPHONY AND LINE TELEPHONY	G.450–G.499
TRANSMISSION MEDIA AND OPTICAL SYSTEMS CHARACTERISTICS	G.600–G.699
DIGITAL TERMINAL EQUIPMENTS	G.700–G.799
DIGITAL NETWORKS	G.800–G.899
DIGITAL SECTIONS AND DIGITAL LINE SYSTEM	G.900–G.999
MULTIMEDIA QUALITY OF SERVICE AND PERFORMANCE – GENERIC AND USER-RELATED ASPECTS	G.1000–G.1999
TRANSMISSION MEDIA CHARACTERISTICS	G.6000–G.6999
DATA OVER TRANSPORT – GENERIC ASPECTS	G.7000–G.7999
PACKET OVER TRANSPORT ASPECTS	G.8000–G.8999
ACCESS NETWORKS	G.9000–G.9999

*For further details, please refer to the list of ITU-T Recommendations.*

# Supplement 65 to ITU-T G-series Recommendations

## Simulations of transport of time over packet networks

### Summary

Supplement 65 to the G-series Recommendations describes the mathematical modelling and simulation analyses to support the development of ITU-T Recommendations on the transport of time over packet networks. The main purpose of this Supplement is to document this work, as it forms the basis for the requirements in the relevant Recommendations.

Both time-domain and frequency domain models and analyses are described.

Since SyncE noise accumulation results are needed as an input to the simulation of time error accumulation, models and simulation analyses for SyncE noise generation and accumulation are also described. The modelling and simulation work described here is limited to steady-state behaviour, non-enhanced clocks, full-timing support from the network, and frequency provided by the physical layer by SyncE. Simulation cases that do not have these limitations are for further study.

### History

Edition	Recommendation	Approval	Study Group	Unique ID*
1.0	ITU-T G Suppl. 65	2018-10-19	15	<a href="http://handle.itu.int/11.1002/1000/13822">11.1002/1000/13822</a>

### Keywords

Mathematical modelling, noise accumulation, simulation, time error accumulation.

---

\* To access the Recommendation, type the URL <http://handle.itu.int/> in the address field of your web browser, followed by the Recommendation's unique ID. For example, <http://handle.itu.int/11.1002/1000/11830-en>.

## FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

## NOTE

In this publication, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this publication is voluntary. However, the publication may contain certain mandatory provisions (to ensure, e.g., interoperability or applicability) and compliance with the publication is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the publication is required of any party.

## INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this publication may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the publication development process.

As of the date of approval of this publication, ITU had not received notice of intellectual property, protected by patents, which may be required to implement this publication. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database at <http://www.itu.int/ITU-T/ipr/>.

© ITU 2019

All rights reserved. No part of this publication may be reproduced, by any means whatsoever, without the prior written permission of ITU.

## Table of Contents

		Page
1	Scope.....	1
2	References.....	1
3	Definitions .....	2
	3.1 Terms defined elsewhere .....	2
	3.2 Terms defined in this Supplement .....	2
4	Abbreviations and acronyms .....	2
5	Conventions .....	3
6	Introduction.....	3
7	Description of scenarios .....	3
8	Mathematical description of a synchronous Ethernet reference chain and simulation models .....	4
	8.1 Time-domain mathematical model, simulator, and simulation results.....	4
	8.2 Frequency-domain mathematical model, simulator, and simulation results ..	44
9	Time-domain mathematical description of a telecom boundary clock (T-BC) and a telecom time slave clock (T-TSC) with SyncE assist for frequency transport, and associated transfer functions and frequency responses .....	60
	9.1 Introduction .....	60
	9.2 T-BC and T-TSC model that neglects SyncE noise generation .....	72
	9.3 T-BC and T-TSC model that includes SyncE noise generation .....	98
10	Frequency domain mathematical description of a T-BC and a T-TSC .....	107
	10.1 Introduction .....	107
	10.2 Model for a T-BC or T-TSC with no filtering.....	107
	10.3 Model for a T-BC or T-TSC with filtering.....	115
	10.4 Modelling of time error accumulation in a T-BC and T-TSC and a chain of clocks.....	119
11	Description of time-domain simulator and implementation of models .....	123
	11.1 Introduction .....	123
	11.2 Overall description of simulator .....	124
	11.3 Modelling of noise generation in the simulator PLL filter .....	132
	11.4 Modelling of timestamping relative to the corrected clock, and the delay request/response mechanism, in the simulator .....	132
12	Steady-state time domain simulation cases and results .....	133
	12.1 Simulation results for HRM3 cases based on single replications of simulations.....	133
	12.2 Simulation results for HRM3 cases based on single replications of simulations.....	151
	12.3 Simulation results for HRM3 cases with timestamping relative to the corrected or uncorrected clock in PTP, based on single replications of simulations.....	160

	<b>Page</b>
12.4 Simulation results for HRM2 cases based on single and multiple replications of simulations.....	189
Appendix I – Sub-Nyquist artefacts and sampling moiré effects when measuring PTP to PTP and PTP to 1 PPS noise transfer .....	207
I.1 Introduction .....	207
I.2 Description of sub-Nyquist artefacts and examples .....	207
I.3 Discussion.....	219
Bibliography.....	220

# Supplement 65 to ITU-T G-series Recommendations

## Simulations of transport of time over packet networks

### 1 Scope

This Supplement describes the mathematical modelling and simulation analyses done to support the development of ITU-T Recommendations on the transport of time over packet networks. The main purpose of this Supplement is to document this work, as it forms the basis for the requirements in the relevant Recommendations. The mathematical models and simulation analyses described here are for cases that assume the following:

- Full timing support, i.e., every node is PTP capable (this means that every node is either a T-BC or T-TSC).
- Frequency is provided by the physical layer, i.e., SyncE.
- SyncE clocks satisfy the requirements of [ITU-T G.8262] (i.e., they are not enhanced clocks).
- PTP clocks are not enhanced.

The network being modelled is in steady-state operation (i.e., there are no SyncE or PTP transients).

Note that analyses for cases where some or all of the above assumptions are relaxed have either been performed, are in progress, or may be performed in the future. The description of this work can be considered for a future amendment or revision of this Supplement.

Both time-domain and frequency domain models and analyses are described.

Since SyncE noise accumulation results are needed as an input to the simulation of time error accumulation, models and simulation analyses for SyncE noise generation and accumulation are also described.

### 2 References

- [ITU-T G.803] Recommendation ITU-T G.803 (2000), *Architecture of transport networks based on the synchronous digital hierarchy (SDH)*.
- [ITU-T G.811] Recommendation ITU-T G.811 (1997), *Timing characteristics of primary reference clocks*.
- [ITU-T G.812] Recommendation ITU-T G.812 (2004), *Timing characteristics of slave clocks suitable for use as node clocks in synchronization networks*.
- [ITU-T G.813] Recommendation ITU-T G.813 (2003), *Timing characteristics of SDH equipment slave clocks (SEC)*.
- [ITU-T G.823] Recommendation ITU-T G.823 (2000), *The control of jitter and wander within digital networks which are based on the 2048 kbit/s hierarchy*.
- [ITU-T G.824] Recommendation ITU-T G.824 (2000), *The control of jitter and wander within digital networks which are based on the 1544 kbit/s hierarchy*.
- [ITU-T G.8251] Recommendation ITU-T G.8251 (2018), *The control of jitter and wander within the optical transport network (OTN)*.
- [ITU-T G.8262] Recommendation ITU-T G.8262/Y.1362 (2018), *Timing characteristics of synchronous Ethernet equipment slave clock*.
- [ITU-T G.8263] Recommendation ITU-T G.8263/Y.1363 (2017), *Timing characteristics of packet-based equipment clocks*.

- [ITU-T G.8271] Recommendation ITU-T G.8271/Y.1366 (2017), *Time and phase synchronization aspects of packet networks*.
- [ITU-T G.8271.1] Recommendation ITU-T G.8271.1/Y.1366.1 (2017), *Network limits for time synchronization in packet networks*.
- [ITU-T G.8273.2] Recommendation ITU-T G.8273.2/Y.1368.2 (2017), *Timing characteristics of telecom boundary clocks and telecom time slave clocks*.
- [ITU-T G.8275.1] Recommendation ITU-T G.8275.1/Y1369.1 (2016), *Precision time protocol telecom profile for phase/time synchronization with full timing support from the network*.
- [IEEE1588] IEEE 1588 – 2008, *IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems*.
- [IEEE 802.1AS] IEEE 802.1AS – 2011, *IEEE Standard for Local and metropolitan area networks–Timing and Synchronization for Time-Sensitive Applications in Bridged Local Area Networks*.
- [IEEE 802.1Q] IEEE 802.1Q – 2014, *IEEE Standard for Local and metropolitan area networks–Bridges and Bridged Networks*.

### 3 Definitions

#### 3.1 Terms defined elsewhere

None.

#### 3.2 Terms defined in this Supplement

This Supplement defines the following term:

**3.2.1 local PTP clock:** The clock of a T-BC or T-TSC that provides the local estimate of the time of its T-GM, i.e., it is synchronized to the time of the T-GM. It is either a physical or mathematical clock, and provides PTP time.

### 4 Abbreviations and acronyms

This Supplement uses the following abbreviations and acronyms:

DFT	Discrete Fourier Transform
EEC	Ethernet Equipment Clock
FPM	Flicker Phase Modulation
HRM	Hypothetical Reference Model
MTIE	Maximum Time Interval Error
NCO	Number-Controlled Oscillator
OTN	Optical Transport Network
PDH	Plesiochronous Digital Hierarchy
PHY	Physical Layer
PLL	Phase-Locked Loop
PRC	Primary Reference Clock
PTP	Precision Time Protocol

SDH	Synchronous Digital Hierarchy
SEC	SDH equipment slave clock
SSU	Synchronization Supply Unit
SyncE	Synchronous Ethernet
T-BC	Telecom Boundary Clock
TDEV	Time Deviation
T-GM	Telecom Grand Master
T-TSC	Telecom Time Slave Clock
TVAR	Time Variance
VCO	Voltage-Controlled Oscillator
WFM	White Frequency Modulation
WPM	White Phase Modulation
ZOH	Zero-Order Hold

## 5 Conventions

None.

## 6 Introduction

This Supplement describes the mathematical modelling and simulation analyses to support the development of ITU-T Recommendations on the transport of time over packet networks. The main purpose of this Supplement is to document this work, as it forms the basis for the requirements in the relevant Recommendations.

Both time-domain and frequency domain models and analyses are described.

Since synchronous Ethernet (SyncE) noise accumulation results are needed as input to the simulation of time error accumulation, models and simulation analyses for SyncE noise generation and accumulation are also described. The modelling and simulation work described here is limited to steady-state behaviour, non-enhanced clocks, full-timing support from the network, and frequency provided by the physical layer by SyncE. Simulation cases that do not have these limitations are for further study.

## 7 Description of scenarios

The hypothetical reference models (HRMs) used for the simulations described in this Supplement are documented in Appendix II of [ITU-T G.8271.1] and will therefore only be briefly described in this Supplement.

Appendix II of [ITU-T G.8271.1] describes the following HRMs (where  $N$  is the total number of time clocks in the HRM):

- 1) Telecom grand master (T-GM), followed by  $N - 2$  telecom boundary clocks (T-BCs), followed by a telecom time slave clock (T-TSC), with synchronous Ethernet not present. This is referred to as HRM1.
- 2) T-GM, followed by  $N - 2$  T-BCs, followed by a T-TSC. The tenth synchronization supply unit (SSU) of an ITU-T G.803 reference chain is collocated with the T-GM. An Ethernet equipment clock (EEC) is collocated with each T-BC and the T-TSC. Upstream of the T-GM and tenth SSU is the remainder of the ITU-T G.803 reference chain (see clause 8.1 and

[ITU-T G.803] for more detail. This is referred to as HRM2. In HRM2, the SyncE and PTP transports take the same path, i.e., they are congruent.

- 3) T-GM, followed by  $N - 2$  T-BCs, followed by a T-TSC. Each T-BC and the T-TSC are collocated with the last EEC of a separate ITU-T G.803 reference chain. The T-GM is collocated with an SSU that is synchronized by a full ITU-T G.803 reference chain. The total number of ITU-T G.803 reference chains is  $N$ . This is referred to as HRM3. In HRM3, the SyncE and PTP transports take different paths, i.e., they are non-congruent.

Appendix II of [ITU-T G.8271.1] describes HRMs where  $N = 12$  and  $N = 22$ . In most of the simulation cases described in this Supplement,  $N = 22$ . Appendix II of [ITU-T G.8271.1] also distinguishes the cases where the T-TSC is a stand-alone equipment and where the T-TSC is embedded in the end application equipment. This distinction is not important for the steady-state simulation cases described in this Supplement (the distinction is described in more detail in [ITU-T G.8271.1]).

## **8 Mathematical description of a synchronous Ethernet reference chain and simulation models**

### **8.1 Time-domain mathematical model, simulator, and simulation results**

#### **8.1.1 Introduction**

The time-domain simulation model for a SyncE reference chain was originally developed for use in jitter and wander accumulation studies of the transport of SyncE and STM-1 clients over a network of the optical transport network (OTN) islands. Initial simulation studies of the wander accumulation for transport of these clients over the OTN island HRMs (documented in Appendix VII.2 of [ITU-T G.8251]) indicated that the wander added by the OTN was sufficiently close to the respective maximum time interval error (MTIE) and TDEV network limits that the effect of input wander needed to be accounted for. The ITU-T G.803 synchronization reference chain was adapted for use with the OTN HRMs. The full ITU-T G.803 reference chain (see clause 8.2.4 of [ITU-T G.803] and Figure 8.5 of [ITU-T G.803]) consists of a primary reference clock (PRC), up to  $K$  SSUs, up to  $N$  SECs between any two successive SSUs, and up to  $N$  SECs following the final SSU. For Option 1 networks,  $K = 10$  and  $N = 20$ , with the constraint that the total number of SECs is limited to 60. For Option 2 networks,  $K$  and  $N$  are for further study. Note that, while ITU-T G.803 describes the reference chains in terms of synchronous digital hierarchy (SDH) networks and uses SECs, the reference chain applies equally to SyncE networks. In this case, the SECs are replaced by EECs. This is acceptable because the relevant EEC and SDH equipment slave clock (SEC) requirements are the same (see [ITU-T G.8262] and [ITU-T G.813], respectively).

The ITU-T G.803 reference chain was adapted for use with the OTN HRMs by assuming that the OTN islands replace one EEC/SEC, and that at least one EEC/SEC follows the OTN islands. This means that the maximum number of EECs/SECs prior to the OTN islands, for Option 1 networks, is 58. In addition, it was realized that, since the EEC/SEC bandwidth is much wider than the SSU bandwidth (i.e., the SSU bandwidth is 3 MHz maximum for Type I and 1 MHz maximum for Types II and III, while the EEC/SEC bandwidth is 1-10 Hz for Option 1 and 0.1 Hz maximum for Option 2); this meant that the wander accumulation would be largest if the EECs/SECs were concentrated as close to the end of the reference chain as possible. The resulting worst-case reference chain is shown in Figure VII.3-1 of [ITU-T G.8251] in the following order:

- PRC
- 8 SSUs
- 20 EECs/SECs
- 1 SSU
- 20 EECs/SECs

- 1 SSU
- 18 EECs/SECs
- OTN islands
- 1 EEC/SEC

Appendix II of [ITU-T G.8271.1] describes the various HRMs for the transport of time via PTP with frequency transport via SyncE. In HRM3, the non-congruent scenario, each T-BC and the T-TSC receive frequency from a separate SyncE reference chain. Each of these reference chains is a full ITU-T G.803 reference chain with 60 EECs, i.e., there are 20 EECs following the final SSU. When the simulation studies for time transport over packet networks began, it was decided to re-use the previous work done for OTN, even though those simulations considered only 18 EECs following the final SSU rather than 20. Those previous results indicated that the accumulation over 2 EECs compared to 20 EECs was small, and that the re-use of previous work might allow results to be obtained more quickly. However, HRM2 was subsequently considered; this HRM is the congruent scenario, and there are 20 EECs following the final SSU. It was therefore necessary to perform new simulations of a SyncE reference chain with 20 EECs following the final SSU, and an 11<sup>th</sup> SSU following the final EEC. However, the simulations and analyses already performed for HRM3 were not re-done, because the effect of the additional two EECs in HRM3 would have a small impact on SyncE noise accumulation.

In the simulation model, each SSU and EEC is modelled as a low-pass filter that filters incoming noise from the previous clock in the chain, with noise (wander) generation added to the result. The PRC is modelled as the sum of phase error due to noise generation plus a frequency offset. The model development was done in several steps. First, noise generation models were developed for Option 1 and 2 EECs, based on the wander generation MTIE and TDEV specification in clause 8.1 of [ITU-T G.8262]. These models were conservative, in the sense that MTIE and TDEV for the models were well above the ITU-T G.8262 MTIE and TDEV wander generation masks (see the following clause for details). Next, the wander generation models were used with low-pass filter models in wander accumulation simulation models. In the initial OTN simulation studies, results were obtained for both Option 1 and Option 2 network; while the ITU-T G.803 values for  $N$  and  $K$  are strictly applicable only for Option 1, they were also used for Option 2 in the absence of any other proposed or suggested values. The initial results for OTN indicated that while the SyncE client requirements could be met at the egress of the OTN for Option 2 they exceeded the requirements for Option 1.<sup>1</sup> However, as noted above, the noise generation model for Option 1 was conservative. It was pointed out that less conservative noise generation models were developed in ETSI (see [b-ETSI01] and [b-ETSI02]) when the Option 1 requirements were initially developed for SDH, and that these models could be used. The ETSI models were subsequently inserted in the simulation models, and the results for SyncE client wander accumulation at the OTN egress were now acceptable. As a result, the ETSI wander generation models for Option 1 were used in all subsequent OTN simulation studies for Option 1, and in all simulations (for Option 1) for time transport using PTP with frequency transport using SyncE.

In the following clauses, the initial Option 1 and Option 2 wander generation models will be described first, followed by the wander accumulation models and simulation results. Finally, the ETSI wander generation models for Option 1 and the corresponding simulation results are described.

---

<sup>1</sup> More precisely, meeting the requirements would have required a client desynchronizer bandwidth and/or additional phase information granularity that would be too small to be practical.

## 8.1.2 Initial EEC/SEC, SSU, and PRC wander generation models

### 8.1.2.1 EEC/SEC wander generation models

Option 1 and 2 input noise models were developed to match, as closely as possible, the respective ITU-T G.813 and ITU-T G.8262 wander generation MTIE and TDEV masks.

Option 1 MTIE mask is shown in Figure 1 of [ITU-T G.813] (Tables 1 and 2 of [ITU-T G.813]) and repeated in Figure 1 of [ITU-T G.8262] (Tables 1 and 2 of [ITU-T G.8262]). Option 1 MTIE mask is reproduced in Figure 1, Table 1 and Table 2 in this Supplement. Option 1 TDEV mask is shown in Figure 2 of [ITU-T G.813] (Table 3 of [ITU-T G.813]) and repeated in Figure 2 of [ITU-T G.8262] (Table 3 of [ITU-T G.8262]). Option 1 TDEV mask is reproduced in Figure 2 and Table 3 in this Supplement.

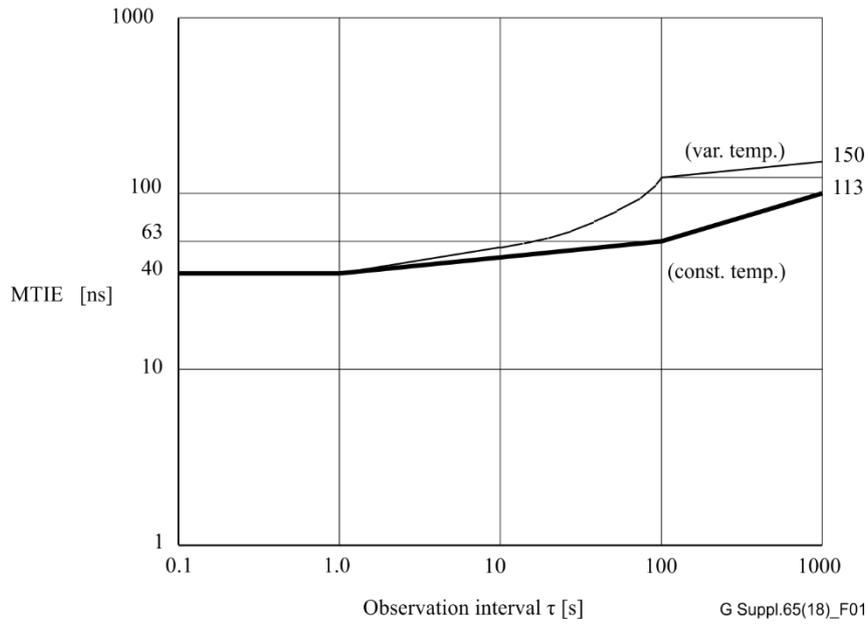
Option 2 MTIE mask is shown in Figure 3 of [ITU-T G.813] (Table 4 of [ITU-T G.813]) and repeated in Figure 3 of [ITU-T G.8262] (Table 4 of [ITU-T G.8262]). Option 2 MTIE mask is reproduced in Figure 1, Table 1 and Table 2 in this Supplement. Option 1 TDEV mask is shown in Figure 2 of [ITU-T G.813] (Table 3 of [ITU-T G.813]) and repeated in Figure 2 of [ITU-T G.8262] (Table 3 of [ITU-T G.8262]). Option 1 TDEV mask is reproduced in Figure 2 and Table 3 of this Supplement.

**Table 1 – Option 1 wander generation MTIE, with constant temperature  
(see Table 1 of [ITU-T G.813] and [ITU-T G.8262])**

MTIE limit	Observation interval $\tau$
40 ns	$0.1 < \tau \leq 1$ s
$40\tau^{0.1}$ ns	$1 < \tau \leq 100$ s
$25.25\tau^{0.2}$ ns	$100 < \tau \leq 1000$ s

**Table 2 – Option 1 additional wander generation MTIE, with temperature effects  
(see Table 2 of [ITU-T G.813] and [ITU-T G.8262])**

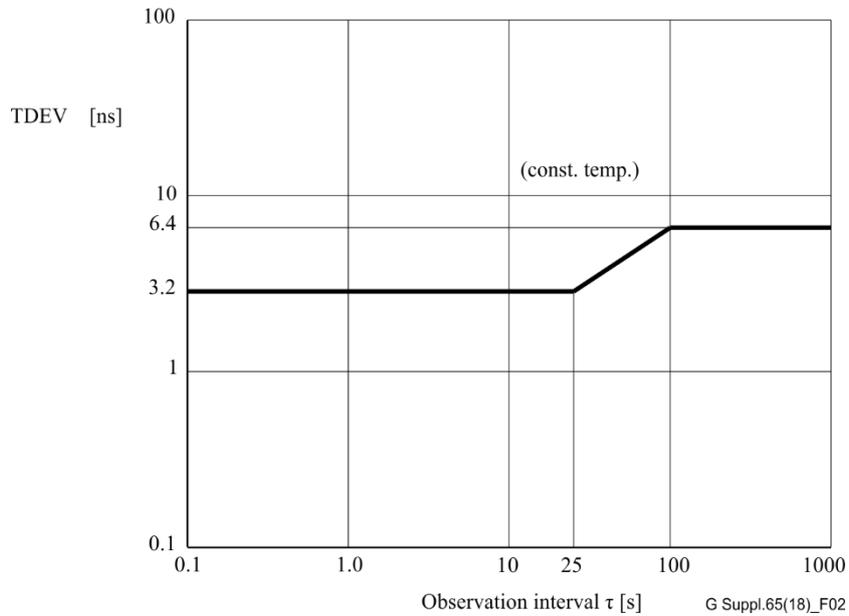
Additional MTIE allowance	Observation interval $\tau$
$0.5\tau$ ns	$\tau \leq 100$ s
50 ns	$\tau > 100$ s



**Figure 1 – Option 1 wander generation MTIE**  
 (see Figure 1 of [ITU-T G.813] and Figure 1 of [ITU-T G.8262])

**Table 3 – Option 1 wander generation TDEV, with constant temperature**  
 (see Table 3 of [ITU-T G.813] and Table 3 of [ITU-T G.8262])

TDEV limit	Observation interval $\tau$
3.2 ns	$0.1 < \tau \leq 25$ s
$0.64\tau^{0.5}$ ns	$25 < \tau \leq 100$ s
6.4 ns	$100 < \tau \leq 1000$ s



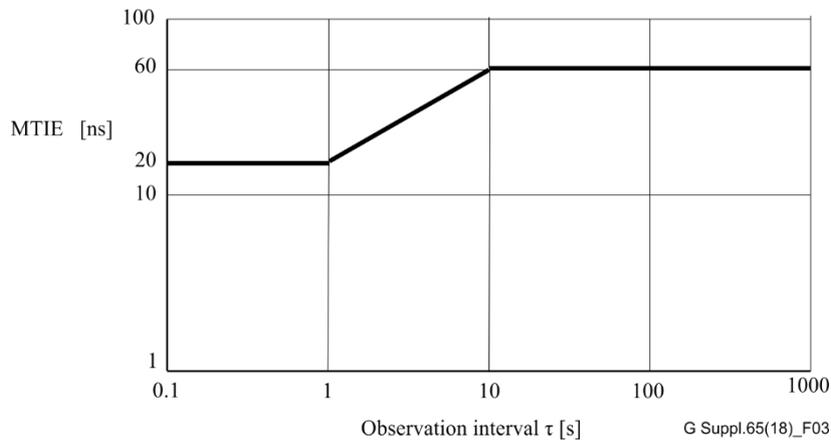
**Figure 2 – Option 1 wander generation TDEV, with constant temperature**  
 (see Figure 2 of [ITU-T G.813] and Figure 2 of [ITU-T G.8262])

**Table 4 – Option 2 wander generation MTIE, with constant temperature**  
 (see Table 4 of [ITU-T G.813] and Table 4 of [ITU-T G.8262])

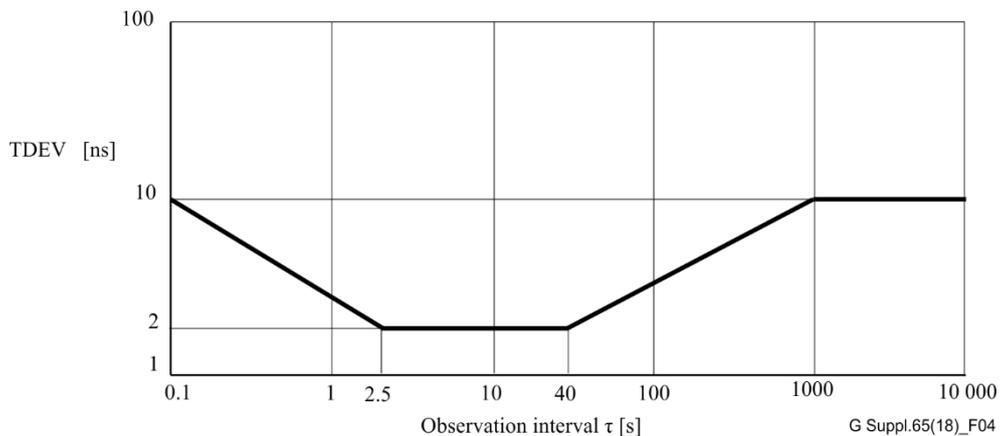
MTIE limit	Observation interval $\tau$
20 ns	$0.1 < \tau \leq 1$ s
$20\tau^{0.48}$ ns	$1 < \tau \leq 10$ s
60 ns	$10 < \tau \leq 1000$ s

**Table 5 – Option 2 wander generation TDEV, with constant temperature**  
 (see Table 5 of [ITU-T G.813] and Table 5 of [ITU-T G.8262])

TDEV limit	Observation interval $\tau$
$3.2\tau^{-0.5}$ ns	$0.1 < \tau \leq 2.5$ s
2 ns	$2.5 < \tau \leq 40$ s
$0.32\tau^{0.5}$ ns	$40 < \tau \leq 1000$ s
10 ns	$1000 < \tau \leq 10\,000$ s



**Figure 3 – Option 2 wander generation MTIE, with constant temperature**  
 (see Figure 3 of [ITU-T G.813] and Figure 3 of [ITU-T G.8262])



**Figure 4 – Option 2 wander generation TDEV, with constant temperature**  
 (see Figure 4 of [ITU-T G.813] and Figure 4 of [ITU-T G.8262])

Previous studies of wander accumulation in Option 2 synchronization reference chains were performed in [b-Garner01] and [b-Garner02], and the results are summarized in Annex B of [b-ANSI01]. As part of this work, it was necessary to develop wander generation models whose MTIE and TDEV met Option 2 masks above. It was found that the MTIE and TDEV masks were not consistent, in the sense that a noise process constructed as a superposition of power-law noise processes that closely matches the TDEV mask (Figure 4 above) will exceed the MTIE mask (Figure 3 above) for a range of observation intervals. Similarly, as noise process constructed as a superposition of power-law noise processes that matches the MTIE mask will be below the TDEV mask for a range of observation intervals. In the previous studies, both sets of models were considered (the former in [b-Garner01] and the latter in [b-Garner02]). For the current purposes, - two models were developed for the EEC/SEC noise generation as follows:

- a) Option 2, Model 1: meets (i.e., matches) TDEV mask, exceeds MTIE mask, and
- b) Option 2, Model 2: meets (i.e., does not fall below) MTIE mask as closely as possible, falls below TDEV mask.

Option 2, Model 1 is the same as the SEC noise generation model of [b-Garner01]; Option2, Model 2 is different from the SEC noise generation model of [b-Garner02] because the model of [b-Garner02] does not actually exceed the MTIE mask in all cases. For the analysis here, the models should exceed the respective masks, but ideally by as little as possible, to ensure that the simulated input noise exceeds any noise that would be present in practice. This ensures that if the simulated OTN client signal accumulated wander meets the respective network limits, the network limits will be met by an actual client signal.

Two models are also developed for Option 1 EEC/SEC noise generation as follows:

- a) Option 1, Model 1: meets (i.e., matches) variable-temperature MTIE mask, exceeds TDEV mask, and
- b) Option 1, Model 2: meets (i.e., does not fall below) TDEV mask as closely as possible, falls below variable-temperature MTIE mask.

As indicated in the introduction to this clause, both these models are conservative, though Option 1, Model 1 is more conservative than Option 1, Model 2. While Option 1, Model 2 falls below the variable-temperature MTIE mask, it does so only for a limited range of observation intervals (see below), and is above the MTIE mask for most of the full range of observation intervals.

Note that, while the Option 2 TDEV mask represents a noise level that is above the Option 2 MTIE mask for some observation intervals, the Option 1 variable-temperature MTIE mask represents a noise level that is above the Option 1 TDEV mask for some observation intervals.

The simulation time step for all four noise generation models is 0.1 s. This is the minimum observation interval for the respective MTIE and TDEV masks. The simulation time was 100,000 s for each case.

Each noise model is constructed by superposing a white phase modulation (WPM), white frequency modulation (WFM), and two flicker phase modulation (FPM) processes. The WFM process is filtered by a first-order, high-pass filter. One of the FPM processes is filtered by a first-order, low-pass filter. Each FPM process is simulated by passing a white noise sequence through a Barnes/Jarvis/Greenhall filter; see [b-Barnes01] and [b-Barnes02] (the implementation of the Barnes/Jarvis/Greenhall filter is as described in [b-Barnes02]; more recently, this implementation has been described in clause I.2.1 of [ITU-T G.8263]). Each WFM process is simulated by generating a white noise sample every 6.5 s

and adding the fraction  $p$  of the sample at each time step the sample to a running total, where  $p$  is equal to the time step in s divided by 6.5 s.<sup>2</sup> All the white noise sequences have Gaussian distribution.

The standard deviations for the WPM sequence and the white noise sequences used to generate the WFM and two FPM sequences, for each of the four models, are given in Tables 6 to 9. These tables also give the corner frequencies for the respective first-order high-pass and low-pass filters that the WFM and one of the FPM sequences are passed through. Note also that when constructing a WPM noise process to match a TDEV mask (i.e., with slope  $-1/2$ ), the white noise standard deviation depends on the sampling interval. It can be shown (e.g., using the theoretical definition of TDEV in clause II.3 of [ITU-T G.810] and directly evaluating the various expected values) that, for a sequence of random samples that are independent and identically distributed (i.i.d.), with standard deviation  $\sigma$ , TDEV is given by

$$\text{TDEV}(\tau) = \sigma \sqrt{\frac{\tau_0}{\tau}} = \frac{\sigma}{\sqrt{n}}, \quad (8-1)$$

where  $\tau_0$  is the sampling interval,  $\tau$  is the observation interval, and  $n = \tau/\tau_0$  is the number of samples minus 1. For the models here, the WPM standard deviation values in Table 6 to Table 9 are chosen for the 0.1 s sampling interval.

**Table 6 – Noise source parameters for Option 1, Model 1  
(meets variable-temperature MTIE mask, exceeds TDEV mask)**

Noise source	Input white noise standard deviation (ns)	Low-pass filter bandwidth (Hz)	High-pass filter bandwidth (Hz)
WPM	0.0	–	–
WFM	2.450	–	$3.183 \times 10^{-3}$
FPM1	7.336	–	–
FPM2	15.96	$3.183 \times 10^{-3}$	–

**Table 7 – Noise source parameters for Option 1, Model 2  
(meets TDEV mask, falls below variable-temperature MTIE mask)**

Noise source	Input white noise standard deviation (ns)	Low-pass filter bandwidth (Hz)	High-pass filter bandwidth (Hz)
WPM	0.0	–	–
WFM	1.750	–	$3.183 \times 10^{-3}$
FPM1	5.240	–	–
FPM2	11.40	$3.183 \times 10^{-3}$	–

<sup>2</sup> Specifically, if  $T$  is the time step size in s and  $A$  is the value of the sample, then the quantity  $pA/(6.5 \text{ s})$  is added at each time step starting at the generation of  $A$  and ending at the time step that precedes the generation of the next sample. It is assumed that the time step  $T$  is an integer sub-multiple of 6.5 s.

**Table 8 – Noise source parameters for Option 2, Model 1  
(meets TDEV mask, exceeds MTIE mask)**

Noise source	Input white noise standard deviation (ns)	Low-pass filter bandwidth (Hz)	High-pass filter bandwidth (Hz)
WPM	3.162	–	–
WFM	1.750	–	$3.183 \times 10^{-3}$
FPM1	2.750	–	–
FPM2	16.00	$3.183 \times 10^{-3}$	–

**Table 9 – Noise source parameters for Option 1, Model 2  
(meets MTIE mask, falls below TDEV mask)**

Noise source	Input white noise standard deviation (ns)	Low-pass filter bandwidth (Hz)	High-pass filter bandwidth (Hz)
WPM	2.135	–	–
WFM	1.181	–	$3.183 \times 10^{-3}$
FPM1	1.856	–	–
FPM2	10.80	$3.183 \times 10^{-3}$	–

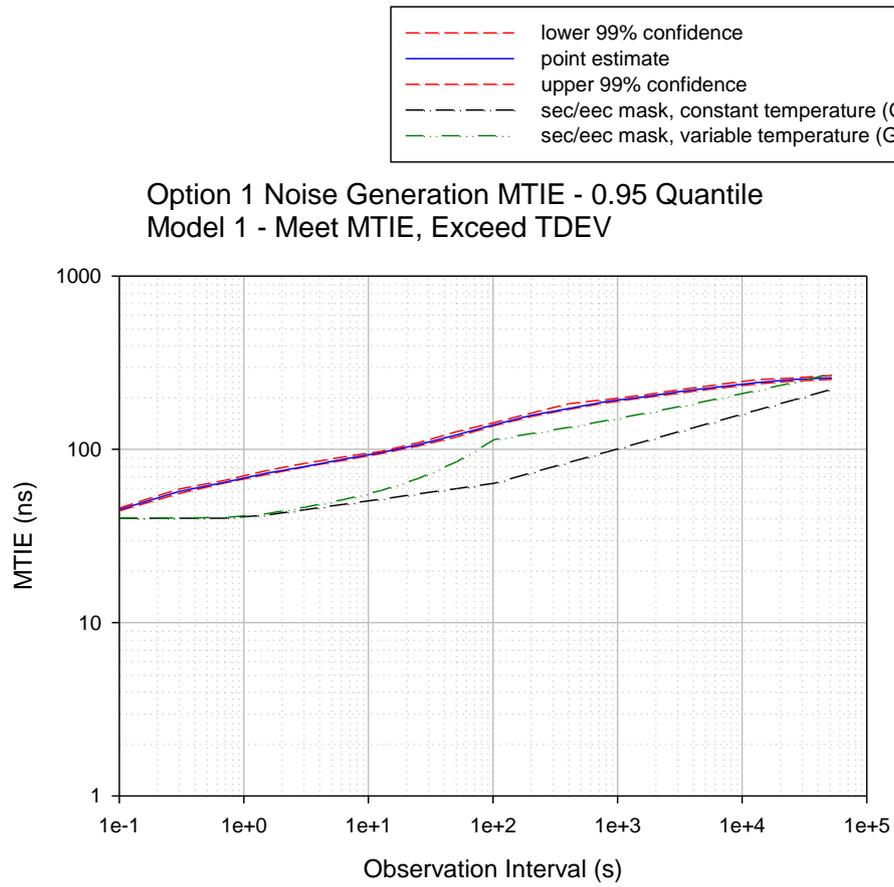
MTIE and TDEV for each of the four models are given in Figures 5 to 12. In each figure, a 99% confidence interval for the 0.95 quantile was obtained by running 300 independent replications of the simulation. For each observation interval value, the 300 MTIE or TDEV values were sorted in ascending order, and the 99% confidence interval for the 0.95 quantile extends from the 275th through 294th values (see Appendix VIII of [ITU-T G.8251] for details).

MTIE for Option 1, Model 1 is approximately as low as possible without having it exceed the variable-temperature MTIE mask (because for the shortest and longest observation intervals it is approximately at the level of the mask, though it is above the mask for other intervals). TDEV for Option 1, Model 1 exceeds the corresponding mask, by 25% or more.

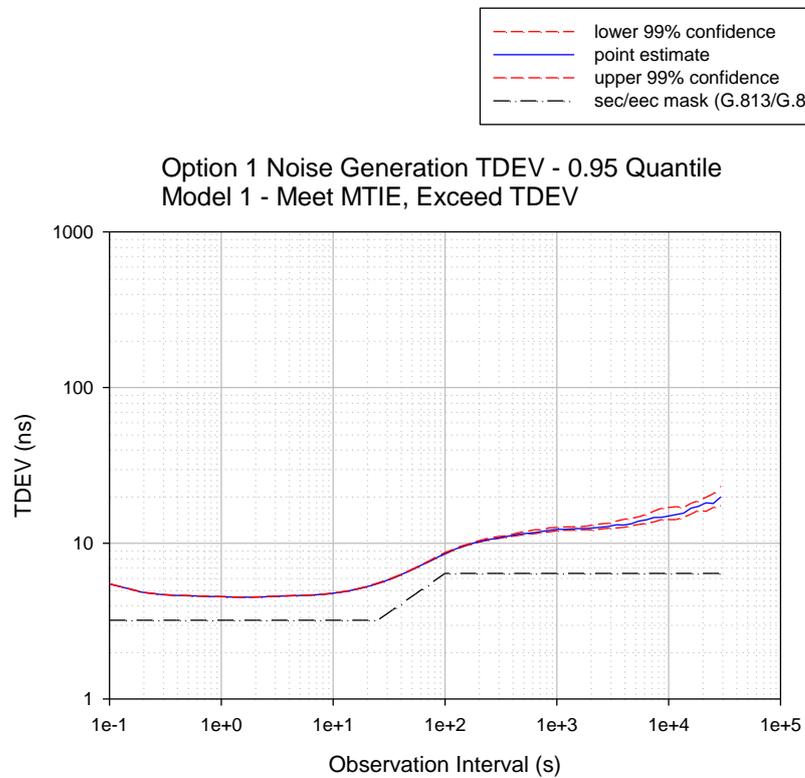
TDEV for Option 1, Model 2 matches the TDEV mask for observation intervals of 100 s or less, and exceeds the mask for longer intervals. It is approximately as low as possible without going below the mask. MTIE is above the MTIE mask for some observation intervals and below the mask for other observation intervals.

TDEV for Option 2, Model 1 matches the TDEV mask for observation intervals of 1000 s or less, and exceeds the mask for longer intervals. It is approximately as low as possible without going below the mask. MTIE is above the MTIE mask for all observation intervals (in the range 0.1 – 100,000 s).

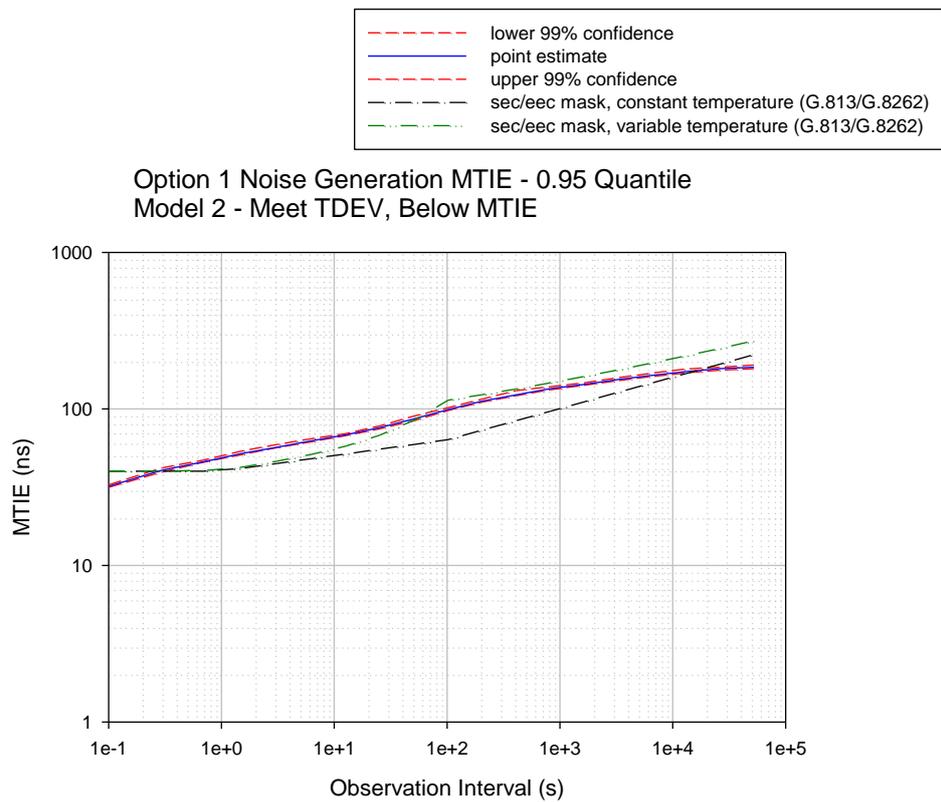
MTIE for Option 2, Model 2 is approximately as low as possible without having it go below the variable-temperature MTIE mask (it is very close to the mask for observation intervals around 10 s, and above the mask for longer and shorter intervals). TDEV for Option 2, Model 2 is below the TDEV mask for all observation intervals ((in the range 0.1 – 100,000 s).



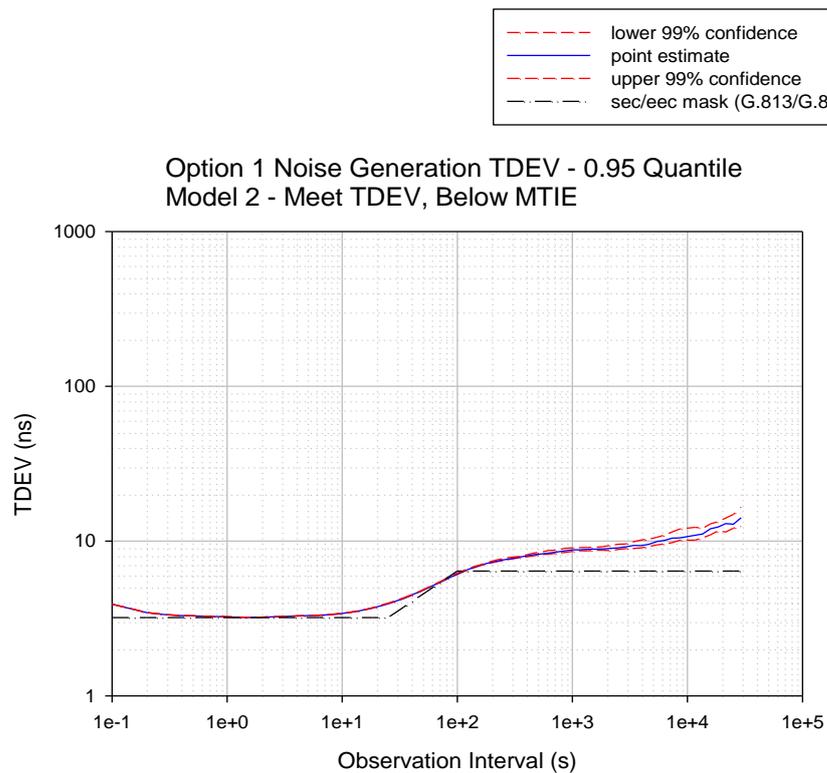
**Figure 5 – Option 1, Model 1 EEC/SEC wander generation MTIE**



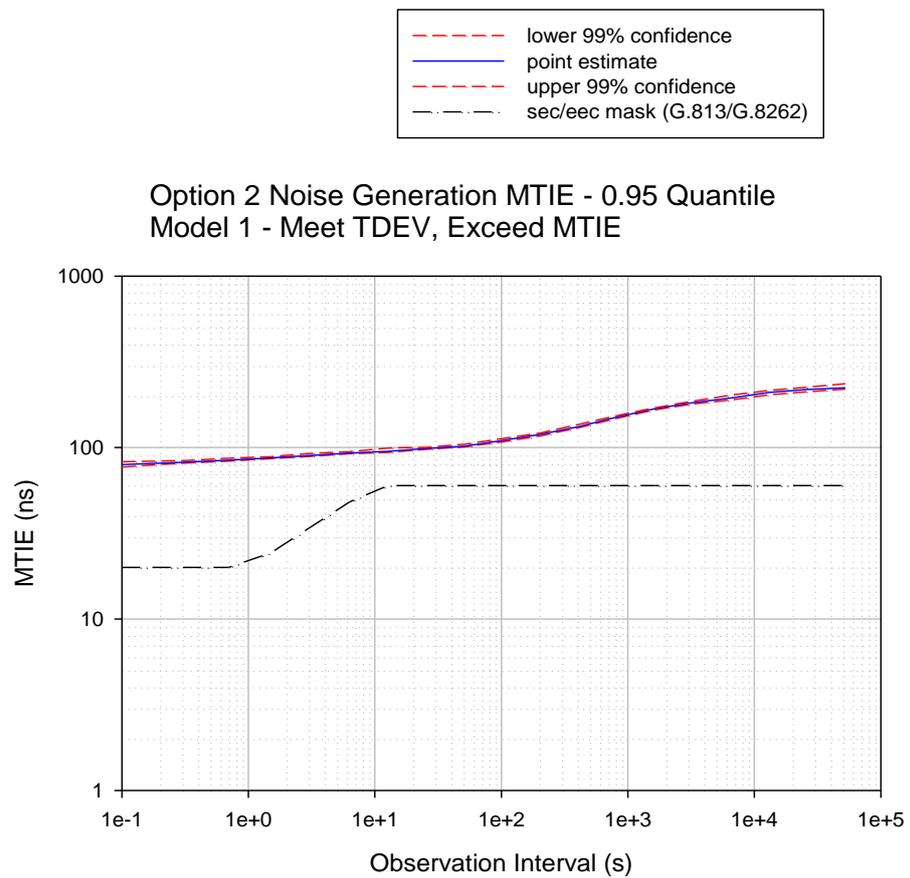
**Figure 6 – Option 1, Model 1 EEC/SEC wander generation TDEV**



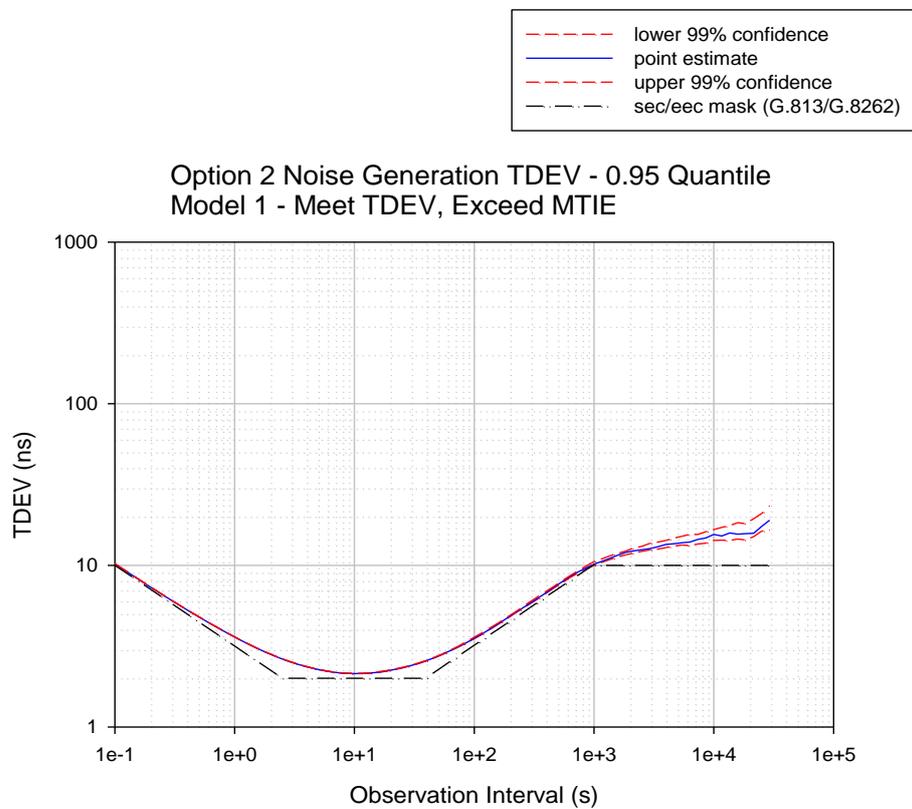
**Figure 7 – Option 1, Model 2 EEC/SEC wander generation MTIE**



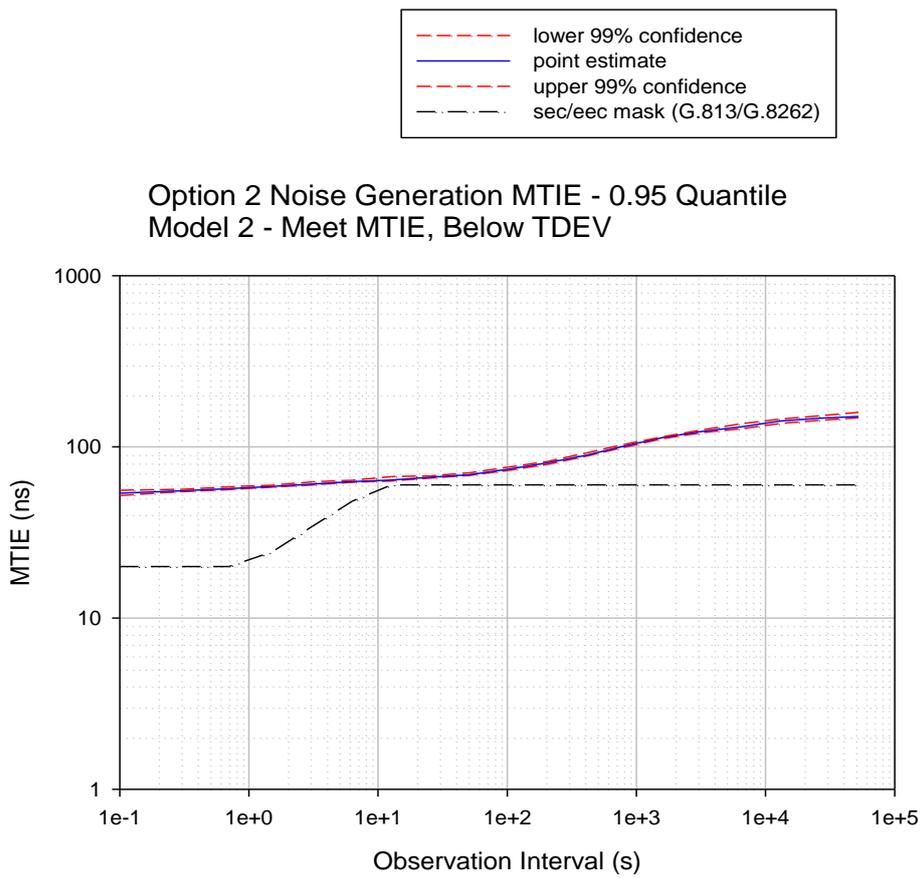
**Figure 8 – Option 1, Model 2 EEC/SEC wander generation TDEV**



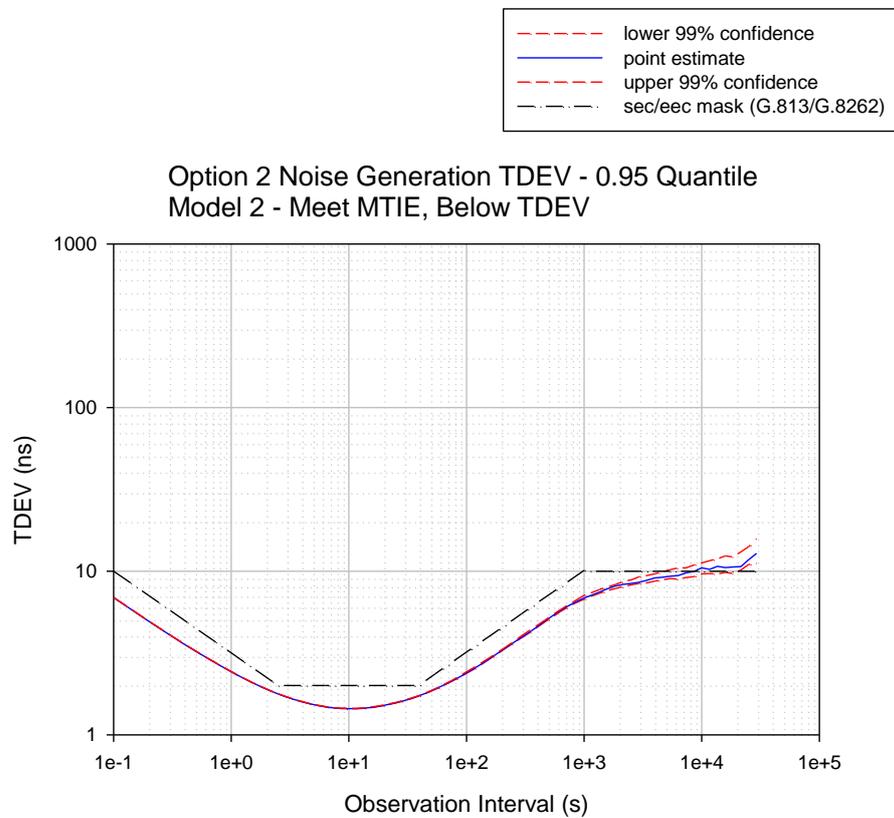
**Figure 9 – Option 2, Model 1 EEC/SEC wander generation MTIE**



**Figure 10 – Option 2, Model 1 EEC/SEC wander generation TDEV**



**Figure 11 – Option 2, Model 2 EEC/SEC wander generation MTIE**



**Figure 12 – Option 2, Model 2 EEC/SEC wander generation TDEV**

### 8.1.2.2 SSU wander generation models

The SSU input noise models were developed to match, as closely as possible, the respective ITU-T G.812 wander generation MTIE and TDEV masks. For Option 1 networks, the masks for a Type I node clock were used. For Option 2 networks, the masks for a Type II and Type III node clock were used.

The MTIE masks are shown in Figure 1 of [ITU-T G.812] (Tables 3, 4 and 5 of [ITU-T G.812]). The MTIE masks are reproduced in Figure 13 and Table 10 to Table 12. The TDEV masks are shown in Figure 2 of [ITU-T G.812] and (Tables 6 and 7 of [ITU-T G.812]). The TDEV masks are reproduced in Figure 14, Table 13 and Table 14.

**Table 10 – Type I node clock wander generation MTIE at constant temperature  
(within  $\pm 1$  K) (see Table 3 of [ITU-T G.812])**

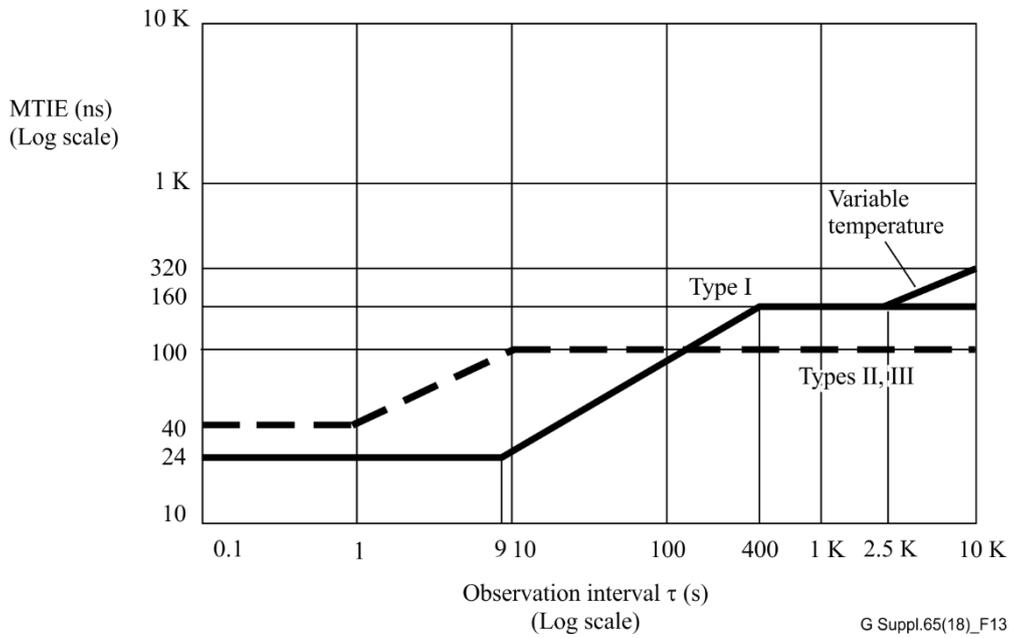
MTIE limit (ns)	Observation interval $\tau$ (s)
24	$0.1 < \tau \leq 9$
$8 \times \tau^{0.5}$	$9 < \tau \leq 400$
160	$400 < \tau \leq 10\,000$

**Table 11 – Type II and II wander generation MTIE at constant temperature  
(within  $\pm 1$  K) (see Table 4 of [ITU-T G.812])**

MTIE limit (ns)	Observation interval $\tau$ (s)
40	$0.1 < \tau \leq 1$
$40 \times \tau^{0.4}$	$1 < \tau \leq 10$
100	$\tau > 10$

**Table 12 – Type I total wander generation MTIE for variable temperature  
(see Table 5 of [ITU-T G.812])**

MTIE limit (ns)	Observation interval $\tau$ (s)
$3.2 \tau^{0.5}$	$2\,500 < \tau \leq 10\,000$
NOTE – For observation periods greater than 10 000 s, the MTIE is expected not to exceed 1 $\mu$ s.	



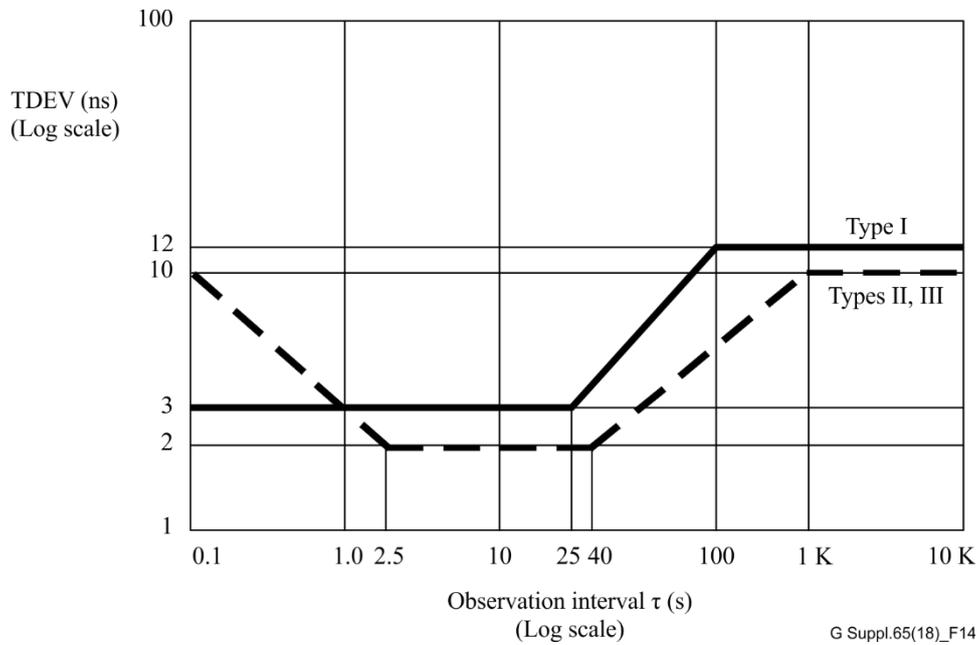
**Figure 13 – SSU wander generation MTIE masks (see Figure 1 of [ITU-T G.812])**

**Table 13 – Type I node clock wander generation TDEV at constant temperature (within  $\pm 1$  K) (see Table 6 of [ITU-T G.812])**

TDEV limit (ns)	Observation interval $\tau$ (s)
3	$0.1 < \tau \leq 25$
$0.12 \tau$	$25 < \tau \leq 100$
12	$100 < \tau \leq 10\,000$

**Table 14 – Type II and III wander generation TDEV at constant temperature (within  $\pm 1$  K) (see Table 7 of [ITU-T G.812])**

TDEV limit (ns)	Observation interval $\tau$ (s)
$3.2 \times \tau^{0.5}$	$0.1 < \tau \leq 2.5$
2	$2.5 < \tau \leq 40$
$0.32 \times \tau^{0.5}$	$40 < \tau \leq 1000$
10	$\tau > 1000$



**Figure 14 – SSU wander generation TDEV masks (see Figure 2 of [ITU-T G.812])**

In contrast with the EEC/SEC wander generation models, it was only necessary to develop one model for each option. This is because the SSU MTIE and TDEV masks for each option are much more consistent with each other compared to the EEC/SEC MTIE and TDEV masks. Note that the SSU models of [b-Garner01] and [b-Garner02] cannot be used here because the Option 2 TDEV mask was changed subsequent to the work of [b-Garner01] and [b-Garner02]. However, the models were developed using the same methodology as for the EEC/SEC models, i.e., as a superposition of a WPM, WFM, and 2 FPM noise processes, with the WFM process high-pass filtered and one of the FPM processes low-pass-filtered.

The standard deviations for the WPM sequence and the white noise sequences used to generate the WFM and two FPM sequences, for the Option 1 and Option 2 SSU wander generation models, are given in Table 15 and Table 16, respectively. These tables also give the corner frequencies for the respective first-order high-pass and low-pass filters that the WFM and one of the FPM sequences are passed through. As with the EEC/SEC models, the sampling interval is 0.1 s and the WPM standard deviation is chosen for this sampling interval. Note that the Option 2 SSU model is the same as for the Option 2 EEC/SEC, Model 1. This is because, for this model, TDEV was at the level of the Option 2 EEC/SEC TDEV mask and MTIE was somewhat above the Option 2 EEC/SEC MTIE mask. For Option 2, the SEC and SSU TDEV masks are the same, and the SSU MTIE mask is somewhat above the level of the EEC/SEC MTIE mask. The increase in level of the SSU mask over the SEC mask is roughly the same as the amount by which MTIE for Option 2, Model 1 exceeds the EEC/SEC MTIE mask.

**Table 15 – Noise source parameters for Option 1 (i.e., Type I) SSU**

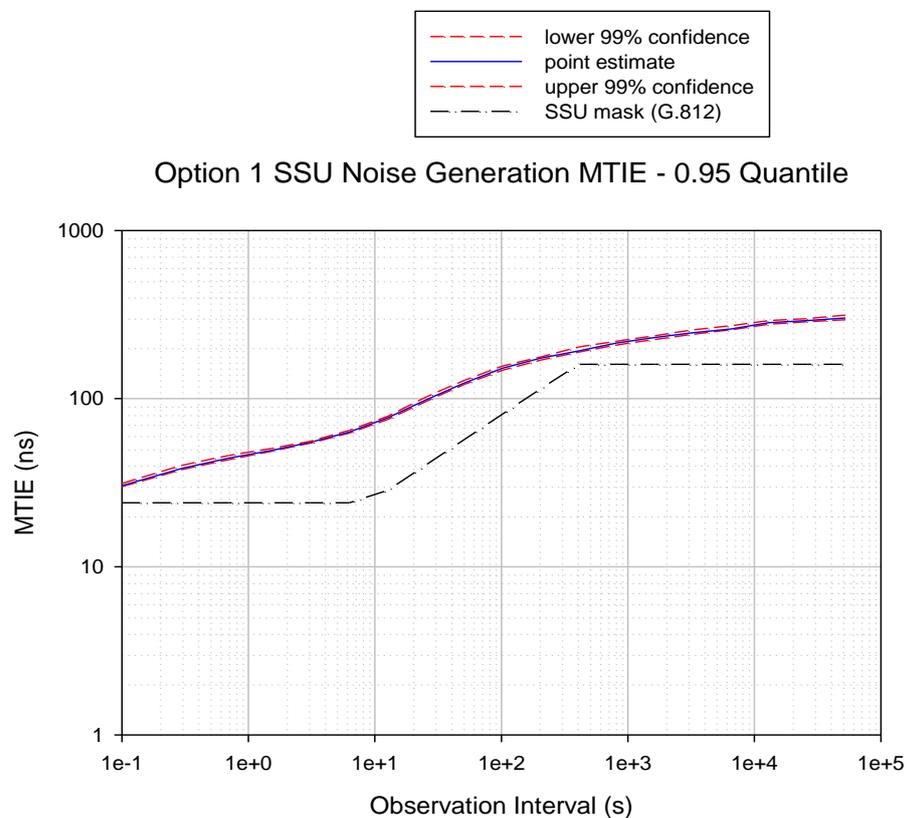
Noise source	Input white noise standard deviation (ns)	Low-pass filter bandwidth (Hz)	High-pass filter bandwidth (Hz)
WPM	0.0	–	–
WFM	0.0	–	$3.183 \times 10^{-3}$
FPM1	4.950	–	–
FPM2	22.00	$6.366 \times 10^{-3}$	–

**Table 16 – Noise source parameters for Option 2 (i.e., Type II and Type III) SSU**

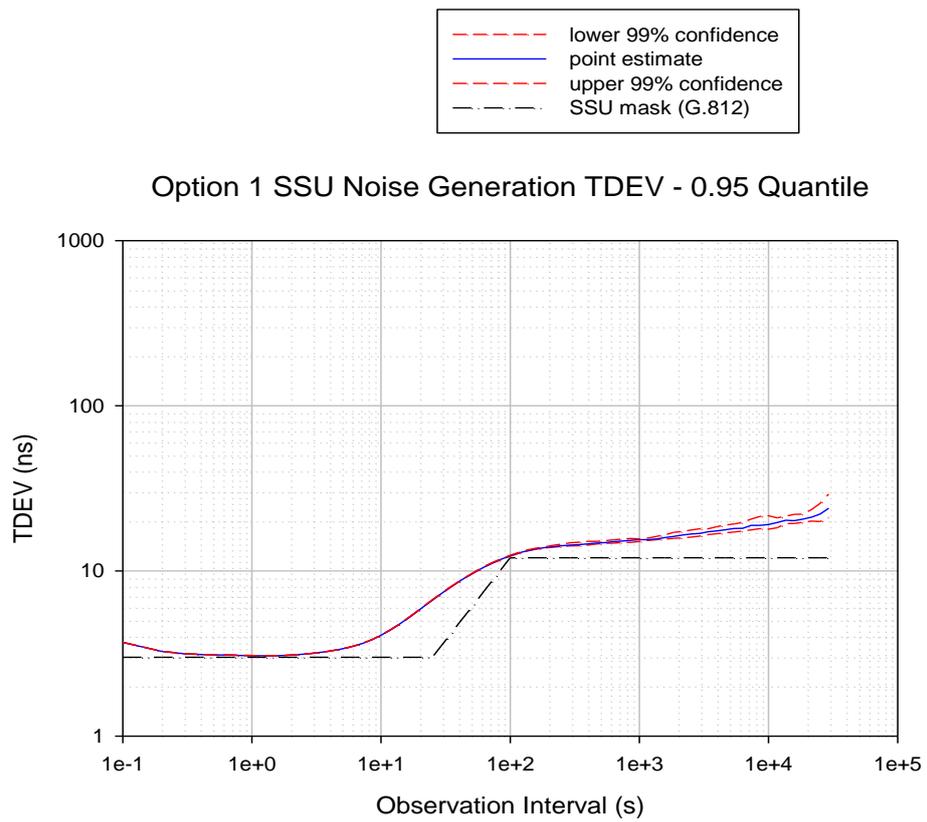
Noise source	Input white noise standard deviation (ns)	Low-pass filter bandwidth (Hz)	High-pass filter bandwidth (Hz)
WPM	3.162	–	–
WFM	1.750	–	$3.183 \times 10^{-3}$
FPM1	2.750	–	–
FPM2	16.00	$3.183 \times 10^{-3}$	–

MTIE and TDEV for each of the two models are given in Figure 15 to 18. As for the EEC/SEC results, 99% confidence intervals are given, based on 300 independent replications of each of the two simulation cases.

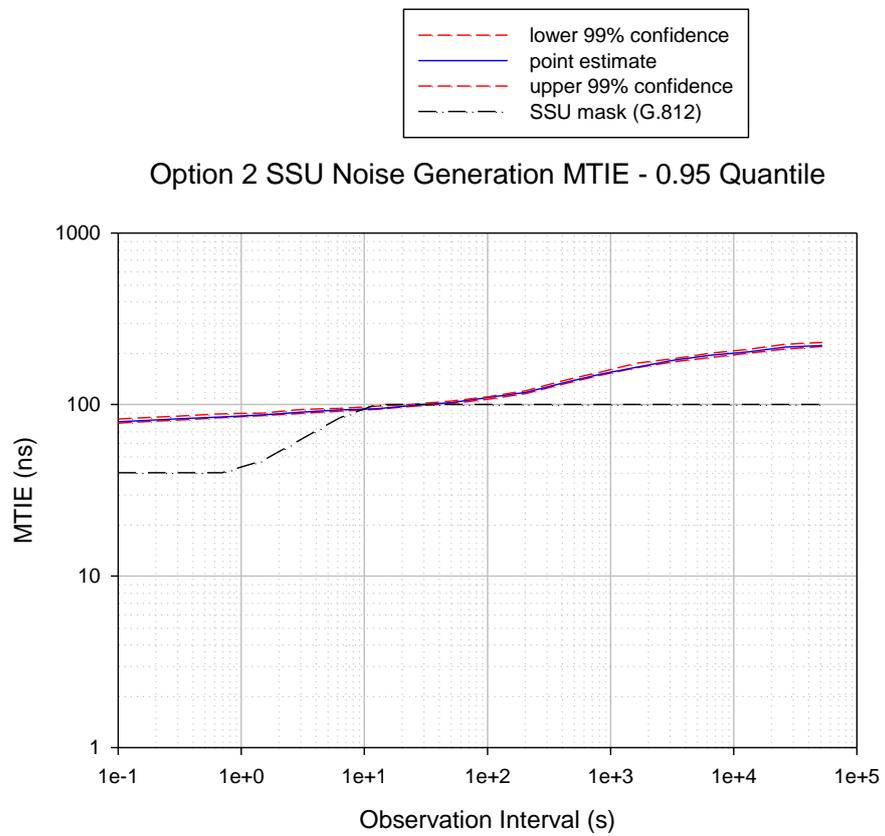
For Option 1, TDEV is approximately as low as possible without going below the mask, and MTIE exceeds the mask by approximately 25% at the point where it comes closest to the mask (around 400 s). For Option 2, both TDEV and MTIE are approximately as close as possible to the mask without going below the mask (though MTIE is very slightly below the mask between 10 and 20 s).



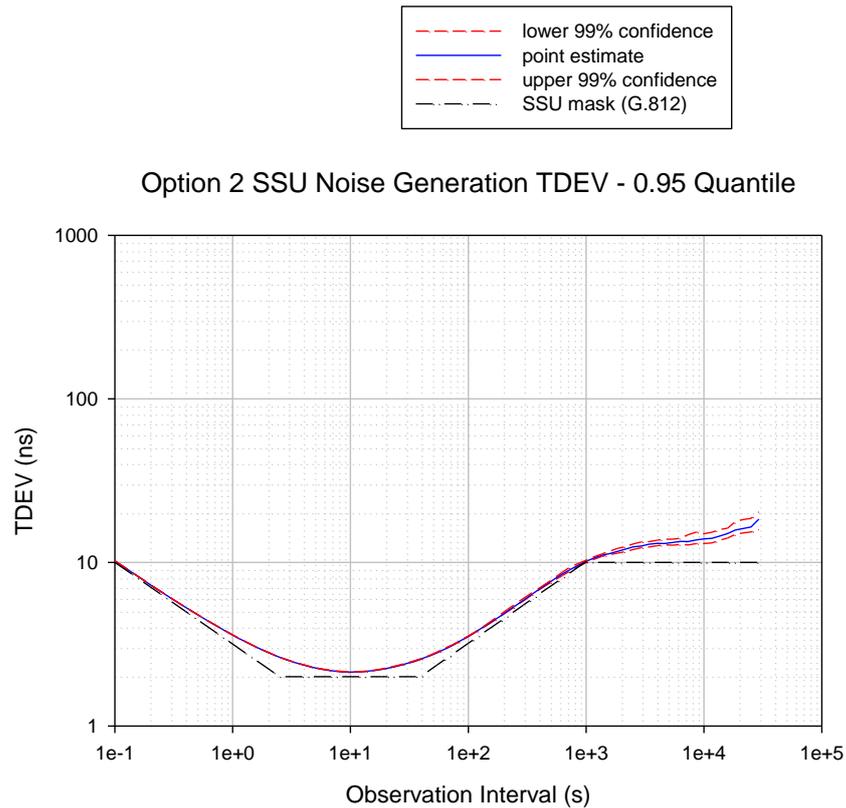
**Figure 15 – Option 1 (i.e., Type I) SSU wander generation MTIE**



**Figure 16 – Option 1 (i.e., Type I) SSU wander generation TDEV**



**Figure 17 – Option 2 (i.e., Type II and III) SSU wander generation MTIE**



**Figure 18 – Option 2 (i.e., Type II and III) SSU wander generation TDEV**

### 8.1.2.3 PRC wander generation models

The PRC input noise models were developed to match, as closely as possible, the respective ITU-T G.811 wander generation MTIE and TDEV masks. For the PRC, there is a single wander generation MTIE mask and a single wander generation TDEV mask in [ITU-T G.811] (i.e., [ITU-T G.811] does not contain separate masks for Option 1 and Option 2 networks). Actually, the primary reference source MTIE mask for Option 2 in [b-ANSI02] is somewhat more conservative at short observation intervals (i.e., 10 ns in [b-ANSI02] versus 25 ns in [ITU-T G.811] for observation intervals less than approximately 1 s). However, this difference is negligible because the PRC will be at the beginning of the ITU-T G.803 reference chain while the OTN islands will be near the end of the reference chain (i.e., after the PRC, 10 SSUs, and 58 SECs).

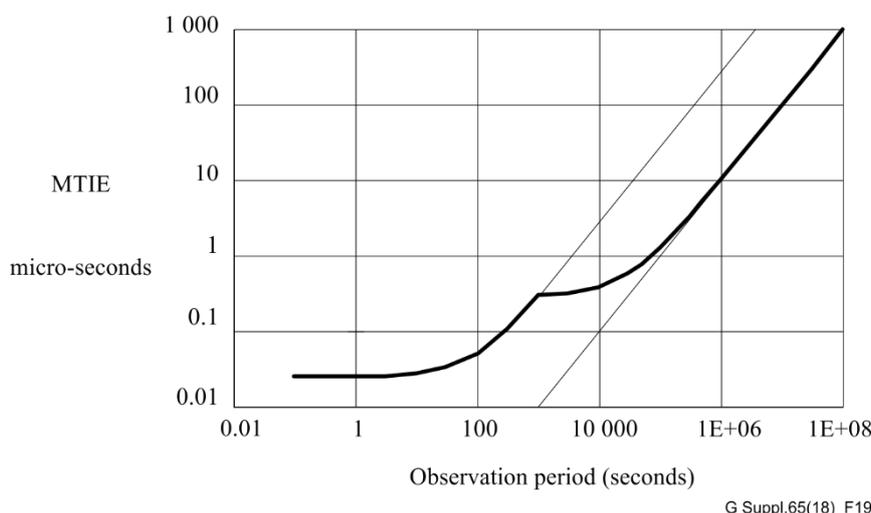
The MTIE mask is shown in Figure 1 of [ITU-T G.811] and the equations that precede the figure. The MTIE mask is reproduced in Figure 19 and Table 17 of this Supplement. The TDEV masks are shown in Figure 2 of [ITU-T G.811] and the equations that precede the figure. The TDEV mask is reproduced in Figure 20 and Table 18.

**Table 17 – PRC wander generation MTIE (see clause 6.1 of [ITU-T G.811])**

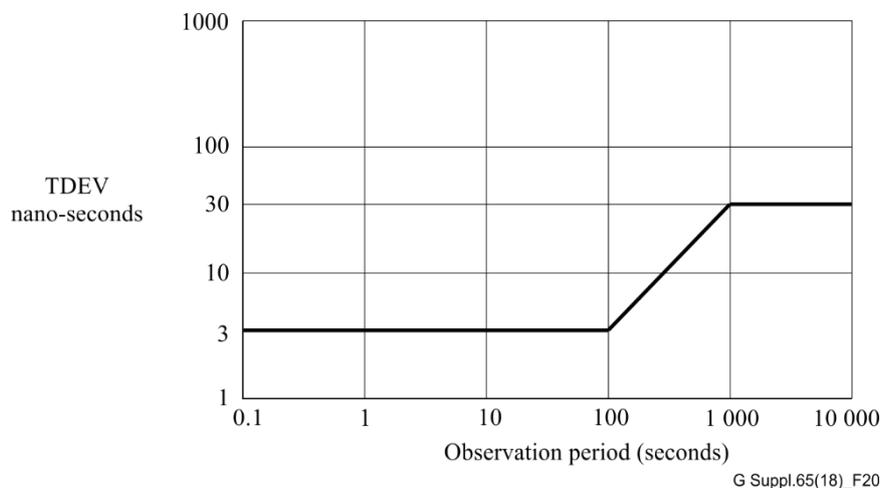
MTIE limit ( $\mu\text{s}$ )	Observation interval $\tau$ (s)
$0.275 \times 10^{-3} + 0.025$	$0.1 < \tau \leq 1000$
$10^{-5} \tau + 0.29$	$\tau > 1000$

**Table 18 – PRC wander generation TDEV (see clause 6.1 [ITU-T G.811])**

MTIE limit (ns)	Observation interval $\tau$ (s)
3	$0.1 < \tau \leq 100$
$0.03 \times \tau$	$100 < \tau \leq 1000$
30	$1000 < \tau < 10000$



**Figure 19 – PRC wander generation MTIE mask (see Figure 1 of [ITU-T G.811])**



**Figure 20 – PRC wander generation TDEV mask (Figure 2 of [ITU-T G.811])**

The PRC noise model is slightly different from the EEC/SEC and SSU noise models. First, the MTIE asymptote of  $10^{-11}$  for long observation intervals is included as an additive, deterministic phase ramp of 0.01 ns/s. This has no impact on TDEV (because its effect cancels when the second difference in the TDEV computation is taken). Second, the model consists of two FPM sources plus an FFM source. The latter is used to represent the portion of the TDEV mask between 100 s and 1000s (this portion of the TDEV mask has a slope of 1 relative to the log-log scale).

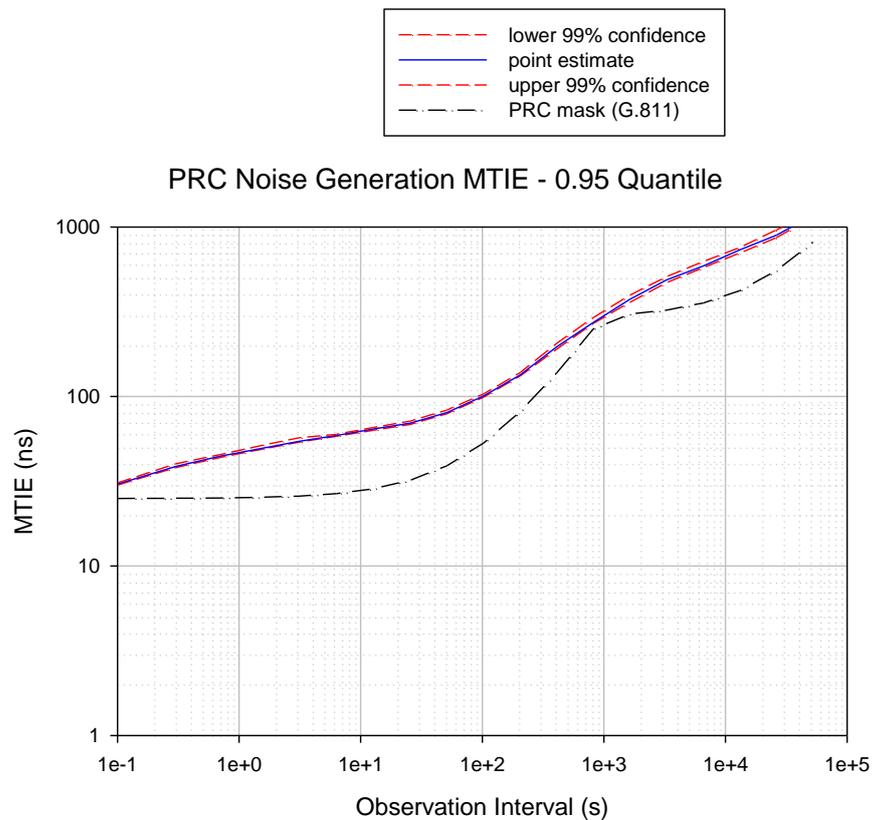
The standard deviations for the WPM sequence and the white noise sequences used to generate the WFM and two FPM sequences, for Option 1 and Option 2 PRC wander generation models, are given in Table 19. This table also gives the corner frequencies for the respective first-order high-pass and low-pass filters that the FFM and one of the FPM sequences are passed through. As with the EEC/SEC and SSU models, the sampling interval is 0.1 s.

**Table 19 – Noise source parameters for PRC**

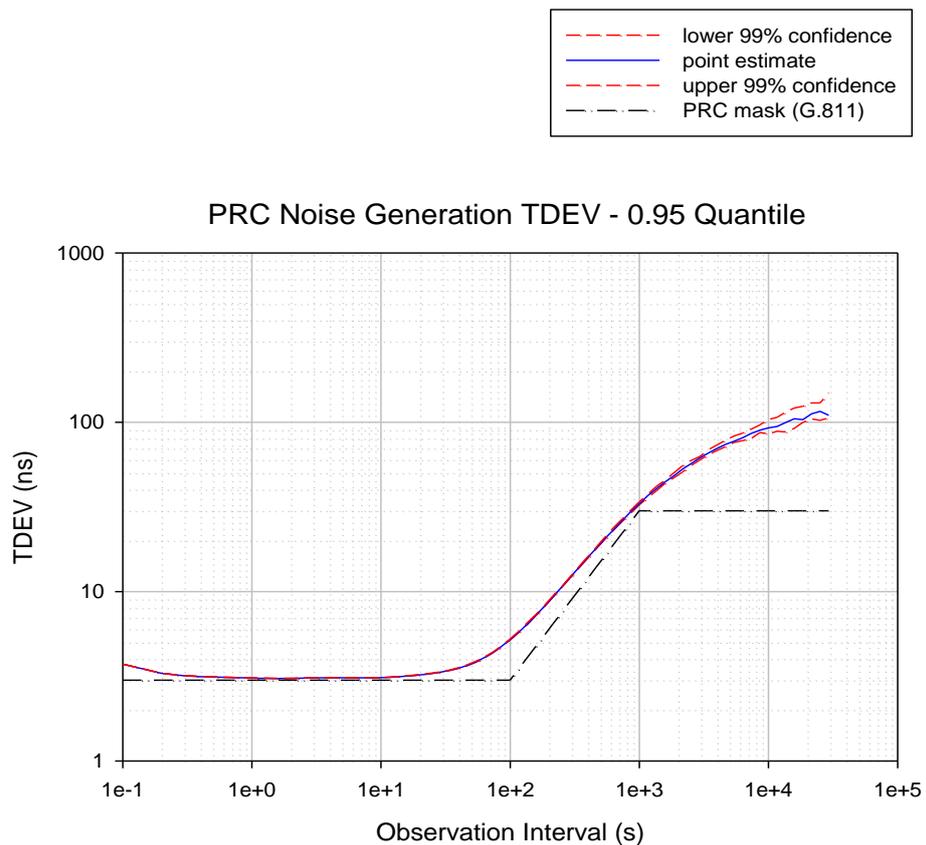
Noise source	Input white noise standard deviation (ns)	Low-pass filter bandwidth (Hz)	High-pass filter bandwidth (Hz)
FFM	0.5250	–	$1.273 \times 10^{-4}$
FPM1	5.0	–	–
FPM2	48.0	$3.183 \times 10^{-4}$	–

MTIE and TDEV are given in Figure 15 and Figure 18. As for the EEC/SEC and SSU results, 99% confidence intervals are given, based on 300 independent replications of each of the two simulation cases.

Both TDEV and MTIE are approximately as close as possible to the mask without going below the mask. See Figures 21 and 22.



**Figure 21 – PRC wander generation MTIE**



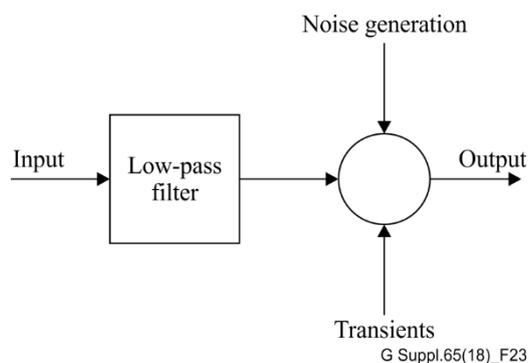
**Figure 22 – PRC wander generation TDEV**

### 8.1.3 Wander accumulation model and initial simulation results

#### 8.1.3.1 Simulation model

The simulation model used here is based on the model of [b-Garner01]. In [b-Garner01], a clock is modelled as shown in Figure 23 below (this figure is taken from [b-Garner01]). The main difference between the model of [b-Garner01] and the model used here is that the low-pass filter model in [b-Garner01] is first-order, while the low-pass filter model here is second-order with 20 dB/decade roll-off. The model used here is taken from Appendix VIII (clause VIII.2.2) of [ITU-T G.8251]. Therefore, the gain peaking of the EECs/SECs and SSUs is modelled here (while it was not modelled in [b-Garner01] and [b-Garner02]). In the model, damping ratio was computed from gain peaking using equation (8-14) of clause 8.2.3.

The clock model assumes that the input is filtered by a second-order, low-pass filter with 20 dB/decade roll-off, and then the phase noise from a noise generation model and the phase variation from a transient model are added. In the simulations here, transients are not considered, because the assumptions on the volume of phase transients used for the wander accumulation simulations of Annex A of [ITU-T G.823] are very conservative for modern SSUs (those assumptions were that the average rate of transients between any two SSUs is 1 transient per 25 days, and that the transient magnitude is 1  $\mu$ s with random polarity).



**Figure 23 – Clock model**

The noise generation is the appropriate noise generation model for the respective clock as described in clause 8.1.2. Clause 8.1.2 describes four noise generation models for the EEC/SEC, two noise generation models for the SSU, and one noise generation model for the PRC. For the SEC, two models were developed for each Option. The models are described below:

- a) Option 1, Model 1: meets (i.e., matches) variable-temperature MTIE mask, exceeds TDEV mask,
- b) Option 1, Model 2: meets (i.e., does not fall below) TDEV mask as closely as possible, falls below variable-temperature MTIE mask,
- c) Option 2, Model 1: meets (i.e., matches) TDEV mask, exceeds MTIE mask, and
- d) Option 2, Model 2: meets (i.e., does not fall below) MTIE mask as closely as possible, falls below TDEV mask.

The two SSU models are for Option 1 and Option 2, respectively. The single PRC model is used for both Option 1 and Option 2.

The chain of clocks is modelled by setting the input of each clock in the chain to the output of the previous clock. The first clock in the chain, i.e., the PRC, has no input from upstream; its output consist of only the PRC noise generation.

### 8.1.3.2 Simulation cases and inputs

Four simulation cases were run, corresponding to the four EEC/SEC noise generation models described above. The assumptions common to all four simulation cases are summarized in Table 20.

**Table 20 – Assumptions common to all simulation cases**

Parameter/Assumption	Value
PRC noise generation model	Model described in clause 8.1.2.3
SSU bandwidth (Hz)	0.001
SSU gain peaking	0.2 dB
Simulation time step (s)	0.1
Simulation time (s)	100,000

The SSU gain peaking is taken from clause 10 of [ITU-T G.812]. The SSU bandwidth is the value for Type I and II clocks in clause 10 of [ITU-T G.812]. The bandwidth for Type I clocks is actually 0.003 Hz; the use of 0.001 Hz here is not expected to have a large impact because both 0.001 Hz and 0.003 Hz are small compared to Option 1 EEC/SEC bandwidth assumption of 10 Hz (see below) and the final SSU is followed by 18 EECs/SECs.

The assumptions specific to each simulation case are given in Table 21.

**Table 21 – Assumptions specific to each simulation case**

Simulation case	SSU noise generation model	EEC/SEC noise generation model	EEC/SEC bandwidth (Hz)	EEC/SEC gain peaking (dB)
1	Option 1 (clause 8.1.2.2)	Option 1, Model 1 (clause 8.1.2.1)	10	0.2
2	Option 1 (clause 8.1.2.2)	Option 1, Model 2 (clause 8.1.2.1)	10	0.2
3	Option 2 (clause 8.1.2.2)	Option 2, Model 1 (clause 8.1.2.1)	0.1	0.2
4	Option 2 (clause 8.1.2.2)	Option 2, Model 2 (clause 8.1.2.1)	0.1	0.2

### 8.1.3.3 Simulation results

Simulation results for cases 1 to 4 are given in Figures 24 to 39. For each case, MTIE and TDEV are given. First for the PRC and selected SSUs and EECs/SECs in the reference chain, and then (in a separate figure) for SSU 10, i.e., the final SSU, and EECs/SECs 41 to 58, i.e., the 18 EECs/SECs that follow the final SSU. In each figure, the upper and lower extent of a 99% confidence interval for the 0.95 quantile, for each clock, are indicated by heavy dashed lines. The solid line between the dashed lines represents the point estimate of the 0.95 quantile, i.e., the midpoint of the confidence interval.

In each figure, the respective network limit MTIE or TDEV mask is shown. These masks are taken from [ITU-T G.823] for Option 1 and [ITU-T G.824] and [b-ANSI02] for Option 2.

Figure 25 gives MTIE results for case 1 (Option 1, Model 1), for the final SSU in the reference chain and each of the 18 EECs/SECs that follows this SSU. The results show steady increase in MTIE at each successive clock. The results for the final 6 EECs/SECs (i.e., EECs/SECs 53 to 58) slightly exceed the MTIE mask for observation intervals around 1.5 s. Note, however, that the EEC/SEC noise generation for Option 1, Model 1 also exceeds the MTIE mask except for observation intervals around 0.1 s and  $10^5$  s (i.e., the longest and shortest observation intervals). See Figure 5. As indicated in clause 8.1.2.1, it was not possible to match the MTIE mask exactly using power-law noise processes, as the model was constructed to attempt to be as low as possible without going below the mask.

Figure 24 gives the MTIE results for case 1, for selected SSUs and EECs/SECs. The results for the PRC and SSUs 1, 4 and 8 are clustered together for observation intervals within the SSU bandwidth. This is because each SSU filters the input noise, but then adds its own noise generation. In the case of longer observation intervals there is noise accumulation. Wander accumulation for shorter observation intervals is seen for EECs/SECs 1, 2 and 20. Since the EEC/SEC bandwidth is 10 Hz, each successive EEC/SEC does not filter the input noise as effectively at observation intervals of 0.1 s or longer, and also adds its own noise generation. However, the wander generation at shorter observation intervals is reduced when an SSU is reached, after a chain of EECs/SECs, due to the narrower SSU bandwidth.

Figures 26 and 27 give TDEV results for case 1. The qualitative behaviour is the same as for the MTIE results. In Figure 27, TDEV steadily increases at each EEC/SEC after the last SSU. In Figure 5, TDEV at the output of the PRC and each SSU is approximately the same at shorter observation intervals. TDEV steadily increases at all observation intervals considered (0.1 s and longer) with each consecutive EEC/SEC in a chain, and is then reduced at shorter observation intervals when an SSU is reached due to the narrower SSU bandwidth. The network limit TDEV mask is exceeded starting with SEC 45. However, Option 1, Model 1 TDEV exceeds the noise generation TDEV mask at all observation intervals, so this result is not unexpected. Since Option 1, Model 1 was constructed to meet the MTIE mask but not the TDEV mask, it is mainly the MTIE results that are of interest for this case.

Finally, note that MTIE and TDEV at the output of the PRC and SSUs are below the network limits at all observation intervals. It is only after some number of SECs that the network limits are exceeded, and this is because the wander generation model exceeds the masks (for all observation intervals for TDEV and for all but the longest and shortest observation intervals for MTIE).

Figures 28 and 29 give MTIE results for case 2 (Option1, Model 2). These results are qualitatively similar to those for case 1, except that the results are now below the MTIE mask for all observation intervals. This is not unexpected, as the wander generation level for Option 1, Model 1 SEC is lower than that of Option 1, Model 2. Figures 30 and 31 give the TDEV results for case 2 (Option1, Model 2). The TDEV results are also qualitatively similar to case 1 results, except that the TDEV mask is exceeded starting with EEC/SEC 49. The mask is first exceeded, for EEC/SEC 49, for observation intervals around 16 s (i.e., where the slope of the TDEV mask changes). Examination of Option 1, Model 2 wander generation model TDEV in Figure 8 indicates that the model is slightly above the mask. This is due to the superposition of power-law noise processes that cannot match an abrupt change in the slope of the mask. However, the flat part of the mask for observation intervals in range 1 to 2 s is exceeded starting with EEC/SEC 52. This is not expected, because Option 1, Model 1 wander generation model matches the wander generation TDEV mask quite well for observation intervals in this range. This indicates that the wander generation, EEC/SEC and SSU bandwidth and gain peaking requirement, network limit, and ITU\_T G.803 reference model are not fully consistent. In any case, it is not clear that consistency was previously checked. Note that Annex A of [ITU-T G.823] does not actually consider the consistency of the wander generation and network limit masks. It uses the wander generation models and ITU-T G.803 reference chain to generate noise processes that time SDH pointer processors, and shows that wander accumulation for plesiochronous digital hierarchy (PDH) clients of SDH is acceptable with such input noise (including the assumptions regarding phase transients).

Figures 32 and 33 give MTIE results for case 3 (Option 2, Model 1). The results in Figure 33 show a monotonic increase in MTIE at longer observation intervals with each successive EEC/SEC after the final SSU. Note that Option 2 EEC/SEC bandwidth is 0.1 Hz, which is much narrower to the minimum observation interval of 0.1 s when compared to the 10 Hz Option 1 EEC/SEC bandwidth. As a result, there is minimal increase in MTIE at shorter observation intervals with successive EECs/SECs (as well as with successive SSUs). This result is shown in Figure 32, where the only increase in MTIE at shorter observation intervals is in going from the PRC to the first SSU. The network limit MTIE mask is exceeded for shorter observation intervals by the output of all the SSUs and EECs/SECs. This is because the EEC/SEC wander generation MTIE mask is exceeded by a large margin at all observation intervals (see Figure 9) and the SSU wander generation MTIE mask is exceeded by a large margin for observation intervals of less than 1 s.

Figures 34 and 35 give TDEV results for Option 2, Model 1. TDEV meets the network limit TDEV mask for all clocks at all observation intervals, except that TDEV at 0.1 s is approximately equal to the mask value of 10 ns because the wander generation TDEV at 0.1 s is 10 ns (see Figure 10). This indicates that Option 2 wander generation TDEV mask, network limit TDEV mask, and EEC/SEC and SSU bandwidth and gain peaking are consistent with the ITU-T G.803 reference model.<sup>3</sup>

Figures 36 and 37 give MTIE results for case 4 (Option 2, Model 2), and Figures 38 and 39 give TDEV results for case 4. The results are slightly below the results for Option2, Model 1, but are qualitatively similar. The MTIE network limit is still exceeded for shorter observation intervals, though not by as much as Option 1, Model 1, and it is not exceeded at the longest observation intervals by any EECs/SECs (it was exceeded by EECs/SECs near the end of the reference chain for Option 2, Model 1). The TDEV mask is met by the output of all EECs/SECs and SSUs, and by a larger margin

---

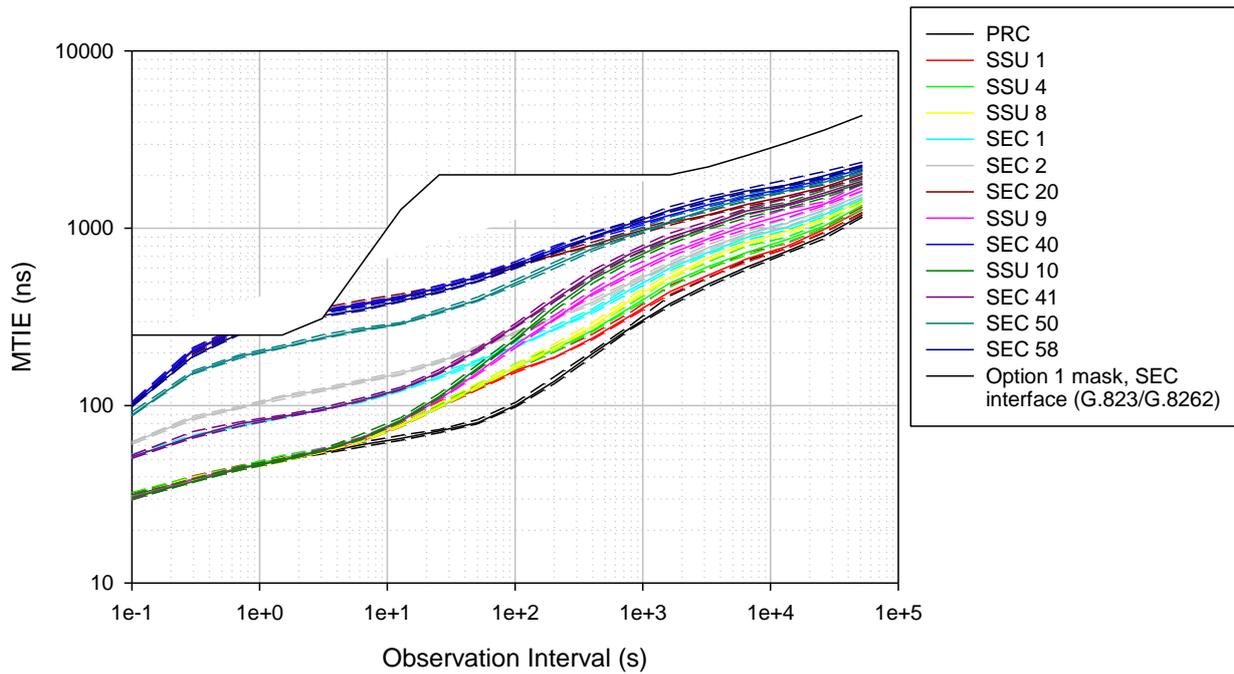
<sup>3</sup> Strictly speaking, the full reference model would have 20 EECs/SECs after the final SSU, while only 18 EECs/SECs are present here. However the qualitative behavior of the results indicates that the TDEV network limit would likely not be exceeded if 2 additional EECs/SECs were added.

than for Option 2, Model 1. Examination of the wander generation model MTIE for Option 2, Model, in Figure 11, indicates that MTIE is close to the wander generation mask for observation intervals around 10 s, but exceeds the mask for longer and shorter intervals by a factor of as much as 2.5. This is the reason why the MTIE mask is exceeded. If the mask were met more closely, at longer and shorter intervals, it is likely the network limit MTIE mask would not be exceeded.

The wander phase history at the output of each of the 69 clocks (1 PRC, 10 SSUs, and 58 EECs/SECs) was saved for each of the 300 replications of each case. Initially, it was retained for use as an input to the OTN client wander accumulation simulations. However, it was later used in the simulations of time transport using PTP with frequency transport via SyncE. Note that not all the data was needed, e.g., the data for EECs and SSUs prior to the final SSU was not needed for the subsequent OTN or PTP simulations. The full set of data for all four simulation cases occupies approximately 2.2 TB of storage.

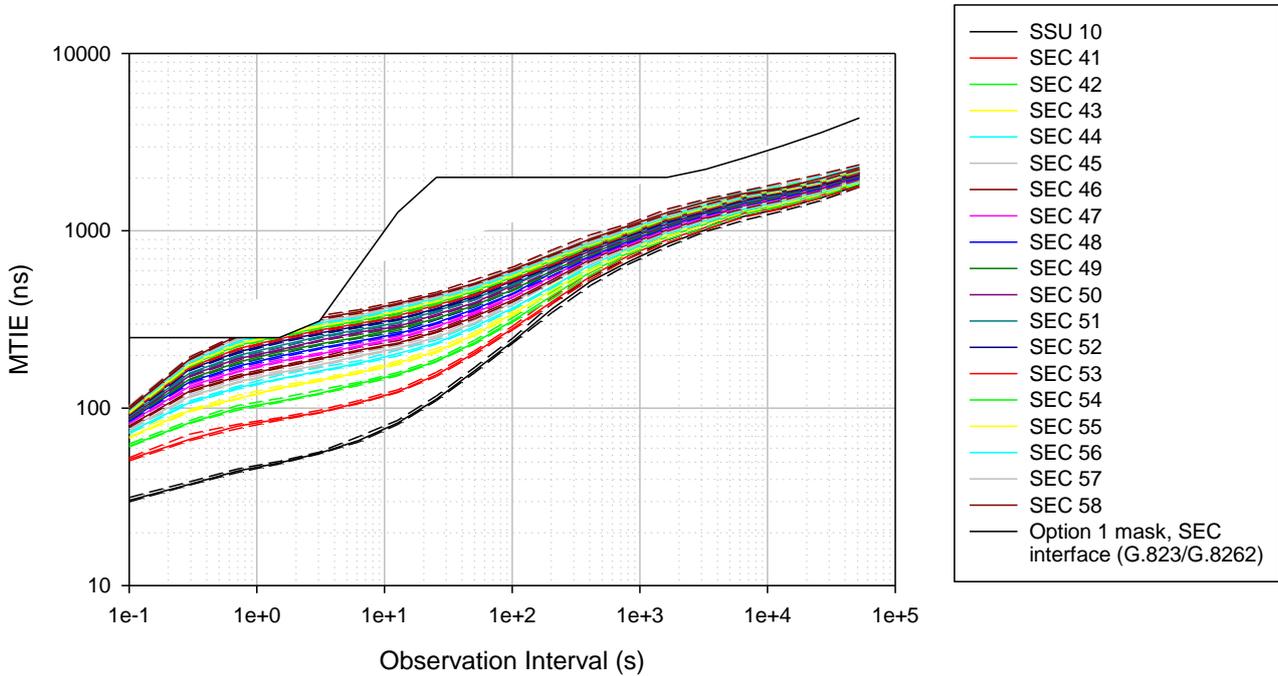
In summary, Option 2 TDEV network limit is met if the Option 2 TDEV wander generation is met, i.e., the Option 2 wander generation TDEV mask, network limit TDEV mask, and EEC/SEC and SSU bandwidth and gain peaking are consistent with ITU-T G.803 reference model. Option 1 and Option 2 MTIE network limits were exceeded for some observation intervals, however, the wander generation MTIE masks were exceeded for these observation intervals. It is likely that the network limits would not be exceeded if the wander generation MTIE for each respective model was closer to the mask (i.e., did not exceed it). Finally, Option 1 TDEV network limit is exceeded for observation intervals in the 1 to 2 s range, for the outputs of EECs/SECs 52 to 58 (i.e., the 12<sup>th</sup> through 18<sup>th</sup> EECs/SECs after the last SSU). This indicates that Option 1 wander generation, EEC/SEC and SSU bandwidth and gain peaking requirement, and network limit, are not fully consistent with ITU-T G.803 reference model. However, note that Annex A of [ITU-T G.823] does not actually consider the consistency of the wander generation and network limit masks. It uses the wander generation models and ITU-T G.803 reference chain to generate noise processes that time SDH pointer processors, and shows that wander accumulation for PDH clients of SDH is acceptable with such input noise (including the assumptions regarding phase transients). Nonetheless, it was this exceedance of Option 1 network limit TDEV mask that led to the use of ETSI model ([b-ETSI01] and [b-ETSI02]), described in clause 8.1.4.

Option 1, 10 Hz SEC/EEC BW  
 SEC noise models meet MTIE mask, exceed TDEV mask



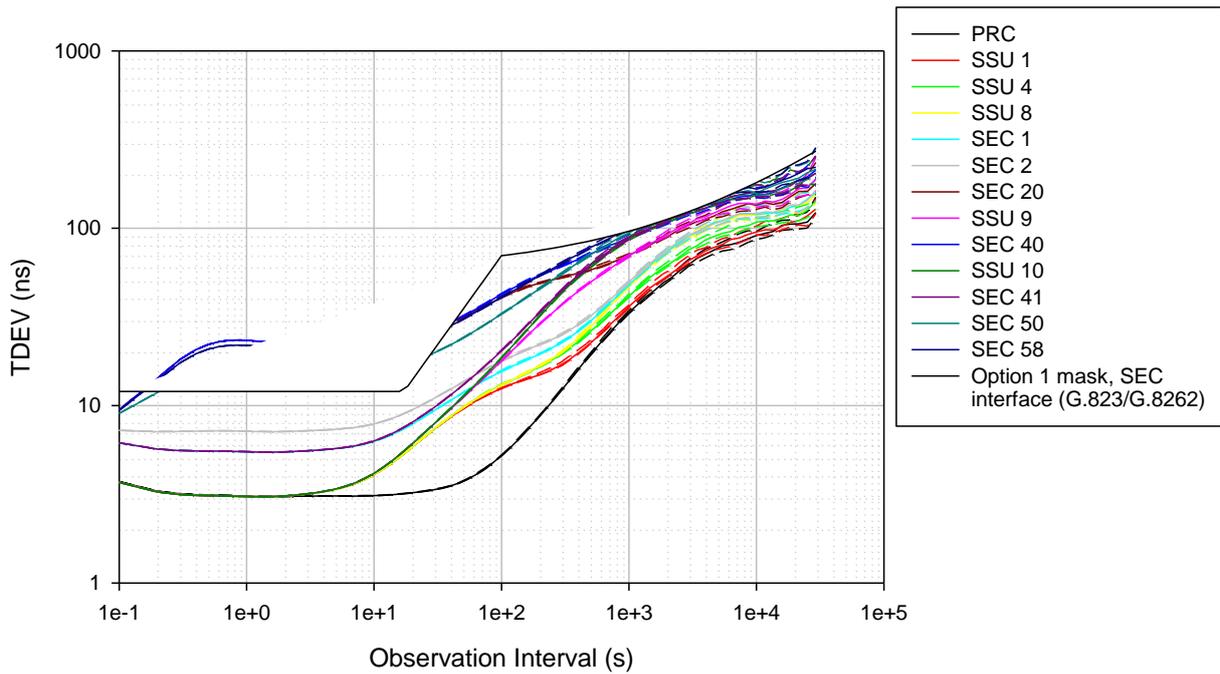
**Figure 24 – Option 1, model 1 MTIE for PRC and selected SECs and SSUs**

Option 1, 10 Hz SEC/EEC BW  
 SEC noise models meet MTIE mask, exceed TDEV mask



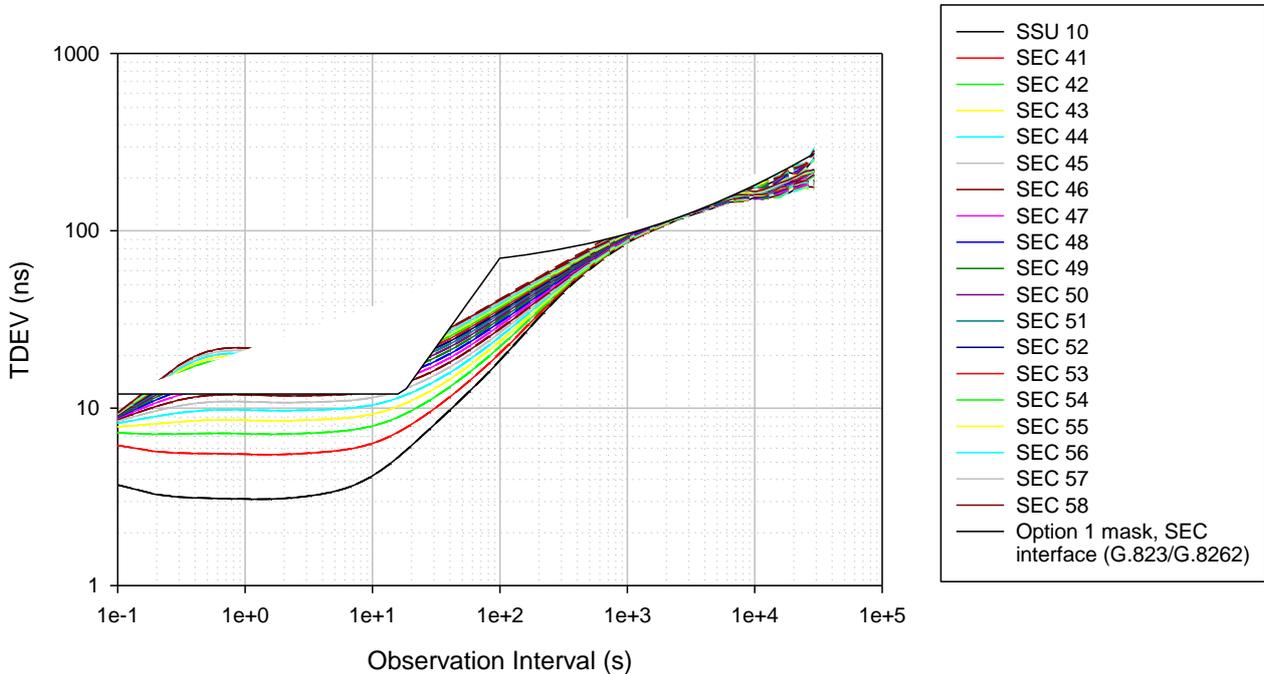
**Figure 25 – Option 1, model 1 MTIE for SSU 10 (final SSU) and EECs/SECs 41 – 58 (EECs/SECs following final SSU)**

Option 1, 10 Hz SEC/EEC BW  
 SEC noise models meet MTIE mask, exceed TDEV mask



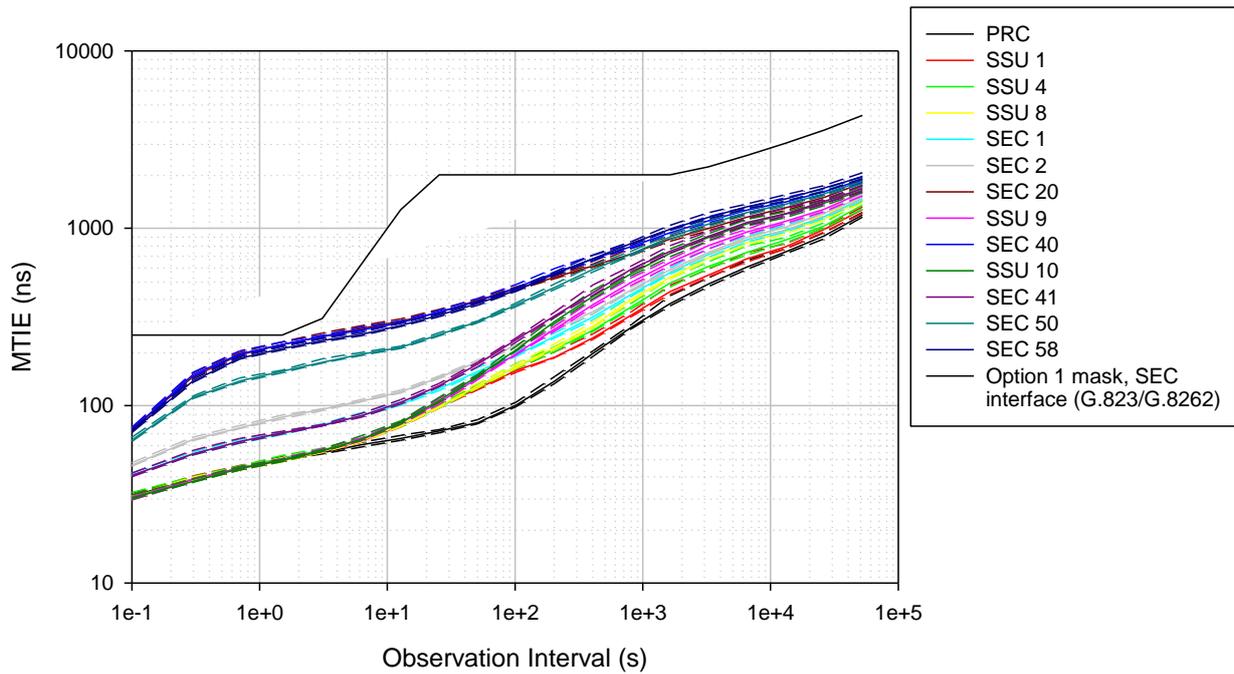
**Figure 26 – Option 1, model 1 TDEV for PRC and selected SECs and SSUs**

Option 1, 10 Hz SEC/EEC BW  
 SEC noise models meet MTIE mask, exceed TDEV mask



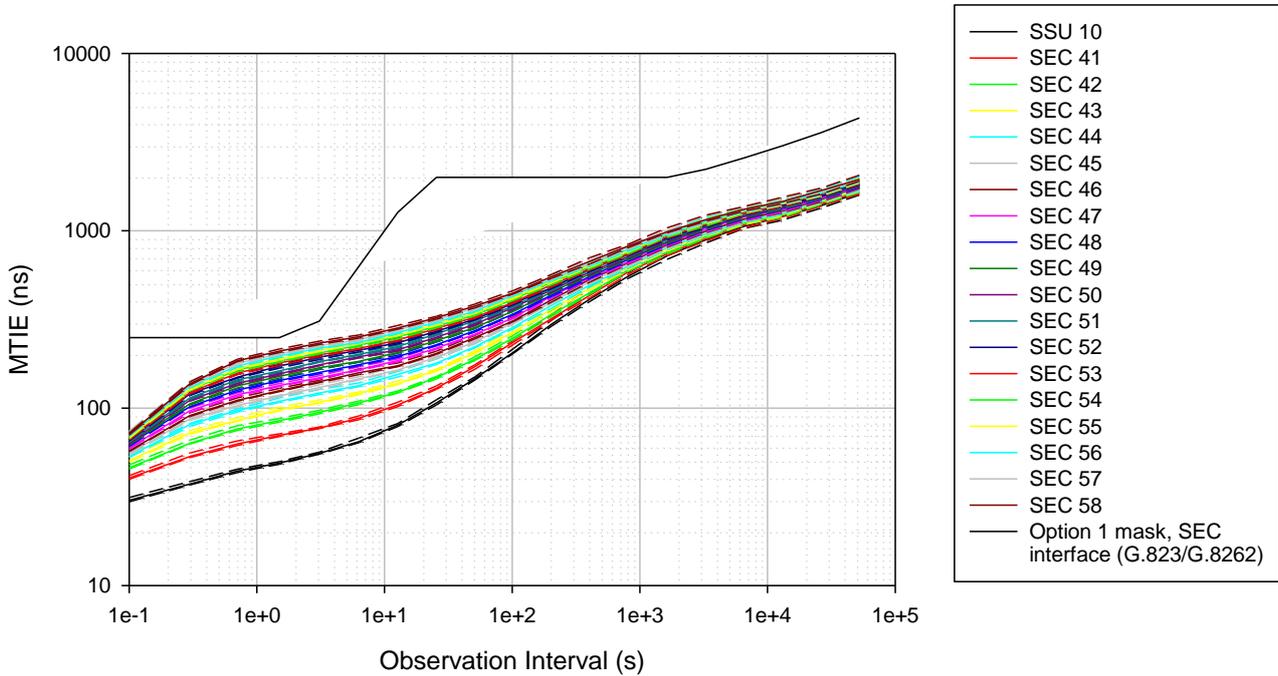
**Figure 27 – Option 1, model 1 TDEV for SSU 10 (final SSU) and EECs/SECs 41 – 58 (EECs/SECs following final SSU)**

Option 1, 10 Hz SEC/EEC BW  
 SEC noise models meet TDEV mask, below MTIE mask



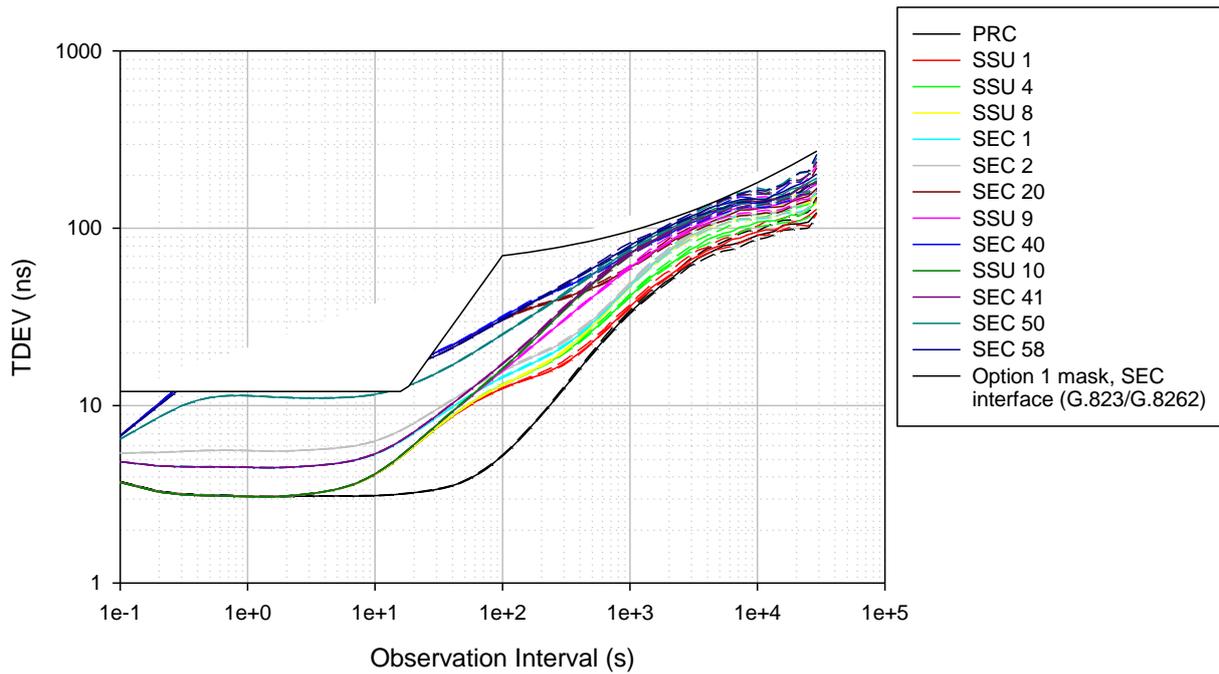
**Figure 28 – Option 1, model 2 MTIE for PRC and selected SECs and SSUs**

Option 1, 10 Hz SEC/EEC BW  
 SEC noise models meet TDEV mask, below MTIE mask



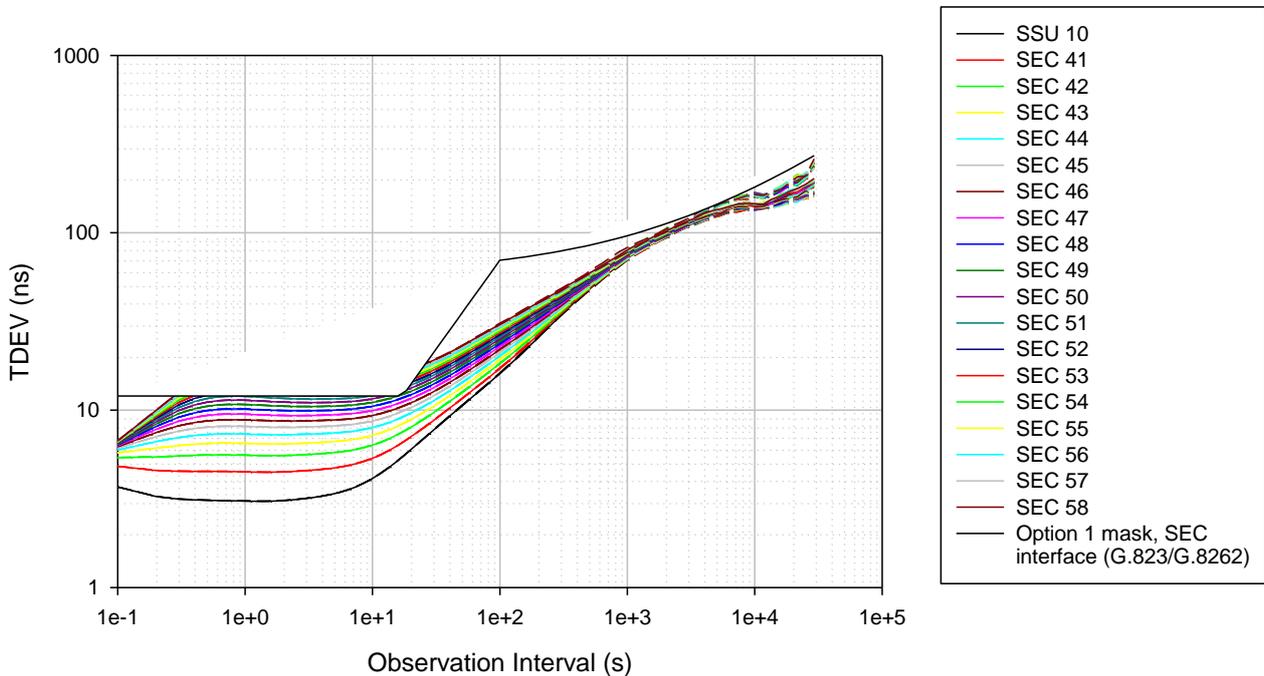
**Figure 29 – Option 1, model 2 MTIE for SSU 10 (final SSU) and EECs/SECs 41 – 58 (EECs/SECs following final SSU)**

Option 1, 10 Hz SEC/EEC BW  
 SEC noise models meet TDEV mask, below MTIE mask



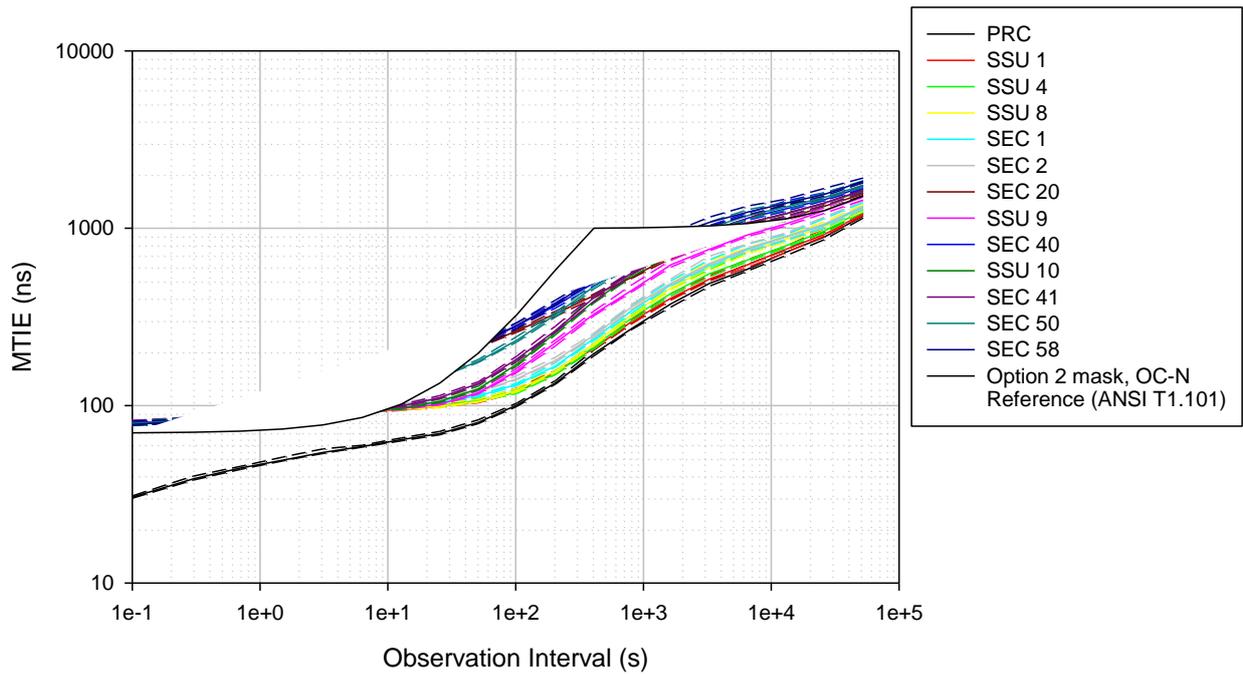
**Figure 30 – Option 1, model 2 TDEV for PRC and selected SECs and SSUs**

Option 1, 10 Hz SEC/EEC BW  
 SEC noise models meet TDEV mask, below MTIE mask



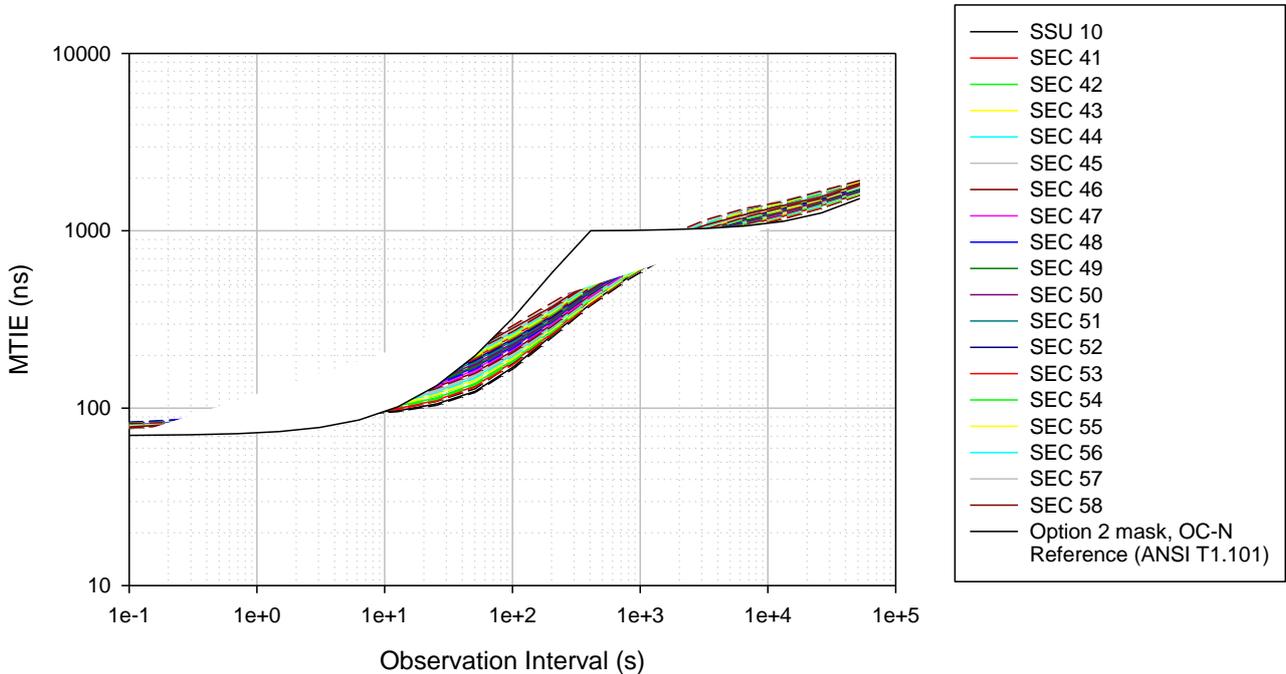
**Figure 31 – Option 1, model 2 TDEV for SSU 10 (final SSU) and EECs/SECs 41 – 58 (EECs/SECs following final SSU)**

Option 2, 0.1 Hz SEC/EEC BW  
 SEC noise models meet TDEV mask, exceed MTIE mask



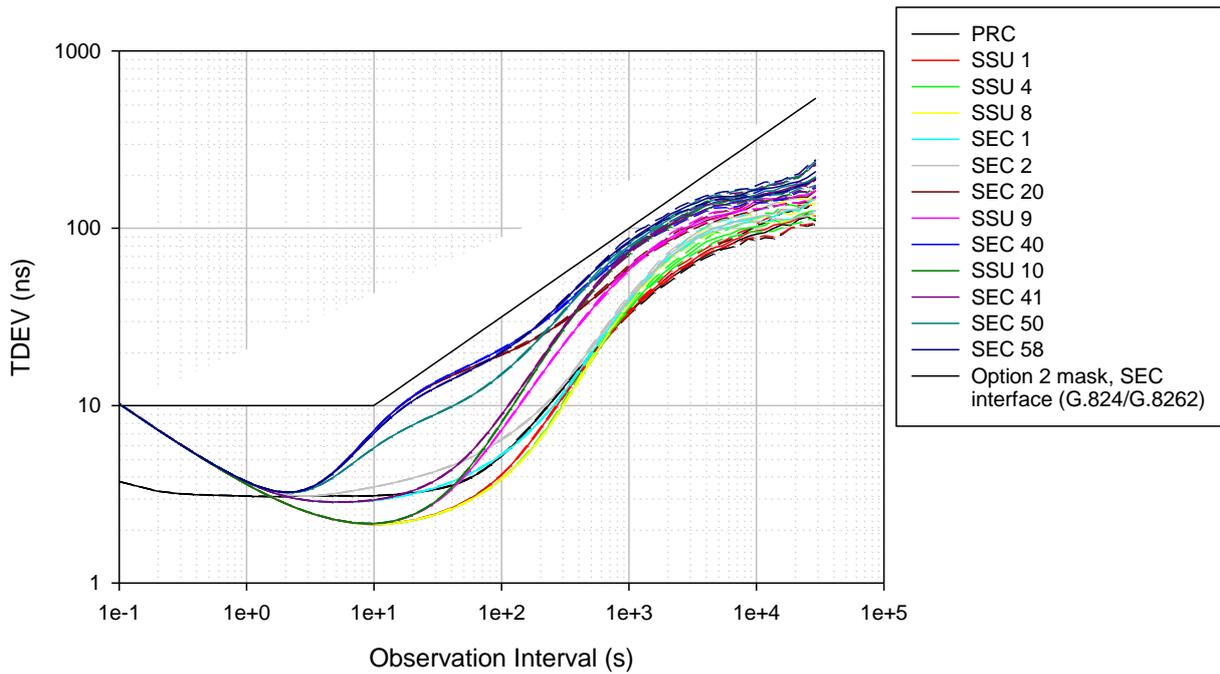
**Figure 32 – Option 2, model 1 MTIE for PRC and selected SECs and SSUs**

Option 2, 0.1 Hz SEC/EEC BW  
 SEC noise models meet TDEV mask, exceed MTIE mask



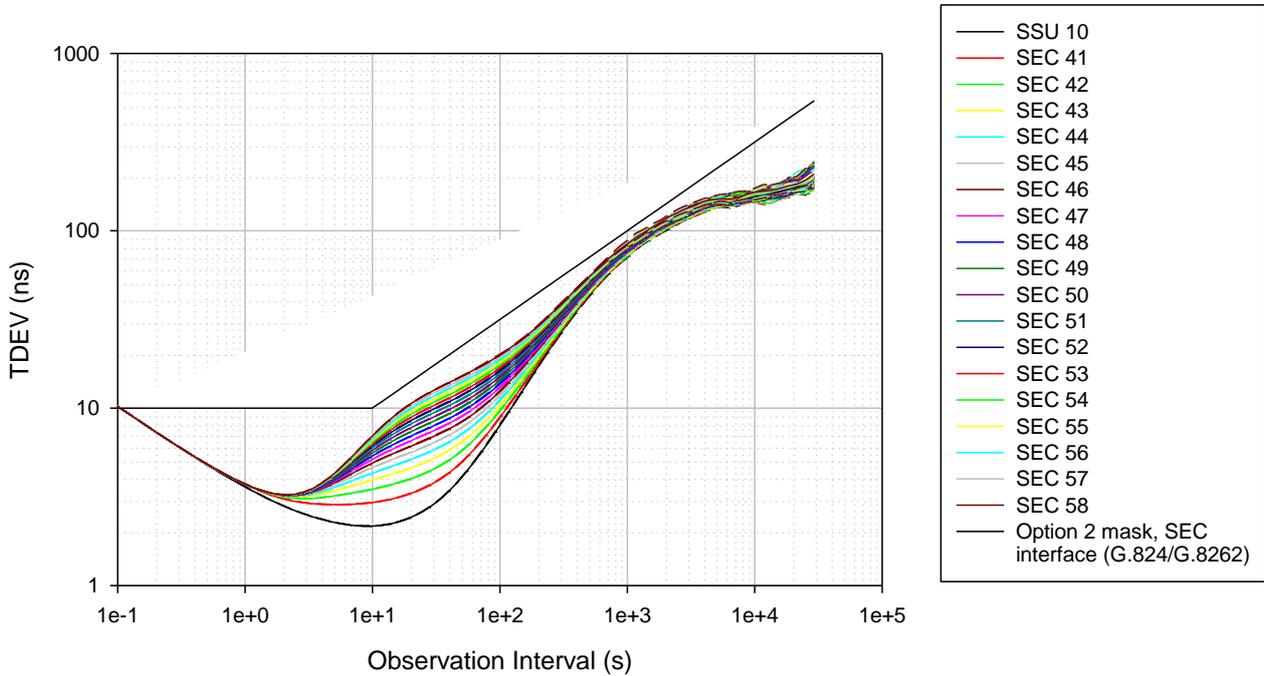
**Figure 33 – Option 2, model 1 MTIE for SSU 10 (final SSU) and EECs/SECs 41 – 58 (EECs/SECs following final SSU)**

Option 2, 0.1 Hz SEC/EEC BW  
 SEC noise models meet TDEV mask, exceed MTIE mask



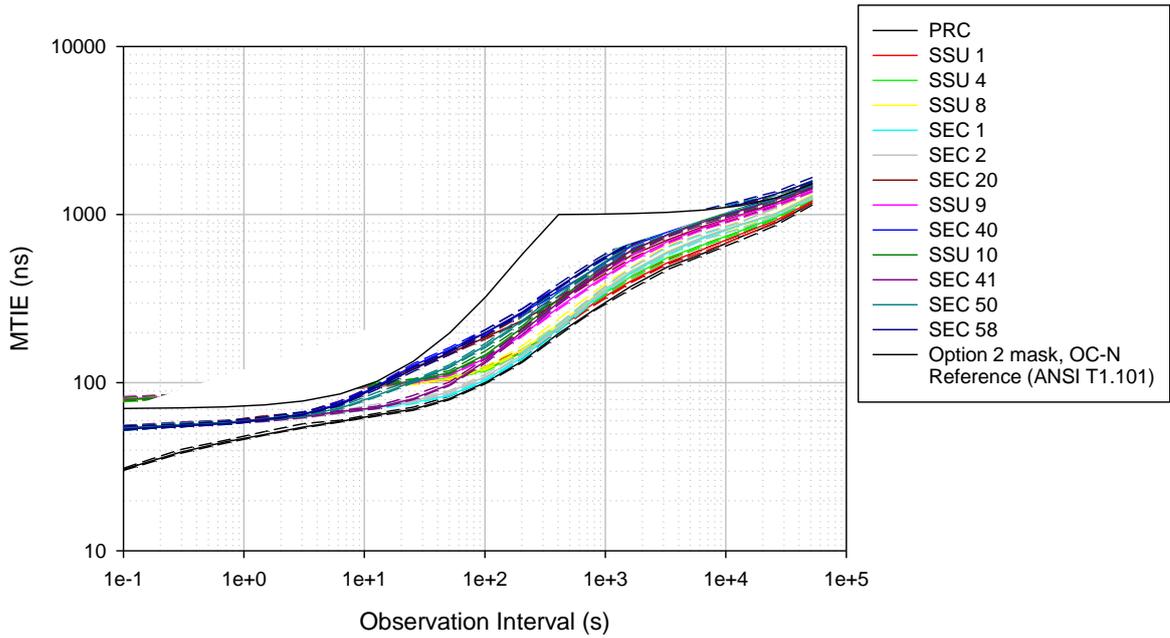
**Figure 34 – Option 2, model 1 TDEV for PRC and selected SECs and SSUs**

Option 2, 0.1 Hz SEC/EEC BW  
 SEC noise models meet TDEV mask, exceed MTIE mask



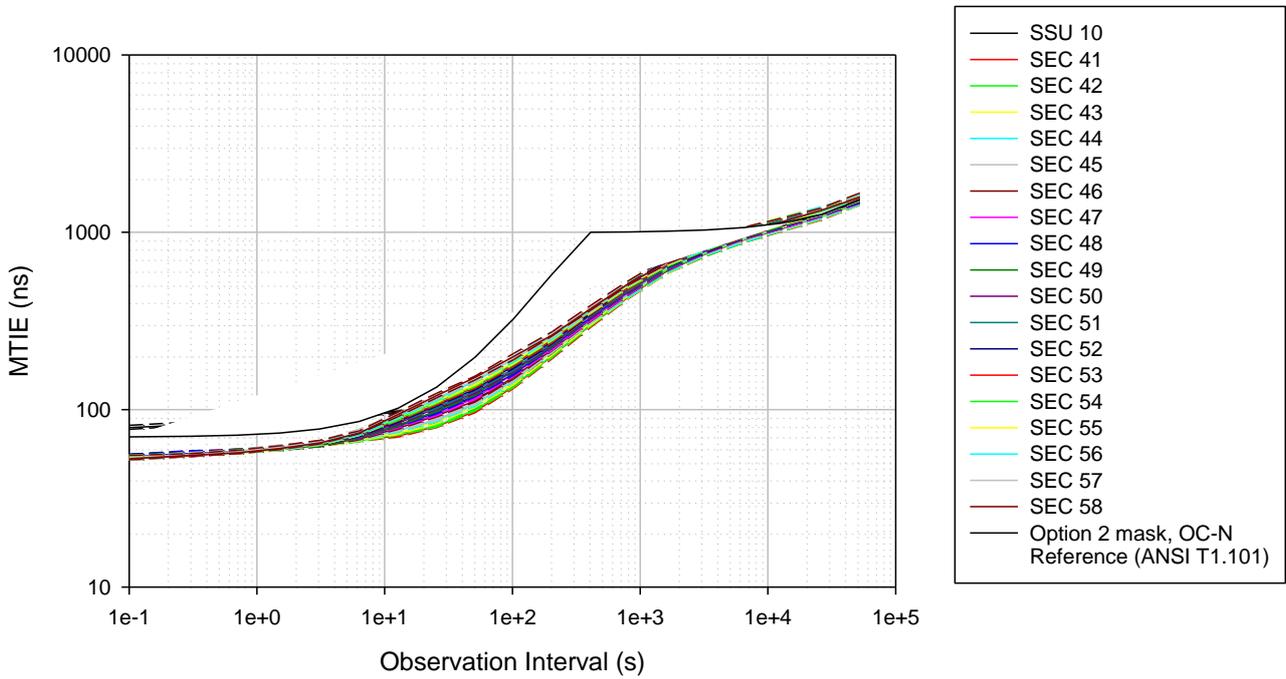
**Figure 35 – Option 2, model 1 TDEV for SSU 10 (final SSU) and EECs/SECs 41 – 58 (EECs/SECs following final SSU)**

Option 2, 0.1 Hz SEC/EEC BW  
 SEC noise models meet MTIE mask, below TDEV mask



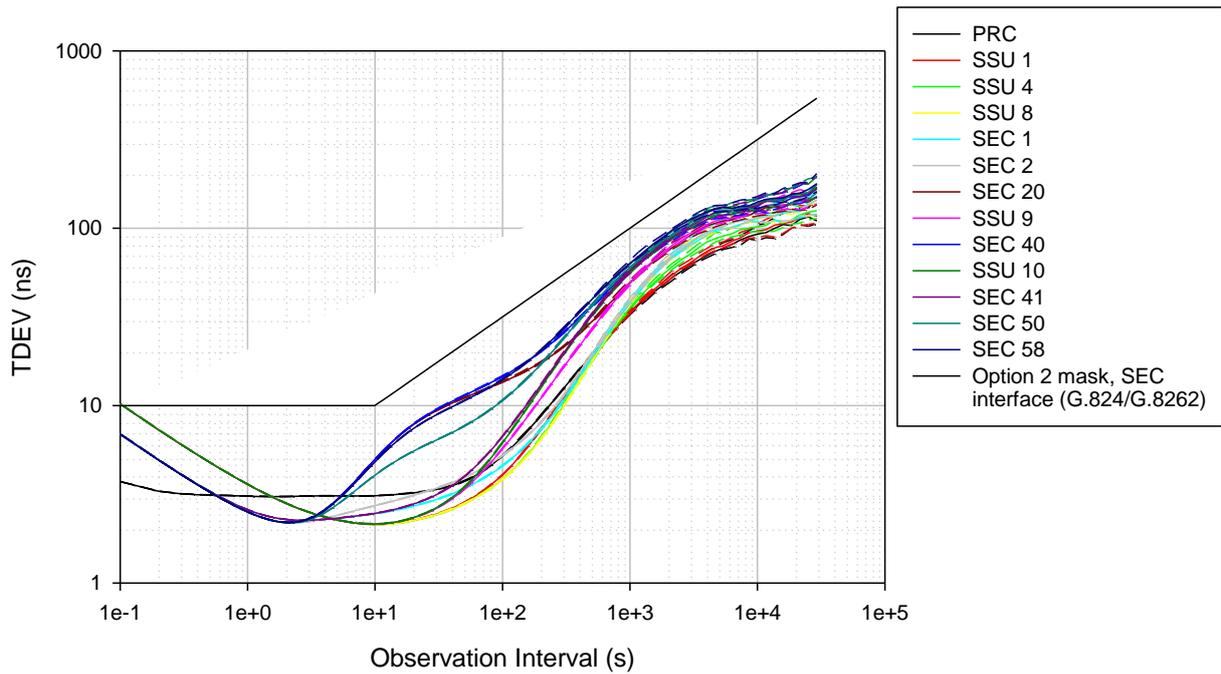
**Figure 36 – Option 2, model 2 MTIE for PRC and selected SECs and SSUs**

Option 2, 0.1 Hz SEC/EEC BW  
 SEC noise models meet MTIE mask, below TDEV mask



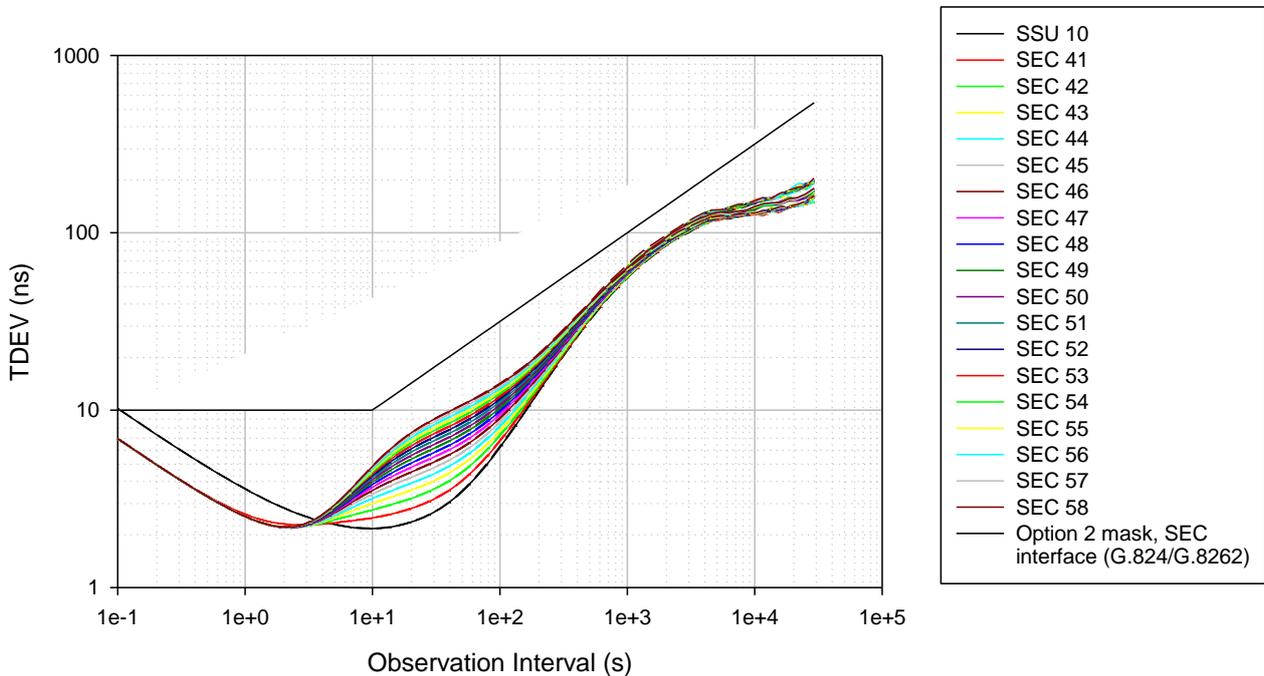
**Figure 37 – Option 2, model 2 MTIE for SSU 10 (final SSU) and EECs/SECs 41 – 58 (EECs/SECs following final SSU)**

Option 2, 0.1 Hz SEC/EEC BW  
 SEC noise models meet MTIE mask, below TDEV mask



**Figure 38 – Option 2, model 2 TDEV for PRC and selected SECs and SSUs**

Option 2, 0.1 Hz SEC/EEC BW  
 SEC noise models meet MTIE mask, below TDEV mask



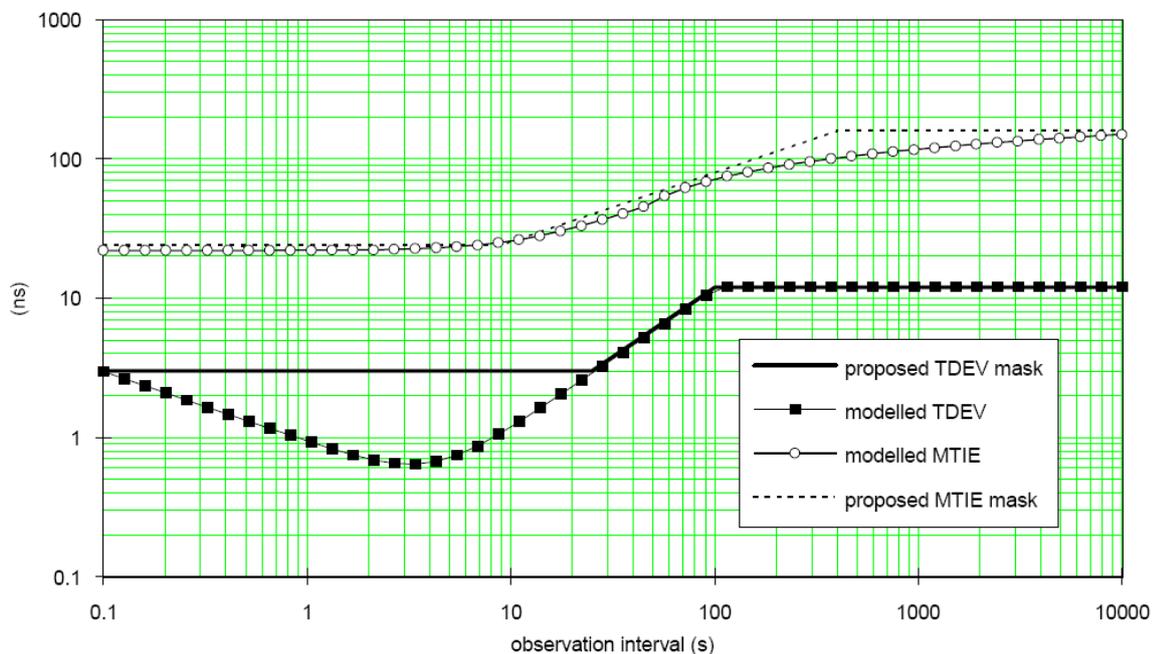
**Figure 39 – Option 2, model 2 TDEV for SSU 10 (final SSU) and EECs/SECs 41 – 58 (EECs/SECs following final SSU)**

## 8.1.4 ETSI Option 1 wander generation models and corresponding wander accumulation simulations and results

### 8.1.4.1 ETSI Option 1 wander generation models

The SEC and SSU noise generation models documented in Annex A of [b-ETSI01] (for the SSU) and Annex C of [b-ETSI02] (for the SEC) both consist of a linear combination of white phase modulation (WPM) followed by a high-pass filter, and flicker phase modulation followed by a low-pass filter. In this sense, the models are mathematically similar to the models used in clause 8.1.2, i.e., those latter models are also constructed as a linear combination of power-law noise processes (with small integer exponents) followed by high-pass and/or low-pass filters. This means that the existing noise accumulation simulator can be used for the ETSI noise generation models, but with different noise components, gains, and filter bandwidths.

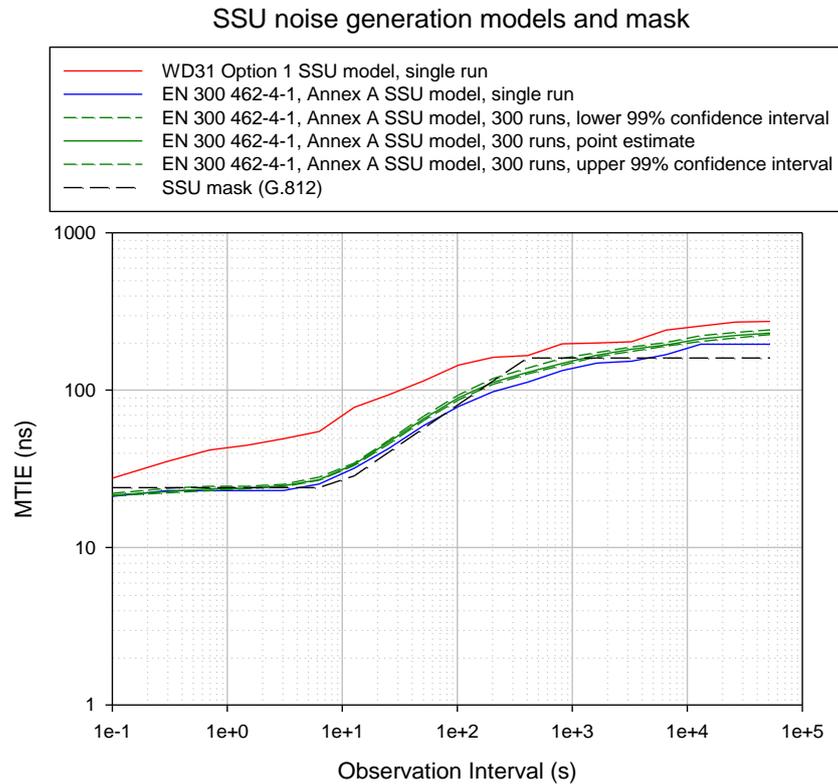
Figure 40 shows MTIE and TDEV for the ETSI SSU noise generation model, along with the MTIE and TDEV noise generation masks (this figure is copied from Figure A.3 of [b-ETSI01]). The model was constructed to match the TDEV mask over longer intervals corresponding to the higher flat level (i.e., greater than 100 s), and for intervals corresponding to the transition between the two flat levels (i.e., 25 – 100 s). However, for shorter intervals TDEV for the model is below the TDEV mask, but MTIE for the model matches the MTIE mask. To implement this model in the noise accumulation simulator, it was necessary to choose a white noise standard deviation value and a corresponding high-pass filter bandwidth, and an input white noise standard deviation value for a Barnes/Jarvis/Greenhall filter to generate flicker noise (see [b-Barnes02]) and corresponding low-pass filter bandwidth. These values were chosen by trial and error (i.e., choosing initial values, generating a sample phase error history, computing TDEV and comparing with the mask, varying one or more parameters to improve the fit, and repeating the process until an acceptable fit was obtained). The resulting SSU noise model parameter values are given in Table 22.



**Figure 40 – ETSI SSU noise generation model, and Option 1 SSU noise generation MTIE and TDEV masks (see Figure A.3 of [ITU-T G.812])**

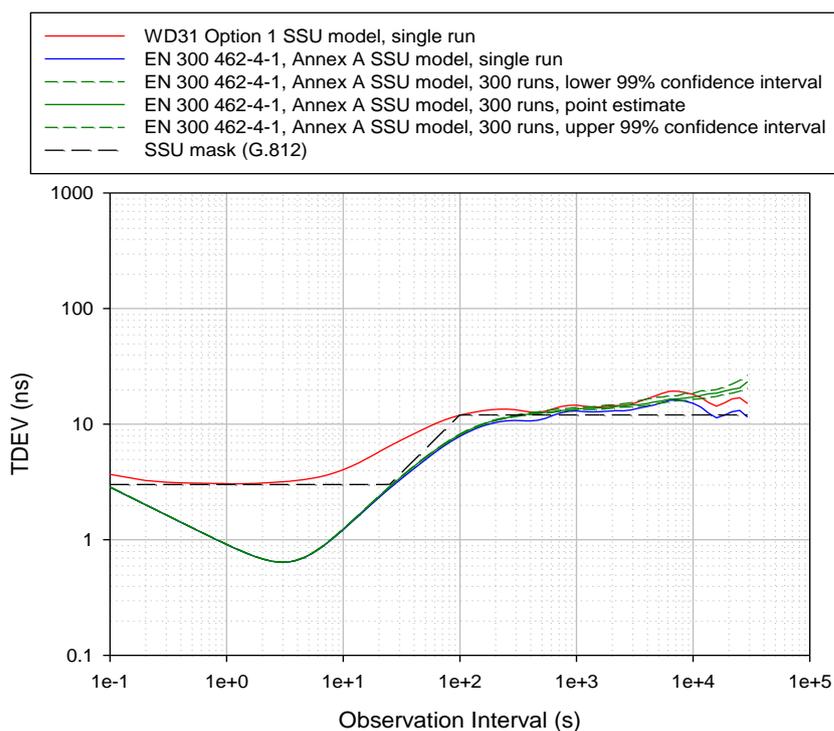
**Table 22 – SSU noise model parameters to match ETSI SSU noise generation model of Figure 40 (Annex A of [ITU-T G.812])**

Noise Source	Input white noise standard deviation (ns)	Low-pass filter bandwidth (Hz)	High-pass filter bandwidth (Hz)
WPM	0.9	–	0.003
FPM	20.0	0.003	–



**Figure 41 – MTIE for ETSI SSU noise generation model implementation, and comparison with Option 1 MTIE mask**

### SSU noise generation models and mask



**Figure 42 – TDEV for ETSI SSU noise generation model implementation, and comparison with Option 1 TDEV mask**

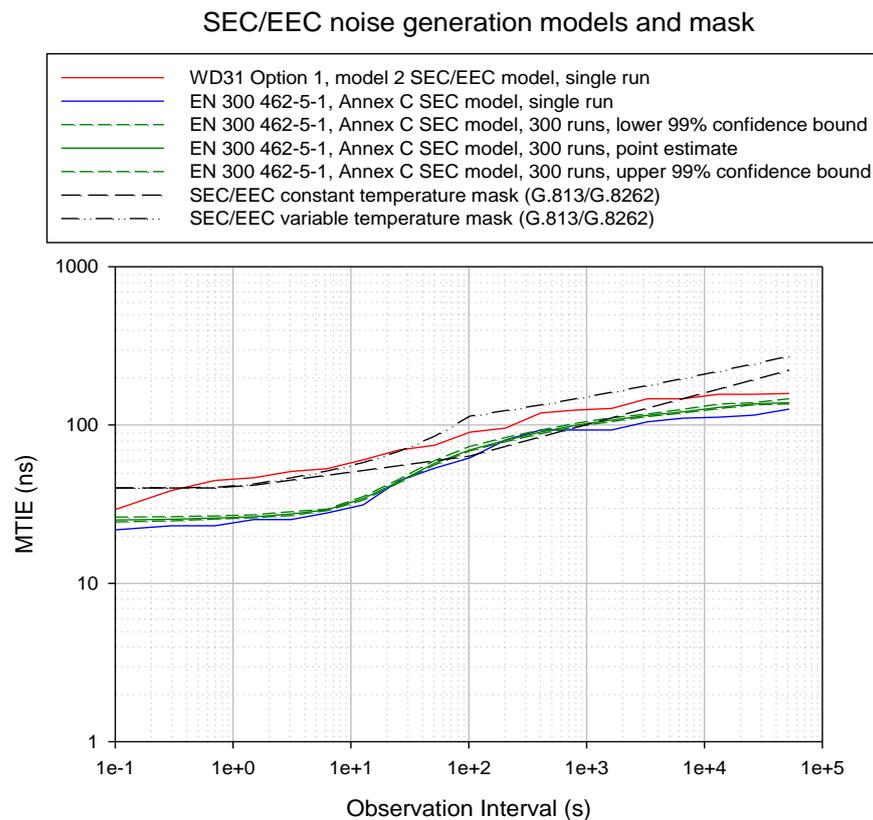
Figures 41 and 42 show MTIE and TDEV, respectively, for the implementation of the ETSI SSU noise generation model developed here (i.e., using the parameters of Table 22). The results are shown for both the single run whose TDEV was compared with the TDEV mask (to obtain a good fit) and a 99% confidence interval for the 0.95 quantile of the measured TDEV obtained from a set of 300 runs. The single run and 99% confidence interval results are in close agreement. Also shown, for comparison, is the TDEV for Option 1 SSU model of clause 8.1.2.2. The TDEV for the ETSI model is somewhat less than that of Option 1 SSU model of clause 8.1.2.2 for observation intervals less than 100 s. This is also the range of observation intervals for which the network limit was exceeded. This indicates that it is possible that the network limit will be met using the ETSI model (which will turn out to be the case). MTIE for the ETSI SSU model is closer to the mask than MTIE for Option 1 SSU model of clause 8.1.2.2 (and is below MTIE for Option 1 SSU model of clause 8.1.2.2). Both MTIE and TDEV for the ETSI SSU model implementation are in reasonable agreement with Figure 40.

A similar procedure was used to develop an implementation for the ETSI SEC model of Appendix C of [b-ETSI02]. For this case, MTIE and TDEV plots were not provided in [b-ETSI02]. However, it was assumed the model was constructed to match the upper flat level and transition between lower and upper flat levels of the TDEV mask, in a manner similar to the SSU model. The noise model parameters are given in Table 23.

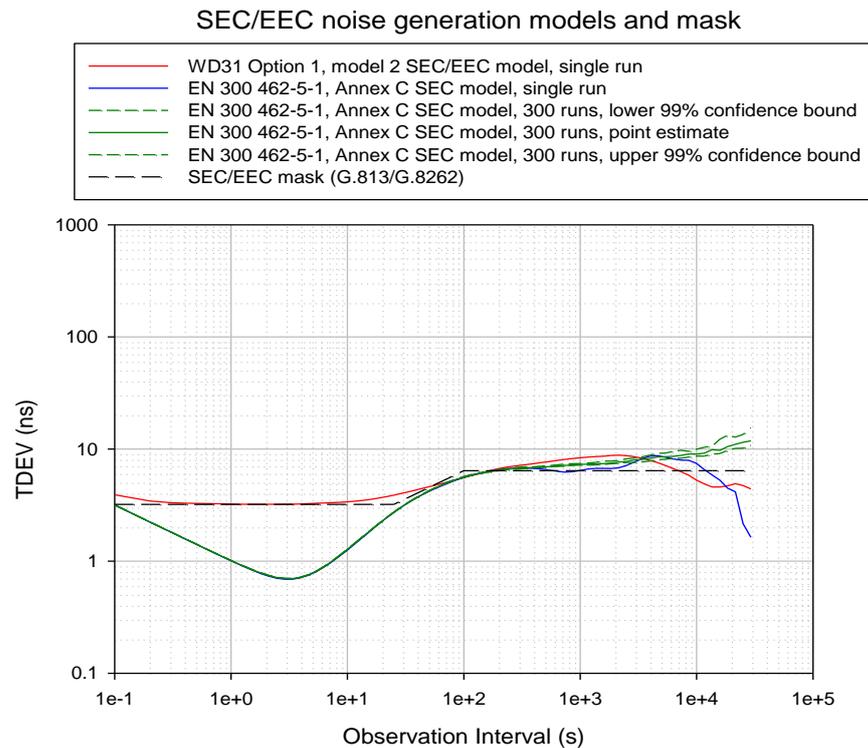
**Table 23 – SEC/EEC noise model parameters, following ETSI SEC noise generation model (Annex C of [b-ETSI02])**

Noise Source	Input white noise standard deviation (ns)	Low-pass filter bandwidth (Hz)	High-pass filter bandwidth (Hz)
WPM	1.0	–	0.006
FPM	10.67	0.006	–

Figures 43 and 44 show MTIE and TDEV, respectively, for the implementation of ETSI SEC noise generation model developed in this Supplement (i.e., using the parameters of Table 23). The results are shown for both the single run whose TDEV was compared with the TDEV mask (to obtain a good fit) and a 99% confidence interval for the 0.95 quantile of the measured TDEV obtained from a set of 300 runs. The single run and 99% confidence interval results are in close agreement. Also shown, for comparison, is TDEV for Option 1, Model 2 of clause 8.1.2.1. The TDEV for the ETSI model is somewhat less than that of Option 1, Model 2 of clause 8.1.2.1 for observation intervals less than 100 s. This is also the range of observation intervals for which the network limit was exceeded. This indicates that it is possible that the network limit will be met using the ETSI model (this will turn out to be the case). MTIE for the ETSI SEC model is closer to the mask than MTIE for Option 1, Model 2 of clause 8.1.2.1 (and is below MTIE for Option 1, Model 2 of clause 8.1.2.1). As with the SSU models, both MTIE and TDEV for the ETSI SEC model implementation are in reasonable agreement with Figure 40.



**Figure 43 – MTIE for ETSI SEC noise generation model implementation here, and comparison with Option 1 MTIE mask**



**Figure 44 – TDEV for ETSI SEC noise generation model implementation, and comparison with Option 1 TDEV mask**

**8.1.4.2 Option 1 wander accumulation over the synchronization reference chain using ETSI Option 1 wander generation models**

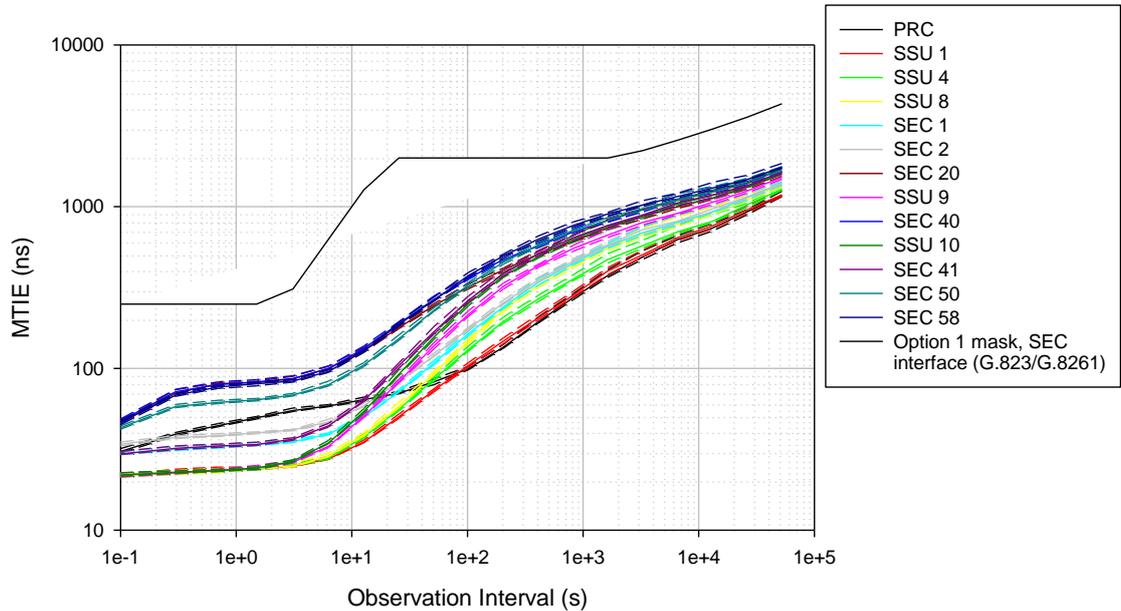
The wander generation models of clause 8.1.4.1 were used to evaluate wander accumulation over the adapted ITU-T G.803 synchronization reference chain described in clause 8.1.1 (i.e., with 18 rather than 20 EECs/SECs following SSU 10). The simulation model described in 8.1.3.1 was used, i.e., each EEC/SEC and SSU was modelled as a second-order, low-pass filter with specified bandwidth and gain peaking, followed by additive noise source (see Figure 23). The model parameters other than those pertaining to SEC and SSU wander generation are given in Table 20.

MTIE and TDEV wander accumulation results shown in Figure 45 to 48. The format follows the presentation of clause 8.1.3. The results are first given for the output of the PRC and selected SSUs and SECs/EECs, and then in a separate figure for the final SSU (SSU 10) and the SECs/EECs that follow the final SSU (SECs/EECs 41 to 58).

The results show that the wander accumulation TDEV is below Option 1 TDEV network limit for all clocks, and for the full range of observation intervals (0.1 – 100,000 s). TDEV is well below the network limit for observation intervals in range 1 to 10 s, which is the range where the network limit was exceeded for Option 1 models of clauses 8.1.2 and 8.1.3. This meant that there would now be a margin for the OTN (in the OTN analyses), and the TDEV network limit would be met at the OTN output. The MTIE results are also well below Option 1 MTIE network limit.

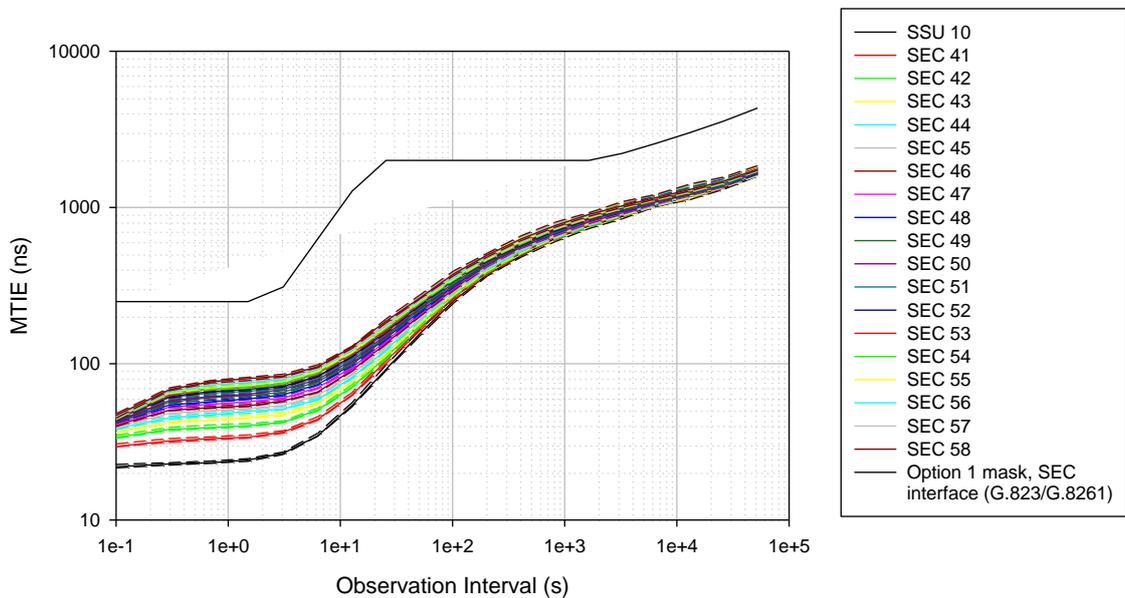
Note also that MTIE and TDEV for the PRC are below MTIE and TDEV for all the EECs/SECs and SSUs for longer observation intervals, i.e., longer than approximately 100 s. However, MTIE and TDEV for the PRC are above MTIE and TDEV for some of the EECs/SECs and SSUs for shorter observation intervals. This is due to the narrow-bandwidth SSU filter (0.001 Hz) combined with the reduced wander generation of the SSUs and SECs/EECs compared to that in clause 8.1.2.

Option 1, 10 Hz SEC/EEC BW  
 SEC noise model follows EN 300 462-5-1, Annex C  
 SSU noise model follows EN 300 462-4-1, Annex A



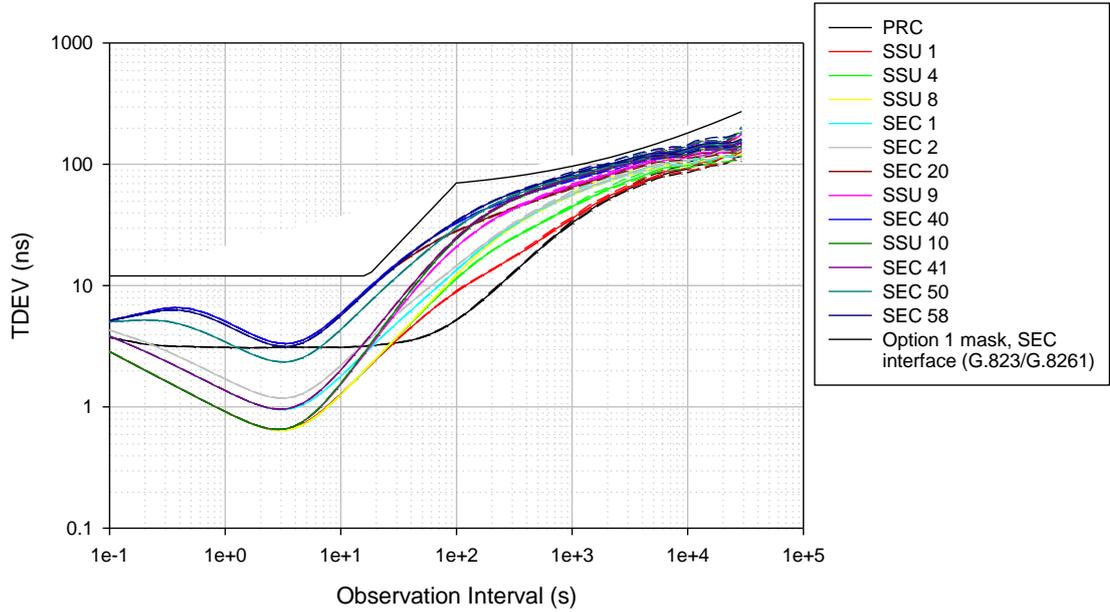
**Figure 45 – MTIE for accumulated wander in synchronization reference chain, for selected SECs/EECs and SSUs**

Option 1, 10 Hz SEC/EEC BW  
 SEC noise model follows EN 300 462-5-1, Annex C  
 SSU noise model follows EN 300 462-4-1, Annex A



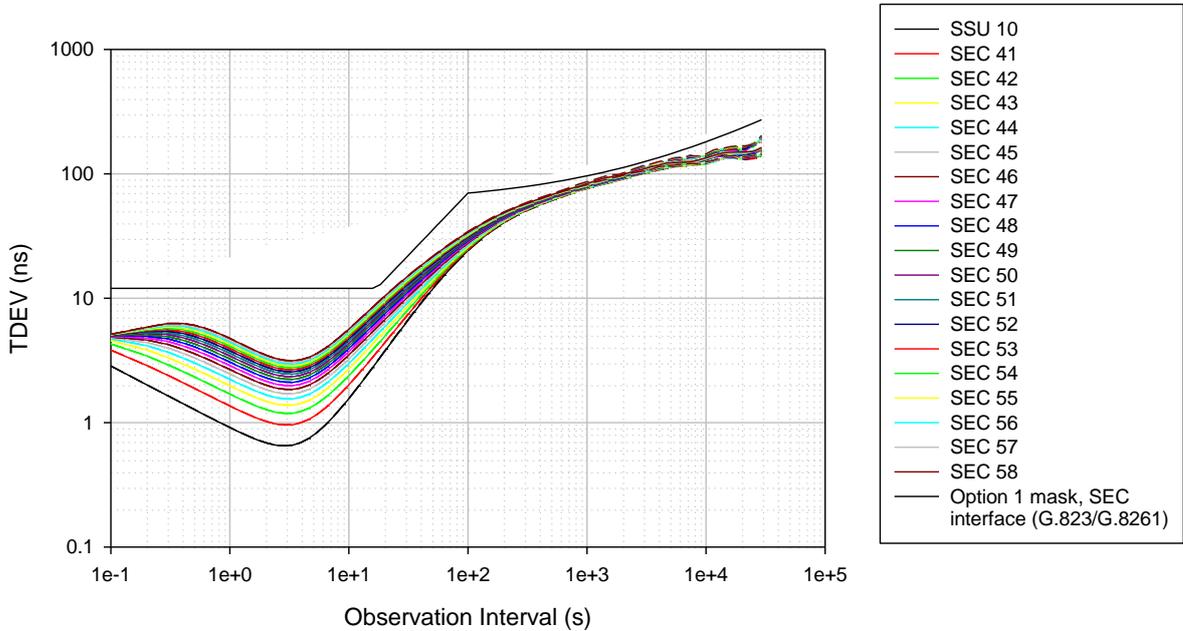
**Figure 46 – MTIE for accumulated wander in synchronization reference chain, for SSU 10 (final SSU) and SECs/EECs 41 – 58 (SECs/EECs following final SSU)**

Option 1, 10 Hz SEC/EEC BW  
 SEC noise model follows EN 300 462-5-1, Annex C  
 SSU noise model follows EN 300 462-4-1, Annex A



**Figure 47 – TDEV for accumulated wander in synchronization reference chain, for selected SECs/EECs and SSUs**

Option 1, 10 Hz SEC/EEC BW  
 SEC noise model follows EN 300 462-5-1, Annex C  
 SSU noise model follows EN 300 462-4-1, Annex A



**Figure 48 – TDEV for accumulated wander in synchronization reference chain, for SSU 10 (final SSU) and SECs/EECs 41 – 58 (SECs/EECs following final SSU)**

## 8.2 Frequency-domain mathematical model, simulator, and simulation results

### 8.2.1 Introduction

The frequency domain model of a SyncE reference chain was developed specifically for use with the frequency domain models for time error accumulation in a chain of T-BCs. As indicated in clause 8.1.1, the time-domain model and simulator for a SyncE reference chain was originally developed for use in jitter and wander accumulation studies of the transport of SyncE and STM-1 clients over a network of OTN islands. Since the time-domain model and simulator for a chain of T-BCs was developed later, the time-domain SyncE model could be reused. The frequency domain models were developed as alternative models, and could be used with the time domain models as a consistency check (i.e., it is expected that, for the same underlying assumptions, the time and frequency domain models should give consistent results).

The frequency-domain model was applied to the same adapted ITU-T G.803 reference chain as the time-domain model. This adapted reference chain is described in clause 8.1.1 where the initial adapted reference chain contained 18 EECs/SECs after the final SSU. The initial chain was used in the time domain simulations for HRM3 of Appendix II of [ITU-T G.8271.1] (see clause 7). However, subsequent time domain simulations for HRM2 of Appendix II of [ITU-T G.8271.1] required the use of a reference chain with 20 EECs following the final SSU and, for some of the analyses involving SyncE rearrangements, a narrower-bandwidth "SSU-like" clock following the 20<sup>th</sup> EEC. While the frequency domain model and simulator could be used to simulate time error accumulation in an HRM2 reference chain, simulations were done only for HRM3. It was therefore not necessary to simulate the additional reference chain with 20 EECs following the final SSU with the frequency domain simulator.

As in the time-domain model, each SSU and EEC is modelled as a low-pass filter that filters incoming noise from the previous clock in the chain, with noise (wander) generation added to the result. The PRC is modelled as a source of phase error due to noise generation. The power spectral density of the output of each clock is related to the power spectral density of the input and the power spectral density of the noise generation of the clock. Using the approximate relation between power spectral density and TDEV given in Appendix I of [ITU-T G.812], the TDEV of the output of a clock may be related to the TDEV of the input and the TDEV of the noise generation of the clock. Using this relation, the TDEV at the output of each clock in the reference chain may be computed. In addition, an approximate relation between the MTIE and the TDEV is developed, to enable the MTIE at the output of each clock to be computed once the TDEV is obtained.

The noise generation in the PRC, SSUs, and EECs is modelled using the noise generation TDEV mask for each clock. The noise generation is assumed to be at the level of each mask, and the respective mask is used in computing the output TDEV of each clock (the details of the computation are given in clause 8.2.2). It was indicated in the description of the time-domain wander generation models (see clauses 8.1.2 and 8.1.4.1) that the MTIE and TDEV for the noise generation are not consistent, i.e., a process that just meets the respective TDEV mask will exceed the MTIE mask for Option 2 and fall below the MTIE mask for Option 1. As a result, Option 2 time-domain model was constructed to marginally exceed both masks without falling below either mask. For Option 1, time-domain noise generation models developed in ETSI (see clause 8.1.4.1) were used. These generally match the MTIE mask and fall below the TDEV mask (these models were used because it was found that, with models that attempted to not fall below either Option 1 noise generation mask, the Option 1 network limits were exceeded at the end of the ITU-T G.803 reference chain). This means that the time-domain and frequency-domain noise generation models do not represent the exact same noise levels which must be taken into account when comparing the results of the two models.

Since, in the frequency domain, the noise (wander) generation is modelled as a random process via its power spectral density, the model does not include transients, frequency offset, or frequency drift. In particular, the PRC frequency offset is not directly included in the model. As the steady-state tracking error of a phase ramp, i.e., frequency offset, is zero for a second-order filter, and since the

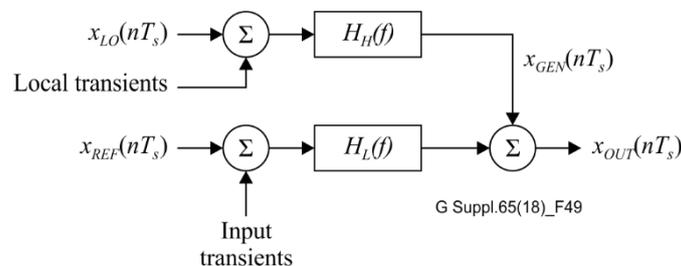
clock filters are linear, the actual output of any clock is equal (in steady-state) to the sum of the output due to the noise generation random processes in the PRC, SSUs, and EECs plus the PRC frequency offset. Therefore, since the time variance (TVAR) and TDEV are not affected by a frequency offset, the TDEV at the output of each clock is not affected by neglecting the PRC frequency offset. However, MTIE is affected by the frequency offset, and therefore the fact that the PRC frequency offset is included in the time-domain model but not the frequency-domain model, it must be accounted for when the MTIE results of the two models are compared.

In the following clauses, the model for propagating TDEV through a chain of clocks will first be described, followed by the approximate model for computing MTIE from the TDEV results and then the results for MTIE and TDEV using the frequency-domain model will be described.

## 8.2.2 Model for TDEV of wander accumulation

### 8.2.2.1 The premise

In the frequency-domain model, the output clock noise in locked mode (i.e., steady-state) is modelled as the sum of two filtered random processes. The model is illustrated in Figure 49. This form of modelling for a loop was previously described and discussed (for example, see Appendix IV of [ITU-T G.8251], [b-Shenoi01], and [b-Bregni]).



**Figure 49 – Model for clock noise in locked mode**

When operating in locked mode, the clock output is aligned to the reference in frequency/phase. The reference contains perturbations (wander) and the action of the loop is to apply a low-pass filter, denoted by  $H_L(f)$ , between the reference and the output. The local oscillator also introduces perturbations, and the action of the loop is to apply a high-pass filter, denoted by  $H_H(f)$ , between the local clock noise and the output. It should be noted that the filter characteristics are not independent. The control loop parameters completely define both  $H_L(f)$  and  $H_H(f)$ . The input and output signals are shown as discrete-time signals, where  $n$  is the sample index and  $T_s$  the sampling time, because modern phase-locked loops (PLLs) operate in the discrete-time domain. The noise on the input reference is  $x_{REF}$ , the local oscillator noise is  $x_{LO}$ , the locally-generated noise (i.e., the noise generation) is  $x_{GEN}$ , and the output is  $x_{OUT}$ . However, the frequency-domain model represents power spectral densities and transfer functions in the continuous-time domain. The model in Figure 49 is mathematically equivalent to the time-domain model of Figure 23 if the input in Figure 23 represents the sum of the input reference and input transients in Figure 1. The noise generation in Figure 23 represents the high-pass filtered input noise in Figure 1, and the transients in Figure 23 represent the high-pass filtered transients in Figure 1.

Input transients can be viewed as perturbations of the reference and are low-pass filtered. Local transients are generally associated with oscillator imperfections and are high-pass filtered. In general, every loop has a voltage-controlled oscillator (VCO) or a number-controlled oscillator (NCO) (or equivalent). The signals that enter the loop at the input to the oscillator experience a low-pass filter and the signals that enter the loop at the output of the oscillator experience a high-pass filter. It is noted that transients are not actually included in the frequency-domain model because they are not easily modelled using a power spectral density. Figure 49 shows where transients would enter *if* they were modelled.

If the clock is not locked to a reference and is effectively free-running, then the input reference noise,  $x_{REF}(nT)$ , is moot and the local oscillator clock noise is present in the clock output directly (no filtering).

Existing recommendations provide limits, i.e., masks, for the noise generation  $x_{GEN}$ . This signal is chosen (mathematically) to have a TVAR/TDEV that is exactly equal to the maximum allowed by the mask for that clock type.

### 8.2.2.2 The analytical modelling approach

The approach taken is to establish upper limits to the relevant metrics describing the clock output. In the process of doing this conservative approximations are made. The general approach is based on the following:

- a. Given a chain of clocks, the input noise (noise in the reference) for clock  $n$  is the output clock noise of clock  $n-1$ .
- b. The locally-generated clock noise,  $x_{GEN}(nT_s)$ , can be modelled by a random signal with TDEV that meets the wander generation mask of the clock type being considered.
- c. The locally-generated clock noise is (statistically) independent of both the clock noise in the reference and the locally-generated noise in all other clocks.
- d. The "start" of a chain is usually an external clock, and for this clock the input reference clock noise is assumed to be such that the output noise of the external clock only satisfies the output mask for that external clock. The start of a chain is, most likely, a PRC.
- e. The TVAR of the clock output is the sum of the TVAR of the local component and the TVAR of the reference component, after the appropriate filtering. That is, the local clock noise and the reference clock noise are assumed to be independent.

It is assumed that the allowable wander generation for clock  $n$  is expressed by the TVAR function (mask)  $W_{GEN}(\tau)$ . This is the mask that describes the limits of  $x_{GEN}(nT_s)$ . It is assumed that this allowable level of noise is for the case where the clock is locked to a wander-free reference and therefore represents the noise at the output of the high-pass filter in Figure 49. In the time-domain model of clause 8.1, certain noise sequences that have TDEV (TVAR) approximately equal to the mask are described, and these sequences are used in the time-domain model. In the frequency-domain model, TDEV (TVAR) can be chosen to be exactly equal to the mask and thereby represent the worst (acceptable) case. Note, however, that this ignores the MTIE mask constraint on noise generation. As noted in the introduction to the frequency-domain model in clause 8.2.1, the time-domain model took into account the wander generation MTIE mask. This difference in the two models must be considered when comparing the results.

The noise at the output of the low-pass filter can be analysed in the following manner. Since there is a simple relationship between TVAR and the power spectral density,  $S_x(f)$ , of a clock noise process  $x(nT_s)$ , the impact of the filter can be introduced in the Fourier frequency domain in terms of the frequency-response of the filter. To see how this is achieved, a relationship must first be established between TVAR and power spectrum.

### 8.2.2.3 Relationship between TVAR and power spectrum

The computation of TVAR (and therefore TDEV) can be viewed as estimating the power of the process after filtering by a band-pass filter. Considering that the power spectrum can also be modelled as the power contained within a certain band of frequencies, it is logical to expect a linkage between TVAR and spectrum. The distinction is that, for power spectrum the band-pass filters are of uniform bandwidth and nominally non-overlapping in frequency. In the case of TVAR, there is a slight overlap and the bandwidth of the filters is not uniform (see [b-Shenoi01] for details). The following equation, based on the expression in Appendix I, equation (I-3) of [ITU-T G.812], provides an approximate relation between power spectral density and TVAR:

$$S_x(f) \approx \left( \frac{K_1}{f} \right) \cdot \sigma_x^2(\tau) \Big|_{\tau = \frac{K_2}{f}} \quad (8-2)$$

In equation (8-2),  $\sigma_x^2(\tau)$  denotes the TVAR of the phase error signal  $x(t)$ . As indicated in [ITU-T G.812], the appropriate values for the constants are  $K_1 = 0.75$  and  $K_2 = 0.3$ . Consequently, the relationships between TVAR (or TDEV,  $\sigma_x(\tau)$ ) and power-spectral density can be written as:

$$S_x(f) \approx \left( \frac{0.75}{f} \right) \cdot \left( \sigma_x \left( \frac{0.3}{f} \right) \right)^2 \quad (8-3)$$

$$TDEV(\tau) = \sigma_x(\tau) \approx \sqrt{\left( \frac{1}{2.5 \cdot \tau} \right) \cdot S_x \left( \frac{0.3}{\tau} \right)}$$

The action of the low-pass filter is to modify the power spectral density of the input, denoted by  $S_x(f)$ , to get the effective contribution at the output, denoted by  $S_y(f)$ , i.e., the component of the output clock noise transferred from the reference clock noise. Specifically,

$$S_y(f) = |H_L(f)|^2 \cdot S_x(f) \quad (8-4)$$

In equation (8-4),  $y(t)$  is the result of filtering the phase error signal  $x(t)$  by the low-pass filter  $H_L(f)$ , and  $\sigma_y^2(\tau)$  denotes TVAR of  $y(t)$ . (In much of the literature the notation  $\sigma_y^2(\tau)$  is used to denote Allan Variance. However, it does *not* denote Allan variance here.) In terms of TVAR

$$TVAR_y(\tau) = \sigma_y^2(\tau) = \left| H_L \left( \frac{0.3}{\tau} \right) \right|^2 \cdot \sigma_x^2(\tau) = \left| H_L \left( \frac{0.3}{\tau} \right) \right|^2 \cdot TVAR_{REF}(\tau) \quad (8-5)$$

Consequently, the TVAR of the clock output can be written as:

$$TVAR_{OUT}(\tau) = W_{GEN}(\tau) + \left| H_L \left( \frac{0.3}{\tau} \right) \right|^2 \cdot TVAR_{REF}(\tau) \quad (8-6)$$

In equation (8-6), the contribution of the locally generated clock noise,  $W_{GEN}(\tau)$ , is set to the limit allowed for the particular clock type.

#### 8.2.2.4 Recursion relationship for power spectrum and TVAR of clock noise

Analysing the behaviour of a chain of clocks is straightforward in an analytical framework once the clock types and filter characteristics are specified. Suppose  $S_0(f)$  is the power spectrum at the input of the first clock in the chain. Denote by  $H_k(f)$  the low-pass characteristic (c.f.  $H_L(f)$ ) of the  $k^{th}$  clock and  $W_k(f)$  the power spectrum of the noise added in the  $k^{th}$  clock (c.f.  $x_{GEN}$ ). Let  $S_k(f)$  denote the power spectrum of the clock noise at the output of the  $k^{th}$  clock. Then

$$S_k(f) = |H_k(f)|^2 \cdot S_{(k-1)}(f) + W_k(f) \quad (8-7)$$

A similar recursion relation for TVAR of clock noise is obtained with the following equation:

$$TVAR_k(\tau) = \left| H_k \left( \frac{0.3}{\tau} \right) \right|^2 \cdot TVAR_{(k-1)}(\tau) + W_{GEN,k}(\tau) \quad (8-8)$$

Equations (8-6) and (8-8) are equivalent;  $TVAR_{OUT}(\tau)$ ,  $TVAR_{REF}(\tau)$ , and  $W_{GEN}(\tau)$  in equation (8-6) are  $TVAR_k(\tau)$ ,  $TVAR_{(k-1)}(\tau)$ , and  $W_{GEN,k}(\tau)$ , respectively, in equation (8-8).

### 8.2.3 Filter frequency response

The equations in the previous clauses are general, i.e., they are valid for any filter frequency response and noise power spectrum. However, in the analyses in this Supplement, the filters are second-order with 20 dB/decade roll-off. The transfer function for such a filter is

$$H(s) = \frac{2\zeta\omega_n s + \omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}, \quad (8-9)$$

where  $\omega_n$  is the undamped natural frequency in rad/s and  $\zeta$  is the damping ratio. Setting  $s = j\omega$ ,  $\omega = 2\pi f$ , and  $\omega_n = 2\pi f_n$ , where  $\omega$  and  $f$  are the Fourier frequency in rad/s and Hz, respectively, and  $f_n$  is the undamped natural frequency in Hz, produces

$$H(f) = \frac{2j\zeta f_n f + f_n^2}{-f^2 + 2j\zeta f_n f + f_n^2}. \quad (8-10)$$

Taking the absolute value of equation (8-10), and squaring, produces

$$|H(f)|^2 = \frac{4\zeta^2 f_n^2 f^2 + f_n^4}{(f_n^2 - f^2)^2 + 4\zeta^2 f_n^2 f^2} = \frac{1 + 4\zeta^2 \cdot \left(\frac{f}{f_n}\right)^2}{\left[1 - \left(\frac{f}{f_n}\right)^2\right]^2 + 4\zeta^2 \cdot \left(\frac{f}{f_n}\right)^2} \quad (8-11)$$

The undamped natural frequency and 3 dB bandwidth are related by (see [b-Gardner])

$$f_{3dB} = f_n \cdot \sqrt{(1 + 2\zeta^2) + \sqrt{(1 + 2\zeta^2)^2 + 1}}, \quad (8-12)$$

Where  $f_{3dB}$  is the 3dB bandwidth in Hz. The relation between damping ration and gain peaking is given by (see [b-Wolaver])

$$H_p = \left[1 - 2\alpha - 2\alpha^2 + 2\alpha(2\alpha + \alpha^2)^{1/2}\right]^{-1/2}, \quad (8-13)$$

where  $\alpha = 1/(4\zeta^2)$  and  $H_p$  is the gain peaking, expressed as a pure number (in cases of practical interest here, its value is slightly greater than 1). This may be solved for  $\alpha$  in terms of  $H_p$ ; the result is

$$\alpha = \frac{(1-q)(1+\sqrt{1-q})}{2q}, \quad (8-14)$$

where  $q = 1/H_p^2$ . The gain peaking in dB,  $H_{p,dB}$ , is given by

$$H_{p,dB} = 20 \log H_p. \quad (8-15)$$

For small gain peaking, i.e., large damping ratio ( $\zeta \gg 1$ ),  $\alpha \ll 1$ , and equation (8-13) may be approximated by

$$H_p \cong 1 + \alpha = 1 + \frac{1}{4\zeta^2}, \quad (8-16)$$

### 8.2.4 Model for MTIE of wander accumulation

Clause 8.2.2.3 describes an approximate relation between TDEV and power spectral density, and uses this to obtain a recursion relation (equation 8-8) for TVAR (TDEV) at the output of a clock in a chain in terms of TVAR (TDEV) at the output of the previous clock in the chain and TVAR (TDEV) of the clock noise generation. This clause describes an approximate relation between TDEV and an upper

limit for MTIE. This material is taken from [b-Anritsu]. With this relation, an upper limit for MTIE at the output of any clock in a chain may be obtain once equation (8-8) is used to propagate TDEV along the chain. The main result, equation (8-28), follows from equations (11) and (12) of [b-Anritsu].

### 8.2.4.1 Relation between TIErms and MTIE upper limit

For a sampled time error function,  $x(i)$ , with sampling interval  $\tau_0$  and observation interval  $\tau = n\tau_0$ ,  $n = 1, 2, \dots, N-1$ , the time interval error, TIE( $i;n$ ), is given by ([ITU-T G.810])

$$\text{TIE}(i;n) = x(i+n) - x(i), \quad (i = 1, 2, 3, \dots, N-n). \quad (8-17)$$

In the following analysis, it is assumed that the distribution for the samples of TIE( $i;n$ ),  $i = 1, 2, 3, \dots, N-n$ , for arbitrary  $n$ , is Gaussian with a mean value of zero. The estimator formula for the root mean square time interval error (TIErms) is defined in [ITU-T G.810] as:

$$\text{TIErms}(n\tau_0) = \sqrt{\frac{1}{N-n} \sum_{i=1}^{N-n} [x(i+n) - x(i)]^2}. \quad (8-18)$$

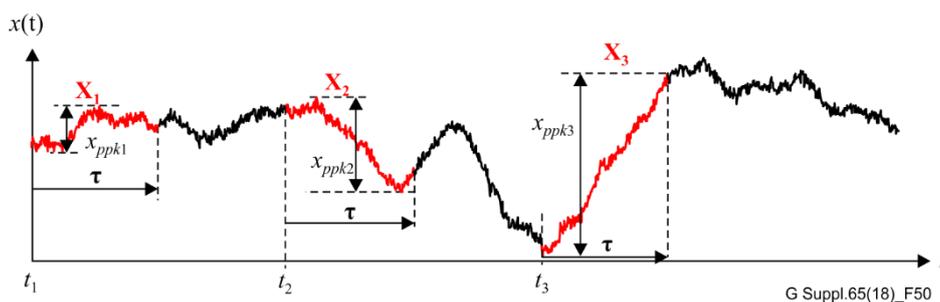
The estimator formula for MTIE for the observation interval  $\tau = n\tau_0$ ,  $n = 1, 2, \dots, N-1$ , is defined in [ITU-T G.810] as:

$$\text{MTIE}(n\tau_0) = \max_{1 \leq k \leq N-n} \left\{ \max_{k \leq i \leq k+n} x(i) - \min_{k \leq i \leq k+n} x(i) \right\}. \quad (8-19)$$

To evaluate the right-hand side of equation (8-19), it is necessary to find the set of  $x(i)$ ,  $i = k, k+1, \dots, k+n$ , with the largest peak-to-peak value of  $x(i)$  among  $n$ -samples. A typical example is shown in Figure 50. The peak-to-peak value,  $x_{ppk3}$ , of the curve  $X_3$ , which increases monotonously in a rough manner, is larger than the values of  $X_1$  and  $X_2$ . The magnitude of  $x_{ppk3}$  is equal to or a little larger than the difference at both ends of the curve  $X_3$ , i.e.  $x_{ppk3} \cong x(t_3 + \tau) - x(t_3)$ . Therefore, when the measurement period  $T (= N\tau_0)$  is sufficiently large, we may replace the right side of equation (8-19) by the approximation:

$$\text{MTIE}(n\tau_0) \cong \max_{1 \leq k \leq N-n} |x(k+n) - x(k)| = \max_{1 \leq k \leq N-n} |\text{TIE}(k;n)|. \quad (8-20)$$

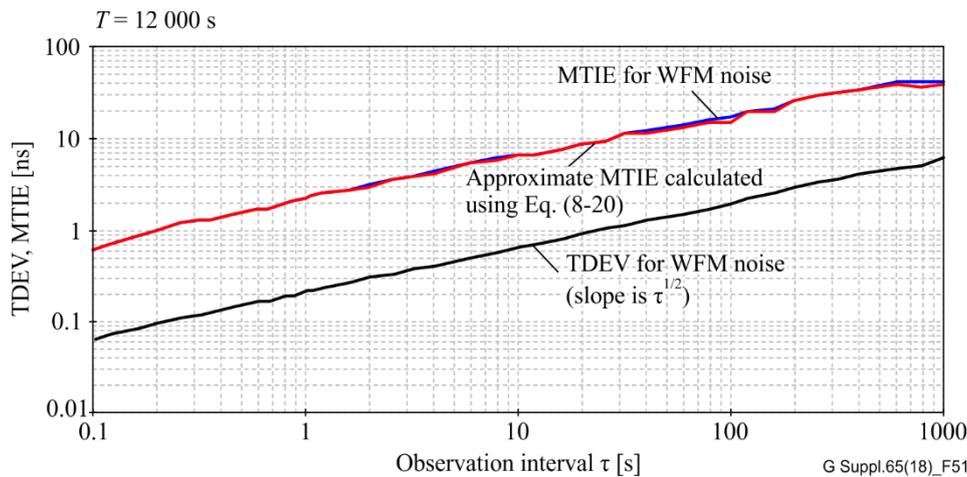
The right side of equation (8-20) represents the maximum of the absolute value of TIE( $k;n$ ),  $k = 1, 2, 3, \dots, N-n$ , for observation interval  $n\tau_0$ .



**Figure 50 – Peak-to-peak values of time error function for interval  $\tau$**

Note that equation (8-20) is valid only for observation intervals that are not longer than the duration of the peak-to-peak excursions. For example, in Figure 50, if equation (8-20) is applied over an observation interval  $\tau$  that is longer than the duration of the peak-to-peak  $X_3$ , then MTIE for that  $\tau$  will be less than  $X_3$ . Some examples of cases where equation (8-20) is not a good approximation to equation (8-19) are given in [b-Anritsu]. Figure 51 shows an example of the TDEV and MTIE for a white frequency modulation (WFM) noise sample. The sample was generated by first generating white noise (i.e., white phase modulation (WPM)) and then accumulating it. The TDEV and MTIE of Figure 51 were calculated by the respective estimator formulas given in [ITU-T G.810]. Figure 51

also shows the approximate MTIE calculated by equation (8-20). These results indicate that the approximate MTIE is very similar to the original MTIE.



**Figure 51 – TDEV and MTIE for WFM noise generated by verification algorithm in Figure V.2 of [ITU-T O.172]**

Next, an upper bound for MTIE in terms of TIErms is obtained. First, the probability that the maximum value among independent  $M$ -samples, with a Gaussian distribution and a standard deviation of  $\sigma$ , is  $X_{\max}$  is evaluated. This is the probability that one sample among  $M$  samples falls between  $X_{\max}$  and  $X_{\max} + dX$ , and all other  $M - 1$  samples are less than  $X_{\max}$ . If the samples are independent of each other, the probability density function  $g(X_{\max})$  is [b-Papoulis]

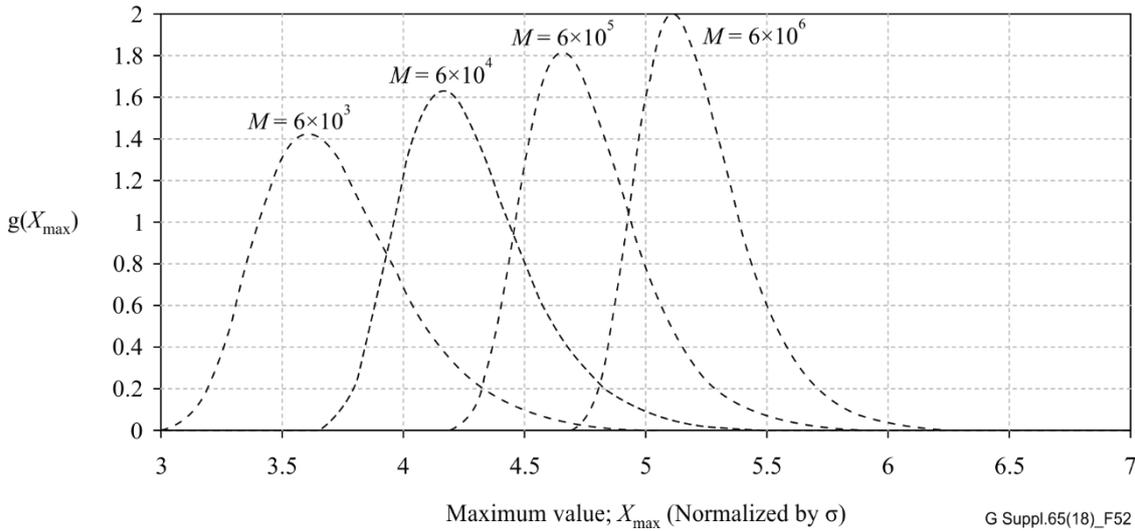
$$g(X_{\max}) = M \left[ \int_{-\infty}^{X_{\max}} f_G(w) dw \right]^{M-1} f_G(X_{\max}), \quad (8-21)$$

where

$$f_G(w) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{w^2}{2\sigma^2}}.$$

The assumption that the samples are independent of each other is equivalent to assuming that the underlying random process is white noise. This is a key approximation, as many noise processes of interest are not white.

For the TDEV wander generation specified in [ITU-T G.812], [ITU-T G.813] or [ITU-T G.8262], the maximum number of samples of TIE( $k;n$ )  $\{k = 1, 2, 3, \dots, N-n\}$  is  $N - n = 6 \times 10^6$ . This is based on a measurement period  $T = 12\tau_{\max} = 120,000$  s and an observation interval  $\tau_{\min} = 0.05$  s, when the sampling period  $\tau_0$  is equal to  $1/50$  s. The rightmost curve in Figure 52 shows the  $g(X_{\max})$  calculated by a computer for  $M = 6 \times 10^6$ . The magnitude of  $g(X_{\max})$  peaks when  $X_{\max} = 5.1\sigma$ , and  $X_{\max}$  is  $\leq 7.0\sigma$  with a probability of 99.99%. Similarly,  $X_{\min}$ , the minimum value of TIE( $k;n$ )  $\{k = 1, 2, \dots, 6 \times 10^6\}$ , is  $\geq -7.0\sigma$  with a probability of 99.99%. Therefore,  $X_{\text{abs}}$ , the maximum of the absolute value of TIE( $k;n$ ), becomes  $\leq 7\sigma$  with a probability of 99.9% obtained by multiplying the probability for  $X_{\max} \leq 7.0\sigma$  and the probability for  $X_{\min} \geq -7.0\sigma$ . When  $M$  is less than  $6 \times 10^6$ , the probability of  $X_{\text{abs}} \leq 7.0\sigma$  increases (Figure 52 also shows the cases  $M = 6000$ ,  $6 \times 10^4$ , and  $6 \times 10^5$ ).



**Figure 52 – Probability density function of maximum value for Gaussian noise**

As a result, the right side of equation (8-21) is expressed by the equation (8-22) using TIErms( $n\tau_0$ ), which is equal to the standard deviation  $\sigma$  of samples of TIE( $k;n$ ).

$$\max_{1 \leq k \leq N-n} |\text{TIE}(k;n)| \leq 7 \times \text{TIErms}(n\tau_0) \quad (\text{with 99.9\% probability}) \quad (8-22)$$

Equation (8-22) is used to define an upper limit for MTIE, i.e., the value that is exceeded with 0.1% probability

$$\text{MTIE}(n\tau_0) \leq 7 \times \text{TIErms}(n\tau_0). \quad (8-23)$$

Note that equation (8-23) is based on the approximations described above (e.g., equation (8-20) and equation (8-21)). In [b-Anritsu], several examples are given where the actual MTIE, computed from the phase time history, exceeds the right-hand side of equation (8-23). These include cases where the distribution of the wander is not Gaussian, and cases where an impulse phase transient is present.

#### 8.2.4.2 Relationship between TDEV and MTIE upper limit

An approximate relation between TDEV and power spectral density is given by equations (8-17) and (8-18) of clause 8.2.2.3

$$S_x(f) \cong \frac{K_1}{f} \left( \text{TDEV} \left( \frac{K_2}{f} \right) \right)^2, \quad (8-24)$$

where  $K_1$  and  $K_2$  are constants (and  $K_1 = 0.75$ ,  $K_2 = 0.3$ ). The relation between TIErms and the phase error power spectral density,  $S_\phi(f)$ , is (see [ITU-T G.810])

$$\text{TIErms}(\tau) = \sqrt{\frac{1}{(\pi v_{nom})^2} \int_0^{f_h} S_\phi(f) \sin^2(\pi\tau f) df}, \quad (8-25)$$

where,  $v_{nom}$  is the nominal frequency of the wandered clock, and  $f_h$  is the measurement system bandwidth. Substituting equation (8-24) into equation (8-25), using  $S_\phi(f) = (2\pi v_{nom})^2 \times S_x(f)$ , produces

$$\text{TIErms}(\tau) \cong \overline{\text{TIErms}(\tau)} \cong \sqrt{4K_1 \int_{\tau_{min}/K_2}^{\tau_{max}/K_2} \left( \text{TDEV} \left( \frac{K_2}{f} \right) \right)^2 \frac{\sin^2(\pi\tau f)}{f} df}, \quad (8-26)$$

where,  $\tau_{min}$  and  $\tau_{max}$  are the smallest and the largest observation interval specified for the TDEV. When a TDEV mask specified in [ITU-T G.812], [ITU-T G.813], or [ITU-T G.8262] is substituted for TDEV( $K_2/f$ ) on the right side of equation(8-26),  $\overline{\text{TIErms}(\tau)}$  represents the approximate TIErms

for the ideal TDEV wander noise. Appendix I of [ITU-T G.812], and clause 8.2.2.3 above, uses  $K_1 = 0.75$  and  $K_2 = 0.3$  in equation (8-26). However, when these values are used in equation (8-26), the calculated  $\overline{\text{TIErms}(\tau)}$  is greatly attenuated at  $\tau = \tau_{\min}$ , the smallest observation interval specified by the TDEV mask. Therefore, the values  $K_1 = 0.84$  and  $K_2 = 0.42$  are used instead (the original values of  $K_1$  and  $K_2$ ) in equation (8-26).

By using  $\overline{\text{TIErms}(\tau)}$  above, the MTIE upper limit for the ideal TDEV wander noise can be expressed as:

$$\text{MTIE}(\tau) \leq 7 \times \overline{\text{TIErms}(\tau)}. \quad (8-27)$$

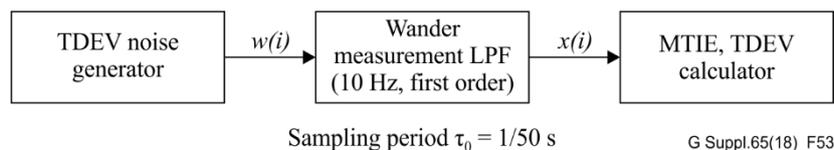
Combining Eqs. (8-26) and (8-27) produces

$$\text{MTIE}(\tau) \leq 7 \cdot \sqrt{4K_1 \int_{K_2/\tau_{\max}}^{K_2/\tau_{\min}} \left( \text{TDEV} \left( \frac{K_2}{f} \right) \right)^2 \frac{\sin^2(\pi f \tau)}{f} df}. \quad (8-28)$$

Note that, while the integral in equation (8-25) is from 0 to the measurement system bandwidth  $f_h$ , the integral in equations (8-26) and (8-28) is from  $K_2/\tau_{\max}$  to  $K_2/\tau_{\min}$ . This is because in practice, the TDEV (TVAR) function is specified over a range of observation intervals, and there are ranges of frequency where the power-spectrum is not specified. Since these frequency ranges correspond to jitter frequencies, which are removed by a low-pass filter, they do not contribute significantly to the estimate of the upper bound on MTIE.

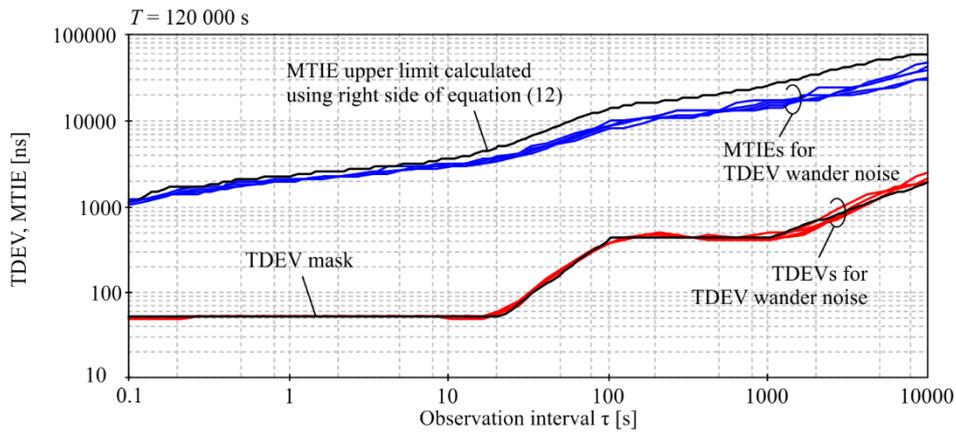
### 8.2.4.3 Verification of MTIE upper limit by computer simulation

This clause gives several examples of the comparison of the upper limit of MTIE computed using Eqs. (8-26) and (8-27) with MTIE computed directly from the phase time history. For each example, a signal whose TDEV meets a respective mask is simulated using the process shown in Figure 53. The TDEV noise generator outputs the sequence  $w(i)$ ,  $i = 1, 2, 3, \dots, N$ , corresponding to the respective TDEV mask specified in ITU-T G.812 or ITU-T G.813, with sampling period  $\tau_0 = 1/50$  s. The time error function  $x(i)$ ,  $i = 1, 2, 3, \dots, N$ , is obtained from the sequence output from the wander measurement LPF and used to calculate MTIE and TDEV. The upper limit of MTIE is computed using equations (8-26) and (8-27), and compared with MTIE obtained for the  $x(i)$  sequence.

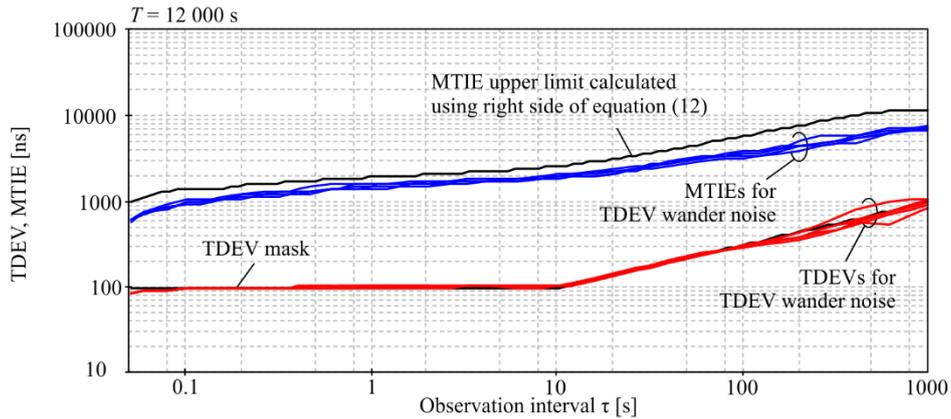


**Figure 53 – Block diagram for verifying MTIE for TDEV wander noise**

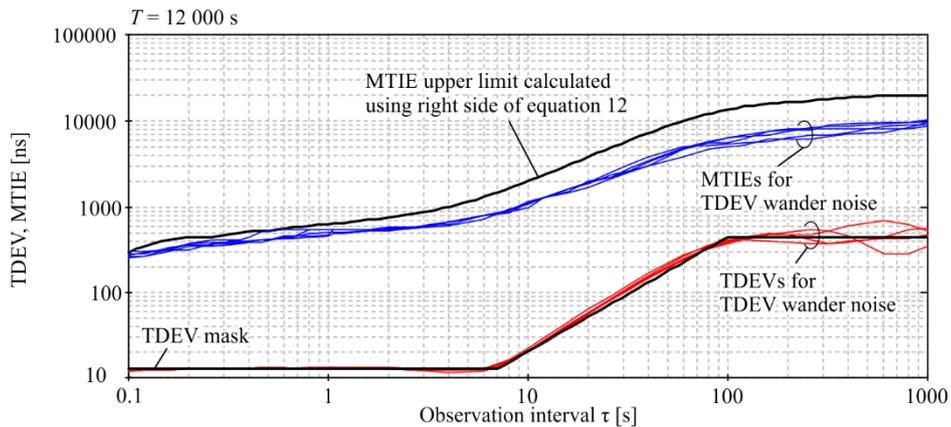
Figure 54 shows the simulation results for the TDEV and MTIE for each TDEV wander noise case. Each TDEV wander noise was generated five times for different initial values, and the TDEVs and MTIEs were calculated using the estimator formulas defined in [ITU-T G.810]. The MTIE upper limit calculated using equations (8-26) and (8-27) is also shown in each figure, where each TDEV mask was substituted as the ideal TDEV for  $\text{TDEV}(K_2/f)$  in equation (8-26). All MTIE curves for each TDEV wander noise are below the MTIE upper limit.



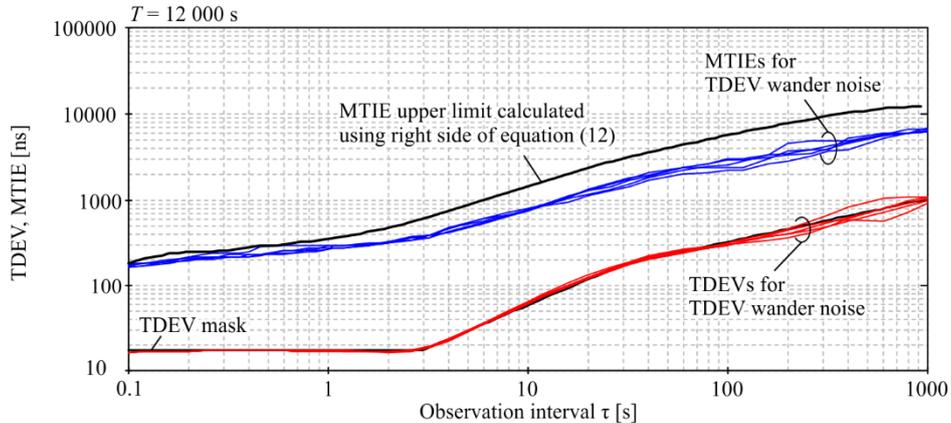
a) TDEV wander noise for Table 11 of [ITU-T G.812]



b) TDEV wander noise for Table 12 (and Table A.9) of [ITU-T G.812]



c) TDEV wander noise for Table 9 of [ITU-T G.813]

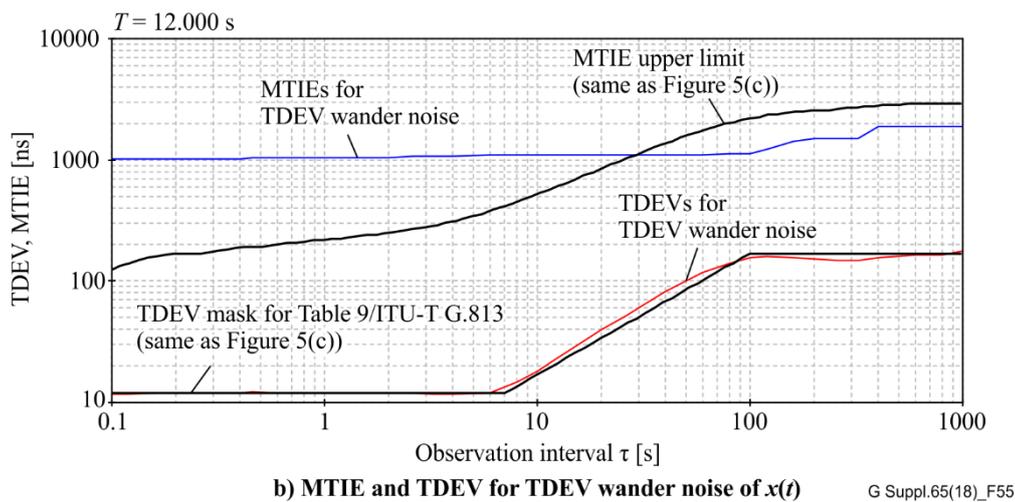
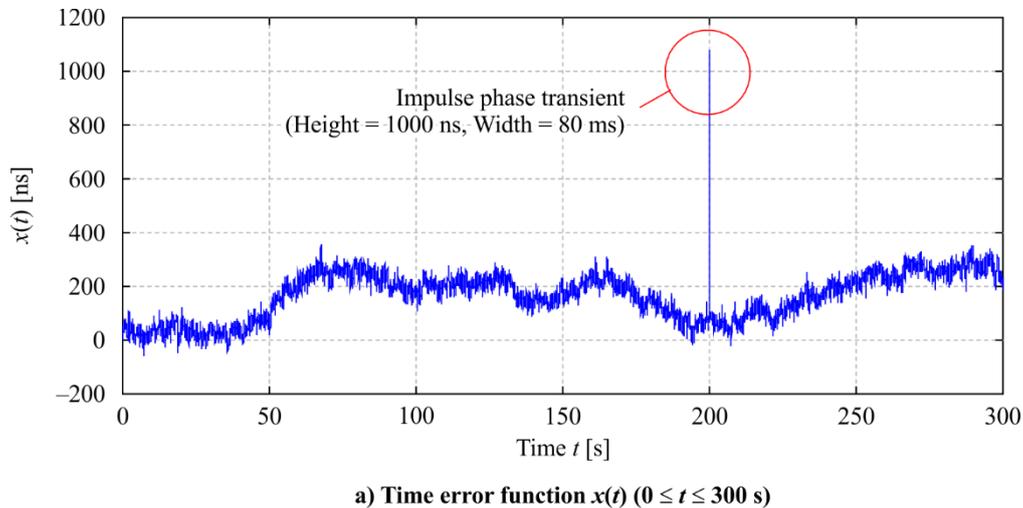


d) TDEV wander noise for Table 11 of [ITU-T G.813]

G Suppl.65(18)\_F54

**Figure 54 – MTIE and TDEV for TDEV Wander noise**

The noise waveform in Figure 55(a) has TDEV at the level of Option 1 SEC wander tolerance specified in Table 9 of [ITU-T G.813] and Figure 6 of [ITU-T G.813], with an intentionally added impulse transient at  $t = 200$  s. TDEV and MTIE for this noise are shown in Figure 55(b), along with the MTIE upper limit given by the right-hand side of equation (8-28). Although the TDEV for the waveform matches the TDEV mask, the MTIE greatly exceeds the MTIE upper limit for observation intervals less than 30 s, due to its sensitivity to the impulse transient. This is because, with the addition of the impulse, equation (8-20) is not a good approximation to equation (8-19).



**Figure 55 – MTIE and TDEV for TDEV Wander noise with impulse transient added (Original TDEV Wander noise model is based on Table 9 of [ITU-T G.813])**

#### 8.2.4.4 Conclusions

The results in Figure 5 verify that MTIE results for the TDEV wander noises are below the MTIE upper limit calculated by the right side of equation (8-27). However, the MTIE for wander noise that is not white or whose distribution is not Gaussian, or for which the approximation of equation (8-20) is not valid (e.g., if an impulse transient is present), may exceed the MTIE upper limit.

#### 8.2.5 Wander accumulation results for frequency domain model

The simulation results in the following clauses are for the HRM described in clause 8.1.1, which is based on ITU-T G.803 reference chain. The results are obtained using the frequency domain model of clauses 8.2.2, 8.2.3 and 8.2.4. In each filter, damping ratio is obtained from gain peaking using the approximation given in equation (8-16) of clause 8.2.3.

### 8.2.5.1 Wander accumulation TDEV results for frequency domain model

Figure 56 to 59 give the results for the TDEV of the HRM described in clause 8.1.1. Figures 56 and 57 give the results for Option 1 SECs and Type I SSUs, and Figures 58 and 59 give the results for Option 2 SECs and Type II and III SSUs. In the first of each pair of figures, the TDEV is shown for the output of the PRC, the output after the 8<sup>th</sup> SSUs, the output after the first SEC after the 8<sup>th</sup> SSU, and the output after the 20<sup>th</sup> SEC after the 8<sup>th</sup> SSU. In the second of each pair of figures, the TDEV is shown for the output of the PRC, the output of the 9<sup>th</sup> and 10<sup>th</sup> SSUs, and the output of SECs 21, 40, 41 and 58 (the latter is the last SEC in the chain). In addition, the appropriate TDEV mask for the SEC interface is shown in all the figures. Option 1 results assume that the SEC and SSU noise generation is at the level of the respective Option 1 TDEV mask, i.e., Table 3 and Figure 2 of clause 8.1.2.1 for the SEC, and Table 13 and Figure 14 of clause 8.1.2.2 for the SSU. The ETSI model TDEV masks (see clause 8.1.4.1) are not used.

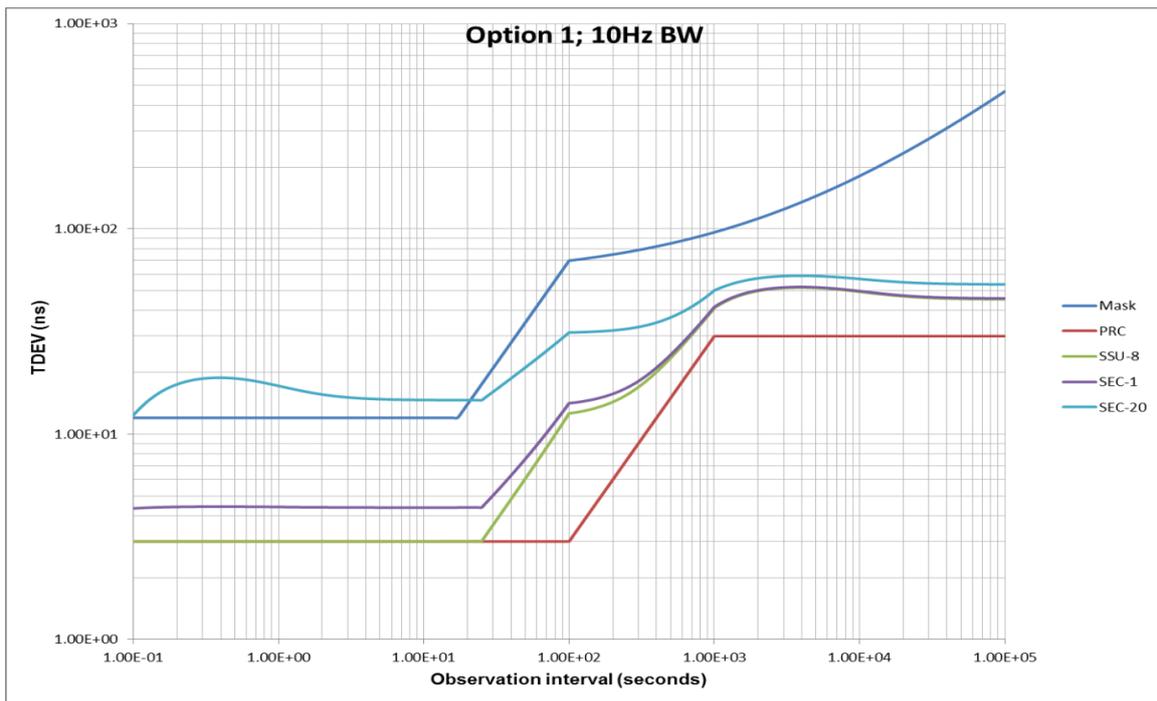
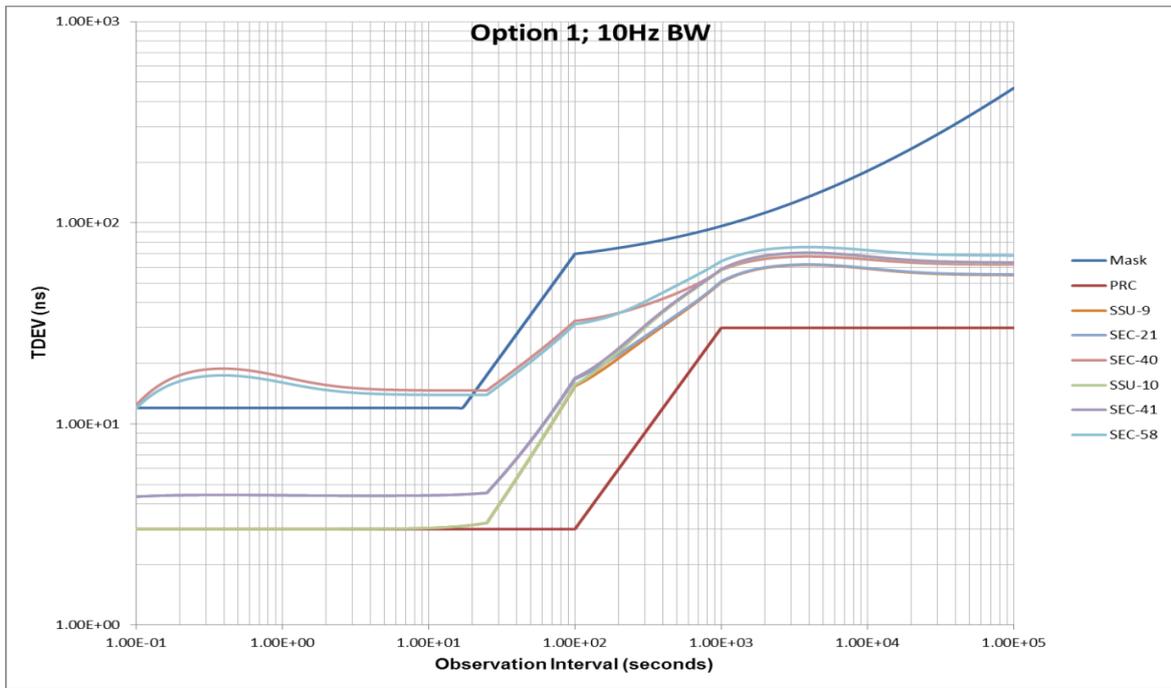
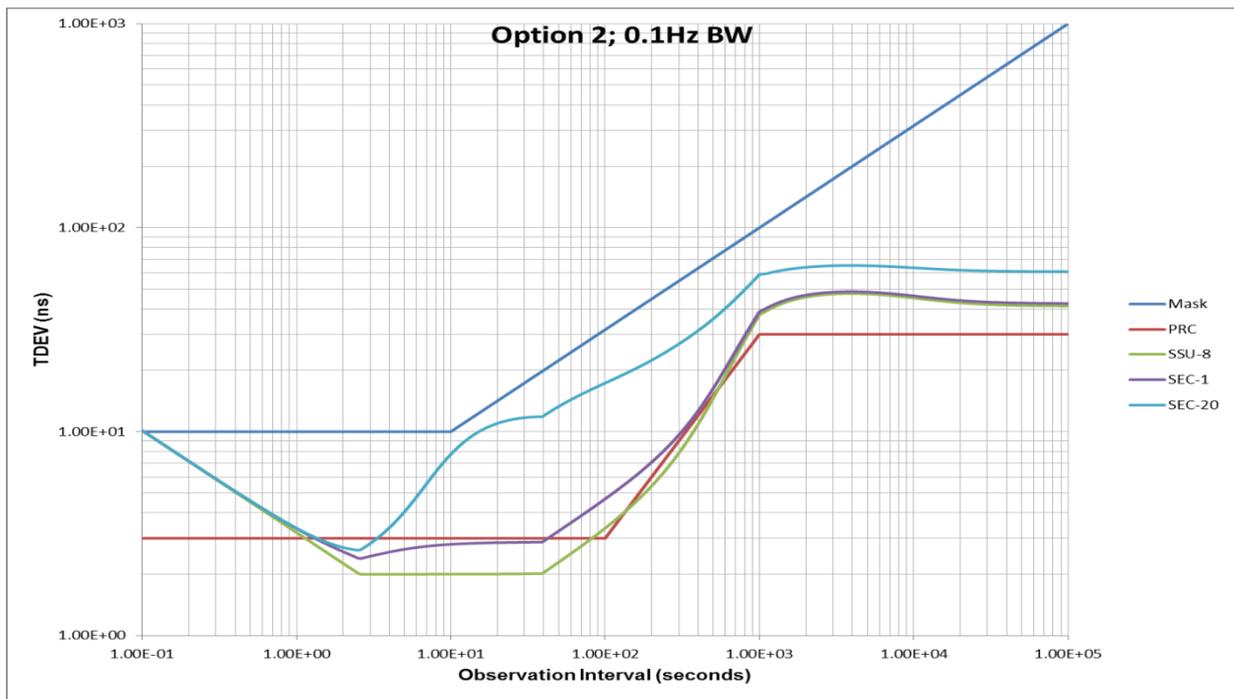


Figure 56 – Option 1 TDEV results for PRC and selected SECs and SSUs



**Figure 57 – Option 1 TDEV results for PRC and selected SECs and SSUs**



**Figure 58 – Option 2 TDEV results for PRC and selected SECs and SSUs**



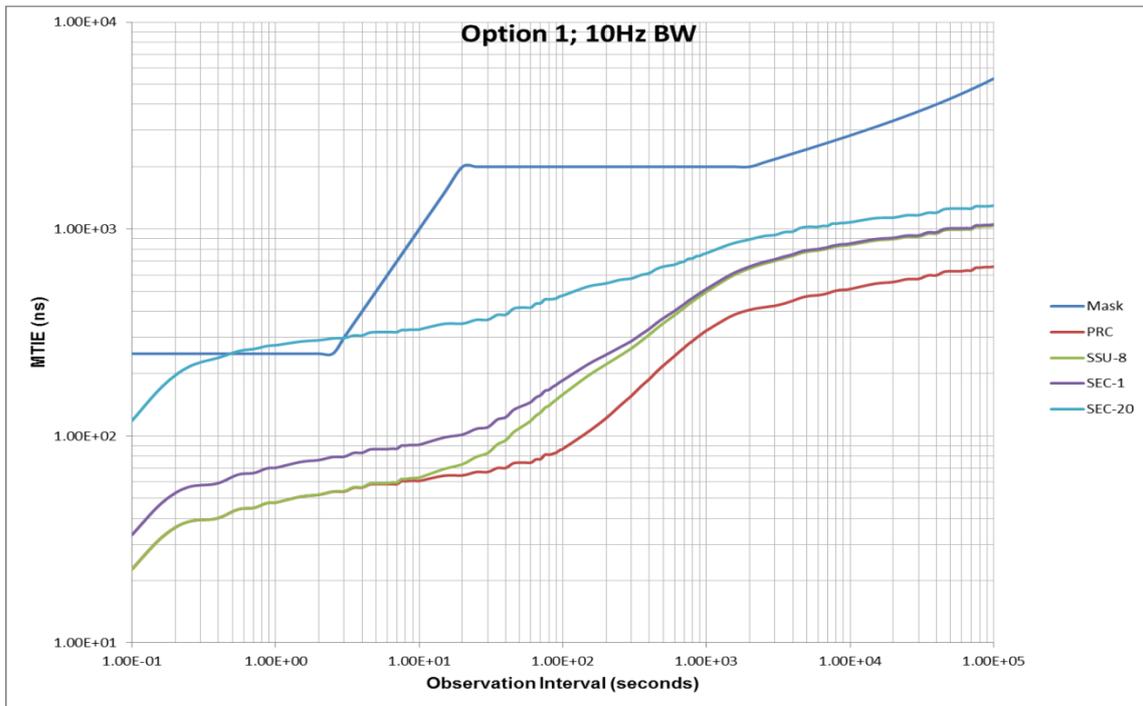
**Figure 59 – Option 2 TDEV results for PRC and selected SECs and SSUs**

The above frequency-domain results may be compared with the time-domain simulation results of clause 8.1. The comparison should be made with the time-domain results that used a noise generation model that was at the level of the respective TDEV mask. In the case of Option 1, the appropriate results are for Option 1, model 2 SEC noise generation, i.e., where the TDEV mask is met but the noise is below the MTIE mask (because the frequency-domain results here assume the noise is at the level of the TDEV mask). In the case of Option 2, the appropriate results are for Option 2, model 1 SEC noise generation, i.e., where the TDEV mask is met but the noise is above the MTIE mask (because the frequency-domain results here assume the noise is at the level of the TDEV mask). The respective time domain results are given in Figures 30 and 31 for Option 1, and in Figures 34 and 35 for Option 2.

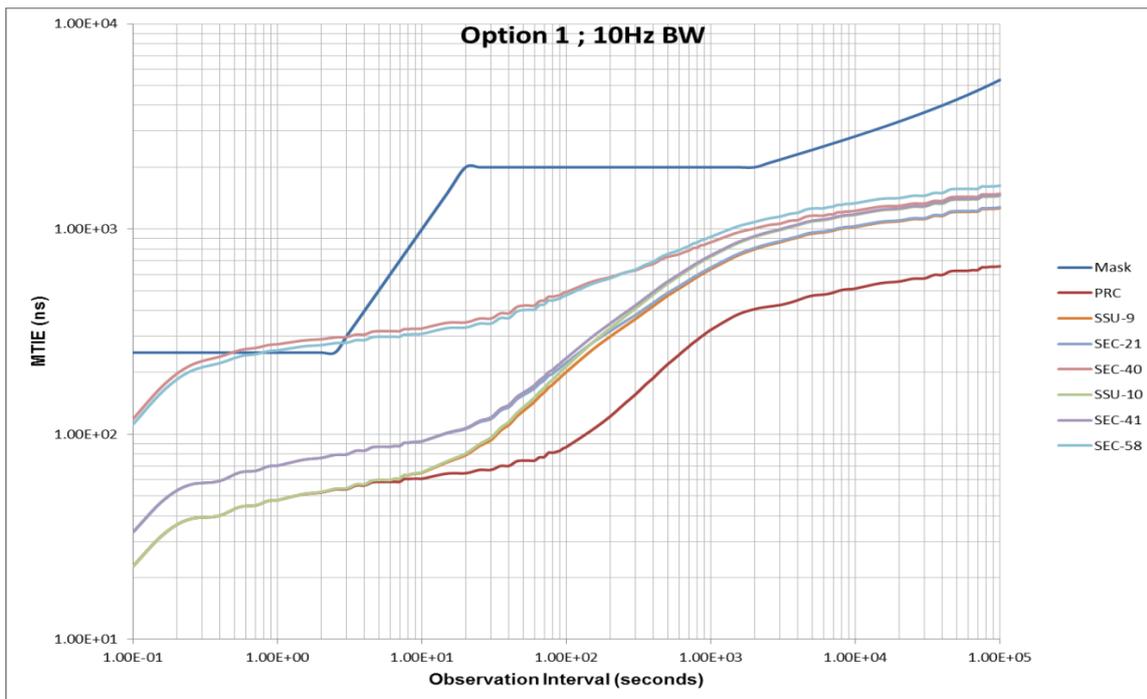
Comparing the results of the frequency domain analytical model and time-domain simulations, it is clear that there is a very high correlation, i.e., agreement.

### 8.2.5.2 Wander accumulation MTIE results for frequency domain model

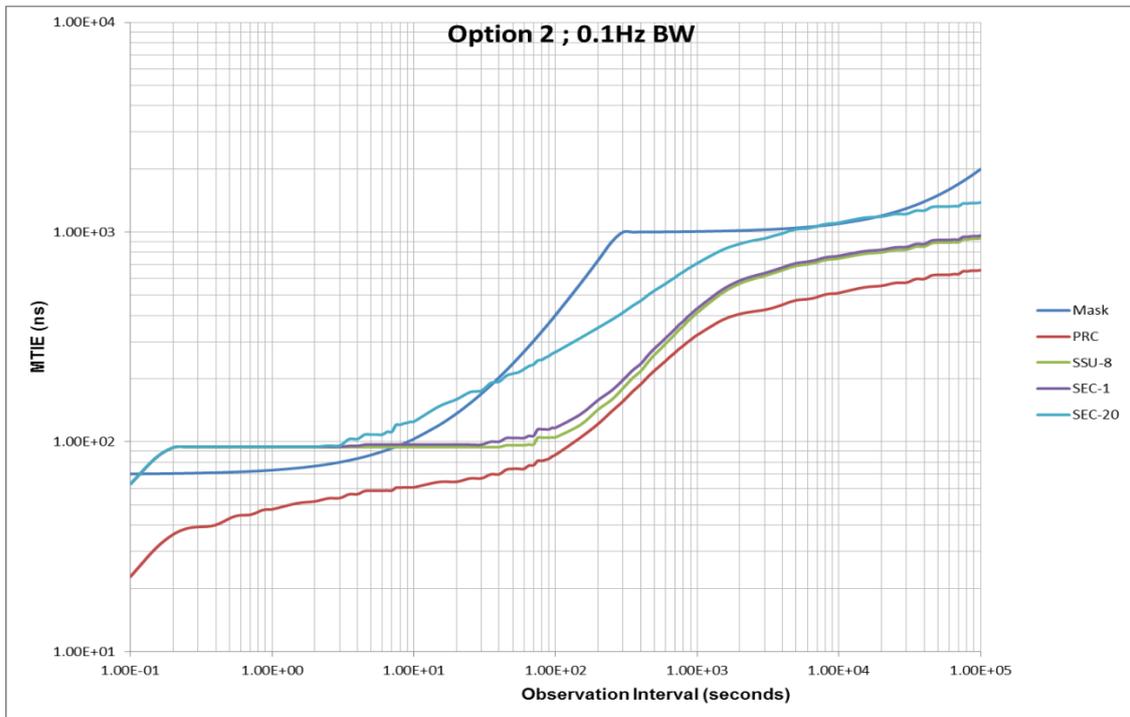
Figures 60 to 63 give the results for the MTIE of the HRM described in clause 8.1.1. Figures 60 and 61 give the results for Option 1 SECs and Type I SSUs, and Figures 62 and 63 give the results for Option 2 SECs and Type II and III SSUs. In the first of each pair of figures, the MTIE is shown for the output of the PRC, the output after the 8<sup>th</sup> SSUs, the output after the first SEC after the 8<sup>th</sup> SSU, and the output after the 20<sup>th</sup> SEC after the 8<sup>th</sup> SSU. In the second of each pair of figures, the MTIE is shown for the output of the PRC, the output of the 9<sup>th</sup> and 10<sup>th</sup> SSUs, and the output of SECs 21, 40, 41 and 58 (the latter is the last SEC in the chain). In addition, the appropriate MTIE mask for the SEC interface is shown in all the figures. As for the TDEV results in the previous clause, Option 1 results assume that the SEC and SSU noise generation is at the level of the respective Option 1 TDEV mask, i.e., Table 3 and Figure 2 of clause 8.1.2.1 for the SEC, and Table 13 and Figure 14 of clause 8.1.2.2 for the SSU. The ETSI model TDEV masks (see clause 8.1.4.1) are not used.



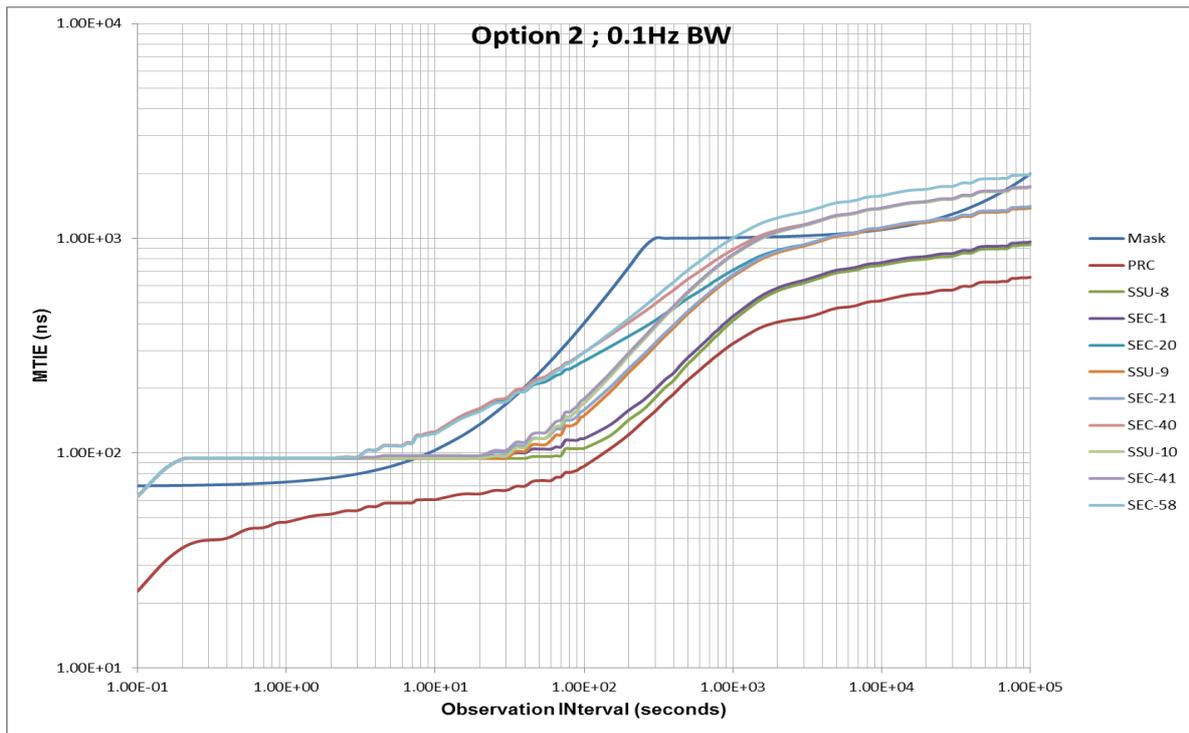
**Figure 60 – Option 1 MTIE results for PRC and selected SECs and SSUs**



**Figure 61 – Option 1 MTIE results for PRC and selected SECs and SSUs**



**Figure 62 – Option 2 MTIE results for PRC and selected SECs and SSUs**



**Figure 63 – Option 2 MTIE results for PRC and selected SECs and SSUs**

The above frequency-domain results may be compared with the time-domain simulation results of clause 8.1. As for the TDEV results in the previous clause, the comparison should be made with the time-domain results that used a noise generation model that was at the level of the respective TDEV mask. In the case of Option 1, the appropriate results are for Option 1, model 2 SEC noise generation, i.e., where the TDEV mask is met but the noise is below the MTIE mask (because the frequency-domain results here assume the noise is at the level of the TDEV mask). In the case of Option 2, the appropriate results are for Option 2, model 1 SEC noise generation, i.e., where the TDEV mask is

met but the noise is above the MTIE mask (because the frequency-domain results here assume the noise is at the level of the TDEV mask). The respective time domain results are given in Figures 30 and 31 for Option 1, and in Figures 34 and 35 for Option 2.

Comparing the results of the frequency domain analytical model and the time-domain simulations, it is clear that there is a very high correlation, i.e. agreement, though in some cases the frequency-domain model results are above the time-domain model results (e.g., SECs 20, 40 and 58 for both Option 1 and Option 2, for observation intervals between 0.1 s and 10 s). This is reasonable, given that the frequency-domain model results for MTIE, equation (8-28), is actually an upper bound. Note also that this level of agreement has occurred even though a number of approximations were used in deriving equation (8-28).

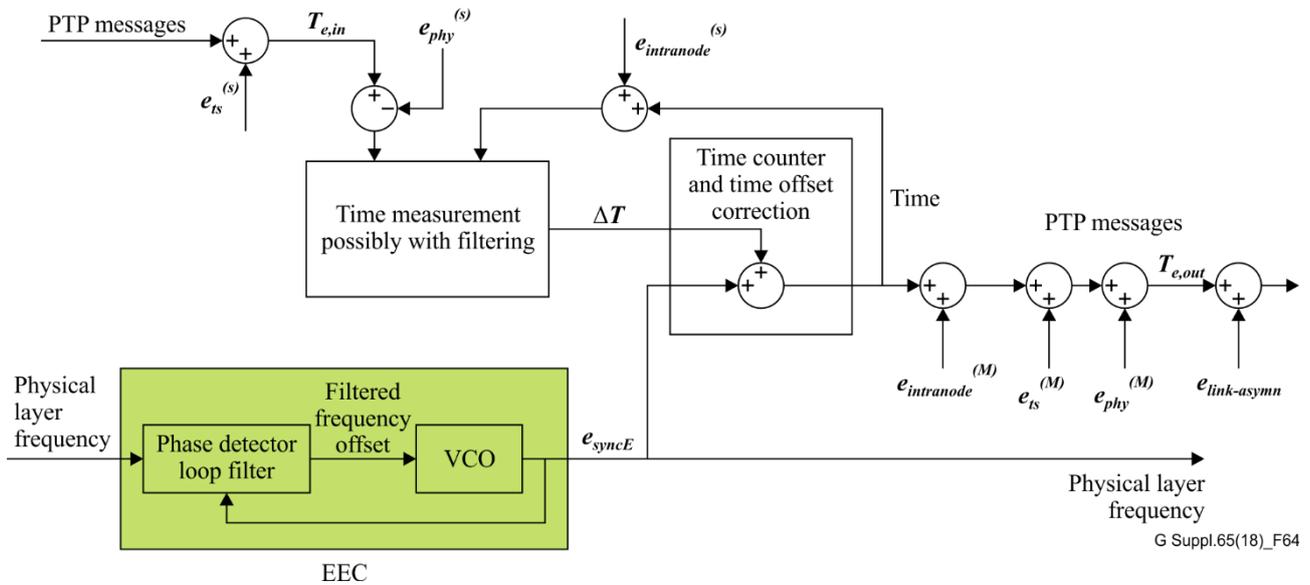
## **9 Time-domain mathematical description of a telecom boundary clock (T-BC) and a telecom time slave clock (T-TSC) with SyncE assist for frequency transport, and associated transfer functions and frequency responses**

### **9.1 Introduction**

This clause describes models for simulating, in the time domain, the transport of time using PTP and frequency using SyncE. Clause 9.1.1 describes, at a high level, the model for a telecom boundary clock (T-BC) and a telecom time slave clock (T-TSC). Much of the description in clause 9.1.1 is taken from Appendix I.1 of [ITU-T G.8271.1]. Clause 9.1.2 describes the measurement of the mean path delay, and the effect of the turnaround time on the measurement (*turnaround time* is defined in clause 9.1.3). Clause 9.1.3 describes the effect of sojourn time on T-BC performance (*sojourn time* is defined in clause 9.1.2). Clauses 9.2 and 9.3 present detailed mathematical models for a T-BC and T-TSC (clause 9.2 presents models that do not include noise generation, and clause 9.3 presents models that include noise generation), along with transfer functions and frequency responses.

#### **9.1.1 High-level model for a telecom boundary clock**

Figure 64 illustrates a telecom boundary clock model for simulating the transport of time using PTP with SyncE assistance. This figure is a copy of Figure I.1 of [ITU-T G.8271.1]. The EEC block represents an Ethernet equipment clock, as specified in [ITU-T G.8262]. The EEC input is a physical layer frequency (i.e., a physical layer signal that is used as a frequency reference), and its output is a local frequency (i.e., a physical layer signal that has a frequency and is local to this node) that is optionally propagated to downstream nodes. The noise process  $e_{syncE}$  represents the SyncE phase noise accumulation in the SyncE HRM (see clause 8).



**Figure 64 – Telecom boundary clock model for simulating the transport of time using PTP with SyncE assistance**

The time counter is incremented by the nominal period of the output clock of the EEC block. For example, if the output clock rate is 125 MHz, then the time counter is incremented by 8 ns with each rising edge of the SyncE output clock. Upon reception and transmission of PTP event messages, the time counter is sampled. The difference between the actual transmission/reception time and the sampled value of the time counter is modelled as  $e_{ts}^{(S)}$  since the transmission/reception event can happen between two rising edges of this clock. The effect of  $e_{ts}^{(S)}$  on the timestamp for reception of a PTP event message is shown added as at the input, and the effect of  $e_{ts}^{(S)}$  on the timestamp for transmission of a PTP event message is shown as added at the output.

The incoming PTP messages contain information that may be used to obtain an estimate of the grandmaster (i.e., PRTC) time. This estimate is not perfect. It contains errors introduced at the grandmaster, the upstream nodes, and upstream links. The error in the incoming estimate of the grandmaster time is represented by  $T_{e,in}$ . The noise process  $e_{phy}^{(S)}$  represents the effect of asymmetry and timestamp sampling uncertainty on the physical layer (PHY) of the input port. The PHY latency asymmetry may be present if timestamping is done at a point other than the reference plane (i.e., the interface between the PHY and the physical medium). Any latency between the point where timestamping actually is done and the reference plane may be compensated for within PTP. However, any uncompensated latencies that result in asymmetry will contribute to  $e_{phy}^{(S)}$ . The noise  $e_{phy}^{(S)}$  is subtracted from the timing information contained in the incoming PTP messages due to the direction of the time distribution (note, that on the master port of the T-BC it is added). Note that the random process  $e_{phy}^{(S)}$  may have a static component and a time-varying component.

The timing information contained in the incoming PTP messages, with the noise due to asymmetry on the input port PHY,  $e_{phy}^{(S)}$ , and the timestamping error,  $e_{ts}^{(S)}$ , is an input to the block labelled time measurement, possibly with filtering. This block compares the local time output of the local clock, which is the accumulation of the syncE phase noise,  $e_{syncE}$ , and the prior time offset correction,  $\Delta T$ , with the timing input that represents an estimate of the grandmaster time (with errors as described in the previous paragraph). This block produces the time offset correction,  $\Delta T$ , between the grandmaster time estimate and the local time. The time measurement block might provide filtering when computing the time offset correction, to reduce the effect of the short-term noise in the observed time error. The filtering characteristics are for further study.

The time counter and time offset correction block produces a local time output (i.e., the output labelled "Time"). The input to the time counter and time offset correction block is the output of the

EEC and the time offset correction of the time measurement block. The counter and time offset correction block may include a low-pass filtering function. This has the same effect as increasing the output frequency of the EEC block.

The local time is sampled upon transmission and reception of PTP event messages on master ports. The sampled value is the accumulation of the SyncE phase noise,  $e_{syncE}$ , the timestamp errors on transmission and reception,  $e_{ts}^{(M)}$  and  $e_{ts}^{(S)}$  respectively, and the offset correction,  $\Delta T$ . The error due to asymmetry of the PHY on the output port,  $e_{phy}^{(S)}$ , is added to the sampled local time to produce the master port output time error,  $T_{e,out}$ . The quantity  $T_{e,out}$  is input to the next PTP node (T-BC or T-TSC) downstream via a link model.

Errors due to intranode transmission,  $e_{intranode}$ , and link asymmetry,  $e_{link-asymm}$ , must also be included. The former affects both the time correction and the T-BC output. The latter is shown as added to the output of the T-BC.

If the blocks to the right of the output of the time counter and time offset correction block are removed, Figure 64 becomes a model for a time telecom slave clock (T-TSC). The time output of the time counter and time offset correction block is the output of the T-TSC.

The following clauses describe more detailed mathematical models for the T-BC of Figure 64. Two cases are considered:

- A) Time-stamping done relative to the corrected time
- B) Time-stamping done relative to the local, uncorrected time.

The mathematical description of the models is contained in the transfer functions. For each of the cases (A) and (B), block diagrams are presented that are more detailed than Figure 64, and contain sufficient detail to allow the transfer function to be derived. The cases are referred to as 'A' and 'B' in the remainder of this clause. For simplicity, models that neglect noise generation in the various filters are presented first. Then, noise generation is added to the models.

### 9.1.2 Effect of delay request/response or peer delay turnaround time on delay measurement and boundary clock performance

#### 9.1.2.1 Description of delay request/response and peer-to-peer delay mechanisms

Much of the discussion of this clause, and Figures 65 and 66, are taken or adapted from [IEEE 1588] and [IEEE 802.1AS]. The mean propagation delay on a link is measured using either the peer-to-peer delay mechanism or the delay request/response mechanism ([IEEE 1588] which specifies that one or the other mechanism is used on a single PTP communication path, i.e., the mechanisms do not mix). The former is used in [ITU-T G.8275.1]. Note that the material in this clause was developed before it was decided that the PTP profile in [ITU-T G.8275.1] would use the delay request/response mechanism).

The peer-to-peer delay mechanism is illustrated in Figure 65. The figure shows the measurement initiated by a port at one end of a link. This port is referred to as the peer delay initiator. The port at the other end of the link is the peer delay responder. When the message exchange is completed, the peer delay initiator can compute the propagation delay. Note that, on an actual link the measurement, and therefore the message exchange, occurs in both directions independently (in addition, the measurements are made on all links that use the peer-to-peer delay mechanism, whether or not they are blocked by non-PTP protocols (e.g., rapid spanning tree protocol (RSTP) [IEEE 802.1Q]). This enables propagation delay to be known to both endpoints of the link. The peer-to-peer delay mechanism is limited to point-to-point links, because the mechanism does not provide for a peer delay initiator to keep track of responses from more than one peer delay responder.

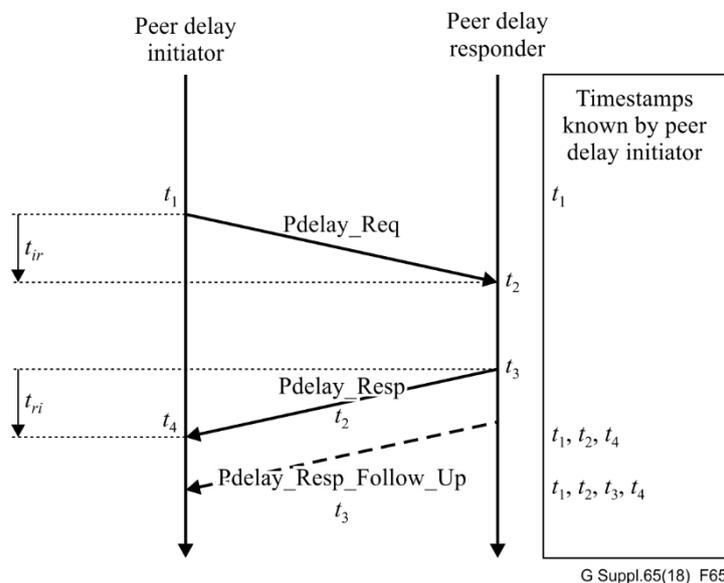
In the case of a two-step clock, the peer delay initiator sends a Pdelay\_Req message and generates a timestamp  $t_1$ . The peer delay responder receives the message and timestamps it with  $t_2$ . At a later time, the peer delay responder sends a Pdelay\_Resp message and timestamps it with  $t_3$ . The peer delay

responder inserts the timestamp  $t_2$  in the Pdelay\_Resp message. At a later time, the peer delay responder sends a Pdelay\_Resp\_Follow\_Up message that carries the value  $t_3$ . The Pdelay\_Resp\_Follow\_Up message is shown as a dashed line in Figure 65 because it is present only in the case of a two-step clock. The peer delay initiator receives the Pdelay\_Resp message and timestamps it with  $t_4$ . When the peer delay initiator receives the Pdelay\_Resp\_Follow\_Up message, it knows all four timestamps:  $t_1$ ,  $t_2$ ,  $t_3$ , and  $t_4$ . It computes the mean propagation delay  $D$  as

$$D = \frac{(t_4 - t_1) - (t_3 - t_2)}{2}. \quad (9-1)$$

Equation (9-1) gives the mean propagation delay, i.e., the sum of the delays in the two directions on the divided by 2. Any delay asymmetry that is present must be measured separately. However, delay asymmetry is outside the scope of this Supplement, and will not be discussed further.

In the case of a one-step clock, the Pdelay\_Resp\_Follow\_Up message is not present. The peer delay responder, on sending the Pdelay\_Resp message, computes the difference  $t_3 - t_2$  and inserts this value in the Pdelay\_Resp message. When the peer delay requester receives the Pdelay\_Resp message, it knows  $t_1$ ,  $t_3 - t_2$ , and  $t_4$ , and may apply equation (9-1).<sup>4</sup>

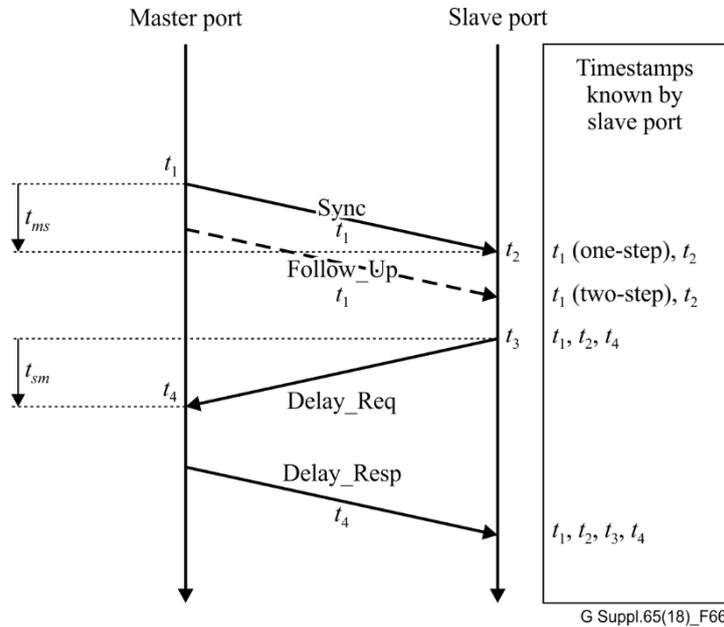


**Figure 65 – Illustration of propagation delay measurement using peer-to-peer delay mechanism**

The delay request/response mechanism is illustrated in Figure 66. The mechanism operates between a PTP master port and each PTP slave port on the communication path (for simplicity, only one slave port is shown in Figure 66). Unlike the peer-to-peer mechanism, the delay request/response mechanism is not limited to point-to-point links. However, the message flows occur such that, when the message exchange is completed, the mean propagation delay is known only to the slave ports. The mean propagation delay is not known to master or passive ports. This means that, if there is a network reconfiguration or grandmaster change that causes the best master clock algorithm to create a new synchronization hierarchy, the propagation delays on some links may have to be measured, and the resulting reconfiguration time may be longer.

<sup>4</sup> In the case of a two-step clock, the peer-to-peer responder may optionally return  $t_3 - t_2$  in the Pdelay\_Resp\_Follow\_Up message instead of returning  $t_2$  in the Pdelay\_Resp message and  $t_3$  in the Pdelay\_Resp\_Follow\_Up message. The details of this, as well as the details of the message fields that carry the timestamp values, are given in clause 11.4 of [IEEE 1588].

In the case of a two-step clock, the master port sends a Sync message on the PTP communication path, and generates a timestamp  $t_1$ . A slave port on the PTP communication path receives the Sync message and timestamps it with  $t_2$ . At a later time, the master port sends a Follow\_Up message that carries the value  $t_1$ . At a time not earlier than  $t_2$ , the slave port sends a Delay\_Req message to the master port, and timestamps it with  $t_3$ . The master port receives the Delay\_Req message at time  $t_4$ , and timestamps it with this value. At a later time, the master port returns the value  $t_4$  to the slave port in a Delay\_Resp message. When the slave port receives the Delay\_Resp message, it knows all four timestamps:  $t_1$ ,  $t_2$ ,  $t_3$ , and  $t_4$ . It computes the mean propagation delay  $D$  using equation (9-1)



**Figure 66 – Illustration of propagation delay measurement using delay request/response mechanism**

In the case of a one-step clock, the Follow\_Up message is not sent. The master port, on sending the Sync message, inserts the timestamp value  $t_1$  in the Sync message as it is transmitted. However, when the message exchange is completed and the slave port has received the Delay\_Resp message, it knows all four timestamps and may compute mean propagation delay using equation (9-1).

### 9.1.2.2 Description of errors in the measurement of mean propagation delay

Two types of errors, referred to as type A and type B, respectively, can arise in the measurement of mean propagation delay.<sup>5</sup>

#### 9.1.2.2.1 Type A errors

Type A errors are due to imperfect knowledge of the frequencies used for timestamping. In Figures 65 and 66, the time interval  $t_3 - t_2$  is the turnaround time, i.e., the time between the receipt of a PTP event message and the sending of a PTP event message (with both messages being used in the mean propagation delay measurement). In equation (9-1), the mean propagation delay is obtained by first subtracting the turnaround time from the time interval between the sending of an event message from the initiator or master and the receipt of an event message by the initiator or master. The result of this subtraction gives the sum of the delays in the two directions. The mean delay is obtained by dividing this result by 2. Since ideally it is desired to measure the mean propagation delay relative to the time

<sup>5</sup> In addition, there may be delay asymmetry, which results in a difference between mean propagation delay and actual propagation delay in each direction. However, as indicated earlier, delay asymmetry is outside the scope of this Supplement.

base of the T-GM, the timestamps  $t_1$ ,  $t_2$ ,  $t_3$ , and  $t_4$  should be taken relative to the T-GM frequency. However, the exact T-GM frequency is not available at the two ends of the link. In practice, the frequency used for timestamping differs from the exact T-GM frequency by some frequency offset  $y$  (in this and subsequent clauses, frequency offset is a pure fraction). The value of  $y$  may be larger or smaller depending on the way the frequency is transferred and on the clocks that are used for timestamp. For example, if frequency is transferred using syncE and is traceable to a primary reference clock (PRC), and if the T-GM is traceable to a primary reference, then  $y$  is on the order of  $10^{-11}$  or better. If frequency is transferred using PTP, then  $y$  depends on the method and algorithm used to do the transfer.<sup>6</sup> If free-running local clocks are used for timestamping, then  $y$  at each end of the link may be relatively large and may need to be measured separately to correct for the resulting error described below.

In describing the error due to the frequencies at the endpoints differing from the T-GM frequency, the error may be divided into two contributions:

- 1) errors due to the frequencies at the endpoints differing from each other, and
- 2) errors due to one of the two endpoint frequencies differing from the T-GM frequency.

It may seem at first that dividing the error into these two components adds unnecessary complication to the description. However, it turns out that in most practical situations the error, when it is appreciable, is due mainly to (1) and the effect of (2) is negligible. This will be explained shortly.

In applying equation (9-1), it is assumed that the slave or responder frequency differs from the master or initiator frequency by  $y_1$ , and the master or initiator frequency differs from the T-GM frequency by  $y_2$ . The measured mean propagation delay is given in equation (9-1). The actual mean propagation delay,  $D_{actual}$ , is given in equation (9-1), but with the time intervals  $t_4 - t_1$  and  $t_3 - t_2$  corrected so they are referred to the T-GM time base.  $D_{actual}$  is given by

$$\begin{aligned} D_{actual} &= \frac{(t_4 - t_1)(1 + y_2) - (t_3 - t_2)(1 + y_1 + y_2)}{2} \\ &= \frac{(t_4 - t_1) - (t_3 - t_2)}{2} (1 + y_2) - 0.5(t_3 - t_2)y_1. \\ &= D(1 + y_2) - 0.5(t_3 - t_2)y_1 \end{aligned} \quad (9-2)$$

The mean propagation delay error,  $\varepsilon_D$ , is shown in equation (9-3):

$$\varepsilon_D = D_{actual} - D = Dy_2 - 0.5(t_3 - t_2)y_1. \quad (9-3)$$

The second term in equation (9-3), which corresponds to (2) above, is usually much larger than the first term in equation (9-3), which corresponds to (1) above. This is because the turnaround time,  $t_3 - t_2$ , is typically much larger than the mean propagation delay. For example, if the group delay of a link is 5 ns/m (this corresponds to a group velocity of  $2 \times 10^8$  m/s. Note that the speed of light in a vacuum is approximately  $3 \times 10^8$  m/s), then the propagation delay for a 1 km link is 5  $\mu$ s. If the PHY delays at the endpoints are on the order of 100 ns, the mean propagation delay is between 5 and 6  $\mu$ s. If the turnaround time is on the order of 1 ms, it is still larger than the mean propagation time by more than a factor of 100, and the second term in equation (9-3) dominates. In order for the first term to be appreciable relative to the second term, the link must be relatively long and/or the turnaround time must be relatively short. For example, if the length is on the order of 100 km or more, and the turnaround time is on the order of 1 ms or less, then the mean propagation delay is 500  $\mu$ s or more and the turnaround time is 1 ms or less (and the two are of the same order of magnitude).

---

<sup>6</sup> Several methods for transferring frequency using PTP are described in [IEEE 1588] and [IEEE 802.1AS]. However, these methods are not described in this Supplement as indicated in clause 9.1, the focus in this Supplement is the case where frequency is transferred using SyncE.

In the case where frequency is transported using syncE and is traceable to a PRC, with the timestamps being taken relative to this frequency, both  $y_1$  and  $y_2$  are at most  $10^{-11}$ . In this case, even if the turnaround time is extremely long, e.g., 1 s, the type A error in mean propagation delay is negligible, e.g., its magnitude is  $0.5(1 \text{ s})(10^{-11}) = 5 \text{ ps}$  in this case. Therefore, for the purpose of simulation type A errors are negligible.

### 9.1.2.2.2 Type B errors

Type B errors are due to errors in the timestamps, i.e., due to each of the values  $t_1$ ,  $t_2$ ,  $t_3$ , and  $t_4$  differing from the actual event of the respective message timestamp point crossing the respective reference plane. There are three contributions to this error:

- 1) phase noise present in the clock (timing signal) used for the timestamping,
- 2) phase measurement granularity, and
- 3) uncompensated ingress or egress latency (see section 7.3.4.2 of [IEEE 1588])

Point (3) is outside the scope of this Supplement. In general, this error must be measured, provided by the vendor, or obtained via some other means. The effect of any long-term frequency offset in (1) is discussed in clause 9.1.2.2.1.

Let  $T_{ts}(t)$  be the timestamp of an event whose actual time is  $t$ . For example, if a Pdelay\_Req message is transmitted by the peer delay initiator and the event of the message timestamp point crossing the reference plane occurs at ideal time  $t$ ,  $T_{ts}(t)$  is the timestamp value  $t_1$  of Figure 65. By *ideal* time, it is understood that it is the time relative to the T-GM. Let  $n(t)$  be the phase noise process of the clock (or timing signal) used for timestamping. Since long-term frequency offset is not discussed in this clause (because, as indicated above, that has already been discussed in clause 9.1.2.2.1), the time of the clock used for timestamping is  $t + n(t)$  when the T-GM time is  $t$ . Let the timestamp granularity be  $\Delta_{ts}$ ; this means that the timestamp clock advances in increments of  $\Delta_{ts}$ , and its reading is always an integer multiple of  $\Delta_{ts}$ . Then, the timestamp that corresponds to  $t$  is

$$T_{ts}(t) = \Delta_{ts} \cdot \text{floor}([t + n(t)] / \Delta_{ts}) \equiv \Delta_{ts} \cdot \lfloor [t + n(t)] / \Delta_{ts} \rfloor, \quad (9-4)$$

where  $\text{floor}(x)$  and  $\lfloor x \rfloor$  denote the greatest integer less than or equal to  $x$  (i.e., these are two alternate notations for the floor function). Since the floor function is an integer, the timestamp  $T_{ts}(t)$  is an integer multiple of  $\Delta_{ts}$ , as required. The timestamp error due to clock phase noise and phase measurement granularity,  $\varepsilon_{ts}$ , is

$$\varepsilon_{ts} = T_{ts}(t) - t = \Delta_{ts} \cdot \text{floor}([t + n(t)] / \Delta_{ts}) - t. \quad (9-5)$$

In a simulation, the phase noise  $n(t)$  is obtained from a noise model. For example, in the case where the frequency is transported using syncE,  $n(t)$  has been obtained by modelling a synchronization reference chain (see clause 8). For the time-domain simulations of clauses 8.1.3.3 and 8.1.4.2, the phase noise history at the output of each clock in the reference chain for each of the 300 independent replications of the simulation has been saved in a separate file. The respective phase noise values may be used to generate timestamps at a respective EEC in the reference chain using equation (9-4). Note that the phase noise sample times are not necessarily the same as the times that timestamps are needed, because the Sync and peer delay (or Delay\_Req) message rates are not necessarily the same as the sampling rate in the syncE noise accumulation simulations. This means that interpolation is needed. This is discussed in clause 9.1.4.

Applying equation (9-1) using equation (9-4), the measured mean propagation delay is

$$D = \frac{(T_{ts}(t_4) - T_{ts}(t_1)) - (T_{ts}(t_3) - T_{ts}(t_2))}{2}. \quad (9-6)$$

The mean propagation delay error,  $\varepsilon_D$ , is given by

$$\begin{aligned}\varepsilon_D &= D - D_{actual} = \frac{(\varepsilon_{ts}(t_4) - \varepsilon_{ts}(t_1)) - (\varepsilon_{ts}(t_3) - \varepsilon_{ts}(t_2))}{2} \\ &= \frac{\varepsilon_{ts}(t_4) + \varepsilon_{ts}(t_2) - \varepsilon_{ts}(t_1) - \varepsilon_{ts}(t_3)}{2}\end{aligned}\quad (9-7)$$

Some insight to equation (9-7) may be gained by considering the case where the phase noise is very small compared to the timestamp granularity. In this case,  $\varepsilon_{ts}$  can take on any value between  $-\Delta_{ts}$  and 0. The mean propagation delay error,  $\varepsilon_D$ , is maximized when  $\varepsilon_{ts}(t_4) = \varepsilon_{ts}(t_2) = 0$  and  $\varepsilon_{ts}(t_3) = \varepsilon_{ts}(t_1) = -\Delta_{ts}$ . In this case,  $\varepsilon_D = \Delta_{ts}$ . The mean propagation delay error,  $\varepsilon_D$ , is minimized when  $\varepsilon_{ts}(t_4) = \varepsilon_{ts}(t_2) = -\Delta_{ts}$  and  $\varepsilon_{ts}(t_3) = \varepsilon_{ts}(t_1) = 0$ . In this case,  $\varepsilon_D = -\Delta_{ts}$ . Therefore, when there is no phase noise, the error due only to phase measurement granularity is distributed between  $-\Delta_{ts}$  and  $\Delta_{ts}$ . Often, an assumption is made that this distribution is uniform. However, even if this assumption is correct, the error due to phase measurement granularity cannot be simulated by simply generating a stream of independent random samples that are uniformly distributed in the range  $[-\Delta_{ts}, \Delta_{ts}]$ . The reason is that, as is seen from equation (9-5), the error  $\varepsilon_{ts}(t)$  is correlated with  $t$ . equation (9-4) produces a timestamp value that is always an integer multiple of  $\Delta_{ts}$ . There would be no guarantee of always getting such an integer multiple if the error were generated from a uniform distribution in the range  $[-\Delta_{ts}, \Delta_{ts}]$  and added to  $t$ .

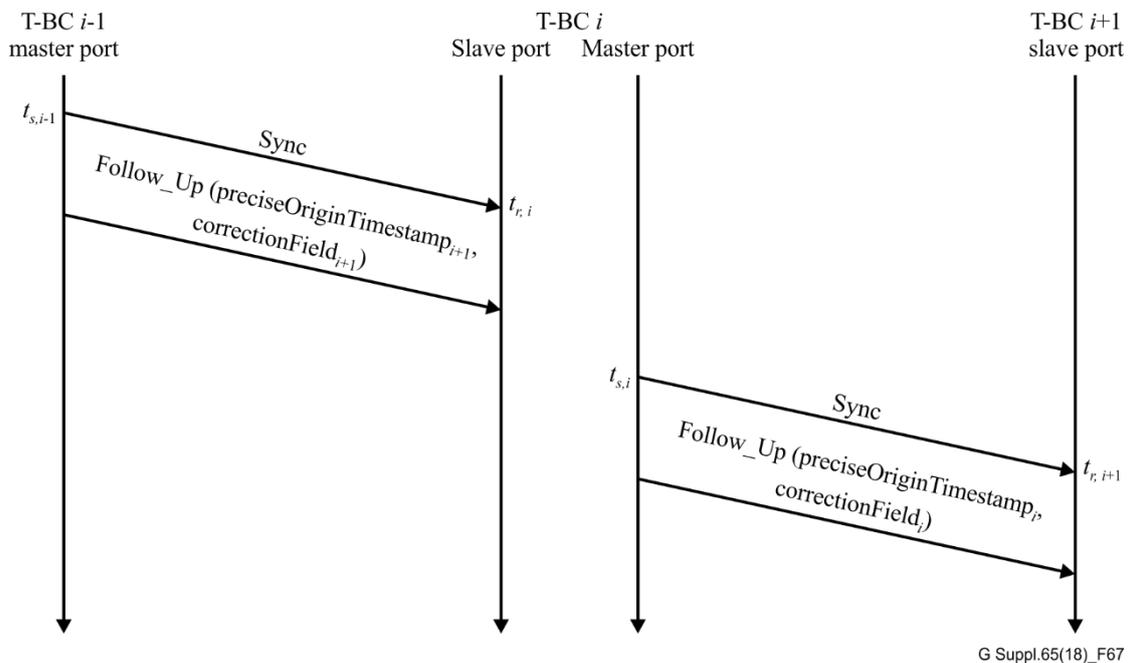
Therefore, for the purpose of simulation type B errors must be taken into account.

### 9.1.3 Effect of sojourn time on boundary clock performance

#### 9.1.3.1 Description of processing performed by a T-BC in transporting time

Some of the discussion of this clause, and Figure 67, is adapted from clause 11.1.3 of [IEEE802.1AS].

Consider the processing performed by a T-BC when it transports synchronization. Figure 67 shows an arbitrary T-BC, labelled with the index  $i$ , in a chain of T-BCs. The preceding T-BC is labelled  $i-1$  and the succeeding T-BC is labelled  $i+1$ . For simplicity, Figure 67 shows the case where the T-BCs are two-step clocks, i.e., both Sync and Follow\_Up messages are shown. If any T-BC were a one-step clock, the only change to Figure 67 would be that the T-BC would send Sync but not Follow\_Up and the preciseOriginTimestamp and correctionField values currently shown in the Follow\_Up sent by that T-BC would be shown as an originTimestamp and correctionField in the Sync message. The T-BC would still be capable of processing received Sync and Follow\_Up messages from an upstream two-step clock. The downstream T-BC is capable of processing the Sync message received from the one-step clock.



**Figure 67 – Illustration of transport of synchronization by a T-BC**

In Figure 67, T-BC  $i-1$  sends a Sync message to T-BC  $i$  on a master port and generates a timestamp  $t_{s,i-1}$ . At a later time, T-BC  $i-1$  sends a Follow\_Up message to T-BC  $i$ , which contains a preciseOriginTimestamp and correctionField. The preciseOriginTimestamp value is the T-GM time when the Sync message was sent, i.e., the T-GM time when the timestamp  $t_{s,i-1}$  was generated, except for any subnanosecond portion which is carried in the correctionField. The sum of the preciseOriginTimestamp and correctionField of the Follow\_Up message sent by T-BC  $i-1$  is the full T-GM time when the timestamp  $t_{s,i-1}$  was generated.

The Sync message is received by T-BC  $i$  on its slave port and timestamped with  $t_{r,i}$ . The Follow\_Up message is received at a later time. At a still later time, T-BC  $i$  sends a Sync message to T-BC  $i+1$  on a master port and timestamps it with  $t_{s,i}$ . T-BC  $i$  then sends a Follow\_Up message to T-BC  $i+1$ . T-BC  $i$  sets the preciseOriginTimestamp equal to the T-GM time when the timestamp  $t_{s,i}$  was generated, except for any subnanosecond portion; this is carried in the correctionField. The sum of the preciseOriginTimestamp and correctionField of the Follow\_Up message sent by T-BC  $i$  is the full T-GM time when the timestamp  $t_{s,i}$  was generated.

The full T-GM time at T-BC  $i$  when the timestamp  $t_{s,i}$  was generated is equal to the sum of:

- the preciseOriginTimestamp of the Follow\_Up message received from T-BC  $i-1$ ,
- the correction field of the Follow\_Up message received from T-BC  $i-1$ ,
- the propagation delay,  $D_{i-1,i}$  measured on the link between T-BCs  $i-1$  and  $i$ , and
- the time interval between the receipt of the most recent Sync message from T-BC  $i-1$  and the transmission of the current Sync message to T-BC  $i+1$ .

Item (d) above is equal to  $t_{s,i} - t_{r,i}$ , and is referred to as the *sojourn* time. This term is introduced to avoid confusion with the residence time of a TC. As indicated above, any subnanosecond portion of the sum of (a) – (d) is carried in the correctionField of the Follow\_Up message sent by T-BC  $i$ . The full T-GM time except for any subnanosecond portion is carried in the preciseOriginTimestamp.

Since the T-GM time when T-BC  $i$  sends Sync to T-BC  $i+1$  is given by the sum of (a) – (d) above, and since a corresponding sum is performed at each successive T-BC when sending Sync and Follow\_Up on a master port, it is seen that the T-GM time when T-BC  $i$  sends Sync to T-BC  $i+1$  is also given by the sum of:

- 1) the T-GM time when this synchronization information was sent by the T-GM in a Sync message,
- 2) the sum of the propagation delays on the successive upstream links between T-BC  $i$  and the T-GM, and
- 3) the sum of the sojourn times in T-BC  $i$  and all the T-BCs between the T-GM and T-BC  $i$ .

Therefore, the error in the time transported over the chain of T-BCs includes the sum of the errors in the mean propagation delays on the successive links plus the sum of the errors in the sojourn times of the successive T-BCs.<sup>7</sup> The error in mean propagation delay is discussed in clause 9.1.2. The error in sojourn time is discussed in the following clause.

### 9.1.3.2 Description of the errors in the measurement of sojourn time

As was the case for turnaround time, two types of errors, referred to as type A and type B, respectively, can arise in the measurement of sojourn time.<sup>8</sup>

#### 9.1.3.2.1 Type A errors

The sojourn time for T-BC  $i$ ,  $S_i$ , is given by

$$S_i = (t_{s,i} - t_{r,i})(1 + y_{GM,meas}). \quad (9-8)$$

The quantity  $y_{GM,meas}$  is any measured frequency offset of the T-GM relative to the timebase used to generate the timestamps at T-BC  $i$  (i.e., used to generate  $t_{s,i}$  and  $t_{r,i}$ ). In the case where the frequency is transported using syncE and the syncE reference chain is traceable to a PRC (i.e., is not in holdover), it is assumed that the syncE frequency is equal to the T-GM frequency and  $y_{GM,meas} = 0$ . In the case where the frequency is transported using PTP,  $y_{GM,meas}$  must actually be measured, but this is not discussed further in this Supplement.

Equation (9-8) gives the measured sojourn time. The actual sojourn time,  $S_{i,actual}$ , is given by equation (9-8), but with  $y_{GM,meas}$  replaced by  $y_{GM,actual}$

$$S_{i,actual} = (t_{s,i} - t_{r,i})(1 + y_{GM,actual}). \quad (9-9)$$

The sojourn time error,  $\varepsilon_S$ , is

$$\varepsilon_S = S_{i,actual} - S_i = (t_{s,i} - t_{r,i})(y_{GM,actual} - y_{GM,meas}) = (t_{s,i} - t_{r,i})\varepsilon_y, \quad (9-10)$$

where  $\varepsilon_y$  is the error in the measured frequency offset of the T-GM relative to T-BC  $i$ , i.e., the difference between what T-BC  $i$  thinks the frequency offset of the T-GM is and what it really is.

In the case where frequency is transported using syncE,  $y_{GM,meas} = 0$ , and  $\varepsilon_y = y_{GM,actual}$ , i.e., it is equal to the actual frequency offset of the T-GM relative to the syncE reference chain frequency. Since the syncE reference chain is assumed to be traceable to a PRC, and the T-GM is traceable to a primary reference,  $\varepsilon_y$  is at most  $10^{-11}$ . In this case, even if the sojourn time is extremely long, e.g., 1 s, the sojourn time error is negligible, i.e., its magnitude is  $(1 \text{ s})(10^{-11}) = 10 \text{ ps}$ . Therefore, as was the case for turnaround time, the sojourn time error is negligible in the case where frequency is transported via syncE and the syncE reference chain is traceable to a PRC.

Therefore, for the purpose of simulation type A errors are negligible.

---

<sup>7</sup> As indicated above, the error also includes any errors due to uncompensated delay asymmetry and PHY latency, but these errors are not the subject of this clause.

<sup>8</sup> As indicated above, there may also be delay asymmetry, but this is not the subject of this clause.

### 9.1.3.2.2 Type B errors

As was the case for turnaround time, type B errors are due to errors in the timestamps, i.e., due to each of the values  $t_{s,i}$  and  $t_{r,i}$  differing from the actual events of the respective message timestamp point crossing the respective reference plane. There are three contributions to this error, which are the same as for turnaround time:

- 1) phase noise present in the clock (timing signal) used for the timestamping,
- 2) phase measurement granularity, and
- 3) uncompensated ingress or egress latency (see clause 7.3.4.2 of [IEEE 1588])

As was the case for turnaround time, point (3) is outside the scope of this Supplement. The effect of any long-term frequency offset in (i) is discussed in clause 9.1.3.2.1.

The analysis of the timestamp error is the same as in clause 9.1.3.2.1. The timestamp value corresponding to actual time  $t$  is given by equation (9-4), and the timestamp error is given by equation (9-5). The measured sojourn time is obtained by using equation (9-4) in (9-8)

$$S_i = [T_{ts}(t_{s,i}) - T_{ts}(t_{r,i})](1 + y_{GM,meas}) = T_{ts}(t_{s,i}) - T_{ts}(t_{r,i}), \quad (9-11)$$

where  $y_{GM,meas} = 0$  since the case where frequency is transported via a syncE reference chain that is traceable to a PRC is being considered. The actual sojourn time is

$$S_{i,actual} = (t_{s,i} - t_{r,i}). \quad (9-12)$$

The sojourn time error is

$$\varepsilon_S = S_{i,actual} - S_i = \varepsilon_{ts}(t_{r,i}) - \varepsilon_{ts}(t_{s,i}). \quad (9-13)$$

As was the case for turnaround time, insight into equation (9-13) may be gained by considering the case where the phase noise is very small compared to the timestamp granularity. In this case,  $\varepsilon_{ts}$  varies between  $-\Delta_{ts}$  and 0. The minimum value of  $\varepsilon_S$  is  $-\Delta_{ts}$ , and occurs when  $\varepsilon(t_{r,i}) = -\Delta_{ts}$  and  $\varepsilon(t_{s,i}) = 0$ . The maximum value of  $\varepsilon_S$  is  $\Delta_{ts}$ , and occurs when  $\varepsilon(t_{r,i}) = 0$  and  $\varepsilon(t_{s,i}) = -\Delta_{ts}$ . Therefore, when there is no phase noise, the error due only to phase measurement granularity is distributed between  $-\Delta_{ts}$  and  $\Delta_{ts}$ . However, even if this distribution is uniform, this error cannot be generated by generating a stream of independent random samples that are uniformly distributed in the range  $[-\Delta_{ts}, \Delta_{ts}]$ . As is seen from equation (9-5) (and as was the case in the discussion of turnaround time), the error  $\varepsilon_{ts}(t)$  is correlated with  $t$ . equation (9-4) produces a timestamp value that is always an integer multiple of  $\Delta_{ts}$ . There would be no guarantee of always getting such an integer multiple if the error were generated from a uniform distribution in the range  $[-\Delta_{ts}, \Delta_{ts}]$  and added to  $t$ .

*Therefore, for the purpose of simulation type B errors must be taken into account because these errors are not negligible.*

### 9.1.4 Use of interpolation when sampling noise in simulations

For simulations of time error accumulation in a chain of T-BCs for the case where frequency is transported using syncE, the phase error process for the timing signal of the EEC (or SSU) at each T-BC must be obtained. The wander accumulation simulation results of clause 8.1 were saved in files, to be used as an input for the time-domain simulation of a chain of T-BCs. In particular, a time history of phase error was saved for each clock in the reference chain of Figure VII.1 of [ITU-T G.8251] for each of the 300 independent replications, for the Option 1 model described in clause 8.1.4 and the two Option 2 models described in clause 8.1.3. These simulations provide a sufficient amount of phase error data for simulations of chains of T-BCs.

However, as indicated in clause 9.1.2.2.2, the phase noise sample times are not necessarily the times when timestamps are taken. This is because the rates at which the PTP event messages are sent (Sync, Pdelay\_Req, Pdelay\_Resp, Delay\_Req) are not necessarily the same as the sampling rate for the

syncE wander accumulation simulations. Specifically, the wander accumulation simulations used a time step of 0.1 s, while the various message intervals are specified as powers of 2. In addition, [IEEE 1588] specifies in clause 7.7.2.1 that a node shall issue Sync, Pdelay\_Req, and announce messages within  $\pm 30\%$  of the value of the message interval attribute for the respective message, with 90% confidence (the corresponding requirement for Delay\_Req messages is given in clause 9.5.11.2 of [IEEE 1588]).

NOTE – One way to model this is to assume that the time between successive messages is gamma-distributed, with the mean of the gamma distribution equal to the respective mean message interval, and the variance chosen such that the portion of the gamma distribution within  $\pm 30\%$  of the mean has area equal to 0.9. The actual times that the successive event messages of the given type are sent are obtained by generating samples of this gamma distribution. This method is used in the simulator, as described in clause 11.

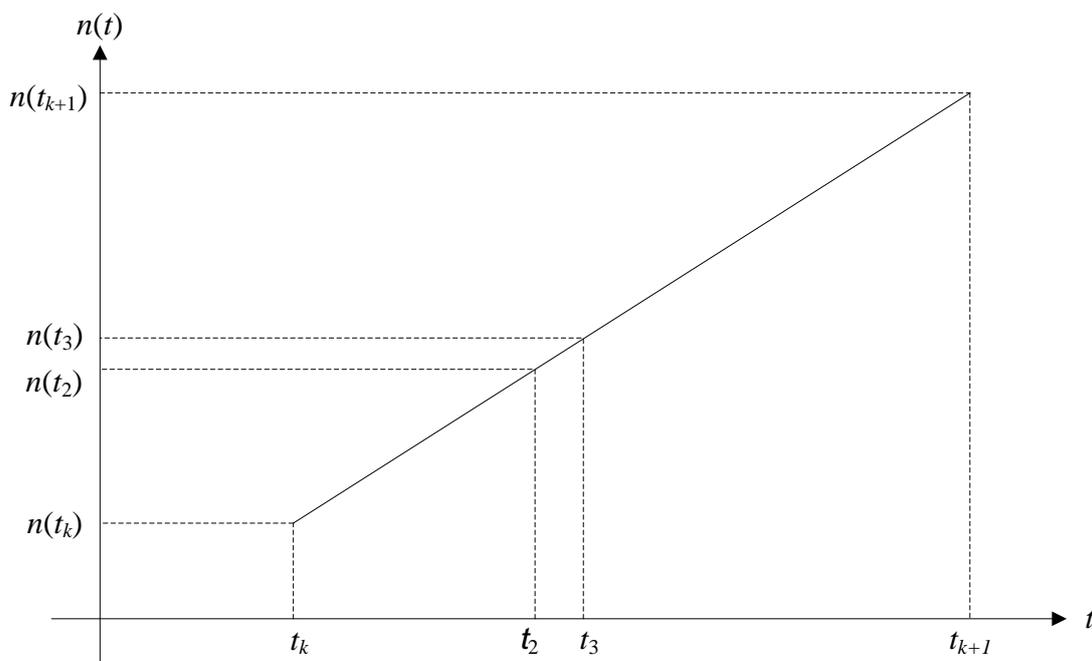
This requirement means that each successive Sync and Pdelay\_Req interval need not be the same, and therefore it is very unlikely that the sending of each successive Sync and Pdelay\_Req message will coincide with a multiple of 0.1 s, even if the mean Sync or Pdelay\_Req interval is a multiple of 0.1 s. Also, the time to transport a Sync, Pdelay\_Req, Pdelay\_Resp, or Delay\_Req message over a link is much less than 0.1 s, and the phase of the timing signal at each node is random relative to the transmission and receipt of PTP messages. It is concluded that the transmission and receipt times of PTP event messages will not necessarily line up with the multiples of 0.1 s when phase noise samples are available. Therefore, some form of interpolation is needed. The reason the sampling time (time step) for the wander accumulation simulations is 0.1 s in the simulations of clauses 8.1.3.3 and 8.1.4.2 is that this is the minimum observation interval for Option 1 EEC and SSU noise generation MTIE and TDEV masks, and also the minimum observation interval for Option 1 MTIE and TDEV network limits. In fact, since wander is defined as a phase variation whose frequency content is less than or equal to 10 Hz, it would not be correct to apply the wander accumulation model for time steps smaller than the inverse of the 20 Hz Nyquist rate, or 0.05 s.

A simple approach is to use linear interpolation to obtain the value of the phase noise at the desired time (i.e., the time of transmission or receipt of a PTP event message). Note that linear interpolation was used in simulations of the transport of SyncE clients over OTN, when applying the wander phase noise as an input to the OTN wander accumulation simulations (these simulations were done previously, but are not described in this Supplement). The linear interpolation would be done between the most recent and next phase noise sample. For example, let  $t_k$  and  $t_{k+1}$  be times when two successive noise samples are available, and assume a noise value is needed at time  $t$ , with  $t_k \leq t \leq t_{k+1}$ . Then, if the noise sample values, read from the respective file of saved noise samples, are  $n_k$  and  $n_{k+1}$ , the noise sample value at time  $t$  obtained using linear interpolation is

$$n(t) = n_k + \frac{n_{k+1} - n_k}{t_{k+1} - t_k} (t - t_k) = \frac{n_{k+1}(t - t_k) + n_k(t_{k+1} - t)}{t_{k+1} - t_k}. \quad (9-14)$$

When computing the difference between two timestamp values, the validity of linear interpolation depends on how close in time the timestamps are. For example, consider Figure 68, which shows a turnaround time interval,  $t_3 - t_2$ , that is much smaller than the time between two successive noise samples,  $t_{k+1} - t_k$ , and is completely contained within one sampling interval. The error in turnaround time due to phase noise is  $n(t_3) - n(t_2)$ . When computed using linear interpolation, as shown in Figure 68, the results is

$$n(t_3) - n(t_2) \Big|_{\text{linear interpolation}} = \frac{t_3 - t_2}{t_{k+1} - t_k} [n(t_{k+1}) - n(t_k)]. \quad (9-15)$$



**Figure 68 – Illustration of linear interpolation for case where the turnaround time is much less than the noise sampling time and is contained within one noise sampling time interval**

For  $t_3 - t_2 \ll t_{k+1} - t_k$ , it is seen that this gives a result that is much less than  $n(t_{k+1}) - n(t_k)$ . It might be argued that a more conservative approximation would be to assume that the entire change in noise from  $n(t_k)$  to  $n(t_{k+1})$  occurs in the interval between  $t_2$  and  $t_3$ , and the value should be  $n(t_{k+1}) - n(t_k)$ . However, since wander frequencies cannot exceed 10 Hz and  $t_3 - t_2 \ll t_{k+1} - t_k = 0.1$  s, the noise is limited in how fast it can change over  $t_3 - t_2$ . If, in the simulations, a conservative estimate (i.e., an overestimate) of the turnaround time is used, and if the actual turnaround time is much less than the inverse of the Nyquist rate for the wander noise (0.05 s), then the estimate of  $n(t_3) - n(t_2)$  obtained using linear interpolation is likely to be conservative. For example, if the actual turnaround time is 1 ms, but a turnaround time of 10 ms is assumed in the simulations, the estimate of  $n(t_3) - n(t_2)$  obtained via linear interpolation is

$$n(t_3) - n(t_2) \Big|_{\text{linear interpolation}} = (0.1)[n(t_{k+1}) - n(t_k)] \quad (9-16)$$

This is likely to be conservative because the actual time interval over which the noise can change is only 0.1 times as long, i.e., 1 ms, and the ratio of this time to the sampling time is 0.01. Note that turnaround time is expected to be small, i.e., on the order of 1 ms, in telecom equipment. Note also that if the delay request/response mechanism is used, a small turnaround time means that the second option described in clause 9.5.11.2 of [IEEE 1588 – 2008] is used, i.e., Delay\_Req should be transmitted as soon as possible following the receipt of a Sync message.

## 9.2 T-BC and T-TSC model that neglects SyncE noise generation

### 9.2.1 Model A, time-stamping relative to the corrected time

Figure 69 shows a detailed block diagram for model A, i.e., where time-stamping is done relative to the corrected time. Figure 69 corresponds to the model of Figure 64, but has several details added that are needed for computing the transfer function. These details pertain to the fact that the timestamp information arrives at discrete instants of time, while the syncE timing information is continuously

available.<sup>9</sup> After the noise  $e_{phy}$  is added to the error in the incoming grandmaster time information,  $T_{e,in}$ , the sum  $u(t)$  is sampled with sampling time  $T_0$ , where  $T_0$  is the sync interval. In the model,  $T_{e,in}$  is the difference between the grandmaster time reflected by the incoming Sync message (and Follow\_Up message if the upstream T-BC is a two-step clock) and the actual grandmaster time, at the time the Sync message is timestamped on transmission from the upstream T-BC. The incoming event message is timestamped relative to the corrected frequency. This is shown by the sampling of the output time  $y(t)$  (the output time is the current best estimate of the grandmaster time). In this model,  $u(t)$  and  $y(t)$  are sampled at the same time. The difference between the sampled input grandmaster time information and current best estimate of the grandmaster time is filtered by a low-pass filter whose transfer function is  $G_A(s)$ . This filter is the optional low-pass filter that might be contained in the 'Time Measurement, possibly with filtering' block of Figure 64. If the filter is not present,  $G_A(s) = 1$ . The filter output,  $p(t)$ , is input to a zero-order hold (ZOH). The output of the ZOH,  $v(t)$ , is subtracted from the syncE timing signal after it has been quantized. The quantization reflects finite phase measurement granularity. The effect of this subtraction is that the grandmaster time input information is used over the succeeding sync interval to correct the time obtained from the syncE signal, to obtain an estimate of the grandmaster time during the sync interval. The low-pass filter  $K_A(s)$  is the optional low-pass filter that might be contained in the 'Counter and Time offset correction' block of Figure 64. If the filter is not present,  $K_A(s) = 1$ .

The filters  $G_A(s)$  and  $K_A(s)$  can be used to prevent instantaneous changes in the estimate of the grandmaster time (i.e., the network synchronized time) at the incoming event message arrivals. However, as will be seen shortly,  $K_A(s)$  filters the incoming timing information and the noise processes  $e_{phy}$ ,  $e_{ts}$ , and  $e_{syncE}$ , while  $G_A(s)$  filters the incoming timing information and the noise processes  $e_{phy}$  and  $e_{ts}$ , but not  $e_{syncE}$ . Therefore,  $K_A(s)$  must be present if it is desired to filter the noise due to quantization. In addition,  $K_A(s)$  can be chosen to filter frequency components of the SyncE noise,  $e_{syncE}$ , and the quantization noise,  $e_{ts}$ , that exceed the Nyquist rate,  $1/(2T_0)$ , that corresponds to the sampling rate (i.e., to prevent aliasing).

In Figure 69, the difference between the timing information from the grandmaster and the timestamp based on the local time is computed using a sign convention where the timing information from the T-GM is positive and the timestamp based on the local time is negative. In the more usual sign convention used in feedback loops, the input has a positive sign and the fed back information has a negative sign. Figure 70 shows the block diagram of Figure 69, but with this more usual sign convention. In Figure 70, the sampled version of  $y(t)$ ,  $y^*(t)$ , is subtracted from the sampled version of  $u(t)$ ,  $u^*(t)$ , and the result  $v(t)$  after applying the filter  $G_A(s)$  and zero-order hold is added to  $n(t)$ .

The process  $n(t)$  is the error in the output of the counter that produces the phase of the SyncE signal. The input to the counter is the quantized SyncE signal with any accumulated phase noise. The quantized SyncE signal with any accumulated phase noise,  $T_{syncE}$ , is given by

$$T_{syncE} = \Delta_{ts} \cdot \text{floor} ([t + e_{syncE}] / \Delta_{ts}), \quad (9-17)$$

where  $\Delta_{ts}$  is the phase measurement granularity,  $e_{syncE}$  is the SyncE phase noise accumulation, and  $t$  is the ideal time relative to the origin of the SyncE counter (timescale) for no SyncE phase noise and granularity of zero. The process  $n(t)$  is equal to the difference between  $T_{syncE}$  and  $t$

$$n(t) = \Delta_{ts} \cdot \text{floor} ([t + e_{syncE}] / \Delta_{ts}) - t. \quad (9-18)$$

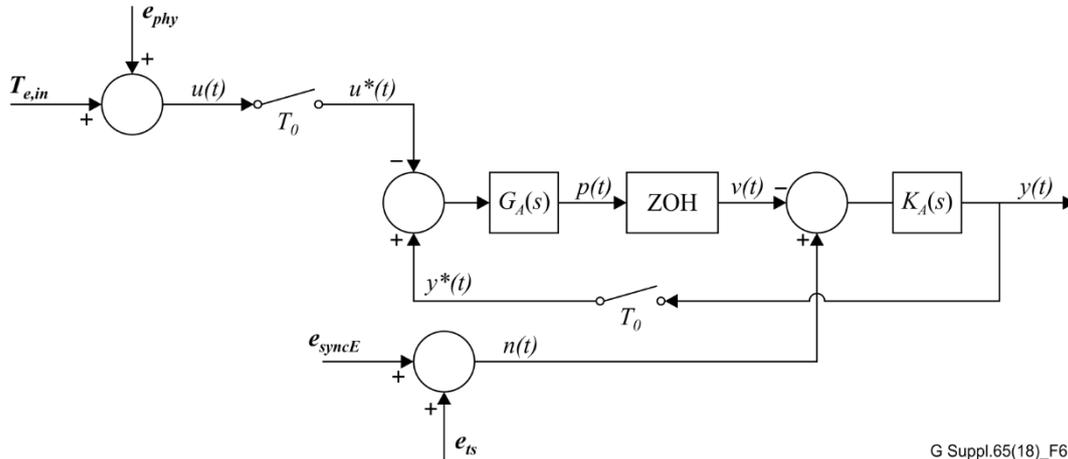
Even though both  $T_{e,in}$  and  $e_{phy}$  are shown as unsampled signals, the input to the T-BC is the sampled sum of these signals,  $u^*(t)$ . Note that the T-BC only has access to the sampled input  $u^*(t)$ . The sampling of  $u(t)$  that is shown is only conceptual. The actual input is  $u^*(t)$ . However, the SyncE phase

---

<sup>9</sup> More precisely, the SyncE timing information is available over a much shorter timescale than the timestamp information. The former is available at time intervals corresponding to the bit period of the SyncE physical layer, while the latter is available at time intervals corresponding to the sync interval.

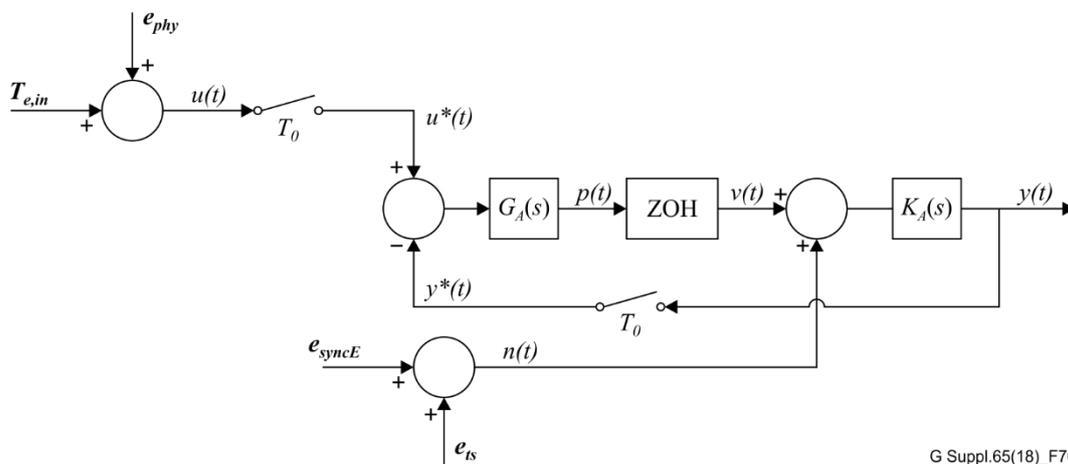
and quantization noise input to the T-BC,  $n(t)$ , is not sampled on input, but rather, the sum of  $v(t)$  and  $n(t)$  is sampled after being filtered by  $K_A(t)$ .

In an actual implementation, some of the blocks, e.g.,  $G_A(s)$ , might be implemented in the discrete domain, and the model of Figures 69 and 70 could have shown these blocks as z-transforms rather than Laplace transforms. However, it is convenient to do the analysis here in the continuous domain so that equivalent 3 dB bandwidth and gain peaking may be obtained. In addition, the order of  $G_A(s)$  and the zero-order hold could be interchanged if desired. The resulting model is mathematically equivalent to the model prior to the interchange.



G Suppl.65(18)\_F69

**Figure 69 – Detailed block diagram for the T-BC model, with timestamping done relative to the corrected time**



G Suppl.65(18)\_F70

**Figure 70 – Recasting of block diagram of Figure 69, with a sign convention where the input timing information from the T-GM is positive**

The transfer function is now computed based on Figure 70. The output  $Y(s)$  is

$$Y(s) = K_A(s)[N(s) + V(s)], \quad (9-19)$$

where  $N(s)$  is the total local time error due to accumulated SyncE noise and error due to timestamp granularity, and  $V(s)$  is the output of the zero-order hold. The transfer function of the zero-order hold is given by [b-Franklin], [b-Ogata]

$$H_{ZOH}(s) = \frac{1 - e^{-sT_0}}{s}. \quad (9-20)$$

The Laplace transform,  $X^*(s)$ , of a sampled signal  $x(kT_0)$ , where  $x(t)$  the unsampled signal, is obtained by first writing the sampled signal,  $x^*(t)$ , as the product of the unsampled signal and a train of impulses

$$x^*(t) = \sum_{k=0}^{\infty} \delta(t - kT_0)x(t), \quad (9-21)$$

and then computing the Laplace transform [b-Franklin], [b-Ogata]

$$X^*(s) = \int_0^{\infty} \sum_{k=0}^{\infty} \delta(t - kT_0)x(t)e^{-st} dt = \sum_{k=0}^{\infty} x(kT_0)e^{-skT_0} = X_z(e^{sT_0}). \quad (9-22)$$

In equations (9-5) and (9-6), it is assumed that  $x(t)$  is zero for  $t < 0$ , so that one-sided Laplace transforms and z-transforms can be used.

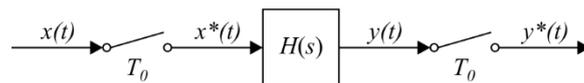
Then, the output  $Y(s)$  is related to the input timing information  $U(s)$  and the SyncE timing information  $N(s)$  by

$$\begin{aligned} Y(s) &= K_A(s) \left\{ N(s) + [U_z(e^{sT_0}) - Y_z(e^{sT_0})]G_A(s) \frac{1 - e^{-sT_0}}{s} \right\} \\ &= K_A(s) \left\{ N(s) + [U^*(s) - Y^*(s)]G_A(s) \frac{1 - e^{-sT_0}}{s} \right\}. \end{aligned} \quad (9-23)$$

This may be rewritten as

$$Y(s) = K_A(s)N(s) + K_A(s)G_A(s) \frac{1 - e^{-sT_0}}{s} U^*(s) - K_A(s)G_A(s) \frac{1 - e^{-sT_0}}{s} Y^*(s). \quad (9-24)$$

To complete the derivation of the transfer function, it is necessary to relate the Laplace transforms of the sampled and unsampled outputs,  $Y^*(s)$  and  $Y(s)$ , respectively. The derivation of this relation follows the derivation given in section 3-5 of [b-Ogata] (this derivation is done in the time domain. An equivalent derivation, in the frequency domain, is given in section 5.4 of [b-Franklin]). Figure 71 shows a system, whose transfer function is  $H(s)$ , with sampled input  $x^*(t)$ . The output of the system,  $y(t)$ , is sampled to produce  $y^*(t)$ .



G Suppl.65(18)\_F71

**Figure 71 – System  $H(s)$  with sampled input**

The Laplace transforms of  $y(t)$  and  $x^*(t)$  are related by

$$Y(s) = H(s)X^*(s). \quad (9-25)$$

Taking the inverse Laplace transform and using the result that the Laplace transform of the convolution of two functions is the product of the Laplace transforms produces

$$y(t) = \int_0^t h(t - \tau)x^*(\tau) d\tau, \quad (9-26)$$

Where  $h(t)$  is the inverse Laplace transform of  $H(s)$ , i.e., the impulse response of the system. Writing  $x^*(t)$  as the product of  $x(t)$  and a train of impulses produces

$$y(t) = \int_0^t h(t-\tau) \sum_{k=0}^{\infty} \delta(\tau - kT_0) x(\tau) d\tau = \sum_{k=0}^{\infty} h(t - kT_0) x(kT_0). \quad (9-27)$$

To obtain the Laplace transform  $Y^*(s)$ , we use equation (9-22)

$$\begin{aligned} Y^*(s) &= \sum_{n=0}^{\infty} y(nT_0) e^{-snT_0} = \sum_{n=0}^{\infty} \sum_{k=0}^{\infty} h((n-k)T_0) x(kT_0) e^{-snT_0} \\ &= \sum_{m=0}^{\infty} \sum_{k=0}^{\infty} h(mT_0) x(kT_0) e^{-s(k+m)T_0} = \sum_{m=0}^{\infty} h(mT_0) e^{-smT_0} \sum_{k=0}^{\infty} x(kT_0) e^{-skT_0} \\ &= H_z(e^{sT_0}) X_z(e^{sT_0}) \\ &= H^*(s) X^*(s) \end{aligned} \quad (9-28)$$

Equation (9-28) is the result that if  $Y(s) = H(s) X^*(s)$ , then

$$Y^*(s) = [H(s) X^*(s)]^* = H^*(s) X^*(s), \quad (9-29)$$

i.e., the result of sampling the output of a system whose input is sampled is equal to the result of applying the sampled input to a system whose impulse response is the sampled impulse response. In the discrete domain, the output is the discrete convolution of the sampled input and sampled impulse response. As indicated in [b-Franklin] and [b-Ogata], if the star (i.e., "\*") operation is viewed as that of sampling a signal or sampling the impulse response of a system, then the result of applying the star operation to the product of a starred and unstarred Laplace transform is the product of the starred Laplace transforms. Note also that (this point is emphasized in [b-Franklin] and [b-Ogata])

$$[H(s) X(s)]^* \neq H^*(s) X^*(s), \quad (9-30)$$

i.e., the sampled output of a system whose input is not sampled is generally not the same as the sampled output of a system whose input is sampled.

For convenience, the transfer function  $Q_A(s)$  is next defined as

$$Q_A(s) = G_A(s) \frac{1 - e^{-sT_0}}{s}. \quad (9-31)$$

Then

$$Q_A^*(s) = \left( G_A(s) \frac{1 - e^{-sT_0}}{s} \right)^* \neq G_A^*(s) \left( \frac{1 - e^{-sT_0}}{s} \right)^* = G_A^*(s). \quad (9-32)$$

Equation (9-24) may be rewritten as

$$Y(s) = K_A(s) N(s) + K_A(s) Q_A(s) U^*(s) - K_A(s) Q_A(s) Y^*(s). \quad (9-33)$$

Applying the star operator to equation (9-24), and using equations (9-29), (9-31) and (9-32), produces

$$Y^*(s) = [K_A(s) N(s)]^* + [K_A(s) Q_A(s)]^* U^*(s) - [K_A(s) Q_A(s)]^* Y^*(s), \quad (9-34)$$

or

$$Y^*(s) \{1 + [K_A(s) Q_A(s)]^*\} = [K_A(s) Q_A(s)]^* U^*(s) + [K_A(s) N(s)]^*. \quad (9-35)$$

Then

$$Y^*(s) = \frac{[K_A(s) Q_A(s)]^*}{1 + [K_A(s) Q_A(s)]^*} U^*(s) + \frac{[K_A(s) N(s)]^*}{1 + [K_A(s) Q_A(s)]^*}, \quad (9-36)$$

and

$$U^*(s) - Y^*(s) = \frac{1}{1 + [K_A(s)Q_A(s)]^*} U^*(s) - \frac{[K_A(s)N(s)]^*}{1 + [K_A(s)Q_A(s)]^*}. \quad (9-37)$$

Substituting equation (9-37) into equation (9-33) produces

$$\begin{aligned} Y(s) &= K_A(s)N(s) + K_A(s)Q_A(s)[U^*(s) - Y^*(s)] \\ &= K_A(s)N(s) + K_A(s)Q_A(s) \left[ \frac{1}{1 + [K_A(s)Q_A(s)]^*} U^*(s) - \frac{[K_A(s)N(s)]^*}{1 + [K_A(s)Q_A(s)]^*} \right] \\ &= \frac{K_A(s)Q_A(s)}{1 + [K_A(s)Q_A(s)]^*} U^*(s) + \left\{ K_A(s)N(s) - \frac{K_A(s)Q_A(s)}{1 + [K_A(s)Q_A(s)]^*} [K_A(s)N(s)]^* \right\}. \end{aligned} \quad (9-38)$$

Equation (9-38) gives the relation between the unsampled output,  $Y(s)$ , and the sampled input,  $U^*(s)$  and unsampled and sampled filtered noises,  $K_A(s)N(s)$  and  $[K_A(s)N(s)]^*$ , respectively. Note that both the unsampled and sampled filtered noises appear in equation (9-38) because the output depends on both the unsampled and sampled filtered noises (the latter via feedback).

Equation (9-38) may be rewritten as

$$\begin{aligned} Y(s) &= H_A(s)U^*(s) + \{K_A(s)N(s) - H_A(s)[K_A(s)N(s)]^*\} \\ &= H_A(s)U^*(s) + \left\{ 1 - H_A(s) \frac{[K_A(s)N(s)]^*}{K_A(s)N(s)} \right\} K_A(s)N(s), \\ &= H_A(s)U^*(s) + H_{n,A}(s)N(s) \end{aligned} \quad (9-39)$$

where the signal transfer function,  $H_A(s)$ , is given by

$$H_A(s) = \frac{K_A(s)Q_A(s)}{1 + [K_A(s)Q_A(s)]^*}, \quad (9-40)$$

and the noise transfer function,  $H_{n,A}(s)$ , is given by

$$H_{n,A}(s) = \left( 1 - H_A(s) \frac{[K_A(s)N(s)]^*}{K_A(s)N(s)} \right) K_A(s). \quad (9-41)$$

The terms "signal transfer function" and "noise transfer function" are used by analogy with corresponding terms used when considering conventional phase-locked loops (PLLs). Here, the signal  $u(t)$  includes the additive noise  $e_{phy}$ , which is filtered by the signal transfer function.

Equation (9-41) indicates that the noise transfer function depends on the ratio of the sampled filtered noise to unsampled filtered noise. In the case where the filter  $K_A(s)$  is an ideal filter such that  $K_A(s)N(s)$  has no frequency components above the Nyquist frequency corresponding to the sampling rate  $1/T_0$ , i.e., no frequency components above  $1/(2T_0)$ , equation (9-41) may be simplified and does not depend on the noise  $N(s)$ . To show this, the relation between the Laplace transform of a sampled signal and the original, unsampled signal is needed. For an arbitrary signal  $X(s)$ , this is given by [b-Franklin], [b-Ogata]

$$X^*(s) = X_z(e^{sT_0}) = \frac{1}{T_0} \sum_{n=-\infty}^{\infty} X(s - \frac{2\pi n}{T_0} j). \quad (9-42)$$

Setting  $s = j\Omega$  in equation (9-25), where  $\Omega$  is the angular analogue frequency, gives the sampling theorem

$$X^*(j\Omega) = X_z(e^{j\Omega T_0}) = \frac{1}{T_0} \sum_{n=-\infty}^{\infty} X(j(\Omega - \frac{2\pi n}{T_0})) = \frac{1}{T_0} \sum_{n=-\infty}^{\infty} X(\frac{j}{T_0}(\omega - 2\pi n)) \quad (9-43)$$

where  $\omega = \Omega T_0$  is the discrete frequency.

If the analogue signal is sampled at twice the Nyquist rate or higher, i.e., if the analogue frequency with the largest magnitude contained in the signal is  $\Omega_0$ , i.e.,  $-\Omega_0 \leq \Omega \leq \Omega_0$ , then

$$\frac{2\pi}{T_0} \geq 2\Omega_0, \quad (9-44)$$

or

$$\Omega_0 T_0 \leq \pi. \quad (9-45)$$

This condition ensures that there is no aliasing, and the Fourier transforms of the sampled and unsampled signals are related by

$$X^*(j\Omega) = X_z(e^{j\Omega T_0}) = \frac{1}{T_0} X(j\Omega) \quad \text{for } -\Omega_0 \leq \Omega \leq \Omega_0. \quad (9-46)$$

Note that the Fourier transform of the unsampled signal is zero for  $\Omega \geq \Omega_0$ , and the Fourier transform of the sampled signal is zero for  $\Omega_0 \leq \Omega \leq \pi/T_0$  and periodic with period  $2\pi/T_0$ . When there is no aliasing, the contributions of the individual terms of the summations in equation (9-43) do not overlap.

To relate the Laplace transforms of the sampled and unsampled signal, set  $s = \sigma + j\Omega$  in equation (9-42), and assume equations (9-44) and (9-45) hold, i.e., there is no aliasing. This means that the rate of transmission of Sync messages is at least twice the Nyquist rate for the filtered noise  $K_A(s)N(s)$ . The Laplace transform of the sampled signal is period in  $\Omega$  with period  $2\pi/T_0$ , and in the case of no aliasing the contributions of the individual terms do not overlap. Then

$$X^*(\sigma + j\Omega) = X_z(e^{(\sigma + j\Omega)T_0}) = \frac{1}{T_0} X(\sigma + j\Omega) \quad \text{for } -\Omega_0 \leq \Omega \leq \Omega_0, \quad (9-47)$$

or

$$X^*(s) = X_z(e^{sT_0}) = \frac{1}{T_0} X(s) \quad \text{for } s = \sigma + j\Omega \text{ and } -\Omega_0 \leq \Omega \leq \Omega_0. \quad (9-48)$$

Applying equation (9-48) to the ratio of the sampled filtered noise to unsampled filtered noise in equation (9-41) produces

$$\frac{[K_A(s)N(s)]^*}{K_A(s)N(s)} = \frac{1}{T_0} \quad \text{for } s = \sigma + j\Omega \text{ and } -\Omega_0 \leq \Omega \leq \Omega_0. \quad (9-49)$$

and

$$\begin{aligned} H_{n,A}(s) &= \left( 1 - \frac{H_A(s)}{T_0} \right) K_A(s) \\ &= \left( 1 - \frac{K_A(s)G_A(s) \frac{1 - e^{-sT_0}}{sT_0}}{1 + [K_A(s)Q_A(s)]^*} \right) K_A(s) \\ &= \left( \frac{1 + [K_A(s)Q_A(s)]^* - \frac{K_A(s)Q_A(s)}{T_0}}{1 + [K_A(s)Q_A(s)]^*} \right) K_A(s) \end{aligned} \quad (9-50)$$

for  $s = \sigma + j\Omega$  and  $-\Omega_0 \leq \Omega \leq \Omega_0$

Note that equation (9-50) holds only for  $-\Omega_0 \leq \Omega \leq \Omega_0$ . This is not a limitation, because it has been assumed that  $K_A(s)N(s)$  has no frequency components above  $\Omega_0$ .

If  $K_A(s)$  and  $G_A(s)$  are low-pass filters, then  $H_A(s)$  has a low-pass characteristic and  $H_{n,A}(s)$  is the product of a low-pass and high-pass filter (i.e, it has a band-pass characteristic). (Note that if  $G_A(s)$  is a low-pass filter,  $Q_A(s)$  is a low-pass filter because the zero-order hold has a low-pass characteristic.) Note that since  $G_A(s)$  enters only as a factor of the product  $Q_A(s)K_A(s)$ , it may not be necessary to have a separate filter  $G_A(s)$ .<sup>10</sup> The same level of filtering of the input timing information from the T-GM and the noise  $e_{phy}$  can be achieved by incorporating all the filtering in  $K_A(s)$ . If this is done, the noises  $e_{syncE}$  and  $e_{ts}$  will be filtered more than they would be if  $G_A(s)$  and  $K_A(s)$  were separate filters. Note, however, that the filtering  $K_A(s)$  must be present if it is desired to filter  $e_{syncE}$  and/or  $e_{ts}$ . While the former will likely have been filtered adequately by the SyncE HRM, the latter, i.e., the quantization noise, will not have been filtered.  $K_A(s)$  must be present if filtering of the quantization noise is needed.

### 9.2.2 Model B, time-stamping relative to the uncorrected time

Figure 72 shows a detailed block diagram for model B, i.e., where timestamping is done relative to the uncorrected time. As in Figures 69 and 70 for model A, the noise  $e_{phy}$  is added to the incoming grandmaster information,  $T_{e,in}$ , and the sum  $u(t)$  is sampled at times that correspond to the arrival of Sync messages. However, since the timestamping is now done relative to the uncorrected time, the SyncE signal, with the quantization noise  $e_{ts}$  added, is filtered by the optional low-pass filter  $M_B(s)$  and then sampled at the same times, and the difference between the two sampled signals is computed at each sampling time. The quantization reflects finite phase measurement granularity. The filter  $M_B(s)$  can serve as an anti-aliasing filter, to remove any frequencies above the Nyquist frequency that corresponds to the sampling rate, prior to sampling. The difference between the input and sampled filtered noise is filtered by a low-pass filter whose transfer function is  $G_B(s)$ . This filter is the optional low-pass filter that might be contained in the 'Time Measurement, possibly with filtering' block of Figure 64. If the filter is not present,  $G_B(s) = 1$ . The filter output,  $p(t)$ , is input to a ZOH. The output of the ZOH,  $v(t)$ , is subtracted from the quantized SyncE timing signal. The effect of this subtraction is that the grandmaster time input information is used over the succeeding sync interval to correct the time obtained from the SyncE signal, to obtain an estimate of the grandmaster time during the sync interval. The output of the subtraction block is filtered by  $K_B(s)$ . The low-pass filters  $K_B(s)$  and  $M_B(s)$  are optional low-pass filters that might be contained in the 'Counter and Time offset correction' block of Figure 64.

As in model A, the filters  $G_B(s)$  and  $K_B(s)$  can be used to prevent instantaneous changes in the estimate of the grandmaster time (i.e., the network synchronized time) at the incoming event message arrivals. The filters  $G_B(s)$  and  $K_B(s)$  for model B are not necessarily the same as the filters  $G_A(s)$  and  $K_A(s)$  for model A. In fact, after the signal and noise transfer functions,  $H_B(s)$  and  $H_{n,B}(s)$ , are derived for model B, it will be clear how  $G_B(s)$ ,  $K_B(s)$ , and  $M_B(s)$  are related to  $G_A(s)$  and  $K_A(s)$  in order for the model A and model B transfer functions to be the same.

The quantity  $n(t)$  is as described in clause 9.2.1.

The signal and noise transfer functions for model B2 are derived in a manner analogous to the derivation in clause 9.2.1 for model A.

Using Figure 72, the output  $Y(s)$  is

$$Y(s) = K_B(s) \left\{ N(s) + \left\{ U^*(s) - [M_B(s)N(s)]^* \right\} G_B(s) \frac{1 - e^{-sT_0}}{s} \right\}, \quad (9-51)$$

<sup>10</sup> As outlined in clause 9.2.3, if  $G(s) = 1$  in one form of the model, e.g., case A,  $G(s)$  is not 1 in the other form of the model (model B) if we want both forms to have the same signal and noise transfer functions.

where the result was used for the transfer function of a zero-order hold, equation (9-20). equation (9-51) may be rewritten as

$$\begin{aligned}
 Y(s) &= K_B(s)Q_B(s)U^*(s) + K_B(s)N(s) - K_B(s)Q_B(s)[M_B(s)N(s)]^* \\
 &= K_B(s)Q_B(s)U^*(s) + \left\{ 1 - K_B(s)Q_B(s) \frac{[M_B(s)N(s)]^*}{K_B(s)N(s)} \right\} K_B(s)N(s), \quad (9-52) \\
 &= H_B(s)U^*(s) + H_{n,B}(s)N(s)
 \end{aligned}$$

where

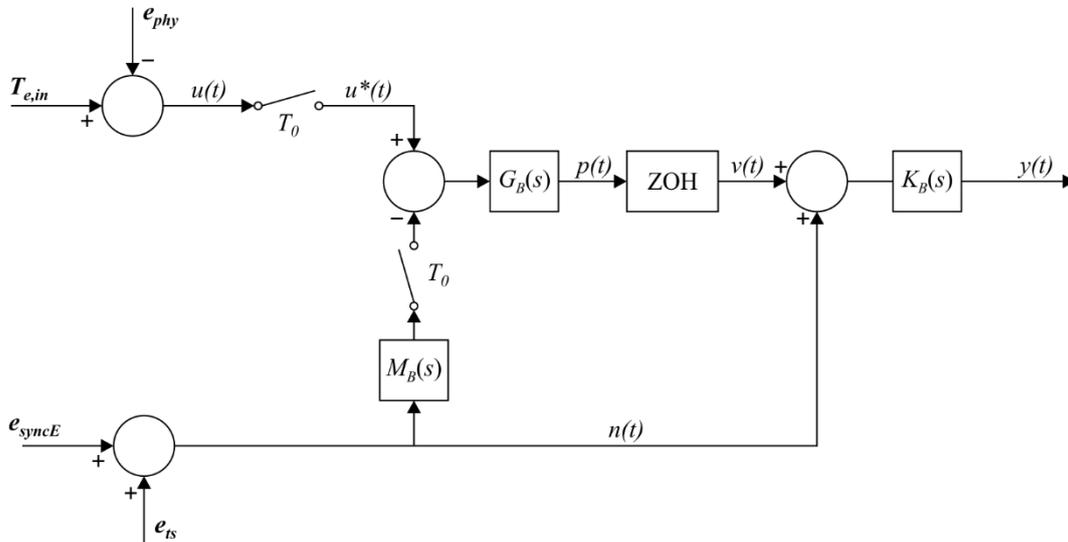
$$Q_B(s) = G_B(s) \frac{1 - e^{-sT_0}}{s}, \quad (9-53)$$

and the signal and noise transfer functions,  $H_B(s)$  and  $H_{n,B}(s)$ , respectively, are given by

$$H_B(s) = K_B(s)Q_B(s) = K_B(s)G_B(s) \frac{1 - e^{-sT_0}}{s} \quad (9-54)$$

and

$$\begin{aligned}
 H_{n,B}(s) &= \left\{ 1 - K_B(s)Q_B(s) \frac{[M_B(s)N(s)]^*}{K_B(s)N(s)} \right\} K_B(s) \\
 &= \left\{ 1 - M_B(s)G_B(s) \frac{1 - e^{-sT_0}}{s} \cdot \frac{[M_B(s)N(s)]^*}{M_B(s)N(s)} \right\} K_B(s)
 \end{aligned} \quad (9-55)$$



G Suppl.65(18)\_F72

**Figure 72 – Detailed block diagram for the T-BC model, with timestamping done relative to the uncorrected time**

In the case where the filter  $M_B(s)$  is an ideal filter such that  $M_B(s)N(s)$  has no frequency components above the Nyquist frequency corresponding to the sampling rate  $1/T_0$ , i.e., no frequency components above  $1/(2T_0)$ , equation (9-55) may be simplified using equation (9-49). The result is

$$\begin{aligned}
 H_{n,B}(s) &= \left\{ 1 - M_B(s)G_B(s) \frac{1 - e^{-sT_0}}{sT_0} \right\} K_B(s) \\
 &\text{for } s = \sigma + j\Omega \text{ and } -\Omega_0 \leq \Omega \leq \Omega_0
 \end{aligned} \quad (9-56)$$

As in clause 9.2.1, the terms "signal transfer function" and "noise transfer function" are used by analogy with corresponding terms used when considering conventional phase-locked loops (PLLs). Here, the signal  $u(t)$  includes the additive noise  $e_{phy}$ , which is filtered by the signal transfer function.

### 9.2.3 Comparison of model A and model B transfer functions

The model A signal and noise transfer functions are given by Eqs. (9-40) and (9-41), respectively. The case B signal and noise transfer functions are given by Eqs. (9-54) and (9-55), respectively. It is important to now consider how  $K_B(s)$ ,  $M_B(s)$ , and  $G_B(s)$  are related to  $K_A(s)$  and  $G_A(s)$  if the signal and noise transfer functions of the two forms of the model are to be the same. With these relations, it will be possible, given one of the forms of the model (model A or model B), to find the filter transfer functions for the other form such that the signal and noise transfer functions are the same. Given this equivalence, simulations can, in principle, be performed using model A or model B with the assurance that, if the filter transfer functions are chosen according to these relations, the performance will be the same using the other form. Note, however, that for desired signal and noise transfer functions, one form may be more convenient than the other.

To derive the relation between model A and model B filter functions such that the signal and noise transfer functions are the same, the signal and noise transfer functions must be equated. Model A signal and noise transfer functions are given by Eqs. (9-40) and (9-41), and model B signal and noise transfer functions are given by equations (9-54) and (9-55). Equating equations (9-40) and (9-54), and equations (9-41) and (9-55), and substituting equation (9-54), produces

$$K_B(s)Q_B(s) = \frac{K_A(s)Q_A(s)}{1 + [K_A(s)Q_A(s)]^*} \quad (9-57)$$

$$\left\{ 1 - H_B(s) \frac{[M_B(s)N(s)]^*}{K_B(s)N(s)} \right\} K_B(s) = \left( 1 - H_A(s) \frac{[K_A(s)N(s)]^*}{K_A(s)N(s)} \right) K_A(s). \quad (9-58)$$

Equations (9-57) and (9-58) may be used to establish the equivalence of models A and B, such that the signal and noise transfer functions are equal. However, note that model A has two filter functions,  $K_A(s)$  and  $G_A(s)$ , while model B has three filter functions,  $K_B(s)$ ,  $G_B(s)$ , and  $M_B(s)$ . Since there are two equations, a unique solution for  $K_A(s)$  and  $G_A(s)$  in terms of  $K_B(s)$ ,  $G_B(s)$ , and  $M_B(s)$  may be found. This means that, if model B is used and  $K_B(s)$ ,  $G_B(s)$ , and  $M_B(s)$  are chosen, an equivalent model A can be found, such that the signal and noise transfer functions are the same as in model B. This is done by solving for  $K_A(s)$  and  $G_A(s)$  in terms of  $K_B(s)$ ,  $G_B(s)$ , and  $M_B(s)$  (except for possibly degenerate cases where the particular model B filter functions result in infinite gain in model A. This will be explained in more detail shortly). However, if model A is used and  $K_A(s)$  and  $G_A(s)$  are chosen, there will in general be more than one mathematically equivalent model B realization, because three filter functions  $K_B(s)$ ,  $G_B(s)$ , and  $M_B(s)$  must be determined (in terms of  $K_A(s)$  and  $G_A(s)$ ) but only two equations are available. This is not a disadvantage, as it simply means there is more freedom to choose the model B filter functions to achieve the desired signal and noise transfer functions.

In view of the above, the equivalent model A filter functions will be determined in terms of the model B filter functions.

Substituting  $H_A(s) = H_B(s)$  in equation (9-58) and simplifying produces

$$K_B(s)N(s) - H_B(s)[M_B(s)N(s)]^* = K_A(s)N(s) - H_B(s)[K_A(s)N(s)]^*. \quad (9-59)$$

Equation (9-59) must be solved for  $K_A(s)$  in terms of  $K_B(s)$  and  $H_A(s)$ . However,  $K_A(s)$  appears as a factor of a term for which the star operator is applied to (recall from clause 9.2.1 that the star operator produces the transfer function of the sampled signal from the transfer function of the unsampled signal). Applying the star operator to both sides of equation (9-59) (and using equation (9-28)) produces

$$[K_B(s)N(s)]^* - H_B^*(s)[M_B(s)N(s)]^* = [K_A(s)N(s)]^* - H_B^*(s)[K_A(s)N(s)]^* \quad (9-60)$$

Solving equation (9-60) for  $[K_A(s)N(s)]^*$  produces

$$[K_A(s)N(s)]^* = \frac{[K_B(s)N(s)]^* - H_B^*(s)[M_B(s)N(s)]^*}{1 - H_B^*(s)} \quad (9-61)$$

Substituting equation (9-61) into equation (9-59) and solving for  $K_A(s)N(s)$  produces

$$K_A(s)N(s) = K_B(s)N(s) + \frac{H_B(s)}{1 - H_B^*(s)} \{ [K_B(s)N(s)]^* - [M_B(s)N(s)]^* \} \quad (9-62)$$

Substituting equation (9-54) for  $H_B(s)$  and dividing by  $N(s)$  produces

$$K_A(s) = K_B(s) + \frac{K_B(s)Q_B(s)}{N(s)[1 - H_B^*(s)]} \{ [K_B(s)N(s)]^* - [M_B(s)N(s)]^* \}, \quad (9-63)$$

or

$$\begin{aligned} K_A(s) &= K_B(s) \left\{ 1 + \frac{K_B(s)Q_B(s)}{[1 - H_B^*(s)]} \cdot \left( \frac{[K_B(s)N(s)]^* - [M_B(s)N(s)]^*}{K_B(s)N(s)} \right) \right\} \\ &= K_B(s) \left\{ 1 + \frac{K_B(s)Q_B(s)}{1 - [K_B(s)Q_B(s)]^*} \cdot \left( \frac{[K_B(s)N(s)]^* - [M_B(s)N(s)]^*}{K_B(s)N(s)} \right) \right\}. \end{aligned} \quad (9-64)$$

Equation (9-64) is the desired result for  $K_A(s)$  in terms of the model B filter functions, such that the model A and B signal and noise transfer functions will be the same.

To obtain the result for  $G_A(s)$  in terms of the model B filter functions, there is the need to solve equation (9-57) for  $Q_A(s)$ , and then obtain  $G_A(s)$  from equation (9-31). Since equation (9-57) contains both  $Q_A(s)$  and  $[K_A(s)Q_A(s)]^*$ , the star operator to equation (9-57) is first applied (i.e., by multiplying both sides of equation (9-41) by  $1 + [K_A(s)Q_A(s)]^*$ , applying the star operator to both sides of the resulting equation, and then dividing both sides by  $1 + [K_A(s)Q_A(s)]^*$ , and then solve for  $[K_A(s)Q_A(s)]^*$

$$[K_B(s)Q_B(s)]^* = \frac{[K_A(s)Q_A(s)]^*}{1 + [K_A(s)Q_A(s)]^*} \quad (9-65)$$

$$[K_A(s)Q_A(s)]^* = \frac{[K_B(s)Q_B(s)]^*}{1 - [K_B(s)Q_B(s)]^*} \quad (9-66)$$

Next substitute equation (9-66) into equation (9-57) to obtain

$$K_B(s)Q_B(s) = \frac{K_A(s)Q_A(s)}{1 + \frac{[K_B(s)Q_B(s)]^*}{1 - [K_B(s)Q_B(s)]^*}} = K_A(s)Q_A(s) \{ 1 - [K_B(s)Q_B(s)]^* \} \quad (9-67)$$

Solving equation (9-67) for  $K_A(s)Q_A(s)$  produces

$$K_A(s)Q_A(s) = \frac{K_B(s)Q_B(s)}{1 - [K_B(s)Q_B(s)]^*} \quad (9-68)$$

Substituting equation (9-64) for  $K_A(s)$  into equation (9-68) produces

$$\begin{aligned}
Q_A(s) &= \frac{Q_B(s)}{\left\{1 + \frac{K_B(s)Q_B(s)}{1 - [K_B(s)Q_B(s)]^*} \cdot \left( \frac{[K_B(s)N(s)]^* - [M_B(s)N(s)]^*}{K_B(s)N(s)} \right) \right\} \{1 - [K_B(s)Q_B(s)]^*\}} \\
&= \frac{Q_B(s)}{\left\{1 - [K_B(s)Q_B(s)]^* + K_B(s)Q_B(s) \cdot \left( \frac{[K_B(s)N(s)]^* - [M_B(s)N(s)]^*}{K_B(s)N(s)} \right) \right\}} . \quad (9-69)
\end{aligned}$$

Using equation (9-31) for  $Q_A(s)$  and equation (9-37) for  $Q_B(s)$  produces the result for  $G_A(s)$  in terms of  $G_B(s)$

$$G_A(s) = \frac{G_B(s)}{\left\{1 - [K_B(s)Q_B(s)]^* + K_B(s)Q_B(s) \cdot \left( \frac{[K_B(s)N(s)]^* - [M_B(s)N(s)]^*}{K_B(s)N(s)} \right) \right\}} . \quad (9-70)$$

Equation (9-64) and (9-70) are the desired transformation between model A and model B, i.e., given model B with specified  $G_B(s)$ ,  $K_B(s)$ , and  $M_B(s)$ , model A will have the same transfer function and frequency response if  $G_A(s)$  and  $K_A(s)$  are chosen using equations (9-64) and (9-70). Note that equations (9-64) and (9-70) were derived without assuming there is no aliasing.

For the special case where the filters  $K_B(s)$  and  $M_B(s)$  are the same, i.e.,  $K_B(s) = M_B(s)$ , equations (9-64) and (9-70) reduce to

$$K_A(s) = K_B(s) \quad (9-71)$$

$$G_A(s) = \frac{G_B(s)}{1 - [K_B(s)Q_B(s)]^*} . \quad (9-72)$$

For this special case, a unique relation for  $G_B(s)$  and  $K_B(s)$  in terms of  $G_A(s)$  and  $K_A(s)$  may be found. The relation for  $K_B(s)$  is obtained trivially from equation (9-71)

$$K_B(s) = K_A(s) . \quad (9-73)$$

To obtain the relation for  $G_B(s)$ , multiply both sides of equation (9-72) by

$$K_B(s) \frac{1 - e^{-sT_0}}{s} , \text{ and use equation (9-73), to obtain}$$

$$K_A(s)Q_A(s) = \frac{K_B(s)Q_B(s)}{1 - [K_B(s)Q_B(s)]^*} . \quad (9-74)$$

Take the star transform of equation (9-74) to obtain

$$[K_A(s)Q_A(s)]^* = \frac{[K_B(s)Q_B(s)]^*}{1 - [K_B(s)Q_B(s)]^*} . \quad (9-75)$$

Solve for  $[K_B(s)Q_B(s)]^*$  to obtain

$$[K_B(s)Q_B(s)]^* = \frac{[K_A(s)Q_A(s)]^*}{1 + [K_A(s)Q_A(s)]^*} . \quad (9-76)$$

Then, from equation (9-76)

$$1 - [K_B(s)Q_B(s)]^* = \frac{1}{1 + [K_A(s)Q_A(s)]^*} . \quad (9-77)$$

Substitute equation (9-77) into equation (9-72) and solve for  $G_B(s)$  to obtain

$$G_B(s) = \frac{G_A(s)}{1 + [K_A(s)Q_A(s)]^*} \quad (9-78)$$

Equations (9-73) and (9-78) are the desired result for  $G_B(s)$  and  $K_B(s)$  in terms of  $G_A(s)$  and  $K_A(s)$ . Note that these equations were derived without assuming there is no aliasing.

Equations (9-71), (9-72), (9-73) and (9-78) indicate that if we use one form of the model and choose  $G(s) = 1$ , the corresponding  $G(s)$  for the other form of the model to obtain the same signal and noise transfer functions will not be 1.

Equation (9-72) indicates that there is a special case, namely the case where  $[K_B(s)Q_B(s)]^* = 1$ , where the model B must be used. This is because  $G_A(s) \rightarrow \infty$  when  $[K_B(s)Q_B(s)]^* \rightarrow 1$ . One example of this is the case  $K_B(s)G_B(s) = 1$ . In this case (the final equality in equation (9-63) is derived in [b-Ogata])

$$[K_B(s)Q_B(s)]^* = [K_B(s)G_B(s) \left( \frac{1 - e^{-sT_0}}{s} \right)]^* = \left( \frac{1 - e^{-sT_0}}{s} \right)^* = 1 \quad (9-79)$$

More generally, model B must be used when  $G_B(s)$ ,  $K_B(s)$ , and  $M_B(s)$  are chosen such that the denominator of equation (9-70) is zero.

The conclusion of this clause is that model A and model B are equivalent, and either may be used for simulation purposes, provided that the filters  $G(s)$  and  $K(s)$  are chosen appropriately to achieve the desired signal and noise transfer functions. The only exception to this is the case where the denominator of equation (9-54) is zero. In this case the equivalent model A filter gain  $G_A(s)$  becomes infinite, and model B must be used.

Depending on the desired filter characteristics, and also on the simulator architecture, one of the forms may be more convenient.

## 9.2.4 Frequency response examples

### 9.2.4.1 Example 1 – model B with no filtering in the 'Time Measurement, possibly with filtering' and 'Counter and Time offset correction' blocks ( $GB = KB = MB = 1$ )

In this example model B for the case of no filtering, i.e., the filters represented by  $G_B(s)$ ,  $K_B(s)$ , and  $M_B(s)$  are not present, is considered. This means that  $G_B(s) = K_B(s) = M_B(s) = 1$ . As noted at the end of clause 9.2.3, in this case the equivalent model A filter gain  $G_A(s)$  is infinite and model B must be used.

The signal transfer functions are, from Eqs. (9-54) and (9-55)

$$H(s) = \frac{1 - e^{-sT_0}}{s} \quad (9-80)$$

$$H_n(s) = 1 - \frac{1 - e^{-sT_0}}{s} \cdot \frac{N^*(s)}{N(s)}$$

If, for the purposes of this example, it is assumed that the noise contains no frequencies above the Nyquist rate, the noise transfer function becomes (using equation (9-49))

$$H_n(s) = 1 - \frac{1 - e^{-sT_0}}{sT_0} \quad (9-81)$$

Note that the signal transfer function has units of time, while the noise transfer function is dimensionless. This is because we defined the signal transfer function input as the sampled input  $u^*(t)$  and the output as the unsampled output  $y(t)$ . In contrast, the noise transfer function input and output were both unsampled ( $n(t)$  and  $y(t)$ , respectively). In the results that follow, the signal transfer function will be expressed in dimensionless form by dividing it by  $T_0$ , i.e.,

$$\frac{H(s)}{T_0} = \frac{1 - e^{-sT_0}}{sT_0}, \quad (9-82)$$

The signal and noise frequency responses are obtained by setting  $s = j\Omega$  (where  $\Omega$  is the fourier frequency) and taking the magnitude (absolute value). The results are

$$\frac{|H(j\Omega)|}{T_0} = \frac{\sqrt{2(1 - \cos x)}}{x} \quad (9-83)$$

$$|H_n(j\Omega)| = \frac{\sqrt{x^2 - 2(\cos x + x \sin x) + 2}}{x}, \quad (9-84)$$

where  $x$  is the dimensionless frequency

$$x = \Omega T_0. \quad (9-85)$$

The signal and noise frequency responses are shown in Figures 73 to 76, both on a linear scale with the magnitudes expressed as pure fractions (Figures 73 and 74), and on a log scale with the magnitudes expressed in dB (Figures 75 and 76). Note that in the limit as  $x \rightarrow 0$

$$\lim_{x \rightarrow 0} \frac{|H(j\Omega)|}{T_0} = 1 \quad (9-86)$$

$$\lim_{x \rightarrow 0} |H_n(j\Omega)| = 0$$

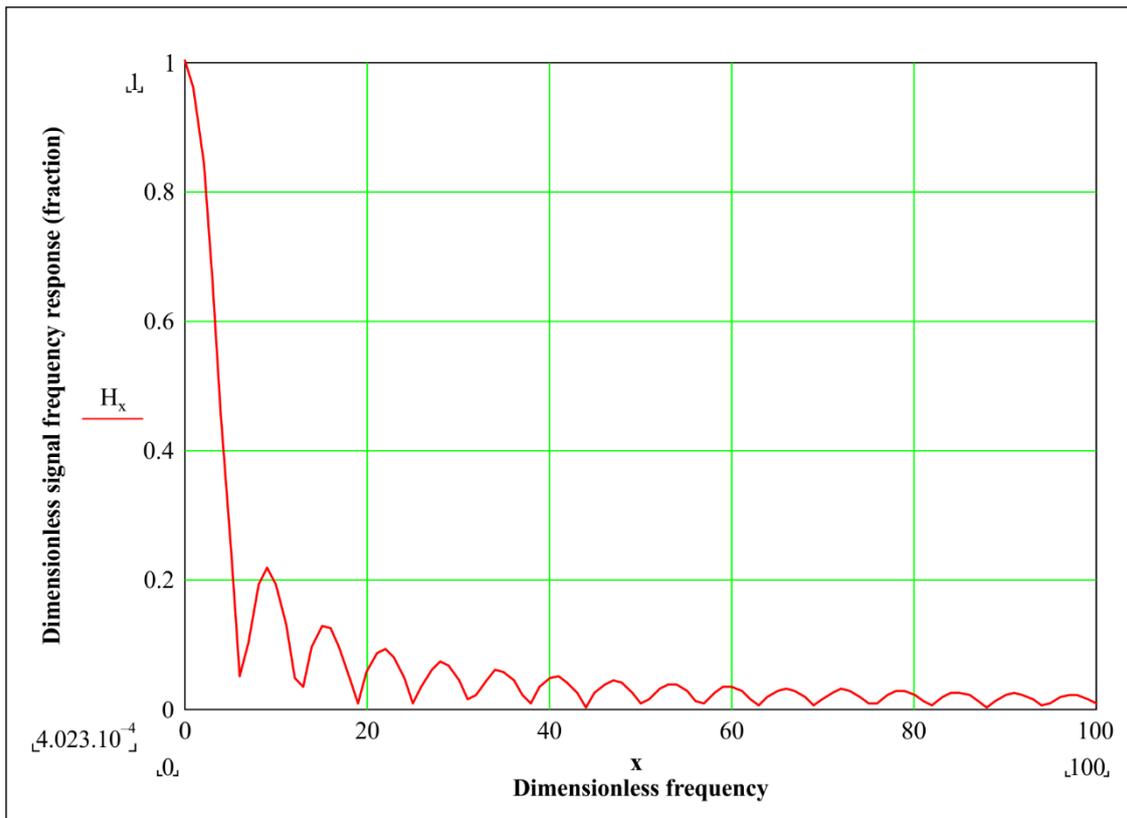
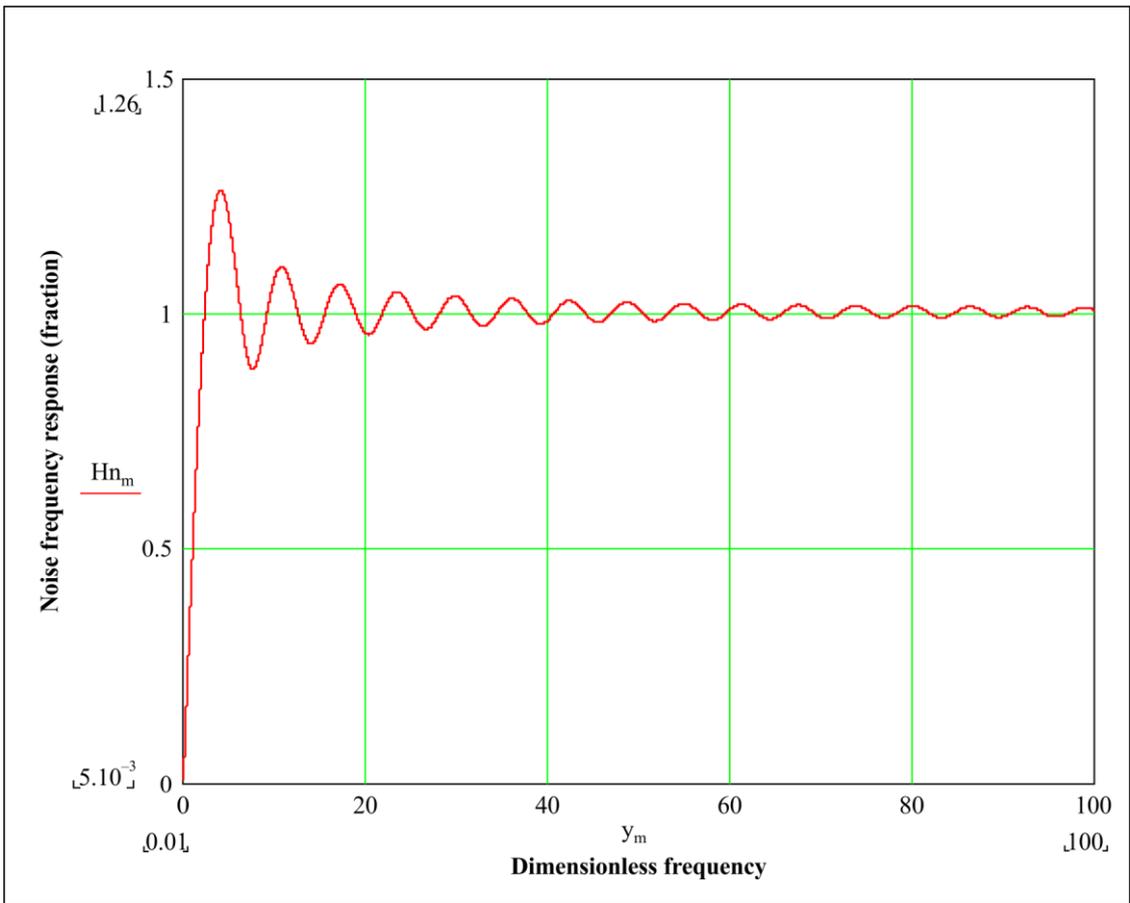
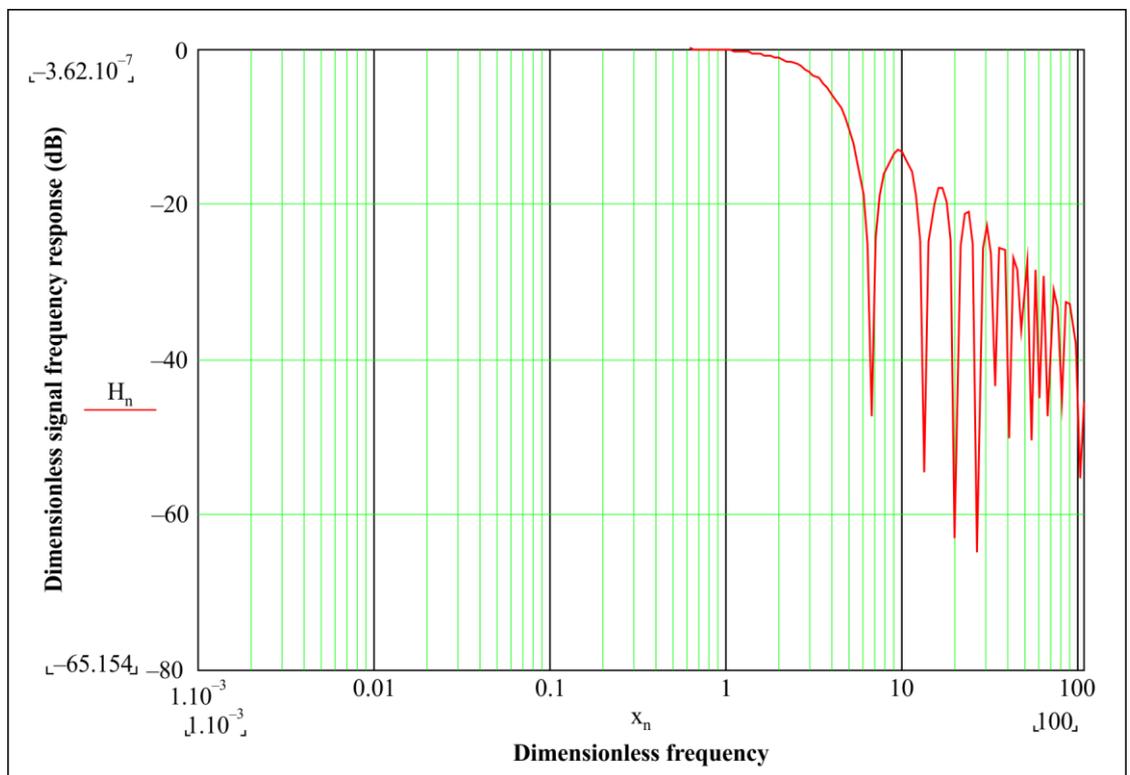


Figure 73 – Example 1 dimensionless signal frequency response (linear scale)



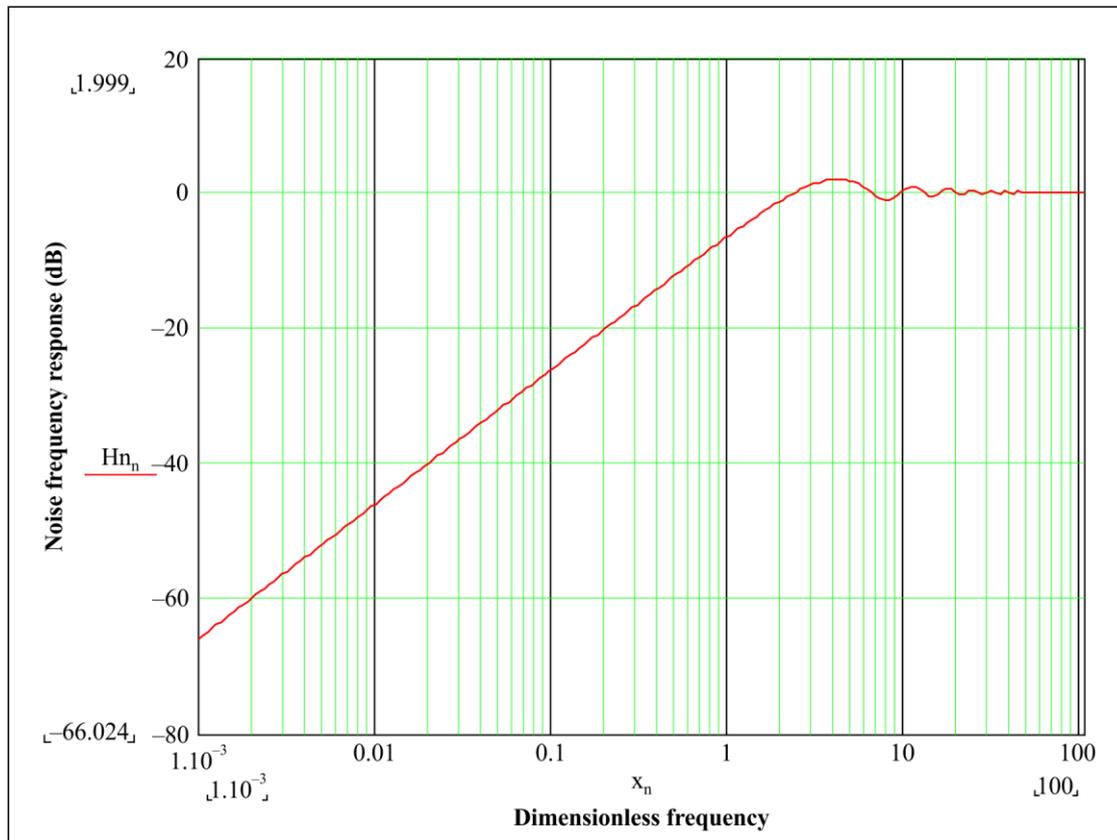
G Suppl.65(18)\_F74

**Figure 74 – Example 1 noise frequency response (linear scale)**



G Suppl.65(18)\_F75

**Figure 75 – Example 1 dimensionless signal frequency response (log scale)**



G Suppl.65(18)\_F76

**Figure 76 – Example 1 noise frequency response (log scale)**

The signal frequency response has a low-pass characteristic, with bandwidth around  $x = 3$  (or  $\Omega = 3/T_0$ ). The noise frequency response has a high-pass characteristic, with corner frequency between  $x = 1$  and  $2$  (i.e.,  $\Omega$  between  $1/T_0$  and  $2/T_0$ ). As expected, there is no low-pass filtering of the quantization noise. Equation (9-55) indicates that, in order for there to be low-pass filtering of the quantization noise, it is necessary that  $K_B(s)$  have a low-pass characteristic. In addition, if  $G_B(s)$  and  $K_B(s)$  are taken to be ideal filters in the range  $-\Omega_0 \leq \Omega \leq \Omega_0$ , i.e.,  $G_B(s) = K_B(s) = 1$  for  $-\Omega_0 \leq \Omega \leq \Omega_0$  and  $G_B(s) = K_B(s) = 0$  for  $|\Omega| > \Omega_0$ , then Eqs. (9-66) and (9-68) hold for  $|x| \leq \pi$ , and the plots of Figures 73 to 76 extend only to the dimensionless frequency of  $\pi$ .

**9.2.4.2 Example 2 – model A with no filtering in the 'Time Measurement, possibly with filtering' and 'Counter and Time offset correction' blocks (GA = KA = 1)**

In this example model A is considered for the case of no filtering, i.e., the filters represented by  $G_A(s)$  and  $K_A(s)$  are not present. Then  $G_A(s) = K_A(s) = 1$ .

The signal and noise transfer functions are, from Eqs. (9-15), (9-24) and (9-25)

$$H(s) = \frac{1 - e^{-sT_0}}{s} = \frac{1 - e^{-sT_0}}{2s} \tag{9-87}$$

$$H_n(s) = 1 - \frac{H(s)}{T_0} = 1 - \frac{1 - e^{-sT_0}}{2sT_0} = \frac{2sT_0 - 1 + e^{-sT_0}}{2sT_0}, \tag{9-88}$$

where the final equality of equation (9-79) has been used to obtain  $[K_A(s) Q_A(s)]^*$  in the denominator of Eqs. (9-40) and (9-41). As in Example 1, it has been assumed (for the purposes of this example)

that the noise contains no frequencies above the Nyquist rate; with this equation (9-49) has been used in equation (9-88) because there is no aliasing.

The signal and noise frequency responses are obtained by setting  $s = j\Omega$  and taking the magnitude (absolute value). The results are (as in example 1, the signal transfer function is divided by  $T_0$  to obtain a dimensionless form)

$$\frac{|H(j\Omega)|}{T_0} = \sqrt{\frac{1 - \cos x}{2x^2}} \quad (9-89)$$

$$|H_n(j\Omega)| = \sqrt{\frac{1 - \cos x + 2x(x - \sin x)}{2x^2}}, \quad (9-90)$$

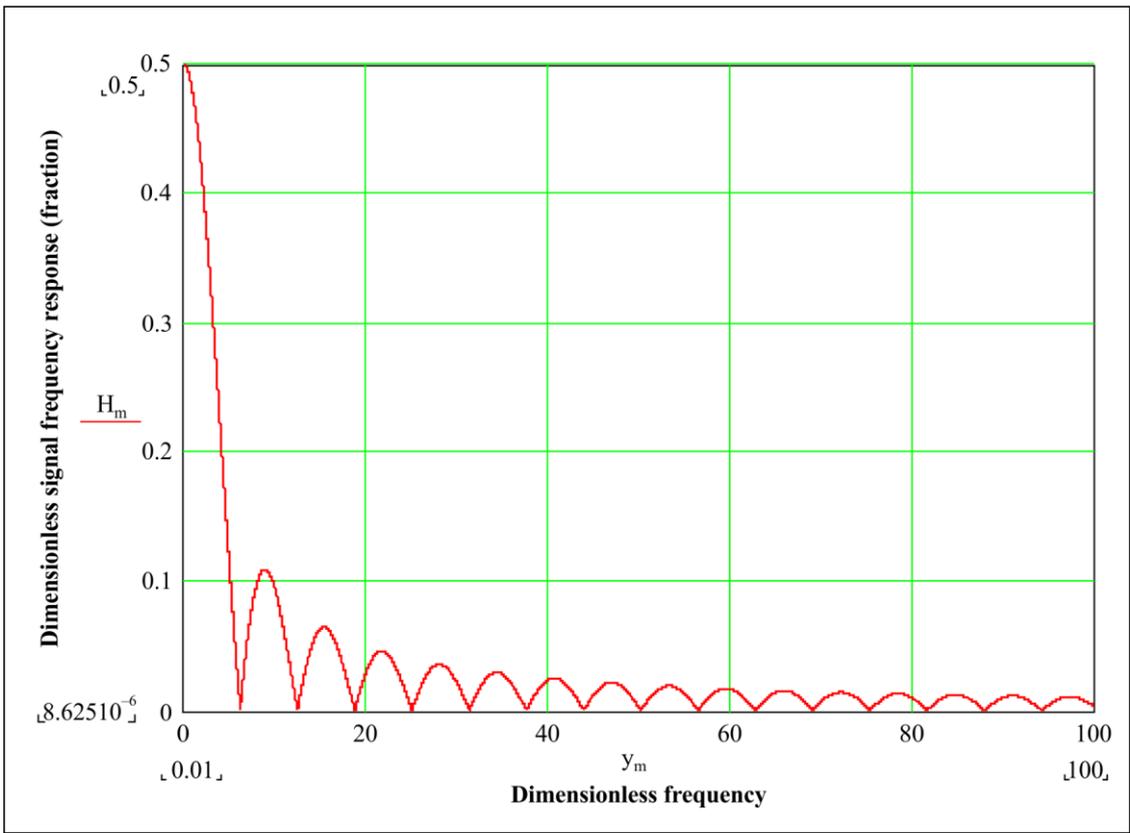
where  $x$  is the dimensionless frequency

$$x = \Omega T_0. \quad (9-91)$$

The signal and noise frequency responses are shown below, both on a linear scale with the magnitudes expressed as pure fractions (Figures 77 and 78), and on a log scale with the magnitudes expressed in dB (Figures 79 and 80). Note that in the limit as  $x \rightarrow 0$

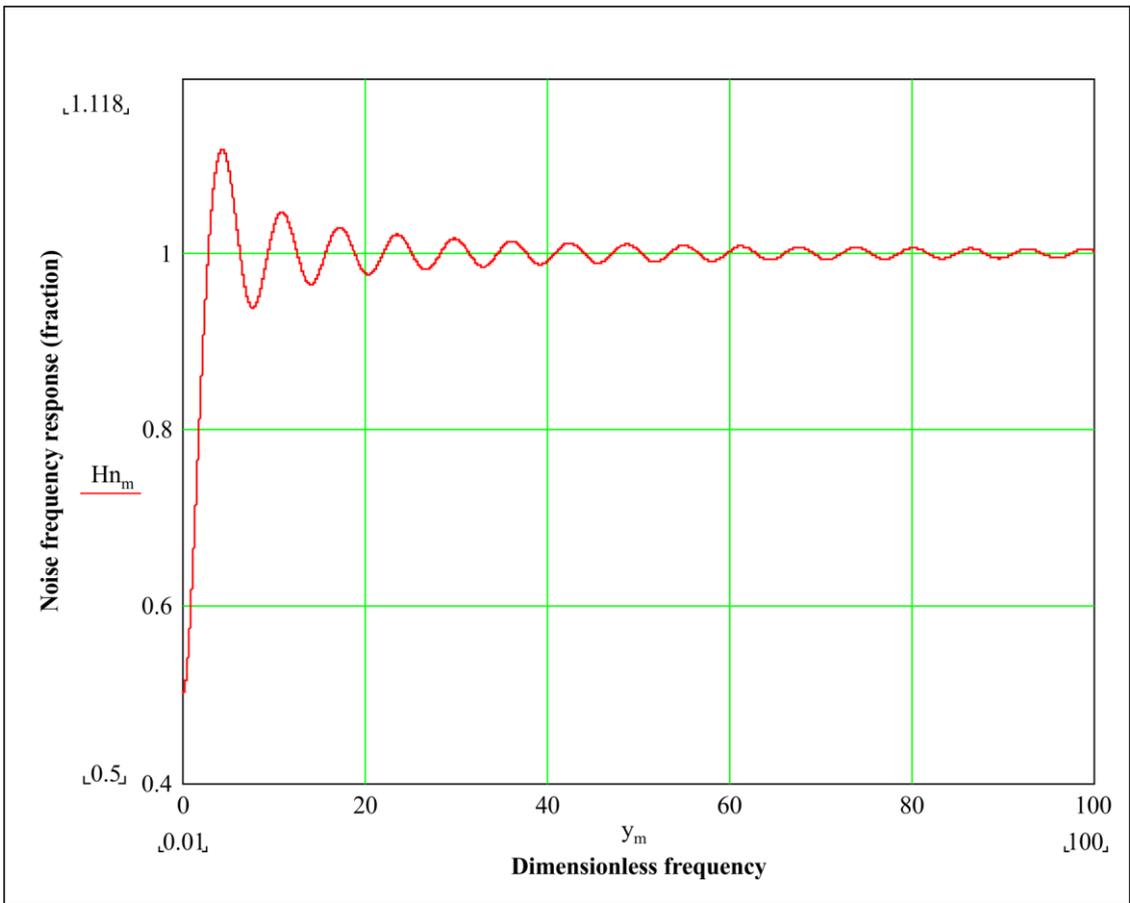
$$\begin{aligned} \lim_{x \rightarrow 0} \frac{|H(j\Omega)|}{T_0} &= \frac{1}{2} \\ \lim_{x \rightarrow 0} |H_n(j\Omega)| &= \frac{1}{2} \end{aligned} \quad (9-92)$$

As in Example 1, the signal frequency response has a low-pass characteristic; however, now there is 6 dB of attenuation at low frequency. The bandwidth is around  $x = 3$ . The noise frequency response has a high-pass characteristic, but now the minimum attenuation at low frequencies is 6 dB.



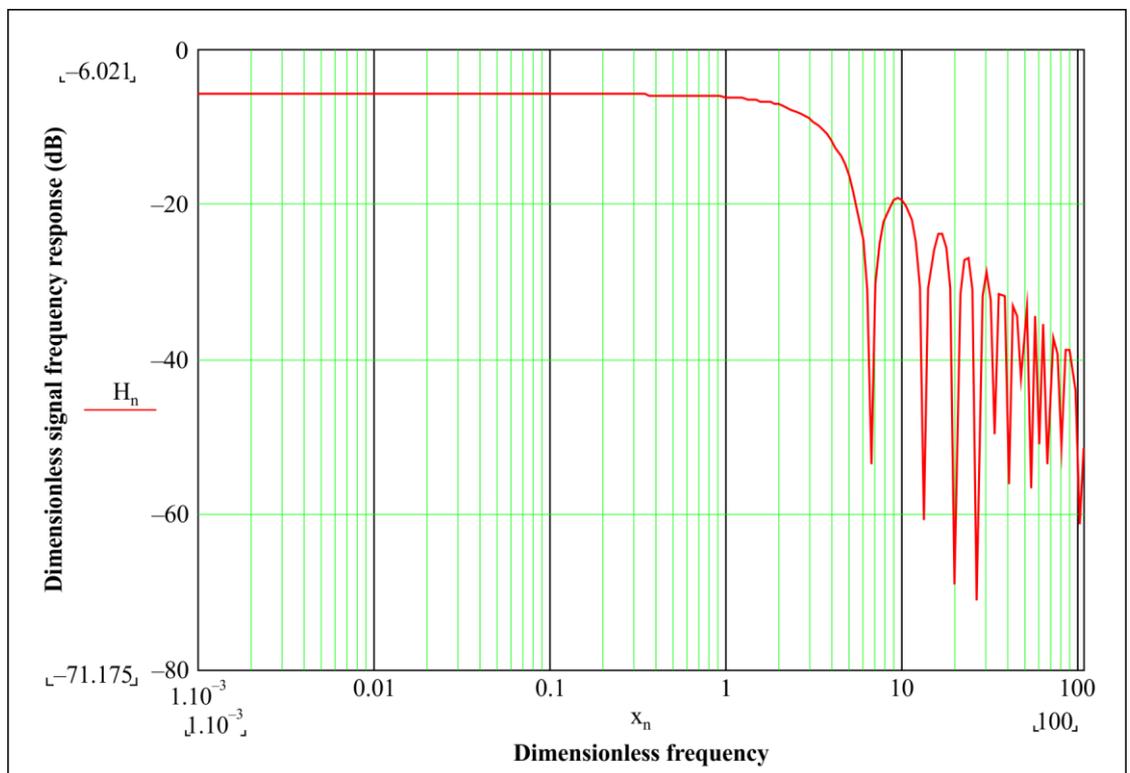
G Suppl.65(18)\_F77

**Figure 77 – Example 2 dimensionless signal frequency response (linear scale)**



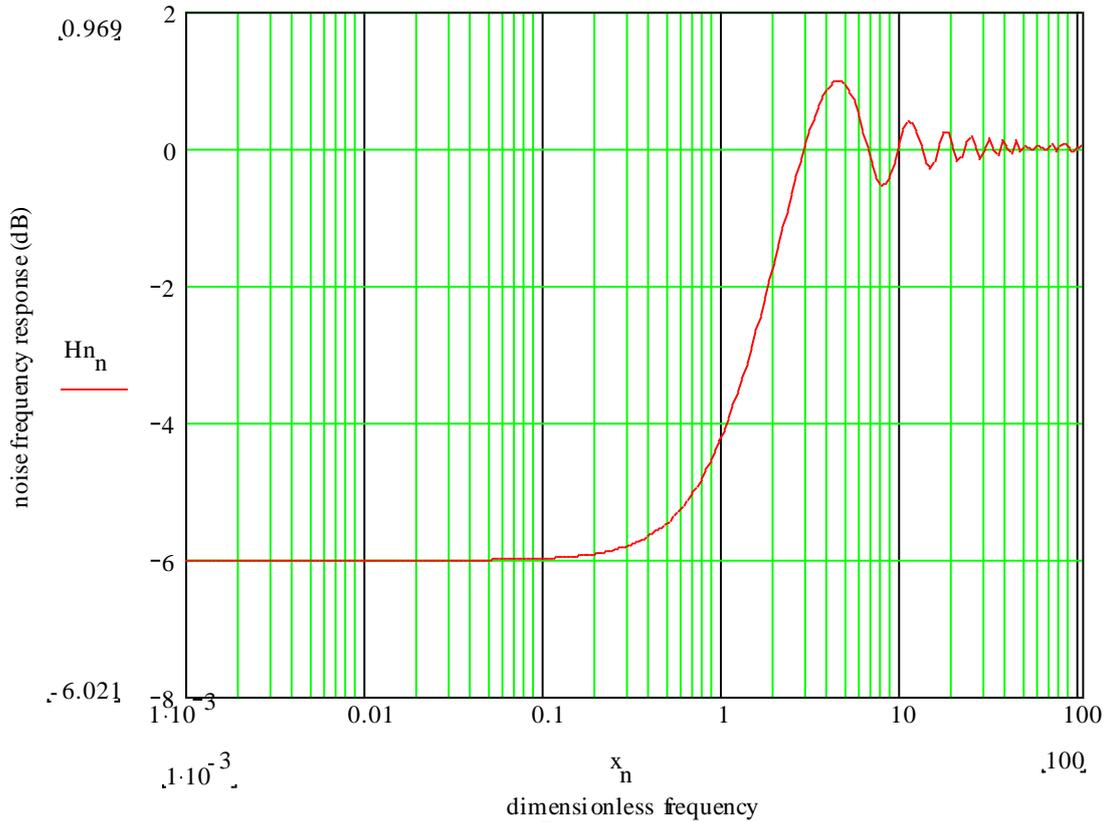
G Suppl.65(18)\_F78

**Figure 78 – Example 2 noise frequency response (linear scale)**



G Suppl.65(18)\_F79

**Figure 79 – Example 2 dimensionless signal frequency response (log scale)**



**Figure 80 – Example 2 noise frequency response (log scale)**

### 9.2.4.3 Example 3 – model B frequency response for case of approximate second-order signal transfer function with 20 dB/decade roll-off

This clause considers a model B example where the signal transfer function is approximately second-order with 20 dB/decade roll-off. Specifically

$$K_B(s) = \frac{2\zeta\Omega_n s + \Omega_n^2}{s^2 + 2\zeta\Omega_n s + \Omega_n^2}, \quad (9-93)$$

$$G_B(s) = 1$$

where  $\Omega_n$  is the analogue undamped natural frequency.

It is also assumed, in this example, that the quantized SyncE phase error,  $N(s)$ , has no frequency components above the Nyquist frequency. In this case,  $M_B(s) = 1$ . Then, using Eqs. (9-53) and (9-54), the signal and noise transfer functions,  $H(s)$  and  $H_n(s)$ , respectively, are

$$H(s) = \frac{2\zeta\Omega_n s + \Omega_n^2}{s^2 + 2\zeta\Omega_n s + \Omega_n^2} \frac{1 - e^{-sT_0}}{s}$$

$$H_n(s) = \left(1 - \frac{1 - e^{-sT_0}}{sT_0}\right) \frac{2\zeta\Omega_n s + \Omega_n^2}{s^2 + 2\zeta\Omega_n s + \Omega_n^2} \quad \text{for } s = \sigma + j\Omega \text{ and } -\Omega_0 \leq \Omega \leq \Omega_0 \quad (9-94)$$

Note that this is only intended as an example. In a real system,  $N(s)$  might contain phase noise components above the Nyquist frequency. For example, the sampling rate for cases 1 – 4 below is 8 Hz, and the sampling rate for cases 5 – 8 below is 1 Hz. The corresponding Nyquist frequencies are 4 Hz and 0.5 Hz, respectively. A real SyncE signal can have wander with frequency components as high as 10 Hz. If  $M_B(s) = 1$  in this case, some of the wander noise would be aliased to frequencies below 10 Hz. In the simulations described in clause 12, the filter  $M_B(s) = 1$ , and the SyncE phase noise input has frequency components above 10 Hz. Therefore, some aliasing of the SyncE wander occurs in

those cases, though some of the aliased wander is subsequently filtered out by the 0.05 Hz endpoint filter (i.e., the filter  $K_B(s)$ ).

Since, in equation (9-53), the signal transfer function depends on the product  $K_B(s)G_B(s)$  rather than on the individual transfer functions, the signal transfer function would be the same if, for example,  $K_B(s) = 1$  and  $G_B(s)$  is taken to be second order with 20 dB/decade roll-off. However, in that case noise transfer function would not have a low-pass factor, and quantization noise (i.e.,  $e_{ts}$ ) components up to the Nyquist frequency would be present. By taking  $K_B(s)$  to have the second-order characteristic, quantization noise components can be filtered. Note that SyncE noise generation is assumed to be zero in clause 9.2 and its subclauses. Models for SyncE noise generation are described in clause 9.3.

This example is a generalization of example 1 of clause 9.2.4.1. In that example, all the model B filter blocks are 1, i.e.,  $K_B(s)$ ,  $M_B(s)$ , and  $G_B(s)$  are not present. In the example here, the signal and noise transfer functions of example 1 are multiplied by a second-order response with 20 dB/decade roll-off.

As in examples 1 and 2, signal and noise frequency responses are computed by setting  $s = j\Omega$  in the transfer functions, where  $\Omega$  is the analogue frequency in rad/s, and taking the magnitude. Defining the digital frequency and digital undamped natural frequency,  $\omega$  and  $\omega_n$  respectively

$$\begin{aligned}\omega &= \Omega T_0 \\ \omega_n &= \Omega_n T_0\end{aligned}\tag{9-95}$$

the resulting frequency responses are

$$\frac{|H(j\Omega)|}{T_0} = \frac{\sqrt{2(1 - \cos \omega)}}{\omega} \frac{\sqrt{4\zeta^2 \left(\frac{\omega}{\omega_n}\right)^2 + 1}}{\sqrt{\left[1 - \left(\frac{\omega}{\omega_n}\right)^2\right]^2 + 4\zeta^2 \left(\frac{\omega}{\omega_n}\right)^2}}\tag{9-96}$$

$$|H_n(j\Omega)| = \frac{\sqrt{\omega^2 - 2(\cos \omega + \omega \sin \omega) + 2}}{\omega} \frac{\sqrt{4\zeta^2 \left(\frac{\omega}{\omega_n}\right)^2 + 1}}{\sqrt{\left[1 - \left(\frac{\omega}{\omega_n}\right)^2\right]^2 + 4\zeta^2 \left(\frac{\omega}{\omega_n}\right)^2}}.\tag{9-97}$$

In example 1, the signal transfer function has a low-pass characteristic, with bandwidth that is on the order of the Nyquist frequency, i.e., the signal transfer function begins to roll off when  $\omega$  is in the range of 1 to 3 (see Figure 75). In addition, the noise transfer function has a high-pass characteristic, with corner frequency on the order of the Nyquist frequency (see Figure 76). In the current example, the signal transfer function has additional low-pass filtering due to the second-order filter  $K_B(s)$ . The noise transfer function has low-pass filtering in addition to the high-pass filtering due to the second-order filter  $K_B(s)$ .

The example is now made more specific by considering sampling times, i.e., sync intervals, of 0.125 s and 1 s, and gain peaking of 0.1 dB. The latter corresponds to damping ratio of 4.3188 (see clauses 8.1 and 8.2.3). Note that this differs from the damping ratio of 4.6465 given in Table VIII.2 of [ITU-T G.8251] because the latter is obtained using an approximation while the value here uses the exact result). For each sampling time, 3 dB bandwidths for  $K_B(s)$  of 0.01 Hz, 0.05 Hz, 0.1 Hz, and 1 Hz are considered. These parameters are summarized in Table 24. The undamped natural frequency is obtained using equation (8-12).

**Table 24 – Summary of parameters for example frequency response cases**

Case	Damping ratio	Sampling time $T_0$ (s)	Sampling rate $1/T_0$ (Hz)	3 dB bandwidth $f_{3dB}$ (Hz)	3 dB bandwidth (digital) = $2\pi f_{3dB}T_0$	Analogue undamped natural frequency $\Omega_n$ (Hz)	Analogue undamped natural frequency $\Omega_n$ (rad/s)	Digital undamped natural frequency $\omega_n = \Omega_n T_0$
1	4.3188	0.125	8	0.01	$7.854 \times 10^{-3}$	$1.1424 \times 10^{-3}$	$7.1780 \times 10^{-3}$	$8.9725 \times 10^{-4}$
2	4.3188	0.125	8	0.05	$3.927 \times 10^{-2}$	$5.7121 \times 10^{-3}$	$3.5890 \times 10^{-2}$	$4.4863 \times 10^{-3}$
3	4.3188	0.125	8	0.1	$7.854 \times 10^{-2}$	$1.1424 \times 10^{-2}$	$7.1780 \times 10^{-2}$	$8.9725 \times 10^{-3}$
4	4.3188	0.125	8	1	0.7854	0.11424	0.71780	$8.9725 \times 10^{-2}$
5	4.3188	1	1	0.01	$6.283 \times 10^{-2}$	$1.1424 \times 10^{-3}$	$7.1780 \times 10^{-3}$	$7.1780 \times 10^{-3}$
6	4.3188	1	1	0.05	0.31416	$5.7121 \times 10^{-3}$	$3.5890 \times 10^{-2}$	$3.5890 \times 10^{-2}$
7	4.3188	1	1	0.1	0.6283	$1.1424 \times 10^{-2}$	$7.1780 \times 10^{-2}$	$7.1780 \times 10^{-2}$
8	4.3188	1	1	1	6.283	0.11424	0.71780	0.71780

Results for signal and noise transfer function, for cases 1 to 4 are shown in Figures 81 and 82, respectively. Results for signal and noise transfer function, for cases 5 to 8 are shown in Figures 83 and 84, respectively. The horizontal axis in each figure is the digital frequency,  $\omega$ , given by equation (9-95). The vertical axis of the signal transfer function plots (Figures 81 and 83) is the quantity  $|H(j\omega)|/T_0$ , given by equation (9-96). It should be recalled that  $|H(j\omega)|$  is the transfer function between  $U^*(s)$ , which is a sampled signal, and  $Y(s)$ , which is an unsampled signal. Consequently, the resulting transfer function contains a factor  $T_0$  and has units of time. Equation (9-96) is divided by  $T_0$  to make the result dimensionless. It may be seen from equation (9-49) that the quantity  $|H(j\omega)|/T_0$  is the transfer function between the analogue signal  $U(s)$ , and  $Y(s)$ , for the case where  $U(s)$  has no frequency components above the Nyquist frequency.

In each figure, the result for no filtering ( $K_B(s) = 1$ ), taken from example 1, is also given. For frequencies well below the Nyquist frequency, the signal transfer function is approximately equal to the second-order frequency response of  $K_B(s)$ . This is given by the second factor (i.e., second fraction) of equation (9-96). This means that in cases where the 3 dB bandwidth of  $K_B(s)$  is well below the Nyquist frequency, i.e., the digital 3 dB bandwidth is much less than  $\pi$ , the frequency response is approximately equal to 1 for frequencies below the 3 dB bandwidth (with 0.1 dB gain peaking just below the 3 dB bandwidth) and rolls off at 20 dB/decade above the 3 dB bandwidth until the frequency approaches the Nyquist frequency. As the frequency approaches the Nyquist frequency, the frequency response for the no-filtering case begins to fall off, and the overall frequency response is less than that given by  $K_B(s)$  (i.e., the second fraction of equation (9-96)). However, if the 3 dB bandwidth of  $K_B(s)$  is well below the Nyquist frequency, the frequency response corresponding to  $K_B(s)$  is already reduced by 30 dB or more when the no-filtering response begins to roll off. Therefore, for cases where the 3 dB bandwidth of  $K_B(s)$  is well below the Nyquist frequency, the no-filtering response, i.e., the first fraction of equation (9-96) may be neglected, and the signal frequency response may be approximated by the frequency response of  $K_B(s)$ .

The question arises as to what the specific criterion is for the 3 dB bandwidth of  $K_B(s)$  to be well below the Nyquist frequency. It is desirable to set the criterion such that  $|K_B(j\omega)|$  is much less than the no-filtering frequency response when the no-filtering frequency response is 3 dB down. The no-filtering frequency response is down by approximately 3 dB when the dimensionless frequency  $\omega$  is approximately equal to 2.8. Then, at this frequency, it is desirable for  $|K_B(j\omega)|$  to be much less than 3 dB. For example, if 20 dB is used for the criterion, i.e., if  $|K_B(j\omega)| < -20 \text{ dB} - 3 \text{ dB} = -23 \text{ dB}$  is considered sufficiently small compared to  $-3 \text{ dB}$ , it is seen from Figure 81, for 0.125 s sampling rate, that the 0.01 Hz, 0.05 Hz, and 0.1 Hz bandwidth cases (cases 1 – 3) meet the criterion, but the 1 Hz bandwidth case (case 4) does not. Likewise, for 1 s sampling rate (Figure 83), the 0.01 Hz and 0.05 Hz

cases (cases 5 – 6) meet the criterion (case 6 just barely), but the 0.1 Hz and 1 Hz cases (cases 7 – 8) do not.

The noise transfer function results in Figures 82 and 84 indicate that the filter  $K_B(s)$  removes a significant amount of high-frequency quantization noise compared to the no-filtering case. The maximum of the noise transfer function is determined by the product of the low-pass  $|K_B(j\omega)|$  and the high-pass no-filtering noise frequency response. For cases where the 3 dB bandwidth of  $K_B(s)$  is much less than the Nyquist frequency, an approximate, simple expression for the maximum of the noise frequency response can be determined. In this case, the maximum occurs at a frequency that is in between the 3 dB bandwidth of  $|K_B(j\omega)|$  and the Nyquist frequency. For frequencies in this range, note that  $\omega \ll 1$  and expand the numerator of the first fraction of equation (9-97) to 2<sup>nd</sup> order in  $\omega$ . In addition, note that  $\omega \gg \omega_n$  and keep only the largest terms in the numerator and the denominator of the second fraction of equation (9-97). The resulting approximation to equation (9-81), for  $\omega_n \ll \omega \ll 1$ , is

$$|H_n(j\Omega)| \cong \frac{\sqrt{\omega^2 - 2\left(1 - \frac{\omega^2}{2} + \frac{\omega^4}{12} + \omega^2 - \frac{\omega^4}{6}\right) + 2}}{\omega} \frac{\sqrt{4\zeta^2 \left(\frac{\omega}{\omega_n}\right)^2}}{\sqrt{\left[\left(\frac{\omega}{\omega_n}\right)^2\right]^2}} \quad (9-98)$$

$$= \frac{\omega}{\sqrt{6}} \frac{2\zeta\omega_n}{\omega} = \frac{2\zeta\omega_n}{\sqrt{6}}$$

Expressed in dB, this becomes

$$20 \log_{10} |H_n(j\Omega)| \cong 20 \log(2\zeta\omega_n) - 7.8. \quad (9-99)$$

A further simplification may be made by noting that the digital 3 dB bandwidth is approximately equal to  $2\zeta\omega_n$  (see equation (IV.2-30) of ITU-T G.8251); the result is

$$20 \log_{10} |H_n(j\Omega)| \cong 20 \log(\omega_{3dB}) - 7.8. \quad (9-100)$$

For example, applying equation (9-100) to case 5, with  $\omega_{3dB} = 0.06283$  (from Table 24) produces – 2 dB. This is close to the –30 dB maximum for the noise frequency response in Figure 84. The approximation becomes less accurate as the 3 dB bandwidth approaches the Nyquist frequency, because it depends on the condition  $\omega_n \ll \omega \ll 1$ .

The results also indicate that there is no benefit to using a filter function  $K_B(s)$  whose 3 dB bandwidth is larger than the Nyquist frequency. In this case, the magnitude of  $K_B(s)$  is very close to 1 for frequencies below the Nyquist frequency. Any further attenuation of the signal or noise (i.e., quantized SyncE noise) below the Nyquist frequency compared to the case of no filter function would be negligible in this case.

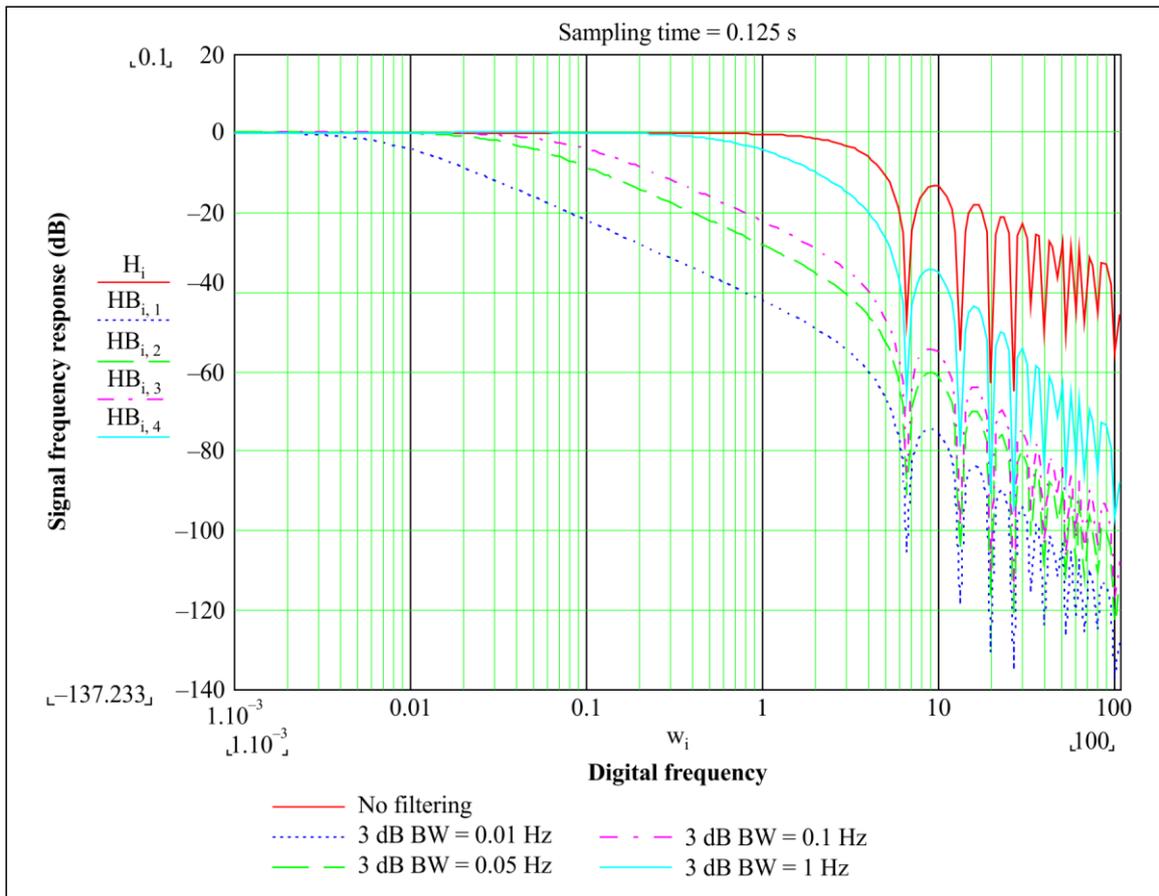
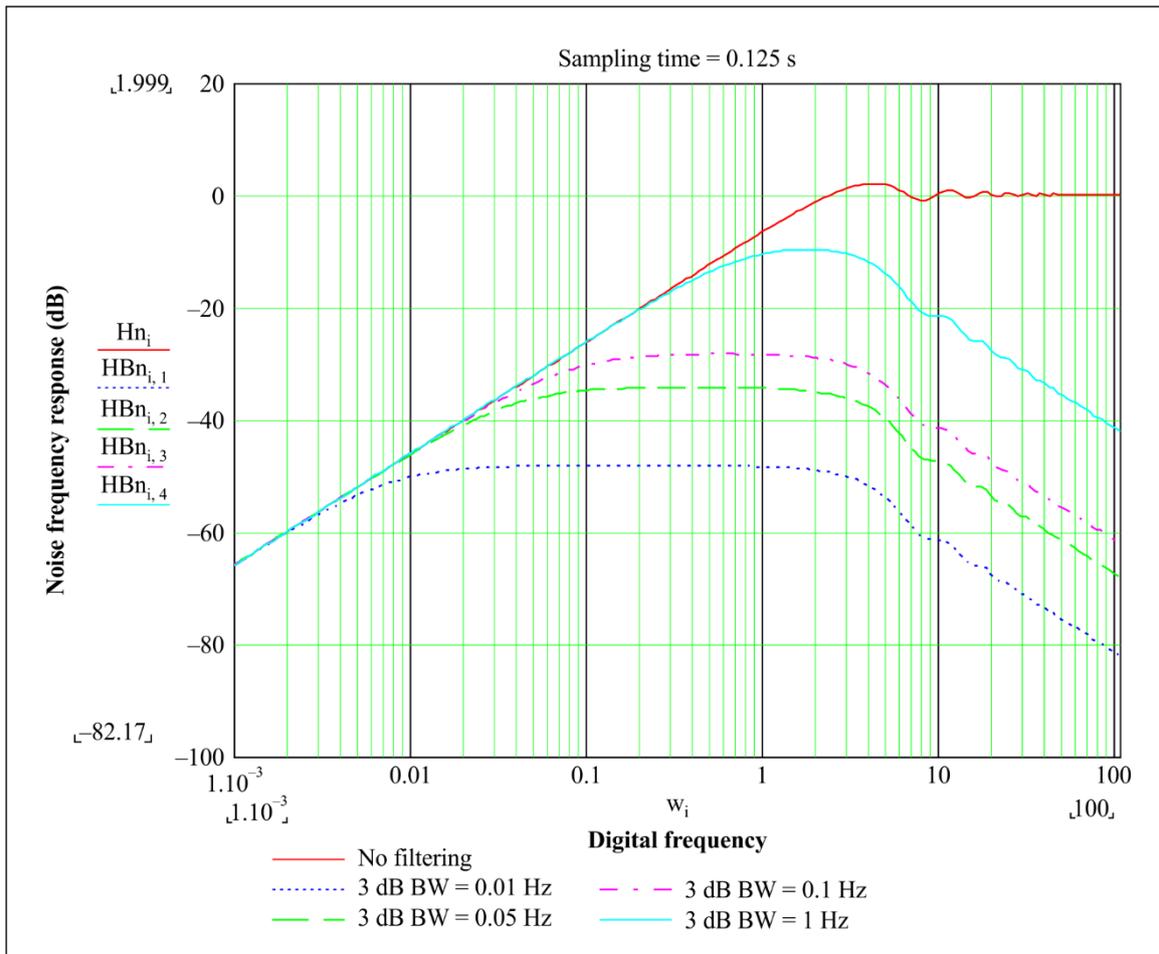
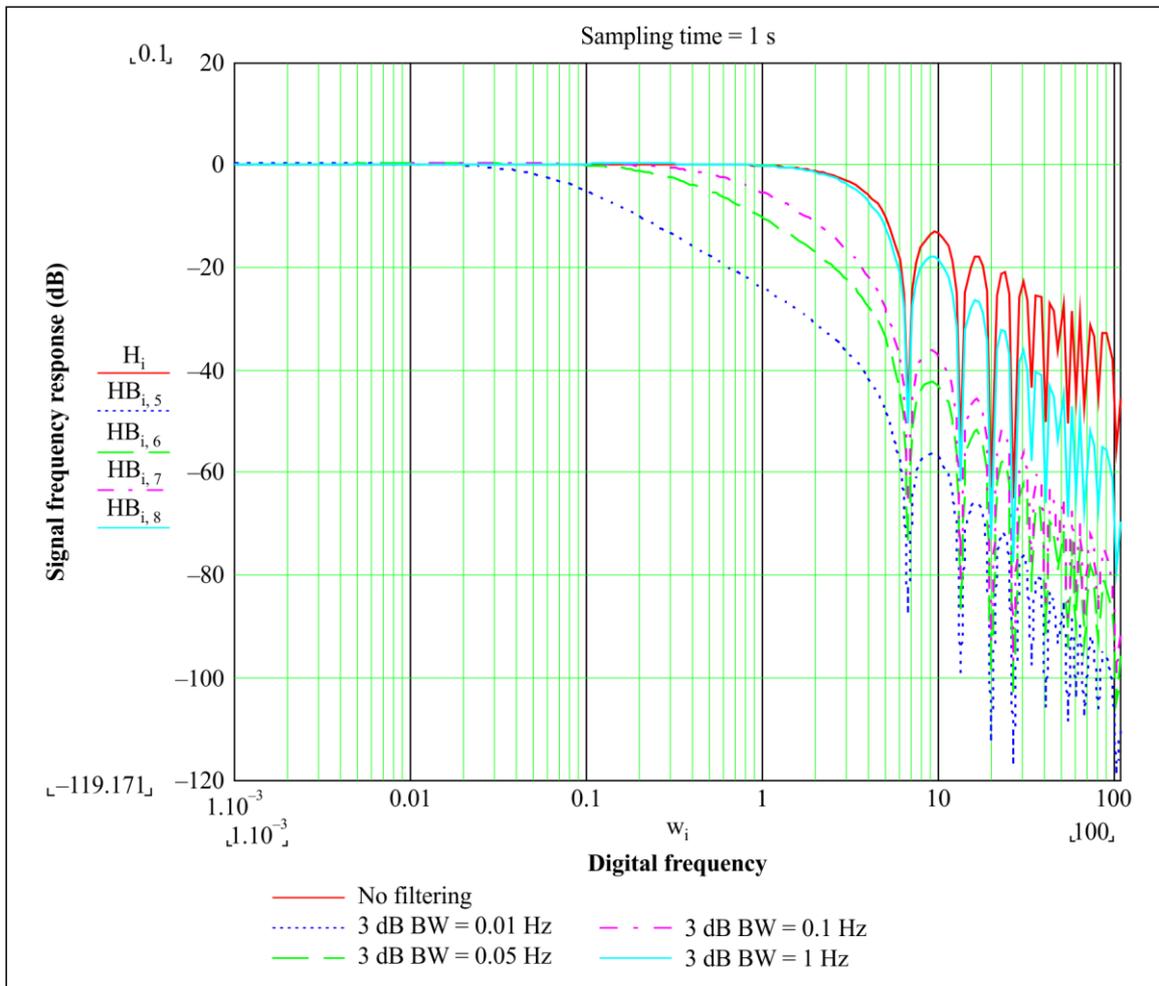


Figure 81 – Signal transfer function for cases 1 – 4 (sampling time = 0.125 s)



G Suppl.65(18)\_F82

**Figure 82 – Noise transfer function for cases 1 – 4 (sampling time = 0.125 s)**



G Suppl.65(18)\_F83

**Figure 83 – Signal transfer function for cases 5 – 8 (sampling time = 1 s)**

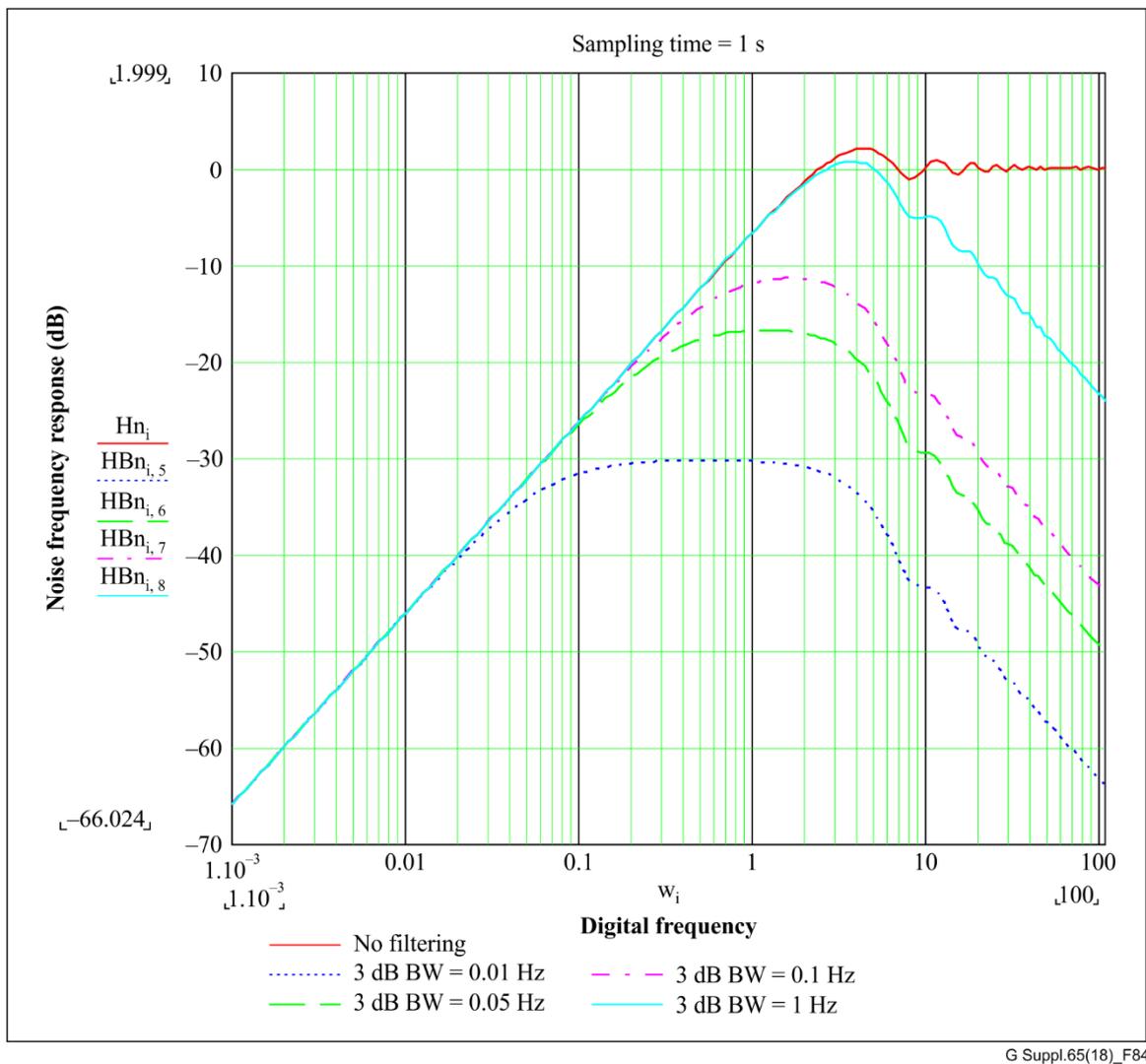


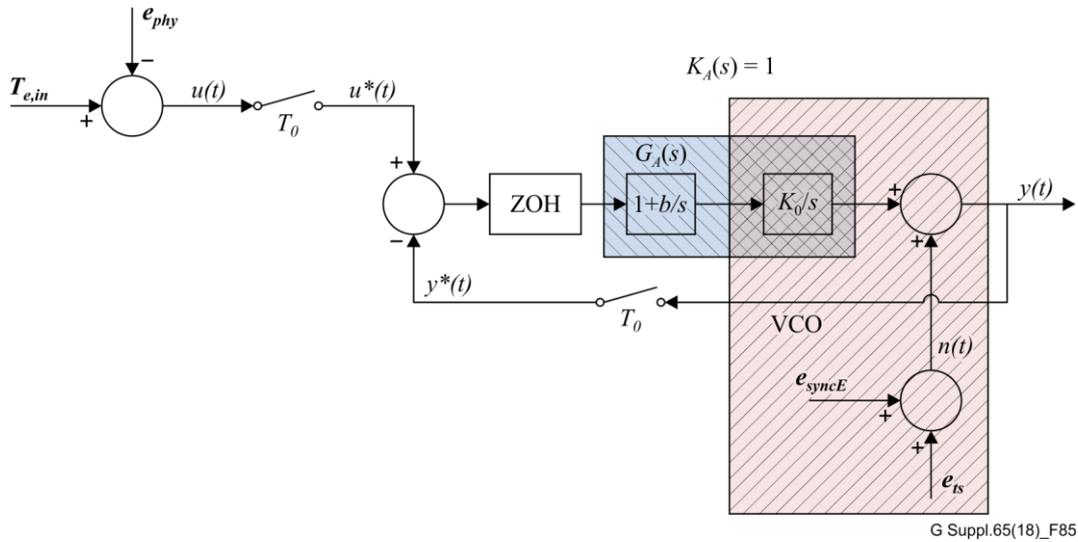
Figure 84 – Noise transfer function for cases 5 – 8 (sampling time = 1 s)

### 9.3 T-BC and T-TSC model that includes SyncE noise generation

#### 9.3.1 Special case of model A – second-order PLL example

The model A filters in Figure 70 are indicated in the most general way, as transfer functions  $G_A(s)$  and  $K_A(s)$ . However, since in model A there is a local oscillator whose physical frequency is adjusted based on the difference between  $u^*(t)$  and  $y^*(t)$ , there must be a VCO function present. In one possible instance of model A, this can be considered to be included in  $G_A(s)$ . In addition, since the difference  $u^*(t) - y^*(t)$  that is output from the subtraction block is a sampled value, the ZOH actually should precede  $G_A(s)$  (but changing the order of the ZOH and  $G_A(s)$  does not change the closed loop transfer function). To gain insight, the special case where the closed loop model A corresponds to a second-order PLL is considered, prior to considering the general case (in clause 9.3.2). The second-order PLL is obtained by considering the case where  $G_A(s)$  includes a PI filter that precedes the VCO function, and assuming that the filter  $K_A(s)$  is not present (i.e.,  $K_A(s) = 1$ ). The resulting special case of model A is shown in Figure 85. In this figure, the VCO frequency is assumed to be synthesized from the SyncE input. The model for the VCO consists of an integrator,  $K_o/s$ , which converts the desired frequency to a phase in the mathematical model, and a noise generation equal to the quantized SyncE phase noise. The noise generation is additive. Figure 85 illustrates that the integrator  $K_o/s$  is part of the general filter function  $G_A(s)$  and is also part of the VCO model. Note that, because  $K_A(s) = 1$ , the SyncE noise in Figure 85 appears as noise generation, i.e., it will be seen shortly that

the noise transfer function has a high-pass characteristic. It must be stressed that Figure 86 is only one particular instance of model A, which is being used as an example.



**Figure 85 – Special case of model A that corresponds to a second-order PLL**

The closed loop signal and noise transfer functions for model A are given by Eqs. (9-40), (9-41) and (9-31). In the case where the filtered noise signal  $K_A(s)N(s)$  has no frequency components above the Nyquist frequency, the noise transfer function simplifies to equation (9-49). For the special case of Figure 85, the filters  $G_A(s)$  and  $K_A(s)$  are given by

$$K_A(s) = 1$$

$$G_A(s) = \left(1 + \frac{b}{s}\right) \frac{K_o}{s} \quad (9-101)$$

Then the signal transfer function is given by (substituting equation (9-101) into equations (9-31) and (9-40))

$$H_A(s) = \frac{\left(1 + \frac{b}{s}\right) \frac{K_o}{s} \frac{1 - e^{-sT_0}}{s}}{1 + \left[\left(1 + \frac{b}{s}\right) \frac{K_o}{s} \frac{1 - e^{-sT_0}}{s}\right]^*} \quad (9-102)$$

It is shown in [b-Franklin] and [b-Ogata] that

$$[(1 - e^{-sT_0})B(s)]^* = (1 - e^{-sT_0})B^*(s) \quad (9-103)$$

Then, to evaluate equation (9-102), the star transforms of  $1/s^2$  and  $1/s^3$  must be evaluated. These are computed in Table 2-1 of [b-Ogata] as

$$\left(\frac{1}{s^2}\right)^* = \frac{T_0 e^{-sT_0}}{(1 - e^{-sT_0})^2} \quad (9-104)$$

$$\left(\frac{1}{s^3}\right)^* = \frac{T_0^2 e^{-sT_0} (1 + e^{-sT_0})}{2(1 - e^{-sT_0})^3} \quad (9-105)$$

Then the signal transfer function is obtained by substituting Eqs. (9-104) and (9-105) into equation (9-102)

$$H_A(s) = \frac{\left(1 + \frac{b}{s}\right) \frac{K_o}{s} \frac{1 - e^{-sT_0}}{s}}{1 + (1 - e^{-sT_0}) \left( \frac{K_o T_0 e^{-sT_0}}{(1 - e^{-sT_0})^2} + \frac{K_o b T_0^2 e^{-sT_0} (1 + e^{-sT_0})}{2(1 - e^{-sT_0})^3} \right)}. \quad (9-106)$$

This may be rewritten as

$$H_A(s) = \frac{(K_o s + K_o b) \frac{1 - e^{-sT_0}}{s}}{s^2 + \frac{K_o T_0 e^{-sT_0} s^2}{1 - e^{-sT_0}} + \frac{K_o b T_0^2 e^{-sT_0} (1 + e^{-sT_0}) s^2}{2(1 - e^{-sT_0})^2}}. \quad (9-107)$$

Equation (9-107) may be related to the analogue transfer function for a second-order system with 20 dB/decade roll-off by expanding each term to the lowest non-vanishing power of  $s$

$$\begin{aligned} \frac{1 - e^{-sT_0}}{s} &\approx \frac{1 - (1 - sT_0)}{s} = T_0 \\ \left(\frac{1 - e^{-sT_0}}{s}\right)^2 &\approx T_0^2 \\ e^{-sT_0} &\approx 1 - sT_0 \approx 1 \end{aligned} \quad (9-108)$$

Then

$$\begin{aligned} H_A(s) &\approx \frac{(K_o s + K_o b) T_0}{s^2 + K_o s + K_o b} = \frac{2\zeta\Omega_n s + \Omega_n^2}{s^2 + 2\zeta\Omega_n s + \Omega_n^2} T_0 \\ \Omega_n &= \sqrt{K_o b} \\ \zeta &= \frac{1}{2} \sqrt{\frac{K_o}{b}} \end{aligned} \quad (9-109)$$

The factor  $T_0$  occurs in equation (9-109) because  $H_A(s)$  relates a sampled input to an unsampled output.

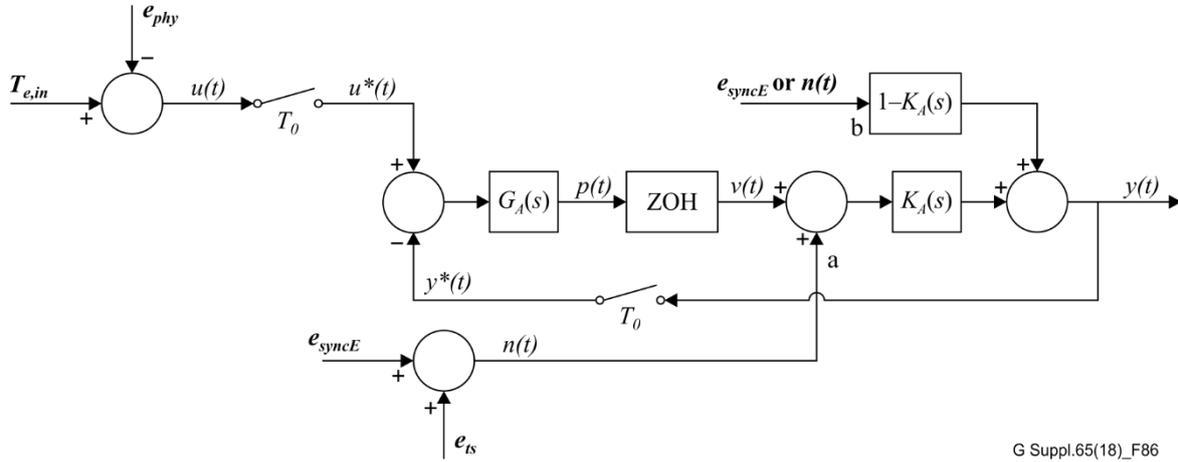
The noise transfer function, for the case where the noise signal  $N(s)$  has no frequency components above the Nyquist frequency, is given by equation (9-49) with  $K_A(s) = 1$ , i.e.,

$$\begin{aligned} H_{n,A}(s) &= 1 - \frac{H_A(s)}{T_0} = \frac{s^2}{s^2 + 2\zeta\Omega_n s + \Omega_n^2} \\ \text{for } s &= \sigma + j\Omega \text{ and } -\Omega_0 \leq \Omega \leq \Omega_0 \end{aligned} \quad (9-110)$$

### 9.3.2 Noise generation in $K_A(s)$ in model A

In clause 9.3.1, model A noise generation is considered for the special case where  $K_A(s) = 1$  and  $G_A(s)$  is chosen to obtain the special case of a second-order PLL for the closed-loop transfer function  $H_A(s)$ . In this clause, the more general case where  $K_A(s)$  is present (i.e.,  $K_A(s) \neq 1$ ) and contains noise generation is considered. For simplicity, the noise generation in  $K_A(s)$  is assumed to be introduced as it would occur in a second-order PLL. Note that this is not the same as the example of clause 9.3.1. In that example,  $K_A(s) = 1$  and  $H_A(s)$  is a second-order PLL.

In model A, the filter  $K_A(s)$  is present to further filter the SyncE phase noise and quantization noise. If  $K_A(s)$  were a second-order PLL, the noise generation would be introduced as shown in Figure 86. As indicated in the previous paragraph, it is assumed that the noise generation is introduced in this manner. Whether the noise generation is either  $e_{syncE}$  or  $n(t)$  depends on the granularity of the VCO function contained in  $K_A(s)$ . The discussion of this point is deferred to clause 9.3.3, where model B noise generation is discussed and the same issue arises for  $K_B(s)$ . At present, only the transfer function is being computed.



**Figure 86 – Model A, with noise generation in the filter  $K_A(s)$**

Let the transfer function between the noise generation at point b, i.e., the  $K_A(s)$  noise generation, and the output be  $H_{n2,A}(s)$ . The transfer function between point a and the output is  $H_{n,A}(s)$ , given by equation (9-41). It is seen from Figure 86 that the transfer function between point b and the output may be obtained from the transfer function between point a and the output by multiplying the latter by  $[1 - K_A(s)]/K_A(s)$ . Then

$$H_{n2,A}(s) = \left( 1 - H_A(s) \frac{[K_A(s)N(s)]^*}{K_A(s)N(s)} \right) (1 - K_A(s)). \quad (9-111)$$

Then, for the case where the noise generation at point b is  $n(t)$ , the total noise transfer function,  $H_{nT,A}(s)$ , is

$$H_{nT,A}(s) = H_{n,A}(s) + H_{n2,A}(s) = \left( 1 - H_A(s) \frac{[K_A(s)N(s)]^*}{K_A(s)N(s)} \right). \quad (9-112)$$

For the case where the filtered noise signal  $K_A(s)N(s)$  has no frequency components above the Nyquist frequency, equation (9-112) simplifies to

$$H_{nT,A}(s) = 1 - \frac{H_A(s)}{T_0} \quad (9-113)$$

for  $s = \sigma + j\Omega$  and  $-\Omega_0 \leq \Omega \leq \Omega_0$

It is seen that in this case the input signal is filtered with transfer function  $H_A(s)$ , which is a low-pass filter, and the noise is filtered with transfer function  $1 - H_A(s)/T_0$ , which is a high-pass filter.  $H_A(s)$  is given by equation (9-40), which is

$$H_A(s) = \frac{K_A(s)Q_A(s)}{1 + [K_A(s)Q_A(s)]^*}, \quad (9-114)$$

where

$$Q_A(s) = G_A(s) \frac{1 - e^{-sT_0}}{s}. \quad (9-115)$$

In the case where the noise generation at point b is  $e_{syncE}$ , separate noise transfer functions must be computed for  $e_{syncE}$  to  $y(t)$  and  $e_{ts}$  to  $y(t)$ . The transfer function for  $e_{syncE}$  to  $y(t)$  is given by equation (9-112), where  $N(s)$  is the Laplace transform of  $e_{syncE}(t) + e_{ts}(t)$ , i.e.,  $E_{syncE}(s) + E_{ts}(s)$ . For the case where the filtered noise signal  $K_A(s)[E_{syncE}(s) + E_{ts}(s)]$  has no frequency components above the Nyquist frequency, equation (9-112) simplifies to equation (9-113). The transfer function for  $e_{ts}$  to  $y(t)$  is given by equation (9-40), where  $N(s)$  is also the Laplace transform of  $e_{syncE}(t) + e_{ts}(t)$ , i.e.,  $E_{syncE}(s) + E_{ts}(s)$ . For the case where the filtered quantization noise signal  $K_A(s)[E_{syncE}(s) + E_{ts}(s)]$  has no frequency components above the Nyquist frequency, equation (9-40) simplifies to equation (9-41).

### 9.3.3 Noise generation in model B

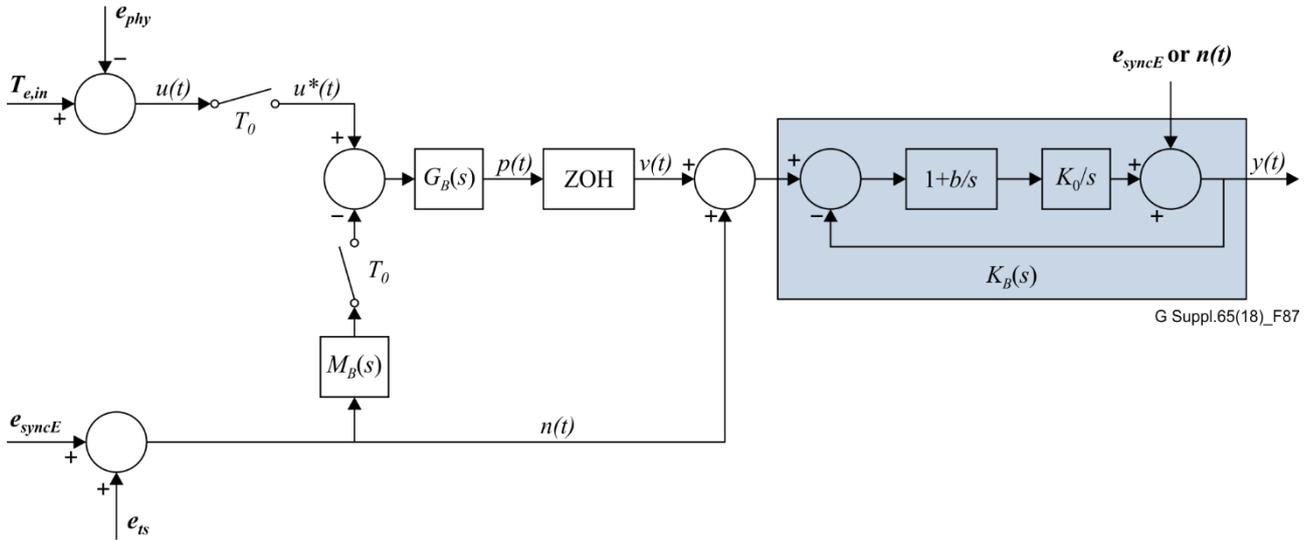
In model B, the physical frequency of the local clock is not adjusted. Instead, the output timestamp values, computed at the master ports, are adjusted based on the incoming timestamp on the slave port. The local clock in this model is the incoming quantized SyncE signal, and timestamping is done relative to this clock. In Figure 72, the incoming time error  $T_{e,in}$  is incoming on the slave port; as in model A, the subtraction of  $e_{phy}$  to produce  $u(t)$ , and the sampling of  $u(t)$  to produce  $u^*(t)$ , are done at the slave port. Also as in model A, the timestamp of the incoming Sync message is taken at the slave port. However, in model B this timestamp is taken relative to the quantized SyncE signal, which is the local clock here (unlike model A, where the local clock is adjusted). In model B, the quantized SyncE clock may be filtered prior to timestamping. The filtering function is  $M_B(s)$ . The difference between the incoming time error and timestamp error value represents the adjustment to the local clock time value needed to synchronize the local clock to the incoming clock on the slave port (i.e., to the grandmaster time). While this adjustment could be made instantaneously, in a more general case it could be filtered by a filter  $G_B(s)$ . A zero-order hold is also shown in the model. As in model A, this should actually precede the filter  $G_B(s)$  because the output of the subtraction operation is a sampled signal. However, as in model A, interchanging the order of  $G_B(s)$  and the ZOH does not change the signal or noise transfer functions. The quantized SyncE clock error signal is added to the filtered adjustments to produce the network clock (i.e., estimate of grandmaster) phase error signal. This can be further filtered by a filter function  $K_B(s)$ .

The filter  $K_B(s)$  filters a clock signal that possibly experiences jumps at the instants of Sync message arrivals on the slave port (but the jumps have been filtered by  $G_B(s)$ ). The frequency of this clock signal in between the Sync message arrival instants is the frequency of the quantized SyncE signal. Therefore, SyncE phase error,  $e_{syncE}$ , is present on the input to  $K_B(s)$ . In addition, since the SyncE signal is quantized, the quantization error  $e_{ts}$  is present. If it is desired to filter the SyncE phase noise  $e_{syncE}$ , then  $K_B(s)$  must be a PLL with a VCO with noise generation that is smaller than  $e_{syncE}$ , i.e., the VCO of  $K_B(s)$  must be inherently more stable than the SyncE signal itself. If it is not, then the noise generation added by the filter will exceed the noise reduction due to filtering the SyncE noise, and the result will be net noise addition. To achieve this,  $K_B(s)$  would need to have a narrower bandwidth and smaller noise generation than an EEC, e.g.,  $K_B(s)$  could have the characteristics of an SSU (or, at least, a clock that is more stable than an EEC). Since it is not assumed that every T-BC or T-TSC is collocated with an SSU function, it will be assumed that the noise  $e_{syncE}$  is not further filtered by  $K_B(s)$ . Regarding the quantization noise  $e_{ts}$ , this could be further filtered by  $K_B(s)$  even if  $e_{syncE}$  is not further filtered by  $K_B(s)$ . This could be done, for example, by choosing  $K_B(s)$  to be a PLL whose time base is obtained from the SyncE signal but whose nominal frequency is sufficiently high that  $e_{ts}$  is small compared to  $e_{syncE}$ .<sup>11</sup> For example, if the bandwidth of  $K_B(s)$  is between 0.01 Hz and 0.1 Hz, the relevant time scale for  $e_{syncE}$  is on the order of the PLL time constant (i.e., the reciprocal of the 3 dB

---

<sup>11</sup> The time base of  $K_B(s)$  in this example might be obtained from the SyncE signal using a clock multiplier PLL whose bandwidth is relatively wide compared to the bandwidth of  $K_B(s)$ .

bandwidth expressed in rad/s), which is 1.6 to 16 s for this range of bandwidth. The accumulated SyncE phase noise, i.e., MTIE, over this range of observation interval, is on the order of 80 to 140 ns (see Figure 46). It appears that a VCO frequency for  $K_B(s)$  of 125 MHz, resulting in a quantization of 8 ns, would produce quantization error that is small compared to  $e_{syncE}$ . The resulting model B, with noise generation added to  $K_B(s)$ , is shown in Figure 87. For now, both noise generation cases (i.e., noise generation due only to SyncE phase noise and noise generation due to both SyncE phase noise and quantization) are shown. However, for cases of practical interest the noise generation due to SyncE phase noise is much larger than that due to quantization.



**Figure 87 – Model B, with the filter  $K_B(s)$  a second-order PLL with noise generation equal to either  $e_{syncE}$  or  $e_{syncE}+e_{ts}$**

As in model A, the output time is sent to each output (master) port ( $y(t)$  is the error in the output time, in the model); This is not shown in Figure 87. In addition, also as in model A, Figure 87 does not show the intra-node delay. This would be added to the SyncE signal after being filtered by  $M_B(s)$ . If the output (master) ports were shown, they would also have an intra-node delay added.

The signal and noise transfer functions for model B (Figure 72) are given by Eqs. (9-36) to (9-38). For the case where the filtered noise signal  $M_B(s)N(s)$  has no frequency components above the Nyquist frequency, equation (9-54) simplifies to equation (9-55). However, equation (9-55) does not account for noise generation added within  $K_B(s)$ . It assumes that the only SyncE phase noise input (with or without quantization) is as shown in Figure 72. If  $K_B(s)$  has the form shown in Figure 87, an additional noise term must be added to the output, given by either  $H_{n2,B}(s)N(s)$  or  $H_{n2,B}(s)E_{syncE}(s)$ , where

$$H_{n2,B}(s) = \frac{1}{1 + \left(1 + \frac{b}{s}\right) \frac{K_o}{s}} = \frac{s^2}{s^2 + K_o s + K_o b} = \frac{s^2}{s^2 + 2\zeta\Omega_n s + \Omega_n^2} = 1 - K_B(s)$$

and

$$K_B(s) = \frac{K_o s + K_o b}{s^2 + K_o s + K_o b} = \frac{2\zeta\Omega_n s + \Omega_n^2}{s^2 + 2\zeta\Omega_n s + \Omega_n^2} \quad (9-116)$$

$$\Omega_n = \sqrt{K_o b}$$

$$\zeta = \frac{1}{2} \sqrt{\frac{K_o}{b}}$$

### A) Noise generation in $K_B(s)$ is $n(t)$

In the case where the noise generation in  $K_B(s)$  is  $n(t)$ , the full noise transfer function is obtained by adding  $H_{n2,B}(s)$  to  $H_{n,B}(s)$ , to obtain

$$\begin{aligned} H_{nT,B}(s) &= \left\{ 1 - M_B(s)G_B(s) \frac{1 - e^{-sT_0}}{s} \cdot \frac{[M_B(s)N(s)]^*}{M_B(s)N(s)} \right\} K_B(s) + 1 - K_B(s) \\ &= 1 - M_B(s)G_B(s) \frac{1 - e^{-sT_0}}{s} \cdot \frac{[M_B(s)N(s)]^*}{M_B(s)N(s)} K_B(s) \end{aligned} \quad (9-117)$$

For the case where the filtered noise signal  $M_B(s)N(s)$  has no frequency components above the Nyquist frequency, equation (9-117) simplifies to

$$\begin{aligned} H_{nT,B}(s) &= 1 - M_B(s)G_B(s) \frac{1 - e^{-sT_0}}{sT_0} K_B(s) \\ &\text{for } s = \sigma + j\Omega \text{ and } -\Omega_0 \leq \Omega \leq \Omega_0 \end{aligned} \quad (9-118)$$

In the special case where  $M_B(s) = 1$ , equation (9-118) becomes

$$H_{nT,B}(s) = 1 - G_B(s) \frac{1 - e^{-sT_0}}{sT_0} K_B(s) = 1 - \frac{H_B(s)}{T_0} \quad (9-119)$$

In this latter case the input signal is filtered with transfer function  $H_B(s)$ , which is a low-pass filter, and the noise is filtered with transfer function  $1 - H_B(s)/T_0$ , which is a high-pass filter. In addition, if  $G_B(s) = 1$ , then the signal and noise transfer functions are

$$H_B(s) = K_B(s) \frac{1 - e^{-sT_0}}{s}, \quad (9-120)$$

and

$$H_{nT,B}(s) = 1 - \frac{H_B(s)}{T_0} = 1 - \frac{1 - e^{-sT_0}}{sT_0} K_B(s). \quad (9-121)$$

### B) Noise generation in $K_B(s)$ is $e_{syncE}(t)$

In the case where the noise generation in  $K_B(s)$  is  $e_{syncE}$ , separate noise transfer functions must be computed for  $e_{syncE}$  to  $y(t)$  and  $e_{ts}$  to  $y(t)$  (as was done for model A). The transfer function for  $e_{syncE}$  to  $y(t)$  is given by equation (9-117), with  $N(s)$  replaced by the Laplace transform of  $e_{syncE}(t) + e_{ts}(t)$ , i.e.,  $E_{syncE}(s) + E_{ts}(s)$ . For the case where the filtered SyncE noise signal  $M_B(s)[E_{syncE}(s) + E_{ts}(s)]$  has no frequency components above the Nyquist frequency, equation (9-117) simplifies to equation (9-118). If, in addition,  $M_B(s) = 1$ , equation (9-118) simplifies to equation (9-119). If, in addition,  $G_B(s) = 1$ , equation (9-119) simplifies to equation (9-121). The transfer function for  $e_{ts}$  to  $y(t)$  is given by equation (9-54), with  $N(s)$  replaced by the Laplace transform of  $E_{syncE}(s) + e_{ts}(t)$ , i.e.,  $E_{syncE}(s) + E_{ts}(s)$ . For the case where the filtered quantization noise signal  $M_B(s)[E_{syncE}(s) + E_{ts}(s)]$  has no frequency components above the Nyquist frequency, equation (9-54) simplifies to equation (9-55).

#### 9.3.4 Special case of model B where $M_B(s) = G_B(s) = 1$ and $K_B(s)$ is a second-order PLL

In the special case where  $M_B(s) = G_B(s) = 1$  and  $K_B(s)$  is a second-order PLL, the signal transfer function, equation (9-53), reduces to

$$H_B(s) = K_B(s) \frac{1 - e^{-sT_0}}{s} = \frac{K_o s + K_o b}{s^2 + K_o s + K_o b} \frac{1 - e^{-sT_0}}{s} = \frac{2\zeta\Omega_n s + \Omega_n^2}{s^2 + 2\zeta\Omega_n s + \Omega_n^2} \frac{1 - e^{-sT_0}}{s}, \quad (9-122)$$

with

$$\begin{aligned}\Omega_n &= \sqrt{K_o b} \\ \zeta &= \frac{1}{2} \sqrt{\frac{K_o}{b}}\end{aligned}\quad (9-123)$$

Using the approximation in equation (9-108)

$$H_B(s) \approx \frac{K_o s + K_o b}{s^2 + K_o s + K_o b} T_0 = \frac{2\zeta\Omega_n s + \Omega_n^2}{s^2 + 2\zeta\Omega_n s + \Omega_n^2} T_0 \quad (9-124)$$

Equations (9-109) and (9-124) indicate that the particular model A of Figure 85 has approximately the same signal transfer function performance as model B with  $M_B(s) = 1$ ,  $G_B(s) = 1$ , and  $K_B(s)$  a second-order PLL as in Figure 87.

For the case where the noise generation in  $K_B(s)$  is  $n(t)$  and has no frequency components above the Nyquist frequency, the noise transfer function is obtained from Eqs. (9-119) and (9-124)

$$H_{nT,B}(s) = 1 - G_B(s) \frac{1 - e^{-sT_0}}{sT_0} K_B(s) = 1 - \frac{H_B(s)}{T_0} = \frac{s^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \quad (9-125)$$

For this case, equations (9-110) and (9-125) indicate that the particular model A of Figure 85 has approximately the same noise transfer function performance as model B with  $M_B(s) = 1$ ,  $G_B(s) = 1$ , and  $K_B(s)$  a second-order PLL as in Figure 87.

For the case where the noise generation in  $K_B(s)$  is  $e_{syncE}(t)$  and has no frequency components above the Nyquist frequency, the noise transfer function between  $e_{syncE}$  and  $y(t)$  is also given by equation (9-108) (i.e., obtained from equations (9-119) and (9-124)). The noise transfer function between  $e_{ts}$  and  $y(t)$ , if  $e_{ts}$  has no frequency components above the Nyquist frequency, is given by equation (9-55).

In the simulations of clause 12, noise generation modelled as in Figure 87, with the noise generation within  $K_B(s)$  equal to  $e_{syncE}$ . The equivalent transfer function between  $e_{syncE}$  and  $y(t)$  in these simulations is given by equation (9-125).

### 9.3.5 Comparison of models A and B for the case where $M_B(s) = 1$ in model B and noise generation is included in both Models

The signal and noise transfer functions for model A, when noise generation is included, are given by equations (9-113) – (9-115)

$$H_A(s) = \frac{K_A(s)Q_A(s)}{1 + [K_A(s)Q_A(s)]^*} \quad (9-126)$$

$$H_{nT,A}(s) = 1 - \frac{H_A(s)}{T_0}, \quad (9-127)$$

$$\text{for } s = \sigma + j\Omega \text{ and } -\Omega_0 \leq \Omega \leq \Omega_0$$

where

$$Q_A(s) = G_A(s) \frac{1 - e^{-sT_0}}{s} \quad (9-128)$$

The signal and noise transfer functions for model B, for the case where  $M_B(s) = 1$  and noise generation is included, are given by equations (9-53) and (9-119)

$$H_B(s) = K_B(s)Q_B(s) = K_B(s)G_B(s) \frac{1 - e^{-sT_0}}{s} \quad (9-129)$$

$$H_{nT,B}(s) = 1 - G_B(s) \frac{1 - e^{-sT_0}}{sT_0} K_B(s) = 1 - \frac{H_B(s)}{T_0}, \quad (9-130)$$

where

$$Q_B(s) = G_B(s) \frac{1 - e^{-sT_0}}{s}. \quad (9-131)$$

Comparing equations (9-127) and (9-130), it is seen that if  $H_A(s) = H_B(s)$ , then  $H_{nT,A}(s) = H_{nT,B}(s)$ . In both models, the noise generation is introduced through a high-pass filter that is related to the signal transfer function in the same way (i.e., one minus the low-pass transfer function divided by the sampling interval). Therefore, it may be ensured that the models have the same transfer functions, and the same performance, by ensuring that the signal transfer functions are equal, i.e.,  $H_A(s) = H_B(s)$ .<sup>12</sup> This condition is

$$K_B(s)G_B(s) \frac{1 - e^{-sT_0}}{s} = \frac{K_A(s)Q_A(s)}{1 + [K_A(s)Q_A(s)]^*} = \frac{K_A(s)G_A(s) \frac{1 - e^{-sT_0}}{s}}{1 + \left[ K_A(s)G_A(s) \frac{1 - e^{-sT_0}}{s} \right]^*}, \quad (9-132)$$

or

$$K_B(s)G_B(s) = \frac{K_A(s)G_A(s)}{1 + \left[ K_A(s)G_A(s) \frac{1 - e^{-sT_0}}{s} \right]^*}. \quad (9-133)$$

Under the assumptions of this clause (i.e., noise generation included as described above and  $M_B(s) = 1$  in model B), the performance of model A depends only on the product  $K_A(s)G_A(s)$ ; likewise, the performance of model B depends only on the product  $K_B(s)G_B(s)$ . If the products are denoted by  $P_A(s)$  and  $P_B(s)$ , respectively, then models A and B have the same signal and noise transfer functions, and therefore the same performance, if

$$P_B(s) = \frac{P_A(s)}{1 + \left[ P_A(s) \frac{1 - e^{-sT_0}}{s} \right]^*}. \quad (9-134)$$

Equation (9-134) shows that, given a model A with filters  $K_A(s)$  and  $G_A(s)$ , then a model B2 (with  $M_B(s) = 1$ ) will have the same performance if the filters  $K_B(s)$  and  $G_B(s)$  are chosen such that the product  $P_B(s) = K_B(s)G_B(s)$  is given by equation (9-134) (and where  $P_A(s) = K_A(s)G_A(s)$ ).

The inverse transformation may be obtained by multiplying equation (9-134) by the factor

$$\frac{1 - e^{-sT_0}}{s},$$

taking the star transformation of both sides, and solving for the quantity

$$\left[ P_A(s) \frac{1 - e^{-sT_0}}{s} \right]^*.$$

---

<sup>12</sup> This is true for the cases where  $H_{nT,A}(s)$  and  $H_{nT,B}(s)$  are transfer functions between  $n(t)$  and  $y(t)$  (i.e., the noise generation in  $K_A(s)$  and  $K_B(s)$  is  $n(t)$ ), and where they are transfer functions between  $e_{syncE}(t)$  and  $y(t)$  (i.e., the noise generation in  $K_A(s)$  and  $K_B(s)$  is  $e_{syncE}$ ).

The result is

$$\left( P_A(s) \frac{1 - e^{-sT_0}}{s} \right)^* = \frac{\left( P_B(s) \frac{1 - e^{-sT_0}}{s} \right)^*}{1 - \left[ P_B(s) \frac{1 - e^{-sT_0}}{s} \right]^*}. \quad (9-135)$$

Then, using equation (9-134)

$$\begin{aligned} P_A(s) &= P_B(s) \left( 1 + \left[ P_A(s) \frac{1 - e^{-sT_0}}{s} \right]^* \right) = P_B(s) \left( 1 + \frac{\left( P_B(s) \frac{1 - e^{-sT_0}}{s} \right)^*}{1 - \left[ P_B(s) \frac{1 - e^{-sT_0}}{s} \right]^*} \right) \\ &= \frac{P_B(s)}{1 - \left[ P_B(s) \frac{1 - e^{-sT_0}}{s} \right]^*}. \end{aligned} \quad (9-136)$$

Equation (9-136) shows that, given a model B2 (with  $M_B(s) = 1$ ) with filters  $K_B(s)$  and  $G_B(s)$ , then a model A will have the same performance if the filters  $K_A(s)$  and  $G_A(s)$  are chosen such that the product  $P_A(s) = K_A(s)G_A(s)$  is given by equation (9-136) (and where  $P_B(s) = K_B(s)G_B(s)$ ).

## 10 Frequency domain mathematical description of a T-BC and a T-TSC

### 10.1 Introduction

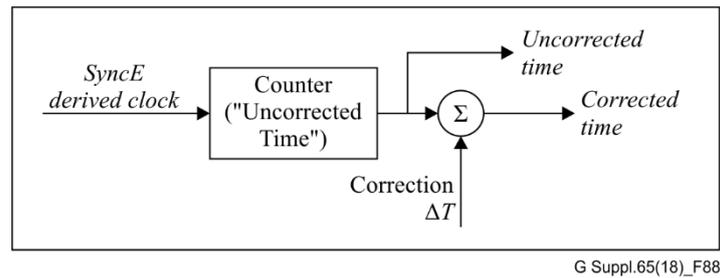
This clause describes models for a T-BC and T-TSC that can be used for simulating, in the frequency domain, the transport of time using PTP and frequency using SyncE. First, in clause 10.2, the case of a T-BC or T-TSC with no filtering is considered. In this case, the time is updated instantaneously when a time synchronization message exchange, i.e., transmission and reception of Sync (and, for the two-step case, Follow\_Up), Delay\_Req, and Delay\_Resp messages. Models for the various sources of time error are described in clause 10.2. Then, in clause 10.3, the case where the T-BC or T-TSC filtering is considered, and models for the filtering are described.

### 10.2 Model for a T-BC or T-TSC with no filtering

In its simplest form, a time clock can be modelled as an accumulator that is routinely incremented. The increment used is representative of "frequency". In essence, time (phase) is considered to be the integral of frequency. This is the model recommended in Appendix I of [ITU-T G.8271.1]. Specifically, time is maintained by a *Time Counter* that is incremented by the nominal period of the clock that provides the frequency assist (e.g. the EEC clock in the case of SyncE frequency assist). The nominal period of the physical clock signal (e.g., 8 ns for the case where the nominal frequency is 125 MHz) introduces a granularity in the time values (8 ns assuming 125 MHz).

In the context of a boundary clock, the error performance can be modelled in the following manner. First, it assume that the nominal packet (i.e., time synchronization message) rate is  $f_0 = 1/T_0$ . That is, the information is available to update/correct the local clock parameters every  $T_0$  units of time. The model presented here assumes this architecture and provides a discrete-time model that, essentially, observes the various signals on a time-grid with interval  $T_0$ . That is, the model subsumes a sampling rate of  $f_0$ .

Second, the model is suitable for analysing the error performance of the T-BC. In a T-BC the "slave" clock develops the time based on communication with an upstream "master" and serves as a "master" providing this time to downstream slaves that could be BCs or ordinary clocks (T-TSCs).



**Figure 88 – Representation of simple time-clock**

The structure in Figure 88 represents the action of the block labelled "Time Counter and time-offset correction" in Figure I.1 of [ITU-T G.8271.1] (which is reproduced in Figure 64 in this Supplement). The general method of operation is as follows:

- a. The frequency assist signal, assumed to be synchronous Ethernet, operates a counter that serves as the *uncorrected time*. It should be noted that the epoch of the uncorrected time is somewhat arbitrary and the uncorrected time could be significantly offset from the master. However, the presumption is that since SyncE is typically traceable to a PRC, the counter increments are reasonably accurate.
- b. The PTP message exchange provides a means for establishing the correction term,  $\Delta T$ , that is an estimate of the time-offset between the master and the uncorrected time at the slave (i.e., the master time minus the uncorrected time at the slave), at the point in time that the exchange of timing messages occurs.

In this clause, we assume that the correction is "jammed"<sup>13</sup>. In other words, there is a register (not explicitly shown in Figure 88) containing the current value of the correction term, and this is added to the uncorrected time value whenever a time value is requested from the clock. Every valid time synchronization message exchange provides a correction, and the correction register is updated. This is generally done in software and the correction value is simply a variable stored in memory. Essentially there is no filtering of time error updates.

From an error analysis viewpoint, the accumulated time error in the physical layer clock, denoted here by  $e_{syncE}$ , is manifested in its entirety in the *uncorrected time*. The contribution to the *corrected time* is only the phase error that accumulates since the last correction. Denote by  $E_{syncE}$  the time error contribution of the physical layer clock in the corrected time. Considering that the correction is performed "instantaneously" every  $T_0$  units of time (the sampling interval) one can write

$$E_{syncE}(t) = e_{syncE}(t) - e_{syncE}(nT_0) ; nT_0 < t \leq (n+1)T_0 \quad (10-1)$$

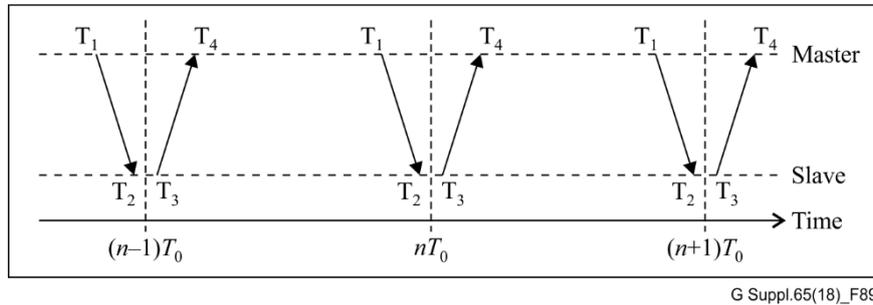
That is, the time error contribution of the physical error clock is high-pass filtered. The value of time error at the sampling points is indeterminate because of the assumption of instantaneous correction. It is conventional to choose either the limit from the left-hand side (as in equation (10-1)) or the right-hand side for consistency but the specific choice should not affect any analytical conclusions made.

*Cautionary note:* If the time-clock in the T-BC is corrected instantaneously in accordance with the method described above, then it is possible (even though it may be unlikely) for the corrected time in Figure 88 to have negative jumps. Such a behaviour may be problematic in many situations. A companion contribution addresses the case where the structure is such that the time clock does not

<sup>13</sup> Clause 10.3 further generalizes this to a "filtered" update mechanism.

have instantaneous jumps. Specifically, all changes in the time clock are achieved by adjusting a numerically controlled oscillator (NCO). Such a behaviour is equivalent to filtering. Changing the time-clock instantaneously is equivalent to no filtering.

The sequence of events is depicted in Figure 89. For convenience, the PTP message transmissions depicted are those whose time-of-departure and time-of-arrival are time-stamped.



**Figure 89 – Sequence of events in time synchronization**

Denoting by  $T_j$  ( $j = 1, 2, 3, 4$ ) the timestamps associated with the ingress and egress instants of the timing event messages, the time offset of the slave clock with respect to the master is computed as:

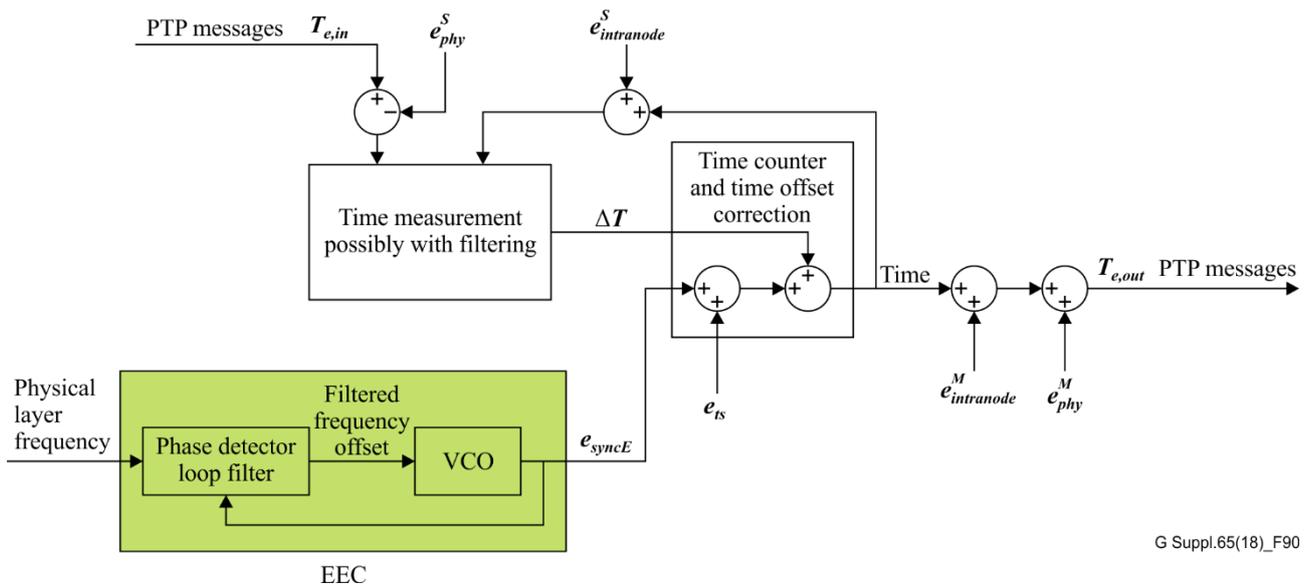
$$\Delta T(nT_0) = \frac{(T_4(nT_0) + T_1(nT_0)) - (T_3(nT_0) + T_2(nT_0))}{2} \quad (10-2)$$

Implicit in equation (10-2) are the assumptions that:

- This offset estimate assumes symmetry. In other words, it assumes that the transit delay in the master-to-slave direction is exactly equal to the transit delay in the slave-to-master direction. This assumption of reciprocity is required to match the number of unknowns to the number of equations. The error due to asymmetry is captured in the term  $e_{link-asym}$  in Appendix I (clauses I.6 and I.7) of [ITU-T G.8271].
- The clocks at the master and the slave are very stable over the interval of the message exchange. The error terms associated with this assumption are not addressed in [ITU-T G.8271].
- The clocks at the master and slave are syntonized. If they are not syntonized, there is an additional error, given by equation (9-3) of clause 9.1.2.2.1. However, for the case where the physical layer clocks at the master and slave are provided by SyncE signals that are traceable to a PRC, the frequency offset between the master and slave is at most  $2 \times 10^{-11}$ , and the resulting error will be negligible compared to other sources of error.

The non-simultaneity of the message transmissions introduces an error. Specifically, the determination of the slave time offset from the master makes the assumption that the slave clock time offset is the same at both the time of arrival (of the Sync message) and the time of departure (of the Delay\_Req message). However, it is expected that this error is small compared to all the other sources and can likely be ignored.

The various sources of error are described in Appendix I of [ITU-T G.8271]. In particular, Figure I.1 of [ITU-T G.8271.1] is shown in Figure 90:



G Suppl.65(18)\_F90

**Figure 90 – Telecom boundary clock model for simulating the transport of time using PTP with SyncE assistance (From Figure I.1 of [ITU-T G.8271.1])**

One reason for associating the time-stamps as shown in equation (10-2) is to recognize that  $T_1$  and  $T_4$  are "Master" time-stamps and  $T_2$  and  $T_3$  are "Slave" time-stamps. In the case of a boundary clock it is possible for the master and slave portions of the boundary clock to reside on different cards and also possible for them to have different line rates. Having different line rates means there could be different time-stamping clock rates for the two sections of the boundary clock (such as 125 MHz for GigE and 25 MHz for 100 Mbit/s). Furthermore, in analysing the error in the slave side of a T-BC, the master time-stamps  $T_1$  and  $T_4$  are actually generated in the upstream boundary clock. It is assumed that the error associated with these time-stamps is all contained in the term  $T_{e,in}$  shown in the Figure 90. Consequently, the term  $T_{e,out}$  must include the time-stamping error for  $T_1$  and  $T_4$ .

The error sources in a boundary clock can be segregated into the "slave" side and the "master" side. The "slave" side errors are contributions that affect the time accuracy of the clock itself. The "master" side errors are contributions that affect the time accuracy of the downstream clock. In the discussion in the following clauses, a revision of Figure 90 is obtained. The revision also illustrates the layered aspect of the boundary clock, where the (physical) frequency-layer is distinct from the (PTP) time-layer. The revision, with minor modifications, is also contained in Appendix I of [ITU-T G.8271.1].

### 10.2.1 Time-stamping granularity noise

Clause I.7 of [ITU-T G.8271], identifies several noise contributions that play a role in determining the performance of telecom boundary clocks from a timing perspective. Clause I.7.3 specifically addresses the noise contribution arising from timestamp granularity.

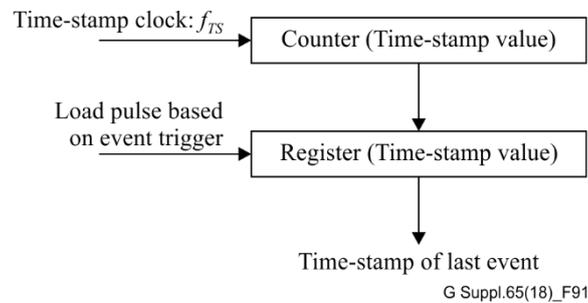
As in clause 10.2, it is considered that the nominal message rate to be  $f_0$  and the nominal message interval to be  $T_0$ , and that these are related as  $f_0 = 1/T_0$ .

It should be recognized that the timestamp error can be separated into three categories, the overall error being the combination of the contributions from the three categories. One category is the wander (and other clock noise contributions such as frequency offset, frequency drift, and random noise components) present in the clock that is responsible for the time-stamping. These components are generally "slow" in nature compared to the actual clock rate. In this case this category is considered as a comprise to components that have a Fourier frequency extent that is less than half the nominal message rate (message rate =  $f_0 = 1/T_0$ ).

The second category comprises clock noise as well but of higher Fourier frequency components such as jitter. Included in this category are all clock noise contributions that have a Fourier frequency

extent ranging from a low of half the message rate to the theoretical maximum of one-half the actual clock rate. In practice, the time-stamping clock will be reasonably smooth, and from a practical standpoint the upper frequency limit will be significantly lower than the theoretical maximum. This second category is often ignored, which could lead to erroneous conclusions. If adequate care is not exercised, the sampling theorem shows that all the power of this noise is folded back into the "signal band" because of *aliasing*.

The third category, and one of the primary subjects of this contribution, is the effect of quantization. The action of time-stamping can be described by the following functional model:



**Figure 91 – Conceptual view of time-stamp generation**

As shown in Figure 91, the time-stamping clock (clock rate =  $f_{TS}$ ) runs a counter. The least-significant bit is associated with a time-equivalency of  $\Delta_{TS} = 1/f_{TS}$ . (Note: In some implementations the increment is other than unity, allowing a different weight (in time units) than  $\Delta_{TS} = 1/f_{TS}$ .) That is, the essential granularity of the time-stamp is  $\Delta_{TS}$ . The event to be time stamped generates a "load-pulse" that samples the counter and stores the value in the timestamp register. In practice, there are methods used to prevent *metastable* conditions but these are not indicated in Figure 88. This is exactly the model proposed in clause I.7.3 of [ITU-T G.8271].

Note that it is not correct to assume that the time-of-arrival of a message is synchronous to an edge of the time-stamping clock ( $f_{TS}$ ). Consider the case of timestamp  $T_2$ , representing the slave's estimate of the arrival time of a message sent by the master. The slave's time-stamping mechanism does not influence what time the master sends the message, and also the slave does not have any control over the transit delay of the message over the medium. In the slave, the signal processing associated with the demodulation and clock and data recovery and determination of the appropriate bit then initiates the strike of the timestamp, which is the notion of the load pulse in Figure 91.

In the case of a transmit timestamp the situation is somewhat different. It is possible, even likely, that the "event" enters the PHY at an edge of the time-stamping clock ( $f_{TS}$ ) or at a consistent offset therefrom. This consistent offset may or may not be properly accommodated in the time-stamp generation and insertion into the message. Note also that the signal processing associated with the modulation, which is achieved at a much higher sampling rate than the time-stamping clock, can introduce a certain uncertainty in delay. In the best case, this delay is fixed, and in the worst case the variable component could change from message to message though in all cases the variable component will be between 0 and  $\Delta_{TS}$ .

Note that all events that occur between  $n \cdot \Delta_{TS}$  and  $(n+1) \cdot \Delta_{TS}$  will map into the same timestamp value. This is the notion of *quantization* or *granularity* of the timestamp. Put another way, there is an uncertainty of  $\Delta_{TS}$  associated with a time-stamp.

It is important to distinguish between clock error and time-stamping granularity error. The creation of the time-stamp, for example  $T_2$ , involves two issues. One is the issue of the imprecision of associating the actual time-of-arrival with the current time-stamp counter and second is the fact that the time-stamp counter (the slave time-clock itself) is in error. It is the first item (imprecision) that is the time-stamping granularity error and the second item is the clock error.

Such quantization has been studied extensively in the area of digital signal processing and, in particular, analogue-to-digital conversion. The effective noise source is modelled as a white noise sequence (spectral characteristic) with a uniform probability density function (pdf) (amplitude statistics) of extent  $[0, \Delta_{TS}]$ . Being sampled data, the Fourier frequency extent is basically between  $[-0.5 \cdot f_0, +0.5 \cdot f_0]$ , and for frequencies outside this range one has to use the periodic continuation (*spectral replication*, the counterpart of *aliasing*).

The efficacy of the quantizing model has been verified in general studies on quantization, particularly in analogue-to-digital conversion. The model works well in general. There are some situations where the model is less efficient. For example, if the message rate is a perfect integer sub-multiple of the clock rate then the time-stamping error will be correlated from event to event. However, in the case of time-stamping messages used in message-based timing methods, there is adequate variation from message to message to randomize the time-stamping error. Furthermore, any jitter/wander on the time-stamping clock will also introduce a randomization of the time-stamping error.

In practice, time-stamps are taken in pairs. Specifically, the time-offset-from-Master (*ofm*) as computed by the slave is given by (the implicit sampling interval  $T_0$  has been suppressed in the notation below):

$$ofm(n) = \frac{(T_4(n) + T_1(n)) - (T_3(n) + T_2(n))}{2} \quad (10-3)$$

where  $T_4(n)$  and  $T_1(n)$  are the time-of-arrival and time-of-departure time-stamps struck at the master and  $T_2(n)$  and  $T_3(n)$  are the time-of-arrival and time-of-departure time-stamps struck at the slave. The index  $(n)$  represents the notion of the *ofm* being calculated using the  $n^{th}$  message exchange.

On the slave side of the boundary clock, the time-stamping granularity noise can be written as

$$\varepsilon_{TS}^{(S)}(n) = -\frac{1}{2} \cdot (\varepsilon_{ts}^{T3}(n) + \varepsilon_{ts}^{T2}(n)) \quad (10-4)$$

where  $\varepsilon_{ts}^{T3}(n)$  and  $\varepsilon_{ts}^{T2}(n)$  are the timestamp granularity noise contributions for time-stamps  $T_3(n)$  and  $T_2(n)$ , respectively. The negative sign is solely for convenience. When considering errors, the more relevant metrics are related to "power" and for that the sign is moot. Likewise, on the master side of the boundary clock, the time-stamping granularity noise can be written as

$$\varepsilon_{TS}^{(M)}(n) = \frac{1}{2} \cdot (\varepsilon_{ts}^{T4}(n) + \varepsilon_{ts}^{T1}(n)) \quad (10-5)$$

where  $\varepsilon_{ts}^{T4}(n)$  and  $\varepsilon_{ts}^{T1}(n)$  are the time-stamping granularity noise contributions for time-stamps  $T_4(n)$  and  $T_1(n)$ , respectively.

Figure I.1 of [ITU-T G.8271.1] combines the different time-stamping granularity error contributions into a single source, " $\varepsilon_{TS}$ ". However, for analytical studies, these are separated into two sources in Figure I.2 of [ITU-T G.8271.1] (and in the discussion below). Three reasons for this are:

- a. The time-stamping granularity error associated with time-stamps  $T_2$  and  $T_3$ , namely  $\varepsilon_{TS}^{(S)}$ , may be filtered in the particular boundary clock under consideration (e.g., boundary clock  $n$ ). However, time-stamps  $T_1$  and  $T_4$  are struck on the master side using the local PTP time clock, namely the corrected clock, and consequently the granularity error, namely  $\varepsilon_{TS}^{(M)}$ , is not processed by the filtering function in this device (though it may be filtered in a downstream device (e.g., boundary clock  $n+1$ )).
- b. The time-stamping granularity error contribution  $\varepsilon_{TS}^{(S)}$  impacts the local PTP clock in the particular boundary clock under consideration (e.g., boundary clock  $n$ ). However, the time-stamping granularity error contribution  $\varepsilon_{TS}^{(M)}$  does not impact the local PTP clock in the particular boundary clock under consideration, but impacts the next boundary clock in the chain (e.g., boundary clock  $(n+1)$ ).

- c. From a testing perspective, measurements made by external equipment can assess time-stamping granularity on a "master" port and thus will reflect just the error of the origin, or precise-origin, timestamp in the transmitted *Sync* or *Follow\_Up* messages, respectively.

As discussed above, assuming that suitable randomization techniques are applied, the behaviour of  $\{\varepsilon_{ts}^{T2}(n)\}$  and  $\{\varepsilon_{ts}^{T4}(n)\}$  are well characterized as uniformly distributed, white noise random processes (times of arrival). In the case of times of departure,  $\{\varepsilon_{ts}^{T1}(n)\}$  and  $\{\varepsilon_{ts}^{T3}(n)\}$ , treating them as uniformly distributed, white noise, random processes is a conservative assumption.

Considering that the four sources are independent, the time-stamping granularity noise at the slave side of the T-BC,  $\{\varepsilon_{TS}^{(S)}(n)\}$ , can be modelled as a white noise sequence with a triangular probability density function with support  $(0, \Delta_{TS}^{(S)})$ . Here  $\Delta_{TS}^{(S)}$  represents the period of the time-stamping clock on the slave side of the T-BC. Similarly, the time-stamping granularity noise at the slave side of the T-BC,  $\{\varepsilon_{TS}^{(M)}(n)\}$ , can be modelled as a white noise sequence with a triangular probability density function with support  $(0, \Delta_{TS}^{(M)})$ . Here  $\Delta_{TS}^{(M)}$  represents the period of the time-stamping clock on the master side of the T-BC.

### 10.2.2 Slave side errors

Appendix I of [ITU-T G.8271] identifies several noise contributions that play a role in determining the performance of T-BCs from a timing perspective. As described therein, the error sources that affect the time error of the slave clock include the time error entering the slave clock, which represents the output time error of the upstream master (T-BC), errors in the local PHY, and time-stamping granularity error.

The time error in the corrected time of the T-BC (slave clock) is given by:

$$e_T(t) = E_{syncE}(t) + \varepsilon_{TS}^{(S)} - e_{phy}^{(S)} + e_{intranode}^{(S)} + T_{e,in} ; nT_0 < t \leq (n+1)T_0 \quad (10-6)$$

In equation (10-6), the terms without an explicit time dependence indicated are the values associated with the time instant  $nT_0$ , representing the  $n^{th}$  message exchange. The error terms with the  $(S)$  superscript represent items intrinsic to the equipment that impair the ability of the slave clock to align with the master, thereby introducing a time clock error.

### 10.2.2 Master side errors

The time error in the output PTP messages can be developed based on Figure 90, as

$$T_{e,out}(t)^{(A)} = e_T(t) + e_{intranode}^{(M)} + e_{phy}^{(M)} \quad (10-7A)$$

where  $e_T(t)$  represents the time error in the "corrected time" (see Figure 88 and equation (10-6)). The superscript  $(A)$  signifies that this is one choice of description of the output time error components. Also note that the terms without an explicit time dependence indicated are the values associated with the time instant  $nT_0$ , representing the  $n^{th}$  message exchange. The error terms with the  $(M)$  superscript represent items intrinsic to the equipment that impair the ability of the master clock to put the correct value for time in the time-stamps, thereby introducing a time-clock error in the downstream slave clock.

In the model described here, the time-stamping granularity is added to this error. In particular, as a "master", the T-BC generates timestamps " $T_1$ " and " $T_4$ " for the exchange with a downstream slave. That is, the form shown in equation (10-7B), below is used:

$$T_{e,out}(t)^{(B)} = e_T(t) + e_{intranode}^{(M)} + e_{phy}^{(M)} + \frac{1}{2} \cdot (e_{ts}^{T1} + e_{ts}^{T4}) \quad (10-7B)$$

In equation (10-7B) the subscript " $ts$ " identifies the error as related to time-stamping granularity and the superscripts " $T1$ " and " $T4$ " represent the timestamps affected by the granularity.

However, the link asymmetry is not shown explicitly in Figure I.1 of [ITU-T G.8271.1]. Here we assume that it is added into the effective time error at the output of the "master" as in equation (10-7C):

$$T_{e,out}(t)^{(C)} = e_T(t) + e_{intranode}^{(M)} + e_{phy}^{(M)} + \frac{1}{2} \cdot (e_{ts}^{T1} + e_{ts}^{T4}) + e_{link-asym} \quad (10-7C)$$

With this addition,  $T_{e,out}$  at T-BC M is associated with  $T_{e,in}$  at T-BC M+1 (which could be a T-TSC).

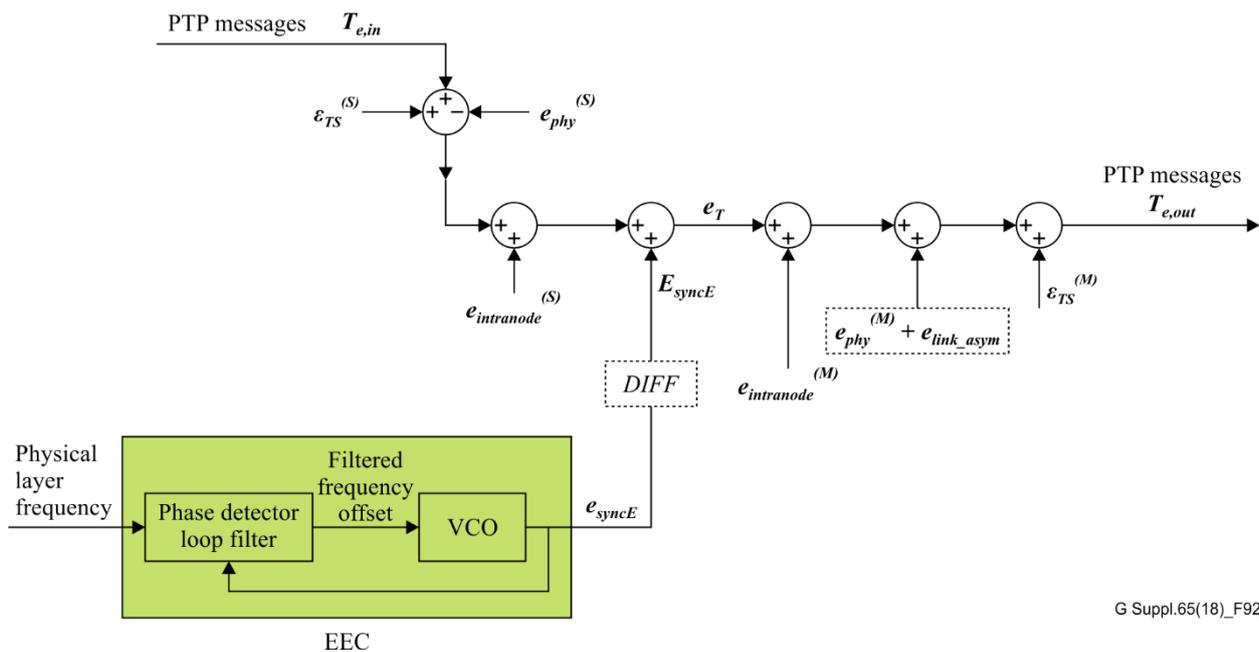
The time-stamping granularity errors can be grouped as:

$$\begin{aligned} \varepsilon_{TS}^{(M)} &= \frac{1}{2} \cdot (e_{ts}^{T1} + e_{ts}^{T4}) \\ \varepsilon_{TS}^{(S)} &= \frac{1}{2} \cdot (e_{ts}^{T2} + e_{ts}^{T3}) \end{aligned} \quad (10-8)$$

### 10.2.3 Modified Figure II.1 of [ITU-T G.8271] (no filtering)

Based on the above, Figure I.1 of [ITU-T G.8271.1] can be modified as follows for the case where there is no time filtering. A separate figure is required to incorporate the case where there is time filtering which is described in clause 10.3.

It is assumed that the intra-node time error is most likely known *a priori* by the manufacturer and can be approximately compensated for. This term, indicated in Figure 92, represents the compensation error.



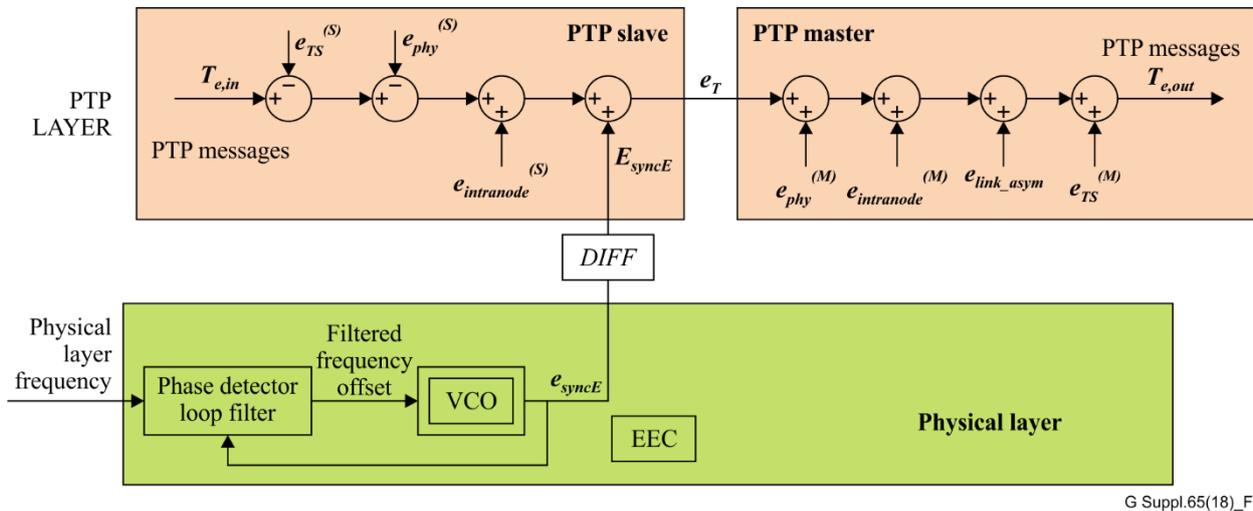
G Suppl.65(18)\_F92

**Figure 92 – Modified version of Figure I.1 of [ITU-T G.8271.1], showing error behaviour for the case of no time filtering**

The primary modifications to Figure I.1 of [ITU-T G.8271.1] are:

- The error contribution arising from the physical layer clock is not the accumulated phase error  $e_{syncE}$ , but a differenced version,  $E_{syncE}$ . This reflects the fact that the local time-clock is corrected every  $T_0$  units of time in the PTP layer.
- The time-stamping granularity noise has been split into two sources, one that affects the slave time-clock of the T-BC under analysis and another that affects the time-clock of the downstream clock.

Considering that the time-clock (PTP layer) and frequency clock (physical layer) are separate, and that the PTP layer does not make corrections to the physical layer clock (frequency), the figure can be redrawn as shown in Figure 93 below to indicate the logical separation of layers and the logical separation of slave side and master side of the T-BC.



G Suppl.65(18)\_F93

**Figure 93 – Modified version of Figure I.1 of [G.8271.1], showing error behaviour for the case of no time filtering and also indicating the distinction between physical layer and PTP layer**

In clause 10.3, this modified version of Figure I.1 of [ITU-T G.8271.1] is further refined to account for the general manner in which a time-clock with frequency assist will be affected by different error sources and by filtering.

### 10.3 Model for a T-BC or T-TSC with filtering

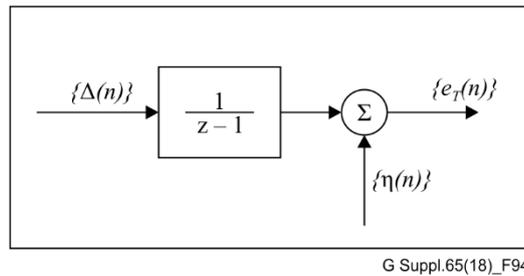
In clause 10.2, a model for the error behaviour of a boundary clock for the case where there is no time filtering is introduced. The boundary clock is assumed to have a physical layer assist mechanism for the time (wall-clock) that is set using PTP. That is, there are two layers of synchronization. The physical layer synchronization, such as synchronous Ethernet, provides a "frequency" clock that provides the nominal progression of the wall-clock time. The PTP layer provides the time correction.

When the time (wall-clock) layer does not include filtering, it is possible that the wall-clock could have negative time jumps. For some applications, these negative jumps may be unacceptable. In this clause the inclusion of filtering is considered. However, the analysis is generic and does not specify any particular implementation. The various error sources considered are from [ITU-T G.8271], as described in clause 10.2.

The basic assumptions and background for time clocks are the same as in clause 10.2.

The time-offset of the slave clock with respect to the master can be used to establish a control loop whereby the counter increment is adjusted from its nominal value (e.g., 8 ns) so that the time-clock of the slave can then align with the master in a gradual manner, as opposed to instantaneously. One important consequence is that the progression of time values can be constrained to always have a positive slope.

In the following analysis, the signals are considered as discrete-time signals with a sampling rate of  $f_0 = (1/T_0)$ . Signal values between points on this sampling grid can be obtained by interpolation (see the literature on digital signal processing, such as [b-Shenoi01]).



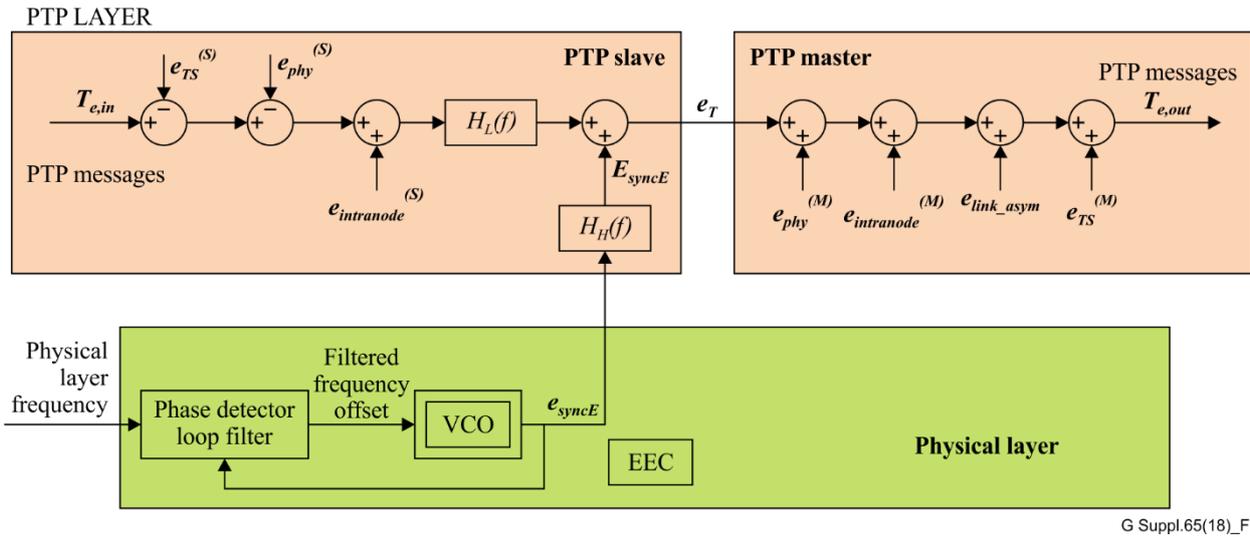
**Figure 94 – Representation of time-clock, showing error components**

Figure 94 shows a representation of a time-clock, with error components. The  $1/(z - 1)$  block represents the time counter. The structure in Figure 94 represents the action of the counter used to establish the time-clock when viewed on the chosen sampling grid. For example, if the underlying sampling rate is (nominally) 125 MHz, and a simple binary counter is considered, the least-significant bit of the counter will represent a time unit of (nominally) 8 ns. If an accumulator or "integrator" is used instead of a binary counter, then in each clock cycle the accumulator can be incremented by  $\Delta(n)$  where  $\Delta(n)$  is the increment which is nominally 8 ns but can be made larger or smaller to emulate a clock rate that is greater or less than 125 MHz, respectively. The error component that arises from the fact that the actual clock rate is not exactly 125 MHz is indicated by the error term  $\{\eta(n)\}$ .

The system is actually closed loop exemplified, as shown in Figure 96. The increment  $\{\Delta(n)\}$  is derived from the PTP input and includes the time error in the input ( $T_{e,in}$ ), the time-stamping ( $e_{TS}^{(S)}$ ), and physical layer asymmetry ( $e_{phy}^{(S)}$ ) (see Figure 92 and Figure 93, and note that the link asymmetry is assumed to be part of the input). In Figure 96 shows that the effective frequency response is low-pass in nature (for proper loop action). The noise component resulting from the clock frequency inaccuracy,  $\{\eta(n)\}$ , arises from the physical layer accumulation modified by a high-pass filter characteristic. The low-pass and high-pass characteristics are depicted in Figure 95 as  $H_L(f)$  and  $H_H(f)$ , respectively.

Note that the model in Figure 95 has differences from the models of clauses 9.2.1 (see Figure 70) and 9.2.2 (see Figure 72). In particular, Figure 95 explicitly indicates the high-pass filtering function that is applied to the contribution from the clock noise originating in the physical layer synchronization chain, while this high-pass filtering is not present in clauses 9.2.1 and 9.2.2. However, the high-pass filtering is introduced in clause 9.3 (see Figures 85, 86 and 87), and this difference is not present between these figures and Figure 95.

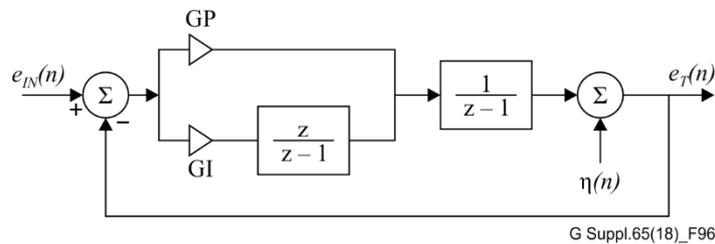
Also note that there is a relationship between the high-pass and low-pass filter characteristics. They are not independent. For example, the two filters have the same cut-off frequency (the same is true in clause 9.3).



**Figure 95 – Model of boundary clock identifying the error sources**

The following example indicates one way in which filtering can be included. The assumption made here is that the offset estimate is used in a closed-loop control architecture to adjust the slave clock. This is especially appropriate since the local frequency assist may have a frequency offset relative to the master PTP clock that is serving the slave clock (part of the boundary clock under consideration). The model provided here assumes a discrete-time, second-order loop with proportional plus integral control. It can be characterized by two parameters GP and GI, which represent the "proportional gain" and "integral gain". The servo model described is based on the formulation in [b-Eidson] (see section 5.2.1, pp 146 – 156 of [b-Eidson]) and [b-Rogers] (see section 3.5, pp. 58 – 61 of [b-Rogers]).

From an error signal perspective, the effective control loop is shown in Figure 96.



**Figure 96 – Control loop viewpoint of slave clock (in boundary clock)**

The input (error) signal is comprised of the various contributions described in clause 10.2, that is,

$$e_{IN}(n) = T_{e,in}(n) + e_{phy}^{(S)}(n) + e_{TS}^{(S)}(n) + e_{intranode}^{(S)} \quad (10-9)$$

The transfer function between the input noise (in the reference) and the output is low-pass in nature and is given by:

$$H_L(z) = \frac{z(GP + GI) - GP}{z^2 + z(-2 + GP + GI) + (1 - GP)} \quad (10-10)$$

The transfer function between the local noise (effectively the physical layer time error) and the output is high-pass in nature and is given by:

$$H_H(z) = \frac{(z - 1)^2}{z^2 + z(-2 + GP + GI) + (1 - GP)} \quad (10-11)$$

The behaviour of the loop is governed by just two parameters,  $GP$  and  $GI$ . A good treatment of the dynamics of the loop is provided by [b-Eidson] (see also [b-Rogers]; the models of [b-Eidson] and [b-Rogers] are equivalent, but use different notation). It is well known that for stability, the poles of the system must lie inside the unit circle in the  $Z$ -plane. As shown in [b-Eidson], the condition for stability is ([b-Rogers] provides a mathematically equivalent expression)

$$GP < 2 - 0.5 \cdot GI \quad (10-12)$$

The following table provides the 3-dB bandwidth, passband gain-peaking, and maximum gain of the high-pass filter for some values of  $GP$  and  $GI$ . For convenience, two values of  $GI$  are chosen for each value of  $GP$ , corresponding to  $0.1 \cdot GP$  and  $0.01 \cdot GP$ .

**Table 25 – Peaking (dB) and 3-dB cut-off frequency for various values of  $GP$  and  $GI$**

$GP$	$GI$	3 dB freq. (relative to sampling rate)	Pass-band peaking (dB)	High-pass filter peak gain (dB)
0.1	$0.1 \cdot GP$	0.031	3.4	1.57
0.1	$0.01 \cdot GP$	0.019	0.61	0.45
0.15	$0.1 \cdot GP$	0.042	2.7	0.87
0.15	$0.01 \cdot GP$	0.028	0.43	0.68
0.2	$0.1 \cdot GP$	0.053	2.2	0.96
0.2	$0.01 \cdot GP$	0.038	0.34	0.92
0.25	$0.1 \cdot GP$	0.065	1.9	1.22
0.25	$0.01 \cdot GP$	0.048	0.28	1.16
0.3	$0.1 \cdot GP$	0.077	1.7	1.49
0.3	$0.01 \cdot GP$	0.06	0.24	1.42
0.35	$0.1 \cdot GP$	0.091	1.5	1.76
0.35	$0.01 \cdot GP$	0.072	0.21	1.68
0.4	$0.1 \cdot GP$	0.11	1.4	2.05
0.4	$0.01 \cdot GP$	0.086	0.18	1.95
0.45	$0.1 \cdot GP$	0.12	1.3	2.34
0.45	$0.01 \cdot GP$	0.10	0.17	2.22
0.5	$0.1 \cdot GP$	0.14	1.2	2.65
0.5	$0.01 \cdot GP$	0.12	0.15	2.51

The calculation of frequency response is straightforward (various mathematical analysis and modelling packages provide the facility to plot the frequency response). Establishing the 3-dB cut-off frequency and peaking (maximum gain in the pass-band) is also straightforward.

It should be noted that since the message rate determines the rate of timing information transfer, there is a natural "sampling frequency",  $f_s$ , associated with the signal processing, which is equal to or less than the message rate. This natural sampling frequency introduces certain constraints, namely, that all Fourier frequencies of interest are between  $0.5 \cdot f_s$  and  $-0.5 \cdot f_s$ .

Another point to note is that the desired frequency response cannot be arbitrary. Since there are just two parameters that can be controlled,  $GP$  and  $GI$ , it is not feasible to arbitrarily specify the three properties, namely, 3-dB cut-off frequency, pass-band gain peaking, and maximum high-pass filter gain, independently.

## 10.4 Modelling of time error accumulation in a T-BC and T-TSC and a chain of clocks

### 10.4.1 The model

This clause provided a simplified analytical method for estimating the time error accumulation in a chain of clocks.

Considering that the time-clock (PTP layer) and frequency clock (physical layer) are disjoint, and that the PTP layer does not make corrections to the physical layer clock (frequency), the general configuration of a T-BC (without filtering) can be shown as in Figure 92 (see clause 10.3, above). For the case where there is no time-clock filtering and in Figure 95 (see clause 10.3) for the case where there is time-clock filtering.

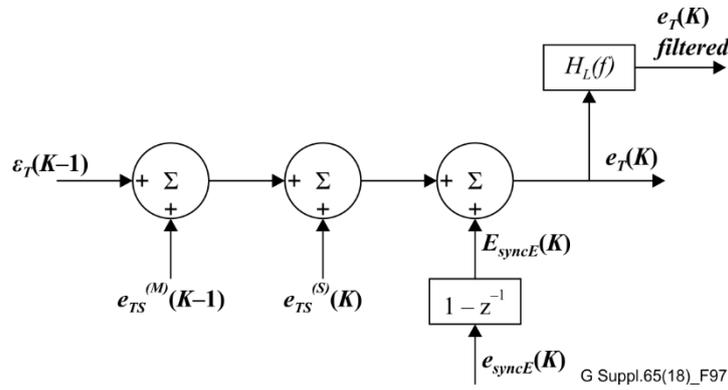
In this simplified analysis, the following assumptions are relevant.

1. The following analysis does not take into account time errors that are related to intranode propagation or to asymmetry, either in the PHY or the link directly. However, an approach to incorporating any known characteristics of the PHY and/or link asymmetry and/or intranode propagation time error is provided.
2. The time error at the grandmaster (T-GM) output is assumed to be zero. That is, the T-GM output is "perfect". This is permissible since all time errors can be considered relative to the T-GM.
3. The packet exchanges are assumed to be on a nominal time-grid of  $T_0$  seconds. That is, the slave time-clock correction is performed every  $T_0$  seconds. The analysis therefore considers all signals as discrete-time signal with an underlying sampling interval of  $T_0$ .
4. The granularity of the time-stamping clock is  $\Delta_{TS}$  and this is the same in all time-stamping modules. Commonly, the granularity for GigE is taken as 8ns and for 100 Mbit/s as 40 ns.
5. All additive random noise sources are assumed to be independent and therefore their strength adds in power (incoherent accumulation).
6. The PRC at the start of a chain will have a frequency offset of  $\pm 1 \times 10^{-11}$  (the PRC limit).
7. The end-point filtering is assumed to be a second-order characteristic with 3dB cut-off frequency of 0.05Hz and a peaking of 0.2dB (assuming 0.1dB will result in a small difference).

The physical layer clock (SyncE) is itself part of a chain of clocks involving SSUs and SEC/EEC equipment. The model for the physical layer assumed here is described below.

The SDH/SyncE reference chain is a full ITU-T G.803 reference chain with the EECs as close to the end of the chain as possible (i.e., a PRC, followed by 8 SSUs, followed by 20 EECs, followed by an SSU, followed by 20 EECs, followed by an SSU, followed by 18 EECs). The number of T-BCs in the HRM is 20 ( $N = 21$ ). This chain has been well studied and the principles developed there will be applied in this clause. The HRM is described in detail in Appendix II.1.2 of [ITU-T G.8271.1] under the 'Non-congruent scenario' heading, and is shown in Figure II.4 of [ITU-T G.8271.1] and Figure II.5 of [ITU-T G.8271.1].

For purposes of analysing the time error in a slave clock (part of a T-BC or the end-point T-TSC), the following signal processing structure is appropriate:



**Figure 97 – Signal Processing viewpoint of time-clock error accumulation in T-BC #K. Sampling interval is the interval between clock corrections, ( $T_0$ )**

#### 10.4.2 Effective maximum value of time error

The signal  $\varepsilon_T(K)$  represents the time-clock error of (the slave side of) Boundary Clock  $K$ . The filtered version is appropriate for the end-point clock where the low-pass characteristic corresponds to a second-order filter with 3 dB cut-off frequency of 0.05 Hz and peaking of 0.2 dB. If  $\mu$  and  $\sigma$  are the mean and standard deviation of the (filtered) error, then the maximum absolute time error is (approximately):

$$e_{\max} \approx |\mu| + 3.5 \cdot \sigma \quad (10-13)$$

The error accumulation can be viewed in terms of mean value and variance. The mean values of the errors are additive and not affected by the low-pass filtering that is assumed to be present in the end-point clock. The factor 3.5 relating peak-to-rms is a common assumption. A more conservative result is obtained by using a larger factor.

#### 10.4.3 Effective mean value of time error

The effective mean value of the error resulting from the time-stamping granularity errors is zero because they are opposite in sign in the master and slave time-stamps.

The effective mean value of the time-error in T-BC # $K$  is the mean value of  $E_{\text{sync}E}(K)$  and is equal to  $\Delta f(K) \cdot T_0$  where  $\Delta f(K)$  is the frequency offset of PRC # $K$  and is  $\pm 1 \times 10^{-11}$ . Given the sampling interval is of the order of 1 second, the mean time error introduced by the PRC frequency offset is of the order of 0.01ns and is considered small enough to be ignored.

That is, the mean value of time-error is essentially zero. In practice, the asymmetries in PHY and link will contribute to this mean error and it may not be zero. One approach to addressing such constant errors is provided later.

#### 10.4.4 Effective variance of time error

The independence assumption implies that the variance of the different components of error will add to each other. Consequently, we can consider them separately.

The assumption of uniform distribution and white-noise behaviour of the time-stamping granularity error implies that each time-stamping module has a variance of  $(\Delta_{TS})^2/12$  and a flat spectrum where  $\Delta_{TS}$  is the granularity of the time-stamping clock.

The path between the point of injection of time-stamping granularity noise and the final slave clock includes a low-pass function in each intervening boundary clock. Consequently, the noise-gain associated with white noise (related to, for example, the time-stamping granularity noise) is given by:

$$\gamma(N) = \frac{2}{f_0} \cdot \int_0^{f_0/2} \left( \sum_{k=1}^N |H_L(f)|^{2(N-k+1)} \right) df \quad (10-14)$$

#### 10.4.5 Contribution of time-stamping granularity to the time error

In equation (10-14) the formula is derived by considering all the white noise sources to be white and uncorrelated with each other. The time-stamping granularity noise in T-BC#1 passes through  $N$  instances of the low-pass filter. The noise in T-BC#2 passes through  $(N-1)$  instances of the low-pass filter. The noise in T-BC# $N$  passes through one instance of the low-pass filter.

The variance contribution from the time-stamping granularity error at the end of  $N$  BCs is therefore (with  $\Delta_{TS} = \sim 8\text{ns}$ ):

$$\sigma_{TS}^2 = m \cdot \gamma(N) \cdot \left( \frac{\Delta_{TS}^2}{12} \right) \quad (10-15)$$

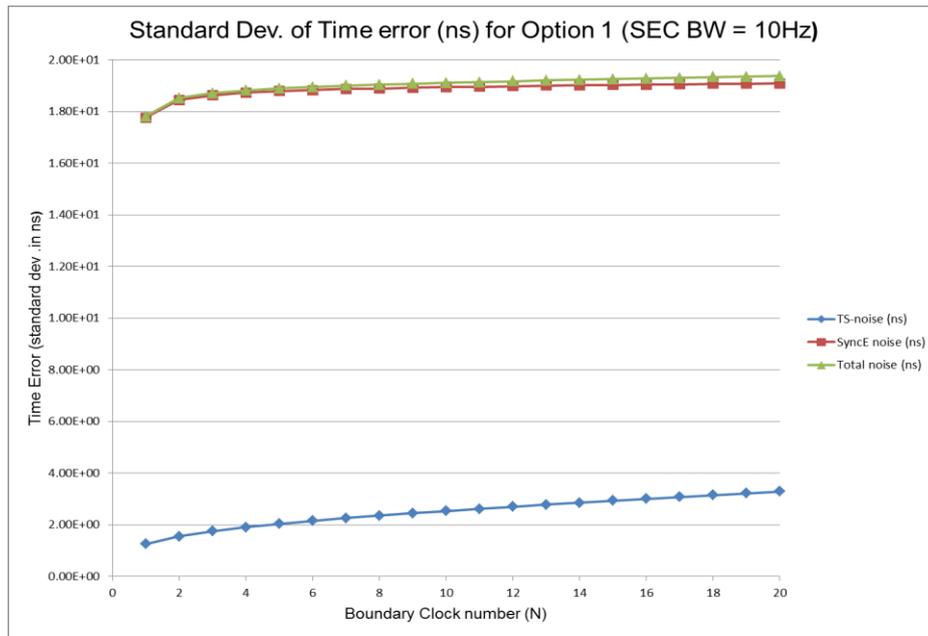
The factor of " $m$ " arises from the number of different time-stamping sources. Solely for convenience it is assumed that  $m = 2$ , that is, considering all four time-stamps. If it is assumed that there is no time-stamping granularity error in time-of-departure time-stamps (i.e.  $T_1$  and  $T_3$ ), then only the error introduced by time-of-arrival time-stamps (i.e.,  $T_2$  and  $T_4$ ) is considered and  $m = 1$ . The reason that  $m = 2$  for the case of all four time-stamps is that the time-stamps are considered in pairs.

For the variance contribution of the SyncE chain, the results of clause 8.2.2 are used to establish the power spectrum  $S_E(f)$  of  $e_{syncE}(K)$ . Recognizing that the physical layer noise component sees a high-pass function,  $H_H(f)$ , followed by a succession of low-pass filters, the contribution is established from the physical layer at the end of  $N$  BCs as:

$$\sigma_{PL}^2 = \frac{2}{f_0} \cdot \int_0^{f_0/2} \left( S_E(f) \cdot H_H(f) \cdot \sum_{k=1}^N |H_L(f)|^{2(k-1)} \right) df \quad (10-16)$$

#### Case 1. SSU Type 1; SEC Option 1; SEC Bandwidth = 10Hz:

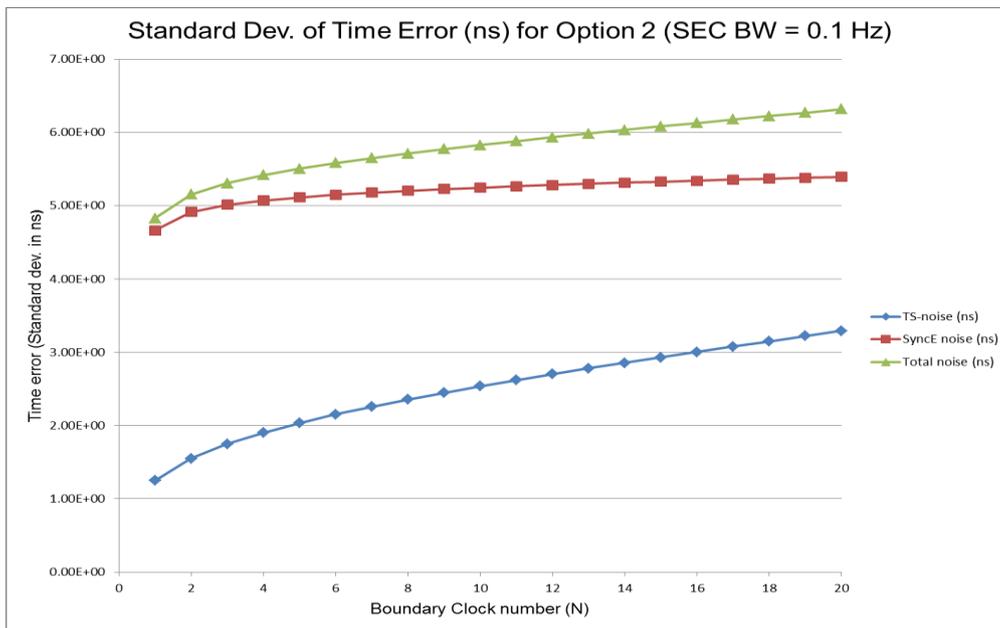
The evaluation of equation (10-15) and equation (10-16) yields the following results. The graph indicates the contribution of the time-stamping granularity error and the contribution of the physical layer noise and the total. The contribution from the "static" sources is not included here.



**Figure 98 – Contribution of time-stamping granularity error and physical layer and total (standard deviation in ns). This is for "Option 1" with SEC bandwidth of 10 Hz**

**Case 2. SSU Type 2; SEC Option 2; SEC Bandwidth = 0.1Hz:**

The evaluation of equation (10-15) and equation (10-16) yields the following results. The graph indicates the contribution of the time-stamping granularity error and the contribution of the physical layer noise and the total. The contribution from the "static" sources is not included here.



**Figure 99 – Contribution of time-stamping granularity error and physical layer and total (standard deviation in ns). This is for "Option 2" with SEC bandwidth of 0.1 Hz**

The graphs in Figures 98 and 99 represent the expected growth of the variance (standard deviation) of time error in a chain of boundary clocks that results from the random (zero-mean) contribution of time-stamping error and physical layer frequency assist noise. To establish the impact of these noise sources on the maximum absolute time error, the contribution is estimated as  $3.5 \sigma$  where  $\sigma$  is the standard deviation of the (total) noise. Note that the impact of filtering of the time error in the

boundary clock has a significant impact on the accumulation of "ac component" errors such as the randomized time-stamping granularity error and the contribution from the physical layer.

It is illuminating to see the impact of error contributions that do not have a zero-mean. The analysis is identical to that provided in [b-Shenoi02] where the boundary clock model assumed that there was no time-error filtering. That is,  $H_L(f) \equiv 1$ . In that case  $H_H(f) = \text{"DIFF"}$  (first-order difference).

The error sources related to PHY and link asymmetry and intranode propagation time (estimate error) introduce a fixed error. That is, the error is non-zero mean but has very small (negligible) variance. The error introduced is unaffected by the low-pass filtering assumed in the final clock. That is, the error sources add coherently and contribute to the mean of the time error. As indicated in equation (10-13) the mean of the error adds directly to the maximum absolute time error.

Denote by  $\mu_k$  the contribution of the PHY asymmetries (master-side and slave-side) of T-BC#k plus the link asymmetry associated with the link between T-BC#k and T-BC#(k+1) and the intranode propagation error contributions. With reference to Figure 88,

$$\mu_k = e_{PHY}^{(M)} - e_{PHY}^{(S)} + e_{link-asymm} + e_{intranode}^{(M)} + e_{intranode}^{(S)} \quad (10-17)$$

In equation (10-17) the right-hand side corresponds to the errors attributed to T-BC#k. The contribution to the time error at the  $N^{th}$  T-BC is, therefore:

$$\mu = \sum_{k=1}^N \mu_k \quad (10-18)$$

Consider the case where all the BCs introduce the same asymmetry error, say  $\alpha$  (ns). Then

$$\mu = N \cdot \alpha \quad (10-19)$$

The impact on the maximum absolute time error is indicated in equation (10-20) and equation (10-21) for Case 1 and Case 2, respectively.  $W_{opt-1}(N)$  and  $W_{opt-2}(N)$  represent the standard deviation of the total noise from the "ac signals" as indicated in Figures 98 and 99.

### Case 1

$$e_{max} \leq \alpha \cdot (N) + 3.5 \cdot W_{opt-1}(N) \text{ ns} \quad (10-20)$$

### Case 2

Note that  $W_{opt-1}(N)$  is of the order of 20ns.

$$e_{max} \leq \alpha \cdot (N) + 3.5 \cdot W_{opt-2}(N) \text{ ns} \quad (10-21)$$

Note that  $W_{opt-2}(N)$  is of the order of 6ns.

## 11 Description of time-domain simulator and implementation of models

### 11.1 Introduction

This clause describes the time-domain simulator. The simulator is based on the T-BC and T-TSC models of clause 9. Clause 11.2 gives an overall description of the simulator, with a focus on timestamping relative to the uncorrected clock (model B of clause 9) using the peer-to-peer mechanism and without modelling noise generation. Clause 11.3 describes the additions needed to model noise generation in the PLL filter. Clause 11.4 describes the modelling of timestamping relative to the corrected clock (model A of clause 9), using either the delay request/response or peer-to-peer mechanism. In all cases, it is assumed that frequency transport is provided by SyncE, i.e., the simulator does not model the pure PTP case. Both HRM2 (SyncE transport congruent to PTP transport, see clause 7) and HRM3 (SyncE transport not congruent to PTP transport, see clause 7) can

be handled by the simulator because the SyncE phase noise results are simulated separately and saved in files.

## 11.2 Overall description of simulator

This clause gives an overall description of the time-domain simulator with a focus on timestamping relative to the uncorrected clock (model B of clause 9) using the peer-to-peer mechanism and without modelling noise generation in the PLL filter.

The simulator models a chain of telecom PTP clocks. The first clock is the telecom grandmaster (T-GM). In the model it is a T-TSC. Since the network configuration is a chain and the T-GM is the first clock in the chain, the T-GM has only a single port (which is a master port). Each successive clock, except the final clock, is a T-BC, and has two ports. The final clock has only a single port because it is the last clock in the chain; it is a T-TSC (i.e., an T-TSC). The simulator models filter at the final T-TSC. This is referred to as "endpoint filtering" because it is applied only in recovering the transported time at the end of the chain. The simulator also models filtering in the T-BCs, referred to as "T-BC filtering." If T-BC filtering is done, then the T-GM time reflected in the originTimestamp (or preciseOriginTimestamp) and correctionField values of transmitted Sync messages on the master port of each T-BC is a filtered time. Both T-BC filtering and endpoint filtering can be optionally turned on or off in the simulator. In addition, even if T-BC filtering is turned off, the simulator can compute the filtered time that would be seen at the T-BC. This is equivalent to the time that would be seen at the T-BC if equivalent endpoint filtering were applied there.

The T-BC model and T-TSC model, for the case without filtering, is an implementation of model B of clause 9.2.2. By "no filtering", it is understood that the filters  $G_B(s)$ ,  $K_B(s)$ , and  $M_B(s)$  are set to 1. If filtering is present, then the filters  $G_B(s)$  and  $M_B(s)$  are set to 1, and the filter  $K_B(s)$  is modelled as a second-order, linear filter with 20 dB/decade roll-off. The 3 dB bandwidth and gain peaking of the filter are specified as input parameters to the simulation. The model for this filter is the same as the second-order filter used in simulations for OTN performance, as described in Appendix VIII of [ITU-T G.8251], clause VIII.2.2, except that the exact relation between gain peaking and damping ratio is used (i.e., Eqs. (8-14) and (8-15) of clause 8.2.3) in the current simulator rather than the approximation described in clause VIII.2.2 of [ITU-T G.8251] (see equation (VIII-7) of [ITU-T G.8251], which is the same as equation (8-16) of clause 8.2.3 above). See Appendix VIII of [ITU-T G.8251] for more detail on the second-order filter model.

Clause 9.2.3 shows that model B is mathematically equivalent to model A, in the sense that if filters  $G_A(s)$  and  $K_A(s)$  are chosen for model A, the corresponding filters  $G_B(s)$  and  $K_B(s)$  may be obtained for model B such that models A and B have the same transfer function and frequency response. In addition, if filters  $G_B(s)$  and  $K_B(s)$  are chosen for model B, corresponding filters  $G_A(s)$  and  $K_A(s)$  may be obtained for model A such that models A and B have the same transfer function and frequency response. Clause 9.2.3 provides the respective equations giving  $G_B(s)$  and  $K_B(s)$  in terms of  $G_A(s)$  and  $K_A(s)$ , and vice-versa.

In model B, timestamping is done relative to the sampled (i.e., quantized) uncorrected time of the local clock. In HRM3, the local clock for each T-BC and the final T-TSC is derived from a SyncE reference chain. Each SyncE reference chain consists of a PRC, followed by 8 SSUs, followed by 20 EECs, followed by an SSU, followed by 20 EECs, followed by an SSU, followed by 18 EECs (see clause 8.1.1). The phase error at the final EEC of each SyncE reference chain is modelled as the sum of the phase error due to the frequency offset of the PRC of that chain plus the phase error due to wander accumulation in the chain. These are the only sources of phase error present in the SyncE reference chain (transients are not considered in this Supplement). The frequency offset of the EEC chain is chosen randomly at the initialization of a simulation from a uniform distribution whose range is  $[-10^{-11}, +10^{-11}]$  (the PRC is assumed to have a frequency accuracy of  $\pm 10^{-11}$ ). The T-GM is assumed to be perfect, i.e., T-GM phase error is not modelled here. This also means that the T-GM is not timed by a SyncE reference chain.

It should be noted that, as described in clauses 9.1.2 and 9.1.3, the effect of the frequency offset of the local clock at a T-BC is small for small frequency offsets and/or small sojourn and Pdelay or delay request/response turnaround times. For frequency offsets in the range  $\pm 10^{-11}$ , the discussion in clauses 9.1.2 and 9.1.3 indicates that the resulting phase (time) error is negligible, even for long sojourn times and long turnaround times. Nonetheless, the effect of frequency offset is modelled in the simulator because the effect may not be negligible in future simulation cases that consider one or more SyncE reference chains in holdover. However, the effect is negligible in the cases where none of the SyncE reference chains are in holdover, and it is the type B errors (i.e., time error due to the accumulated phase noise in the SyncE reference chains) that must be taken into account (see clause 9.1 for a description of the "type A/B" terminology).

The wander accumulation in a SyncE reference chain is simulated separately, using the time-domain simulation model of clause 8.1 for the SyncE reference chain of clause 8.1.1. As described there, 300 independent replications of the simulation were performed. The SSU and SEC wander generation models are based on the models previously used in ETSI for Option 1 networks, as described in clause 8.1.4.2. As indicated in clause 8.1, the SyncE wander accumulation simulation results for all 68 clocks in the SyncE reference chain and all 300 replications of the simulation were saved. Therefore, these results can be re-used for different assumptions for the PTP clock chain (e.g., T-BC and/or T-TSC filter bandwidth and gain peaking, timestamp granularity, mean Sync interval, etc.).

For HRM2, the same simulator for the SyncE reference chain is used. However, now it is only necessary to model a single SyncE reference chain (300 times, for multiple replication cases). As indicated in clauses 7 and 8.1.1, now it is necessary that there are 20 EECs following the final SSU, rather than 18, because the PTP clock chain has 20 clocks that need a SyncE reference from the congruent SyncE chain. In addition, future analyses of cases involving SyncE transients/rearrangements (not considered in this Supplement) will require a collocated "SSU-like" clock at the T-TSC. To accommodate these future cases, an 11<sup>th</sup> SSU was added to the SyncE reference chain, following the 60<sup>th</sup> EEC.

The simulations are a combination of discrete time with fixed time step and discrete event. The transmission and receipt of PTP event messages is explicitly modelled, and the transmission and receipt of each such message is modelled as a discrete event. Each endpoint filter is modelled by discretizing a continuous-time model with a fixed time step. This fixed time step is used between each pair of successive events to integrate the endpoint filters. The time step is obtained as follows. First, a maximum time step is specified as an input parameter for the endpoint filter. This is taken to be small compared to the inverse of the 3 dB bandwidth of the filter (e.g., 0.1 times the inverse of the 3 dB bandwidth). For each pair of successive events, the number of time steps between those events is computed such that the time step size will be as large as possible but still less than the maximum time step size specified.

As indicated in clause 9.2.4.3 (see Figure 84), there is no benefit to using a filter bandwidth that is larger than the Nyquist frequency, and the results there indicate that the filter provides the most benefit when its bandwidth is much less than the Nyquist frequency. Since the Nyquist rate (i.e., the sampling rate, or twice the Nyquist frequency) is equal to the average Sync message rate, this means that the filter bandwidth should be somewhat less than the average Sync message rate, or the inverse of the filter bandwidth should be somewhat larger than the mean Sync interval. Since the transmission and receipt of each Sync message is an event, this means that the number of time steps between successive events will not be large in cases of practical interest, and often may be equal to 1. We also note that some discrete-time filter realizations will become unstable if the sampling rate is not sufficiently large compared to the filter bandwidth. For example, section 3.5 of [b-Rogers] gives an example of a discrete-time realization of a second-order PLL that becomes unstable if the sampling rate is less than approximately  $\pi$  multiplied by the 3 dB bandwidth. However, this is not an issue for the simulations, because the second-order PLL model of Appendix VIII of [ITU-T G.8251] uses a convolution of the exact system function (i.e., impulse response for the continuous-time filter) with

the filter input, and only the input is discretized (using a trapezoidal rule approximation over each time-step). This technique is stable regardless of the timestep (though the accuracy will decrease as the timestep gets larger).

The above discussion also means that the time step size will generally be different between different pairs of successive events. This means that the parameters of the filter implementation of Appendix VIII of [ITU-T G.8251] that depend on time step (e.g., the elements of the matrix exponential of equation (VIII-15) of [ITU-T G.8251]) must be re-calculated between each pair of successive events. This is not a burden, because all the filter parameters that do not depend on time step are calculated and saved on initialization, and only the time-step-dependent parameters are re-calculated on each time step.

The simulator explicitly models the sending and receiving of PTP event messages by each clock. For simplicity, the clocks are assumed to be one-step. This avoids having to also model the sending and receiving of PTP general messages. Initially, the peer-to-peer mechanism was modelled for the measurement of propagation delay (the delay request/response mechanism was modelled later, see clause 11.4). The expected time synchronization performance difference (i.e., as measured by peak time error) between chains of one-step and two-step clocks (with all other parameters and conditions the same) is small. This is so because Follow\_Up is transmitted very soon after Sync, Pdelay\_Resp\_Follow\_Up is transmitted very soon after Pdelay\_Resp, and, since the links in the HRM are point-to-point (i.e., every node is a PTP clock), Follow\_Up and Pdelay\_Resp\_Follow\_Up are received very soon after Sync and Pdelay\_Resp, respectively. This means that the turnaround and sojourn times will be small and, since frequency is transferred using SyncE, resulting Type A errors are small (see clauses 9.1.2 and 9.1.3)

The PTP messages whose sending and receipt are modelled are:

- a) Sync
- b) Pdelay\_Req
- c) Pdelay\_Resp

Since the HRM models the transport of time synchronization from the T-GM to the T-TSC, Sync is transmitted only on downstream ports (these are the master ports) and is received only on upstream ports (these are the slave ports). In addition, while in a real network the peer-to-peer mechanism is invoked by both ports of each link, i.e., it is invoked constantly in both directions. In the simulation the measured propagation delay is needed only by the slave port of each node. Therefore, at each node the following events are modelled:

- 1) Transmission of Sync on a master port
- 2) Receipt of Sync on a slave port
- 3) Transmission of Pdelay\_Req on a slave port
- 4) Receipt of Pdelay\_Req on a master port
- 5) Transmission of Pdelay\_Resp on a master port
- 6) Receipt of Pdelay\_Resp on a slave port

Events (1) and (3) occur at the average sync and Pdelay rates, respectively. They are scheduled as follows. [IEEE 1588] specifies in clause 7.7.2.1 that a node shall issue Sync, Pdelay\_Req, and Announce messages within  $\pm 30\%$  of the value of the message interval attribute (i.e., the mean message interval) for the respective message, with 90% confidence (the corresponding requirement for Delay\_Req messages is given in clause 9.5.11.2 of [IEEE 1588]). This is modelled by assuming that the time between successive messages is gamma-distributed, with the mean of the gamma distribution equal to the respective mean message interval (mean Sync interval and mean Pdelay interval for events (1) and (3), respectively). The variance is chosen such that the portion of the gamma distribution within  $\pm 30\%$  of the mean has an area equal to 0.9. The gamma probability density function (pdf) is

$$p_X(x; \lambda, a) = \frac{\lambda(\lambda x)^{a-1} e^{-\lambda x}}{\Gamma(a)}, \quad (11-1)$$

where  $X$  is the random variable, in this case the time interval between transmission of successive messages of a given type,  $x$  is the value of the random variable,  $\lambda$  is the scale parameter,  $a$  is the shape parameter, and  $\Gamma(a)$  is the gamma function

$$\Gamma(a) = \int_0^{\infty} x^{a-1} e^{-x} dx. \quad (11-2)$$

Note that  $\Gamma(a+1) = a\Gamma(a)$  and, when  $a$  is an integer,  $\Gamma(a) = (a-1)!$ . The mean and variance of the gamma distribution,  $\mu$  and  $\sigma^2$ , are related to  $a$  and  $\lambda$  by

$$\begin{aligned} \mu &= \frac{a}{\lambda} \\ \sigma^2 &= \frac{a}{\lambda^2} \\ \lambda &= \frac{\mu}{\sigma^2} \\ a &= \frac{\mu^2}{\sigma^2} \end{aligned} \quad (11-3)$$

The condition that the inter-message time must be within  $\pm 30\%$  of the mean with 90% confidence may be written (assuming the inter-message time is gamma-distributed)

$$\int_{0.7\mu}^{1.3\mu} \frac{\lambda(\lambda x)^{a-1} e^{-\lambda x}}{\Gamma(a)} dx = 0.9. \quad (11-4)$$

Making the change of variable  $u = \lambda x$ , equation (2-4) may be written

$$\int_{0.7\lambda\mu}^{1.3\lambda\mu} \frac{u^{a-1} e^{-u}}{\Gamma(a)} du = 0.9. \quad (11-5)$$

The last two equations of (11-3) indicate that  $a = \lambda\mu$ . Substituting this into equation (11-5) produces

$$\int_{0.7a}^{1.3a} \frac{u^{a-1} e^{-u}}{\Gamma(a)} du = 0.9. \quad (11-6)$$

It may be verified that equation (11-6) is satisfied when  $a = 29.374$  (using, for example, the Chi-Square distribution table in section 26 of [b-Abramowitz] and relations between the incomplete gamma function and Chi-Square distribution in sections 6 and 26 of [b-Abramowitz]).

The actual times that the successive event messages of the given type are sent are generated by generating samples of this gamma distribution. The samples of the gamma distribution are generated by first generating a uniformly-distributed pseudo-random number between 0 and 1 and then using the standard transformation from a uniform distribution, i.e., computing (see section 8.2 of [b-Law])

$$x = F_X^{-1}(u), \quad (11-7)$$

where  $u$  is the sample of the uniform distribution,  $x$  is the sample of the gamma distribution, and  $F_X^{-1}(u)$  is the inverse of the cumulative gamma distribution. The cumulative gamma distribution is

$$F_X(x; \lambda, a) = \int_0^x \frac{\lambda(\lambda t)^{a-1} e^{-\lambda t}}{\Gamma(a)} dt. \quad (11-8)$$

Making the change of variable  $u = \lambda t$  (as above), equation (11-7) becomes

$$F_x(x; \lambda, a) = \int_0^{\lambda x} \frac{u^{a-1} e^{-u}}{\Gamma(a)} du = P(\lambda x; a), \quad (11-9)$$

where  $P(x; a)$  is the incomplete gamma function with shape parameter  $a$  (see clause 6.5 of [b-Abramowitz]). In the case here,  $a = 29.374$  as indicated above. Then, to generate a sample inter-message time when the mean inter-message time is  $\mu$ , with the requirement that the inter-message time is within  $\pm 30\%$  of the mean with 90% confidence, a uniform pseudo-random number in the interval  $[0, 1]$  is generated. The inter-message time sample  $x$  is then obtained from

$$x = \frac{P^{-1}(u; a)}{\lambda}, \quad (11-10)$$

where  $P^{-1}(u; a)$  is the inverse of the incomplete gamma function with shape parameter  $a$ , and  $a = 29.374$ . There exist algorithms for computing the gamma function, incomplete gamma function, and inverse of the incomplete gamma function.

With the above approach, the requirements of clause 7.7.2.1 of [IEEE 1588] are met. However, this does not prevent a single instance (or a few instances) of a very long inter-message interval from occurring. In fact, when the method was initially used in running multiple replications of simulations, it was found that a single intermessage interval could be generated that was sufficiently long that the next message of that messageType would be generated after the end of the simulation (in some cases the simulation time was 11000 s). For example, when this occurred for a Sync intermessage interval at a particular node, it meant that no more Sync messages would be sent by that node for the remainder of the simulation. This behaviour was considered to be undesirable, even though it does not violate the requirement in [IEEE 1588] because that requirement applies only to 90% of the inter-message intervals. Note that if such an interval actually was generated in reality, it would likely result in a receipt timeout for the respective message (assuming the PTP profile defined this timeout; [IEEE 1588 – 2008] defines Announce Receipt Timeout, but does not define timeouts for other messages). In any case, it was decided that, in the model, there should be a reasonable upper limit on the generated message interval. Since the requirement in [IEEE 1588 – 2008] is that the samples of the message interval be within  $\pm 30\%$  of the mean with 90% confidence, it was decided that an upper limit of 2 times the mean would be reasonable. Therefore, the above model for generating message intervals was modified as follows: For each message (in the simulator, Sync and Pdelay\_Req), a sample is generated as described. If the sample value is less than or equal to twice the mean interval value (the mean interval value is an input parameter to the simulation), it is used. If it is more than twice the mean, the interval is set equal to twice the mean interval value. Note that an upper limit of twice the mean intermessage interval is specified for Sync and Announce messages in clause 6.2.8 of [ITU-T G.8275.1].

The simulator maintains an event list, and proceeds from event to event. The simulator contains an event handler for each of the above 6 events. On the completion of handling an event, the event is removed from the event list and the next event is obtained. The simulator integrates the filtered and unfiltered transported time, at each node, from the time of the current event to the time of the next event, using a fixed time step. The filtered and unfiltered time error, i.e., the difference between the filtered and unfiltered time and the grandmaster time, for each node is saved in a file.

The major tasks performed by the event handler for each event are summarized in Table 26. The timestamping of messages is described in more detail shortly. Some of the tasks in Table 26 require scheduling an event whose time relative to the local clock is given. To schedule the event, its time relative to the simulator clock (i.e., its ideal time) must be determined. The determination of the simulator time in terms of the local clock time is described in the clause 11.2.1.

**Table 26 – Summary of major tasks performed by event handlers**

Event number	Event	Major tasks performed by event handler
1	Transmission of Sync on a master port	<p>A) Timestamp the Sync message transmission</p> <p>B) Compute the originTimestamp and correctionField, and write them in the message.</p> <p>B1) If this node is not the T-GM, the sum of these fields is equal to the sum of the originTimestamp and correctionField of the most recently received Sync message, plus the most recently measured propagation delay on this link, plus the elapsed local time between the receipt of the most recent Sync message and the timestamp of (A) (this elapsed local time is equal to the timestamp of (A) minus the timestamp of the receipt of the most recent Sync message) (note that the SyncE signal, used for timestamping, is assumed to have the same frequency as the T-GM, and therefore a rate ratio correction is not necessary; in any case, rate ratio of the SyncE frequency relative to the grandmaster frequency is not measured)</p> <p>B2) If this node is the T-GM, the sum of these fields is equal to the current T-GM time.</p> <p>C) Compute the arrival time of the Sync message at the slave port of the next node, using the transmit PHY, link, and receive PHY delay models</p> <p>D) Schedule the arrival event of the Sync message at the slave port of the next node</p> <p>E) Schedule the next Sync message transmission for this port</p> <p>F) Remove the current event</p>
2	Receipt of Sync on a slave port	<p>A) Timestamp the Sync message receipt</p> <p>B) Save the Sync message receipt timestamp, originTimestamp field, and correctionField</p> <p>C) Update the unfiltered transported time</p> <p>D) Remove the current event</p>
3	Transmission of Pdelay_Req on a slave port	<p>A) Timestamp the Pdelay_Req transmission, and save the timestamp</p> <p>B) Compute the arrival time of the Pdelay_Req message at the master port of the upstream node, using the transmit PHY, link, and receive PHY delay models</p> <p>C) Schedule the arrival event of the Pdelay_Req message at the master port of the upstream node</p> <p>D) Schedule the next Pdelay_Req transmission for this port</p> <p>E) Remove the current event</p>
4	Receipt of Pdelay_Req on a master port	<p>A) Timestamp the Pdelay_Req receipt</p> <p>B) Schedule the transmission event for the corresponding Pdelay_Resp message using the time of the current event and the specified (as an input to the simulation) Pdelay turnaround time</p> <p>C) Remove the current event</p>
5	Transmission of Pdelay_Resp on a master port	<p>A) Timestamp the transmission of the Pdelay_Resp message</p> <p>B) Compute the measured Pdelay turnaround time as the transmission timestamp for the current message minus the</p>

**Table 26 – Summary of major tasks performed by event handlers**

Event number	Event	Major tasks performed by event handler
		<p>arrival timestamp for the corresponding Pdelay_Req message (note that the SyncE signal, used for timestamping, is assumed to have the same frequency as the T-GM, and therefore a rate ratio correction is not necessary. In any case, rate ratio of the SyncE frequency relative to the grandmaster frequency is not measured).</p> <p>C) Write the computed Pdelay turnaround time in the message</p> <p>D) Compute the arrival time of the Pdelay_Resp message at the slave port of the downstream node, using the transmit PHY, link, and receive PHY delay models</p> <p>E) Schedule the arrival event of the Pdelay_Resp message at the slave port of the downstream node</p> <p>F) Remove the current event</p>
6	Receipt of Pdelay_Resp on a slave port	<p>A) Timestamp the Pdelay_Resp receipt</p> <p>B) Compute the measured propagation delay using the receipt timestamp of (A), the saved timestamp for transmission of the corresponding Pdelay_Req message, and the Pdelay turnaround time carried in the message</p>

At initialization, event (1) is scheduled at the master port of the T-GM and each T-BC by generating a time interval  $x$ , given by equation (11-10) for each of these nodes and scheduling event (1) at time  $x$  (after time zero). Also at initialization, event (3) is scheduled at the slave port of the T-GM and each T-BC by generating a time interval  $x$ , given by equation (11-1) for each of these nodes and scheduling event (3) at time  $x$  (after time zero). A subsequent instance of event (1) or event (3) is scheduled as the last task of the handling of the current instance of event (1) or (3), respectively. Event (1) generates event (2) at the next downstream node, and event (3) generates events (4) and (5) and the next upstream node and event (6) at the current node.

The arrival or departure of a message at a port of a node is timestamped using a model for the quantized (i.e., sampled) SyncE signal at that node. The model follows the description given in clauses 9.1.2.2 and 9.1.4. Let the ideal time, i.e., the simulator event clock, be represented by  $t$ . As indicated above, it is assumed the T-GM is ideal. This means that the T-GM time is  $t$  when the simulator clock is  $t$ . It is desired to obtain the quantized SyncE signal time when the ideal time (simulator clock) time is  $t$  (this is because events are scheduled relative to the simulator clock, but events are timestamped relative to the local quantized SyncE clock at the node). If  $y$  is the actual frequency offset of the PRC of this SyncE chain and  $n(t)$  is the SyncE phase noise, then the unsampled syncE time is  $(1+y)t + n(t)$ . The actual frequency offset  $y$  is obtained by sampling a distribution, at initialization, that is uniform over the range  $\pm 10^{-11}$ . As shown in clause 9.1.2.2.1, the effect of  $y$  here is negligible. However, it is modelled in the simulator because it may not be negligible in future cases where holdover of the SyncE reference chain is considered. The SyncE phase noise,  $n(t)$ , is obtained from the saved SyncE phase noise accumulation simulation results (obtained using the time-domain model of clause 8.1). As described there, the sampling time for the SyncE phase error accumulation simulations is 0.1 s, and in general the times of these phase noise samples will not correspond to the times of events in the current simulations. Phase noise values at the times of the events are obtained here using linear interpolation, as described in clause 9.1.4. To obtain the quantized (sampled) SyncE signal time, let  $\Delta_{ts}$  be the timestamp granularity. This means that the timestamp clock advances in increments of  $\Delta_{ts}$ , and its reading is always an integer multiple of  $\Delta_{ts}$ . Then, the timestamp that corresponds to  $t$ ,  $T_{ts}(t)$ , is (see clause 9.1.2.2.2)

$$T_{ts}(t) = \Delta_{ts} \cdot \text{floor}([t(1+y) + n(t)] / \Delta_{ts}) \equiv \Delta_{ts} \cdot \lfloor [t(1+y) + n(t)] / \Delta_{ts} \rfloor, \quad (11-11)$$

where  $\text{floor}(x)$  and  $\lfloor x \rfloor$  denote the greatest integer less than or equal to  $x$  (i.e., these are two alternate notations for the floor function). Since the floor function is an integer, the timestamp  $T_{ts}(t)$  is an integer multiple of  $\Delta_{ts}$ , as required. The timestamp error due to clock phase noise and timestamp granularity,  $\varepsilon_{ts}$ , is

$$\varepsilon_{ts} = T_{ts}(t) - t = \Delta_{ts} \cdot \text{floor}([t(1+y) + n(t)] / \Delta_{ts}) - t. \quad (11-12)$$

As indicated in clause 9.1.2.2.2, the timestamp error  $\varepsilon_{ts}$  is correlated with  $t$ , and therefore cannot be simulated by generating a stream of independent random samples with some distribution. It is seen from equations (11-11) and (11-12) that generating  $\varepsilon_{ts}$  in this manner and adding the result to  $t$  would not guarantee that  $T_{ts}(t)$  would always be an integer multiple of  $\Delta_{ts}$  (in general, it would not be an integer multiple of  $\Delta_{ts}$  in this case).

The simulator can save to output files the filtered and unfiltered time error histories at each node. As indicated above, the filtered time error is computed at each node regardless of whether T-BC filtering is specified (if T-BC filtering is specified, the filtered time is also used to produce the timestamps taken at the T-BC). The filtered and unfiltered time errors may be used to obtain other statistics of interest, e.g., peak of the absolute value of time error over an observation interval, peak-to-peak of the absolute value of phase error over an observation interval, MTIE, TDEV, etc. These computations are done as post-processing tasks.

### 11.2.1 Simulator (ideal) time corresponding to local clock time

When the simulator schedules an event, it is often given the time of the event relative to the local clock. However, to schedule the event the simulator must know the time of the event relative to the simulator, i.e., ideal clock. This means that equation (11-11) must be inverted, i.e., given a time  $T_0$  relative to the local clock, we must find the time  $t$  such that

$$T_{ts}(t) = \Delta_{ts} \cdot \text{floor}([t(1+y) + n(t)] / \Delta_{ts}) \equiv \Delta_{ts} \cdot \lfloor [t(1+y) + n(t)] / \Delta_{ts} \rfloor = T_0. \quad (11-13)$$

In this clause, an approximate solution to this equation is derived, valid when  $y \ll 1$  and  $n(t) \ll t$ .

First, note that if  $T_0$  is an arbitrary time, it may not be an integer multiple of  $\Delta_{ts}$ . However, the final equality in equation (11-13) implies that it is an integer multiple of  $\Delta_{ts}$ . To ensure that  $T_{ts}(t)$  has reached  $T_0$  at the value of  $t$  obtained, equation (11-13) is modified as follows

$$\begin{aligned} T_{ts}(t) &= \Delta_{ts} \cdot \text{floor}([t(1+y) + n(t)] / \Delta_{ts}) \equiv \Delta_{ts} \cdot \lfloor [t(1+y) + n(t)] / \Delta_{ts} \rfloor \\ &= \text{ceiling}(T_0 / \Delta_{ts}) \cdot \Delta_{ts} \equiv \Delta_{ts} \cdot \lceil T_0 / \Delta_{ts} \rceil \end{aligned}, \quad (11-14)$$

where  $\text{ceiling}(x)$  is the smallest integer greater than or equal to  $x$ . Then,

$$\lfloor [t(1+y) + n(t)] / \Delta_{ts} \rfloor = \lceil T_0 / \Delta_{ts} \rceil \equiv N_0. \quad (11-15)$$

Since  $T_0$  and  $\Delta_{ts}$  are given,  $N_0$  is also known. Because of the floor function, equation (11-15) does not have a unique solution. However, the time  $t$  of interest is the earliest time that equation (11-15) is satisfied. If  $n(t) \ll t$ , this occurs approximately when

$$[t(1+y) + n(t)] / \Delta_{ts} = N_0, \quad (11-16)$$

or

$$t(1+y) + n(t) = N_0 \Delta_{ts}. \quad (11-17)$$

Since  $n(t)$  is a random process, the time  $t$  for which equation (11-17) is satisfied is also random, i.e., is a random variable. However, if  $n(t) \ll t$ , the variance of this random variable is much less than its mean. In this case, the mean time at which equation (2-16) is satisfied may be approximated by

finding a zeroth-order solution to equation (11-17), evaluating  $n(t)$  at this zero-order solution for  $t$ , and then finding a first-order correction. To find a zeroth-order solution, rewrite equation (11-17) as

$$t + yt + n(t) = N_0\Delta_{ts}, \quad (11-18)$$

and note that  $yt \ll t$  and  $n(t) \ll t$ . Then, equation (11-18) is may be approximated by

$$t = N_0\Delta_{ts} \equiv t^{(0)}. \quad (11-19)$$

equation (11-19) is the zeroth-order solution for  $t$ . Next, evaluate  $n(t)$  at  $t = t^{(0)}$  in solving equation (11-18) to first order

$$t + yt + n(t^{(0)}) = N_0\Delta_{ts}. \quad (11-20)$$

The solution to this equation is

$$t = \frac{N_0\Delta_{ts} - n(t^{(0)})}{1 + y}. \quad (11-21)$$

However, since  $y \ll 1$ , this may be written to first order

$$t \cong [N_0\Delta_{ts} - n(t^{(0)})](1 - y) \cong N_0\Delta_{ts}(1 - y) - n(t^{(0)}) = N_0\Delta_{ts}(1 - y) - n(N_0\Delta_{ts}). \quad (11-22)$$

In equation (11-22), the product  $n(t^{(0)})y$  has been neglected because this term is of second order.

Equation (11-22) is the desired result. Given a time  $T_0$  relative to the local clock, the quantity  $N_0$  is computed from equation (11-15). The corresponding simulator (ideal) time  $t$  is then computed using equation (11-22).

### 11.3 Modelling of noise generation in the simulator PLL filter

Noise generation is modelled in the simulator PLL filter in accordance with the analytic model of clause 9.2.3 and Figure 87. The noise generation is obtained by filtering the SyncE input to the T-BC or T-TSC by a high-pass filter whose corner frequency and gain peaking are equal to the 3 dB bandwidth and gain peaking, respectively, of the PLL filter. The simulator allows the noise generation to be computed from either the quantized or unquantized SyncE filter (specified via an input parameter). In the actual simulator implementation, the low-pass filtered SyncE (quantized or unquantized, as specified) waveform is computed and subtracted from the SyncE input. Note that if there is no filtering specified, the noise generation of the PLL filter is not added. However, the input SyncE noise of the SyncE signal that maintains the timebase between successive Sync message arrivals (i.e., the SyncE input shown at the lower left in Figures 72 and 87) is still present.

### 11.4 Modelling of timestamping relative to the corrected clock, and the delay request/response mechanism, in the simulator

The simulation model described above was modified to model timestamping relative to the corrected clock, and also to model the Delay Request/Response mechanism. The model described in clause 11 assume timestamping relative to the uncorrected clock. Modelling timestamping relative to the corrected clock is straightforward, as the current estimate of the T-GM time at each node, i.e., the corrected clock, is maintained in the simulator. The Delay Request/Response mechanism is modelled by modifying the functions that handle the events of transmission and reception of Pdelay\_Req and Pdelay\_Resp messages to model the transmission and reception of Delay\_Req and Delay\_Resp messages, and to save the <syncIngressEventTimestamp>, originTimestamp, and correctionField of the most recently received Sync message for use in the mean propagation delay calculation.

A new input parameter, i.e., the Delay Request/Response turnaround time, is needed for the Delay Request/Response mechanism. This is the time interval between the receipt of Delay\_Request and sending of Delay\_Resp at the master port. It may be presumed that this is analogous to the Pdelay turnaround time used in the Pdelay mechanism model; however, there is one important difference.

The Pdelay turnaround time is the time between receipt of Pdelay\_Req and sending of Pdelay\_Resp. If it is small, there is less chance for a Sync message from upstream to arrive between the receipt of Pdelay\_Req and sending of Pdelay\_Resp. In the case where timestamping is done relative to the corrected clock, the result in time error due to a clock correction (i.e., update to the estimate of the grandmaster time) occurring and causing the Pdelay\_Req and Pdelay\_Resp timestamps to differ by a larger than usual amount due to this correction. In the Delay Request/Response mechanism, the four timestamps used to compute mean propagation delay are for the sending of Delay\_Req by the slave port, the receipt of Delay\_Req by the master port, the sending of Sync by the master port, and the receipt of Sync by the slave port. In addition, the time the master port receives Delay\_Req is conveyed to the slave port by the Delay\_Resp message (which is not an event message), and the delay computation is done on receipt of Delay\_Resp using the timestamps of the corresponding Delay\_Req and the most recent Sync message received by the slave port. Therefore, in the Delay Request/Response mechanism the Sync and Delay\_Req messages are analogous to the Pdelay\_Req and Pdelay\_Resp messages of the Pdelay mechanism for purposes of the mean propagation delay computation. This means that the Pdelay turnaround time of the Pdelay mechanism is actually analogous to the time between the receipt of the most recent Sync and the sending of Delay\_Req, and not the time between the receipt of Delay\_Req and sending of Delay\_Resp. In the simulations here, Delay\_Req is assumed to be sent asynchronously to the sending of Sync, and therefore can be as large as one Sync interval.

## 12 Steady-state time domain simulation cases and results

### 12.1 Simulation results for HRM3 cases based on single replications of simulations

#### 12.1.1 Description of simulation cases

The simulation cases include all combinations of the following parameters:

- a) 8 ns and 40 ns timestamp granularity
- b) 0.125 s Sync Interval with 1 s Pdelay interval, and 1 s Sync interval with 8 s Pdelay interval
- c) 0.01 Hz, 0.05 Hz, and 0.1 Hz endpoint filter bandwidth
- d) With/without 0.1 Hz T-BC filter.

The combinations of (a) – (d) give rise to 24 cases. In numbering the cases, it is convenient to let cases 1, 2, 3 and 4 refer to the respective timestamp granularity, Sync interval, and Pdelay interval, and refer to the case number with a respective endpoint filter bandwidth and with/without 0.1 Hz T-BC filtering. For example, we might refer to Case 1 with 0.01 Hz endpoint filtering and 0.1 Hz T-BC filtering.

Table 27 summarizes the parameters and assumptions common to all simulation cases described here.

**Table 27 – Model parameters and assumptions common to all simulation cases**

Parameter	Value
Hypothetical Reference Model	HRM3 of Appendix III of [ITU-T G.8271]
SyncE phase noise accumulation	Results of clause 8.1.4.2
SyncE PRC frequency accuracy	$\pm 10^{-11}$
Model for sending of Sync and Pdelay_Req messages	Model based on gamma distribution, as described in clause 11.2, such that actual message intervals are within $\pm 30\%$ of mean intervals with 90% confidence (Note that sending of Sync message on master port of T-BC is asynchronous with receiving of Sync message on slave port of same T-BC)

**Table 27 – Model parameters and assumptions common to all simulation cases**

Parameter	Value
Actual link propagation time	0.1 ms (total for each link, including transmit and receive PHYs; based on 20 km link and approximately 5 ns/m group delay); taken to be the same in both directions
Link and PHY asymmetries	0 (i.e., no link or PHY asymmetry)
Assumption for grandmaster time	Grandmaster is assumed perfect; time errors computed relative to T-GM
Pdelay turnaround time	10 ms
Simulation time	11,000 s (first 1000 s removed when computing peak or peak-to-peak values, to remove any initial transient)
Endpoint filter 3 dB bandwidth	0.05 Hz (note that results are also obtained for the case of no endpoint filtering)
Endpoint filter gain peaking	0.1 dB
T-BC filtering assumption	when present, 0.1 Hz 3 dB bandwidth with 0.1 dB gain peaking
One-step/two-step behaviour	T-BC and T-TSC models based on one-step behavior
Timestamp granularity	40 ns or 8 ns
Noise generation in each filter (both endpoint filters and, if included, T-BC filters)	Modelled as described in clause 8.1.4.1

Table 28 summarizes all the simulation cases, using the nomenclature described above. The simulations are for HRM3 (see clause 7). Results are given for nodes 2 (the first T-BC after the T-GM) through 22 (the time error at node 1 is zero because the T-GM is assumed perfect).

**Table 28 – Summary of simulation cases**

New case number	Timestamp granularity (ns)	Sync Interval; Pdelay Interval (s)	Endpoint filter bandwidth (Hz)	0.1 Hz T-BC filter present
1	8	0.125; 1	0.1	No
	8	0.125; 1	0.05	No
	8	0.125; 1	0.01	No
2	8	1; 8	0.1	No
	8	1; 8	0.05	No
	8	1; 8	0.01	No
3	40	0.125; 1	0.1	No
	40	0.125; 1	0.05	No
	40	0.125; 1	0.01	No
4	40	1; 8	0.1	No
	40	1; 8	0.05	No
	40	1; 8	0.01	No
1	8	0.125; 1	0.1	Yes
	8	0.125; 1	0.05	Yes

**Table 28 – Summary of simulation cases**

<b>New case number</b>	<b>Timestamp granularity (ns)</b>	<b>Sync Interval; Pdelay Interval (s)</b>	<b>Endpoint filter bandwidth (Hz)</b>	<b>0.1 Hz T-BC filter present</b>
	8	0.125; 1	0.01	Yes
2	8	1; 8	0.1	Yes
	8	1; 8	0.05	Yes
	8	1; 8	0.01	Yes
3	40	0.125; 1	0.1	Yes
	40	0.125; 1	0.05	Yes
	40	0.125; 1	0.01	Yes
4	40	1; 8	0.1	Yes
	40	1; 8	0.05	Yes
	40	1; 8	0.01	Yes

All the results in this clause are for single replications of each simulation case. Clause 12.2 describes cases where multiple independent runs are made. This enables confidence intervals for quantities of interest to be obtained. In the cases of the current clause, the same random number generator initial state is used for all the cases so that results with and without noise generation may be compared.<sup>14</sup>

Link and PHY asymmetry are not considered in the simulation cases described here. The simulator includes models for these asymmetries in that the total actual link delays specified as inputs to the simulator need not be the same in both directions. Here, we are comparing results for different timestamp granularity, Sync and Pdelay interval, endpoint filter bandwidth, and T-BC filter present or not present; asymmetry does not impact this comparison.

In each simulation case of Table 28, the mean Pdelay interval is 8 times the mean Sync interval (i.e., the mean Sync message rate is 8 times the mean Pdelay message rate). The phase measurement granularities of 8 ns and 40 ns correspond to 125 MHz and 25 MHz clock rates, respectively. The maximum endpoint filter bandwidth considered, of 0.1 Hz, was chosen to be small compared to the Nyquist frequency corresponding to the sampling rates. The Nyquist frequency is one-half the sampling rate<sup>15</sup>, i.e., it is 4 Hz for cases 1 and 3 and 0.5 Hz for cases 2 and 4. It is shown in clause 9 that there is no benefit to using an endpoint filter whose 3 dB bandwidth is larger than the Nyquist frequency.

Finally, all the above cases of Table 28 (a total of 24 sub-cases for cases 1 to 4 (6 sub-cases for each case)), but with all the filters having 0.2 dB gain peaking instead of 0.1 dB gain peaking. These additional cases were considered to see the effect of larger gain peaking. The 0.1 dB gain peaking corresponds to the maximum gain peaking requirement for many desynchronizers, e.g., low-order ODU to client and high-order ODU to low-order ODU desynchronizers in OTN [ITU-T G.8251]. However, various clock specifications, e.g., SDH Equipment Clock [ITU-T G.813], EEC [ITU-T G.8262], and SSU [ITU-T G.812] require that the gain peaking not exceed 0.2 dB. The

<sup>14</sup> The same random number generator initial state is used for each corresponding case as single-replication results are compared. If results were compared based on multiple replications, e.g., statistical point estimates and confidence intervals, it would not be necessary to use the same initial state for the random number generator.

<sup>15</sup> Note that the Nyquist rate is twice the Nyquist frequency, i.e., the Nyquist rate is the same as the sampling rate.

simulations with 0.2 dB gain peaking were run to determine if the results would be appreciably different compared to the results for 0.1 dB gain peaking.

### 12.1.2 Simulation results

This clause makes the following comparisons among the maximum absolute value time error results for respective sub-cases of cases 1 to 4 of Table 28:

- a) For each of cases 1 to 4, without T-BC filtering and with 0.1 dB gain peaking, the results for the three endpoint filter bandwidths are compared,
- b) For each of cases 1 to 4, with 0.1 dB gain peaking, the results with and without T-BC filtering for each of the three endpoint filter bandwidths are compared, and
- c) For each of cases 1 to 4, the results of each sub-case with 0.1 dB gain peaking and 0.2 dB gain peaking are compared.

#### 12.1.2.1 Results for cases without T-BC filtering

Figures 1 to 4 show the results for the maximum absolute value of filtered time error, as a function of T-BC/T-TSC node number, for cases 1 to 4 with no T-BC filtering. For each case, the results for no endpoint filtering and endpoint filter bandwidths of 0.1 Hz, 0.05 Hz, and 0.01 Hz are compared. Since time error is computed relative to the T-GM, the time error at the T-GM is zero. Therefore, the time error at the T-GM is not shown, and the results begin at node 2. Table 29 gives numerical results for maximum absolute value time error for the same cases and sub-cases as Figures 1 to 4, for the first T-BC (node 2), a T-BC near the middle of the chain (node 12), and the T-TSC at the end of the chain (node 22). The results in Table 29 are contained in Figures 1 to 4, though it may be difficult to obtain the same resolution visually from the figures.

The results indicate that there are two competing effects of making the endpoint filter bandwidth narrower:

- a) Noise generation increases due to the smaller high-pass filter corner frequency, causing increased time error, and
- b) The narrower endpoint filter bandwidth results in narrower-bandwidth low-pass filtering of the time error, causing it to decrease.

In case 1 (Figure 100), there is less overall time error accumulation due to the smaller granularity and PTP message interval (compared to the other cases). As a result, when the endpoint filter bandwidth decreases to 0.01 Hz the increase in noise generation exceeds the decrease in the time error accumulation due to the filtering. The result is a larger time error accumulation compared to the sub-cases with 0.05 Hz and 0.1 Hz endpoint filter bandwidths.

In case 2 (Figure 101), the unfiltered time error is only moderately larger than in case 1 for a small number of hops (e.g., 2 or 3 hops), but accumulates to a much larger value (e.g., 255 ns for case 2 versus 142 ns for case 1) at 21 hops (i.e., at node 22). For the smaller number of hops, the increased noise generation due to narrowing the endpoint filter exceeds the decrease in the time error accumulation due to the filtering. The overall time error accumulation is larger for 0.01 Hz endpoint filter bandwidth compared to 0.05 Hz bandwidth for a smaller number of hops. For a larger number of hops, the increased noise generation due to narrowing the endpoint filter is less than the decrease in time error accumulation due to the filtering. The overall time error accumulation is smaller for 0.01 Hz endpoint filter bandwidth compared to 0.05 Hz bandwidth for a larger number of hops.

The behaviour in case 3 is similar to case 2. The main difference is that now the increased time error accumulation is due to the larger granularity rather than the larger message intervals. For a smaller number of hops, the increased noise generation due to narrowing the endpoint filter bandwidth exceeds the decrease in the time error accumulation due to the filtering. The overall time error accumulation is larger for 0.01 Hz endpoint filter bandwidth compared to 0.05 Hz. For a larger number of hops, the increased noise generation due to narrowing the endpoint filter bandwidth is less

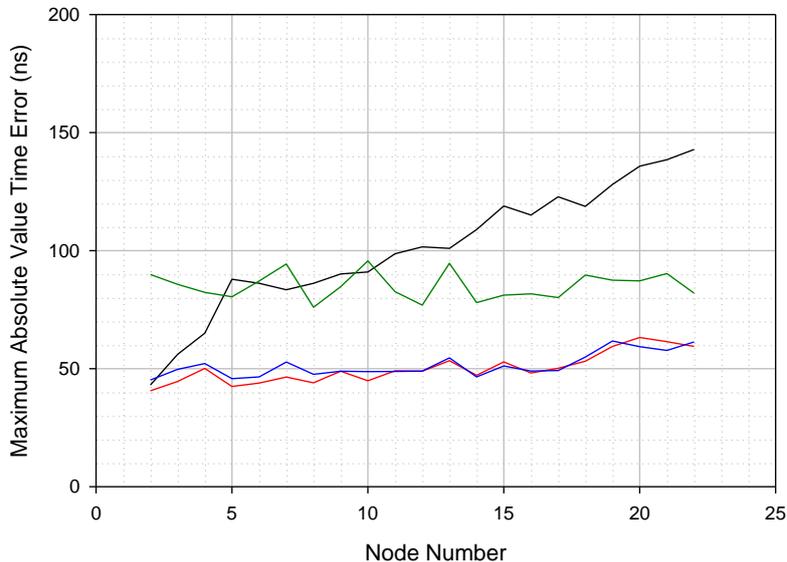
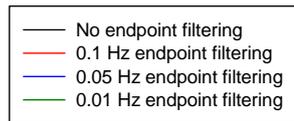
than the decrease in time error accumulation due to the filtering. The overall time error accumulation is smaller for 0.01 Hz endpoint filter bandwidth compared to 0.05 Hz.

In case 4, both the granularity and PTP message interval are increased compared to case 1. This results in somewhat more rapid time error accumulation compared to cases 2 and 3. Now the time error accumulation for 0.01 Hz endpoint filter bandwidth is larger than that for 0.05 Hz bandwidth for a small number of hops, but becomes less than that for 0.05 Hz bandwidth after a smaller number of hops compared to cases 1 to 3. For example:

- in case 1, the time error accumulation for 0.01 Hz bandwidth is larger than that for 0.05 Hz bandwidth for all nodes from 2 to 22,
- in case 2, the time error for 0.01 Hz bandwidth is smaller than that for 0.05 Hz bandwidth for all nodes after node 10 except for node 13,
- in case 3, the time error for 0.01 Hz bandwidth is smaller than that for 0.05 Hz bandwidth for all nodes after node 16, and
- in case 4, the time error for 0.01 Hz bandwidth is smaller than that for 0.05 Hz bandwidth after node 7.

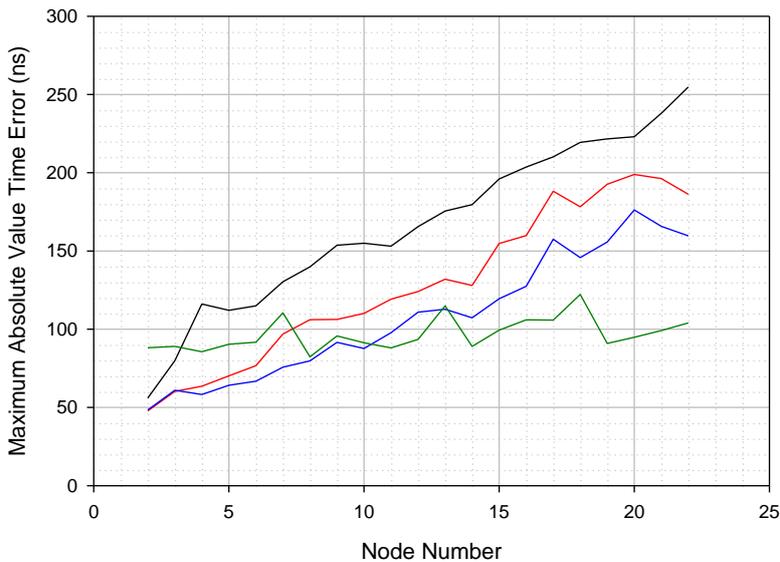
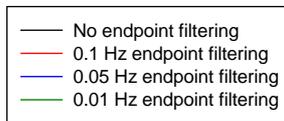
Finally, note that the maximum absolute value time error after endpoint filtering is larger for case 2 than for case 3 for 0.1 Hz and 0.05 Hz bandwidths, and roughly the same for both cases for 0.01 Hz bandwidth. This is in spite of the fact that the unfiltered maximum absolute value time error for case 2 is significantly smaller than for case 3. For example, Table 4 of [b-Garner03] indicates that the maximum absolute value *unfiltered* time error for node 2 is 56 ns for case 2 and 74 ns for case 3. For node 22 it is 256 ns for case 2 and 377 ns for case 3. Figures 5 to 8 of the present contribution show node 22 unfiltered time error history for case 2 (full history in Figure 104 and detail of 2000 – 2100 s in Figure 105) and case 3 (full history in Figure 106 and detail of 2000 – 2100 s in Figure 107). The plots indicate that the frequency content of the time error history is smaller for case 2 than for case 3 and, therefore, more difficult to filter. The reason for the smaller frequency content in case 2 is that the message interval is longer in case 2 compared to case 3, and therefore the sampling rate in case 2 is smaller. This results in less reduction in time error in case 2, for the same filter bandwidth.

HRM3  
 0.1 dB gain peaking for endpoint filters  
 no BC filtering  
 with noise generation in filters  
 8 ns phase measurement granularity  
 Sync interval = 0.125 s  
 Pdelay interval = 1 s



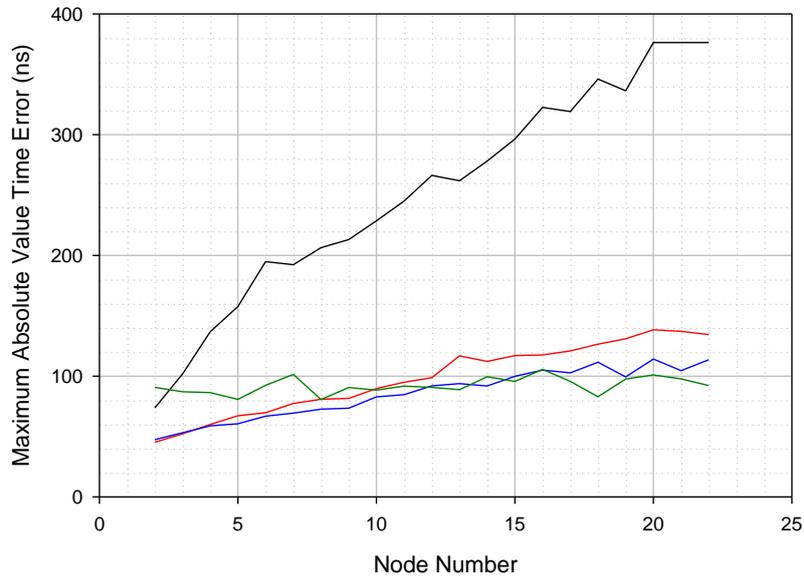
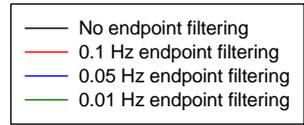
**Figure 100 – Maximum absolute value of filtered time error, case 1, no T-BC filtering**

HRM3  
 0.1 dB gain peaking for endpoint filters  
 no BC filtering  
 with noise generation in filters  
 8 ns phase measurement granularity  
 Sync interval = 1 s  
 Pdelay interval = 8 s



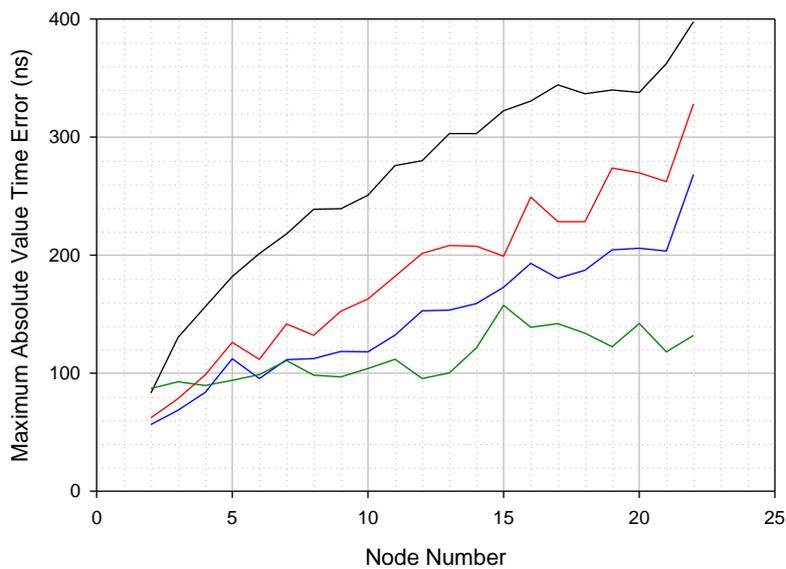
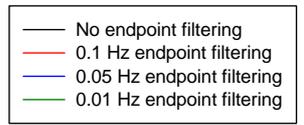
**Figure 101 – Maximum absolute value of filtered time error, case 2, no T-BC filtering**

HRM3  
 0.1 dB gain peaking for endpoint filters  
 no BC filtering  
 with noise generation in filters  
 40 ns phase measurement granularity  
 Sync interval = 0.125 s  
 Pdelay interval = 1 s



**Figure 102 – Maximum absolute value of filtered time error, case 3, no T-BC filtering**

HRM3  
 0.1 dB gain peaking for endpoint filters  
 no BC filtering  
 with noise generation in filters  
 40 ns phase measurement granularity  
 Sync interval = 1 s  
 Pdelay interval = 8 s

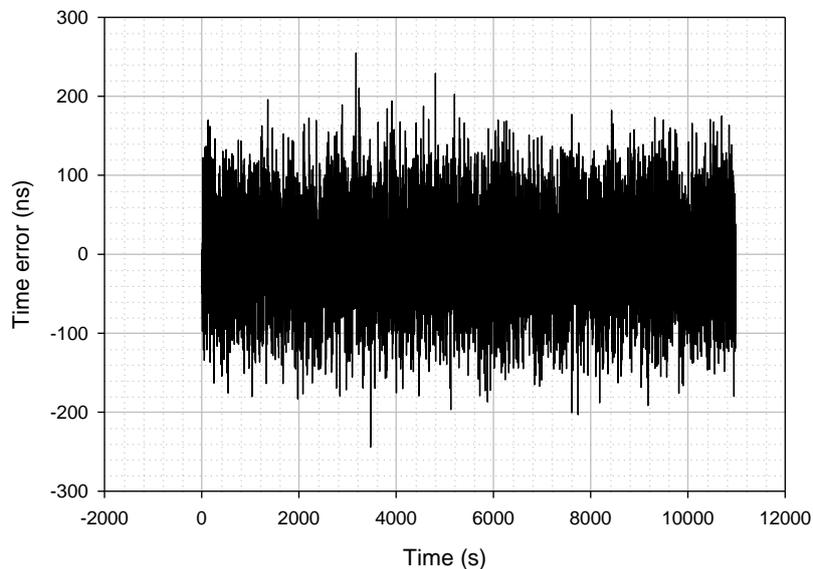


**Figure 103 – Maximum absolute value of filtered time error, case 4, no T-BC filtering**

**Table 29 – Results for maximum absolute value of filtered time error, cases 1 – 4 with no T-BC filtering, for nodes 2, 12 and 22**

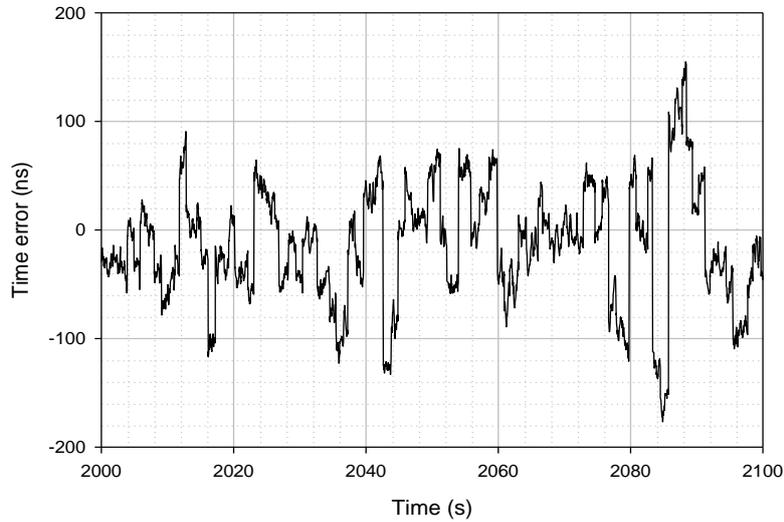
Case/Node	Maximum absolute value of time error (ns)			
	No endpoint filtering	0.1 Hz endpoint filter	0.05 Hz endpoint filter	0.01 Hz endpoint filter
Case 1, Node 2	43.1	40.6	45.1	89.9
Case 1, Node 12	101.6	48.9	48.9	76.9
Case 1, Node 22	142.8	59.4	61.3	82.0
Case 2, Node 2	56.0	47.8	48.4	88.2
Case 2, Node 12	165.6	124.1	110.9	93.5
Case 2, Node 22	254.8	186.3	159.6	104.0
Case 3, Node 2	73.7	45.2	47.4	90.7
Case 3, Node 12	266.3	98.7	91.9	90.1
Case 3, Node 22	376.3	134.4	113.5	92.2
Case 4, Node 2	83.2	62.1	56.4	86.9
Case 4, Node 12	280.0	201.4	152.7	95.3
Case 4, Node 22	397.6	328.0	268.2	131.9

Case 2  
Node 22 (OC that follows final BC)  
No endpoint filtering  
Sync interval = 1.0 s  
Pdelay interval = 8.0 s  
8 ns phase measurement granularity  
Initial transient removed



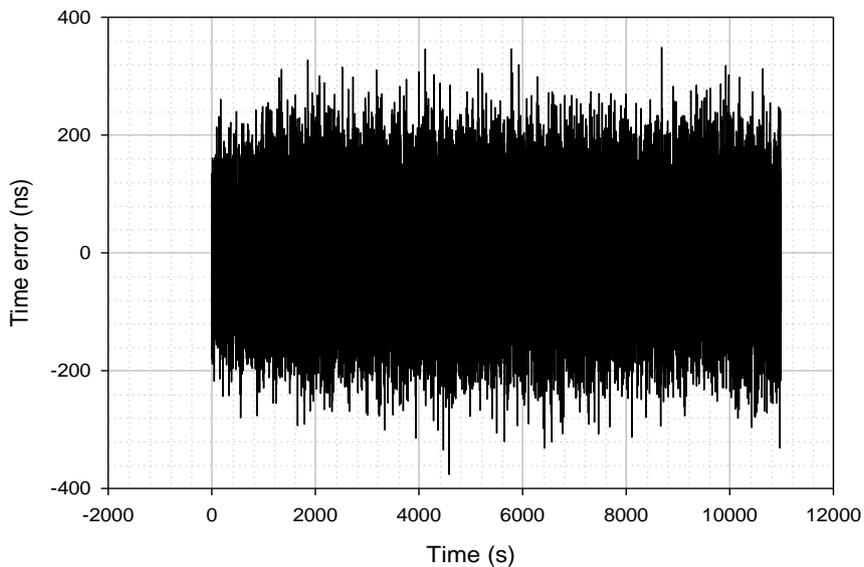
**Figure 104 – Time error history, case 2, no T-BC or endpoint filtering, node 22 (T-TSC that follows final T-BC)**

Case 2  
Node 22 (OC that follows final BC)  
No endpoint filtering  
Sync interval = 1.0 s  
Pdelay interval = 8.0 s  
8 ns phase measurement granularity  
Detail of 2000 - 2100 s



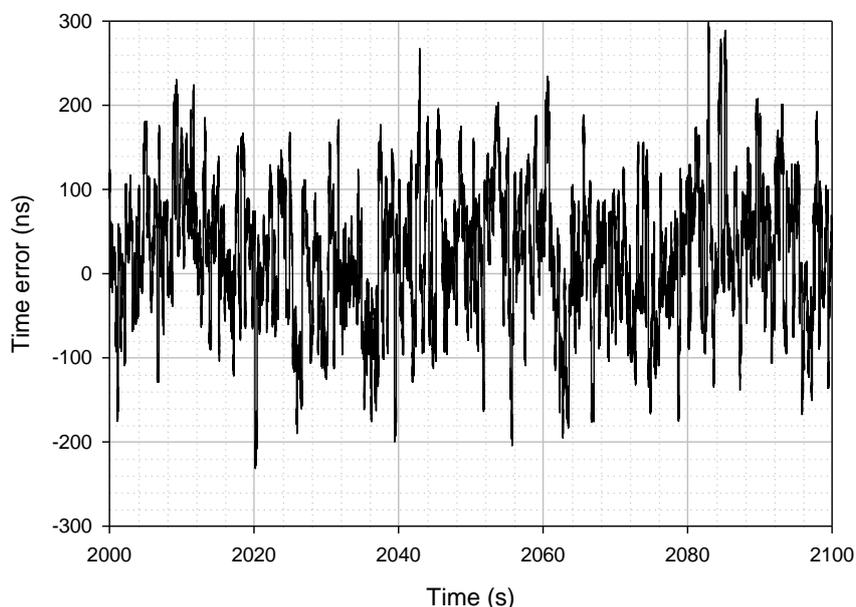
**Figure 105 – Time error history, case 2, no T-BC or endpoint filtering, node 22 (T-TSC that follows final T-BC), detail of 2000 – 2100 s**

Case 3  
Node 22 (OC that follows final BC)  
No endpoint filtering  
Sync interval = 0.125 s  
Pdelay interval = 1.0 s  
40 ns phase measurement granularity  
Initial transient removed



**Figure 106 – Time error history, case 3, no T-BC or endpoint filtering, node 22 (T-TSC that follows final T-BC)**

Case 3  
 Node 22 (OC that follows final BC)  
 No endpoint filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 40 ns phase measurement granularity  
 Detail of 2000 - 2100 s



**Figure 107 – Time error history, case 3, no T-BC or endpoint filtering, node 22 (T-TSC that follows final T-BC), detail of 2000 – 2100 s**

### 12.1.2.2 Results for cases with T-BC filtering

Figures 9 to 12 show the results for the maximum absolute value of filtered time error, as a function of T-BC/T-TSC node number, for cases 1 to 4 with and without T-BC filtering. For each case, the results with and without T-BC filtering are compared for endpoint filter bandwidths of 0.1 Hz, 0.05 Hz, and 0.01 Hz. As outlined in clause 3.1, the time error is computed relative to the T-GM. The time error at the T-GM is zero and is not shown, and the results begin at node 2. Table 30 gives numerical results for maximum absolute value time error for the same cases and sub-cases as shown in Figures 9 to 12, for the first T-BC (node 2), a T-BC near the middle of the chain (node 12), and the T-TSC at the end of the chain (node 22). The results in Table 30 are contained in Figures 1 to 4, though it may be difficult to obtain the same resolution visually from the figures.

The results indicate that there are two competing effects of adding T-BC filtering:

- a) The narrow-bandwidth filtering at each node reduces the time error accumulation up to that node.
- b) Noise generation of the T-BC filter tends to increase the time error. However, this is filtered at the next node (and the noise generation of the last T-BC is filtered at the endpoint filter). Note that the noise generation of the T-BC filter is statistically the same as that of the 0.1 Hz endpoint filter (and therefore statistically less than that of the 0.05 Hz and 0.01 Hz endpoint filters).

Therefore, it is expected that T-BC filtering will have a larger benefit in cases where the time error accumulation is larger, e.g., (i) larger granularity, (ii) larger PTP message intervals, or (iii) time error due to SyncE rearrangements. Note that SyncE rearrangements have not yet been analysed or simulated (this will be done in future work). It is expected that a SyncE rearrangement will result in time error over the interval between receipt of successive Sync messages, and that T-BC filtering can reduce this error if the T-BC filter time constant is of the order of the Sync interval or larger.

In case 1, there is relatively small difference in time error accumulation when 0.1 Hz T-BC filtering is added, for the same endpoint filter bandwidth. As outlined in clause 3.1, the increased noise generation for 0.01 Hz endpoint filter bandwidth results in larger time error accumulation compared to 0.05 Hz and 0.1 Hz endpoint filter bandwidths. This result holds both with and without T-BC filtering.

In cases 2 and 3, there is a larger decrease in time error accumulation at a larger number of hops when T-BC filtering is added, for the same endpoint filter bandwidth. However, the decrease due to T-BC filtering is larger for 0.1 Hz and 0.05 Hz bandwidth endpoint filters, compared to 0.01 Hz bandwidth endpoint filter. This is due to the larger noise generation of the 0.01 Hz endpoint filter compared to 0.05 Hz and 0.1 Hz endpoint filters.

In case 4, where both granularity and PTP message intervals are larger, the decrease in time error accumulation due to T-BC filtering is larger than in cases 2 and 3. As in cases 2 and 3, the decrease due to T-BC filtering is smaller for the case of 0.01 Hz endpoint filter bandwidth, due to the larger noise generation of that endpoint filter.

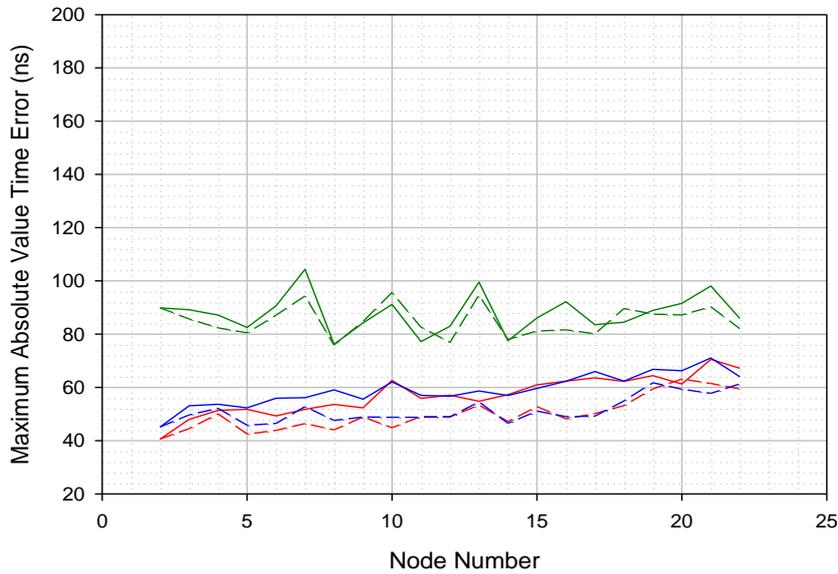
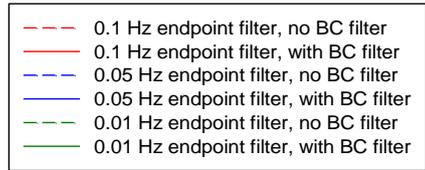
In addition, all the cases 1 to 4 show very small difference in the results for 0.1 Hz endpoint filtering and 0.05 Hz endpoint filter, when BC filtering is present. Furthermore, the results for 0.01 Hz endpoint filtering, when T-BC filtering is present, are either worse than the results for 0.05 Hz and 0.1 Hz endpoint filtering or are of the same order as the results for 0.05 Hz and 0.1 Hz endpoint filtering (e.g., in case 1 the 0.01 Hz endpoint filter results are worse than the 0.05 Hz and 0.1 Hz results for 1 to 21 hops. In cases 2 to 3 the former are worse for smaller numbers of hops and are of the same order for larger numbers of hops). This means that if T-BC filtering is present, there is no benefit to making the endpoint filter bandwidth narrower than 0.1 Hz, and the performance may be worse if the bandwidth is narrowed to 0.01 Hz. This conclusion is favourable because it means, in the case where T-BC filtering is present, (a) the endpoint filter bandwidth need not be too narrow, and (b) the endpoint filter bandwidth can have the same value as the T-BC filter bandwidth (i.e., the two clocks need not have different bandwidth requirements).

In conclusion, for cases with T-BC filtering:

- a) There seems to be little benefit of 0.1 Hz T-BC filtering for case 1.
- b) There may be more benefit of 0.1 Hz T-BC filtering for cases 2 and 3.
- c) The benefit of T-BC filtering is largest for case 4.
- d) If 0.1 Hz T-BC filtering is present, there is no benefit to making the endpoint filter bandwidth narrower than 0.1 Hz, i.e., the endpoint and T-BC filter bandwidths can be the same.

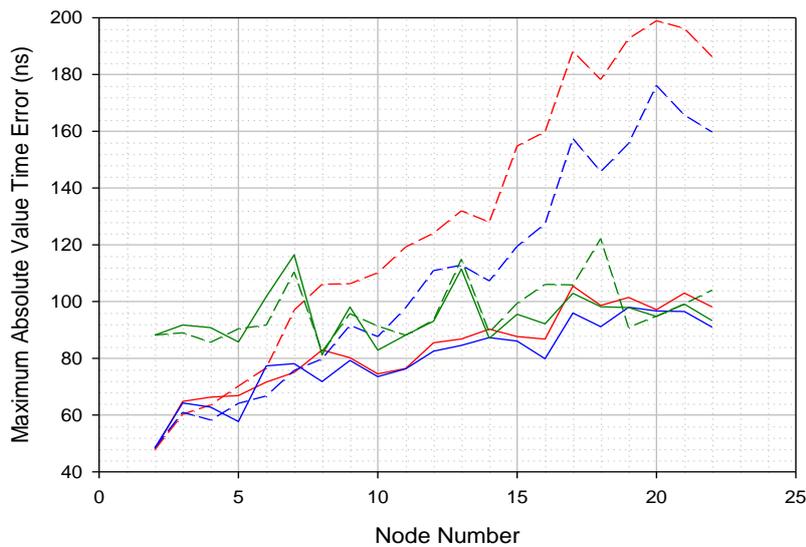
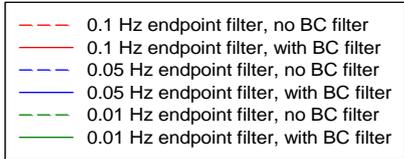
As indicated above, cases with SyncE rearrangements have not yet been analysed. There may be benefit of T-BC filtering for these cases if the T-BC filter time constant is of the order of the Sync interval or larger. In this case, conclusion (d) above is relevant.

HRM3  
 0.1 dB gain peaking for endpoint and BC filters  
 0.1 Hz BC filter bandwidth  
 with noise generation in all filters  
 8 ns phase measurement granularity  
 Sync interval = 0.125 s  
 Pdelay interval = 1 s



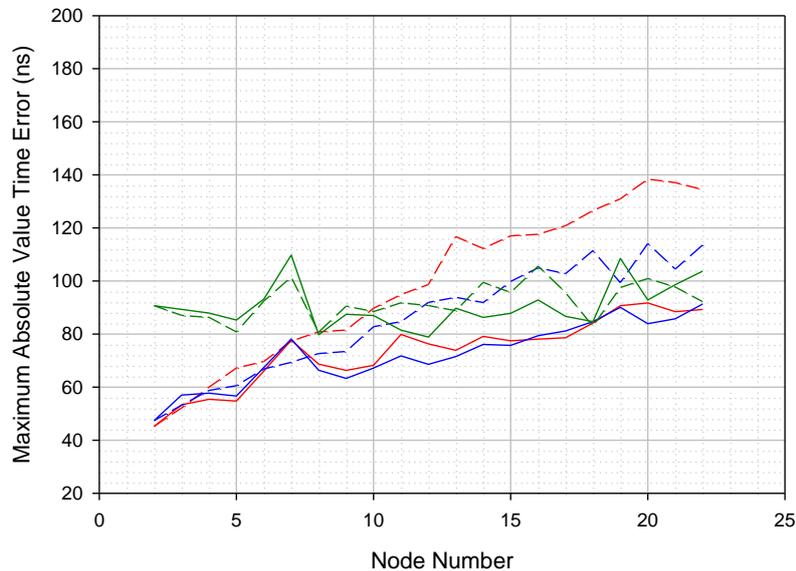
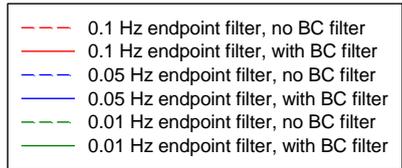
**Figure 108 – Case 1, comparison of results with and without T-BC filtering**

HRM3  
 0.1 dB gain peaking for endpoint and BC filters  
 0.1 Hz BC filter bandwidth  
 with noise generation in filters  
 8 ns phase measurement granularity  
 Sync interval = 1 s  
 Pdelay interval = 8 s



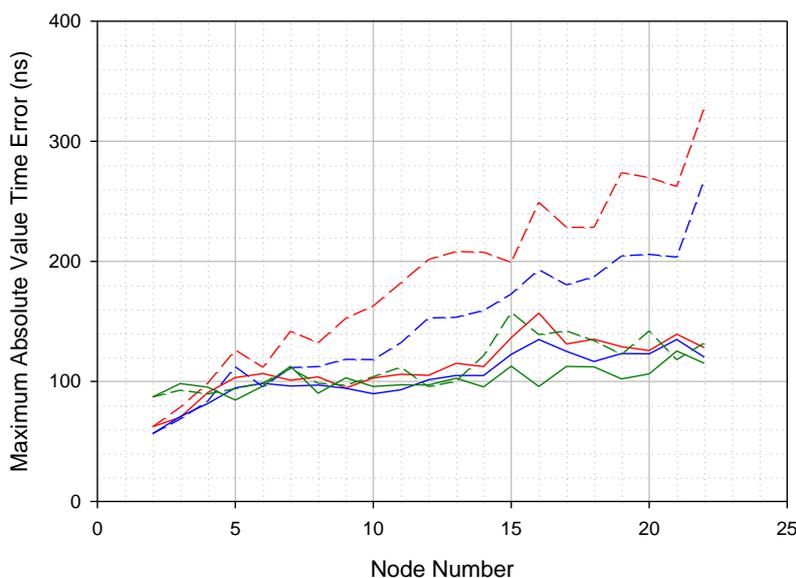
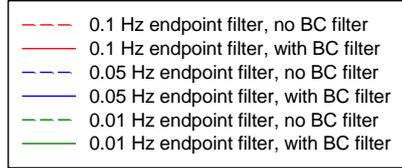
**Figure 109 – Case 2, comparison of results with and without T-BC filtering**

HRM3  
 0.1 dB gain peaking for endpoint and BC filters  
 0.1 Hz BC filter bandwidth  
 with noise generation in all filters  
 40 ns phase measurement granularity  
 Sync interval = 0.125 s  
 Pdelay interval = 1 s



**Figure 110 – Case 3, comparison of results with and without T-BC filtering**

HRM3  
 0.1 dB gain peaking for endpoint and BC filters  
 0.1 Hz BC filter bandwidth  
 with noise generation in all filters  
 40 ns phase measurement granularity  
 Sync interval = 1 s  
 Pdelay interval = 8 s



**Figure 111 – Case 4, comparison of results with and without T-BC filtering**

**Table 30 – Results for maximum absolute value of filtered time error, cases 1 to 4, with and without T-BC filtering for the respective endpoint filter bandwidths, for nodes 2, 12 and 22**

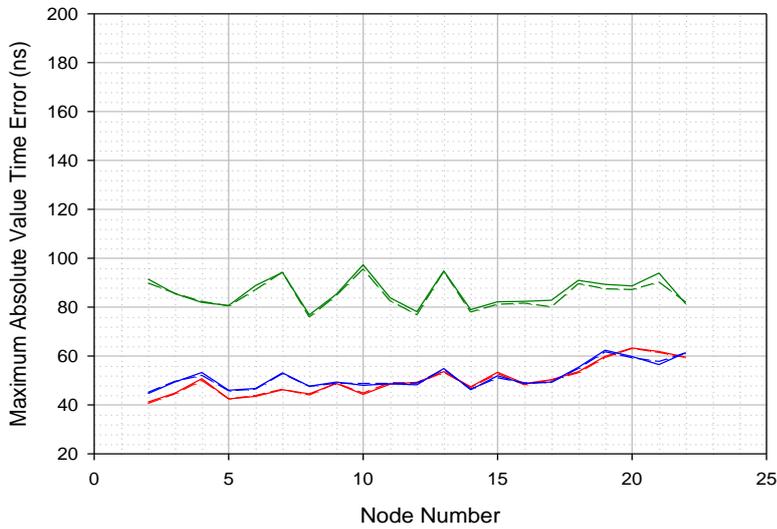
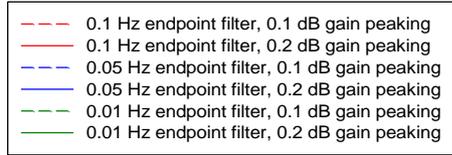
Case/Node	Maximum absolute value of time error (ns)					
	0.1 Hz endpoint filter		0.05 Hz endpoint filter		0.01 Hz endpoint filter	
	No T-BC filter	0.1 Hz T-BC filter	No T-BC filter	0.1 Hz T-BC filter	No T-BC filter	0.1 Hz T-BC filter
Case 1, Node 2	40.6	40.6	45.1	45.1	89.9	89.9
Case 1, Node 12	48.9	57.1	48.9	56.7	76.9	83.0
Case 1, Node 22	59.4	67.2	61.3	64.2	82.0	85.9
Case 2, Node 2	47.8	47.8	48.4	48.4	88.2	88.2
Case 2, Node 12	124.1	85.5	110.9	82.5	93.5	93.1
Case 2, Node 22	186.3	98.1	159.6	91.0	104.0	93.2
Case 3, Node 2	45.2	45.2	47.4	47.4	90.7	90.7
Case 3, Node 12	98.7	76.3	91.9	68.5	90.1	78.9
Case 3, Node 22	134.4	89.3	113.5	91.3	92.2	103.7
Case 4, Node 2	62.1	62.1	56.4	56.4	86.9	86.9
Case 4, Node 12	201.4	105.0	152.7	101.3	95.3	97.0
Case 4, Node 22	328.0	128.0	268.2	120.2	131.9	115.0

### 12.1.2.3 Comparison of results for 0.1 dB and 0.2 dB gain peaking

This clause considers the effect of increasing the gain peaking in all the filters (i.e., T-BC filters, if present, and endpoint filters) from 0.1 dB to 0.2 dB. As indicated in clause 2, the gain peaking for both T-BC and endpoint filters has not been decided. The 0.1 dB gain peaking corresponds to the maximum gain peaking requirement for many desynchronizers. However, various clock specifications require that the gain peaking not exceed 0.2 dB. The simulations with 0.2 dB gain peaking were run to determine if the results would be appreciably different compared to the results for 0.1 dB gain peaking.

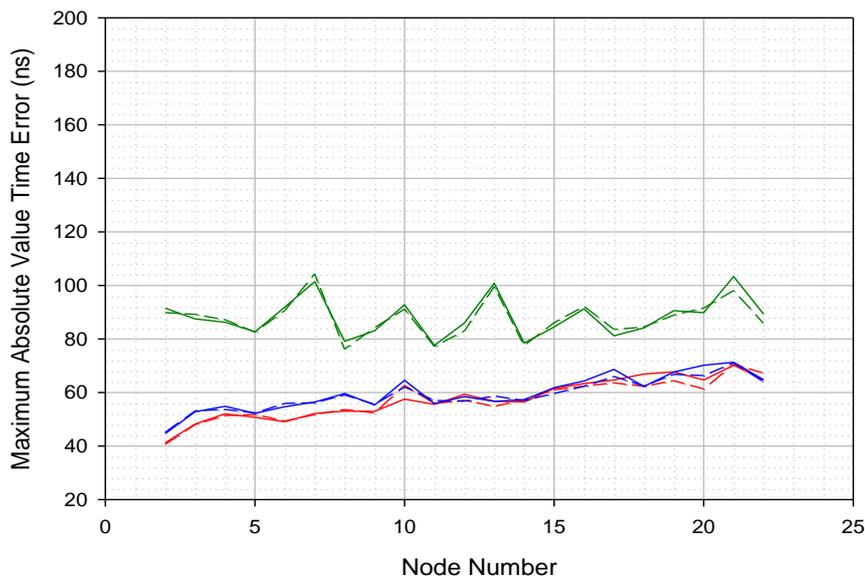
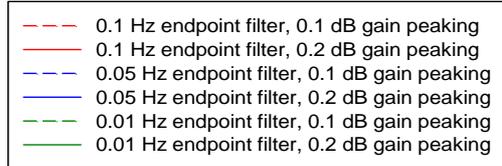
Figures 13 to 20 give the results for 0.1 dB and 0.2 dB gain peaking, for cases 1 to 4 with and without T-BC filtering (Figures 13 to 14 contain the results for case 1 with and without T-BC filtering, Figures 15 to 16 contain the results for case 2 with and without T-BC filtering, Figures 17 to 18 contain the results for case 3 with and without T-BC filtering, Figures 19 to 20 contain results for case 4 with and without T-BC filtering, respectively). In all the sub-cases, the corresponding results for 0.1 dB and 0.2 dB gain peaking are very close (in many instances they are almost identical). The conclusion is that the difference in performance between 0.1 dB and 0.2 dB gain peaking is negligible, at least for an HRM of 22 nodes (or fewer) considered in this Supplement.

HRM3  
 no BC filtering  
 with noise generation in all filters  
 8 ns phase measurement granularity  
 Sync interval = 0.125 s  
 Pdelay interval = 1 s



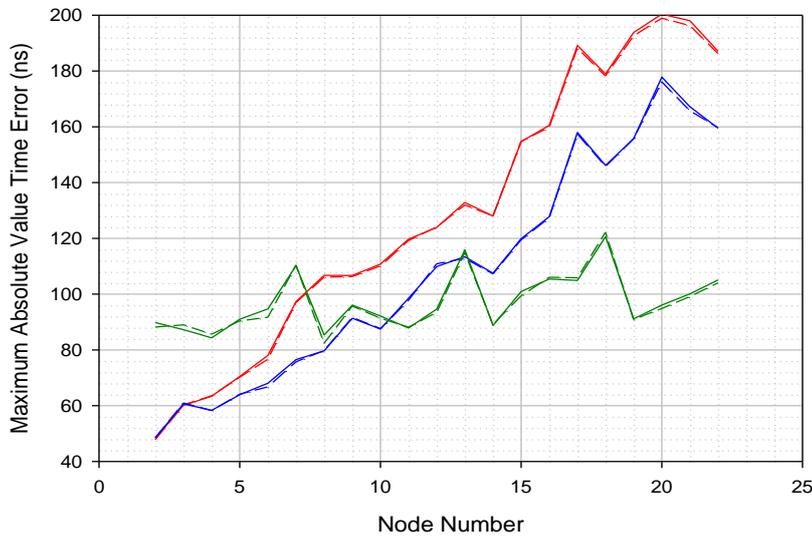
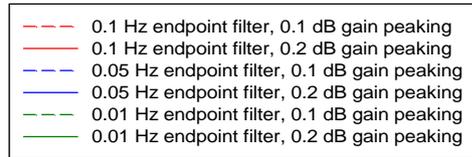
**Figure 112 – Case 1, no T-BC filtering, comparison of results for 0.1 dB and 0.2 dB gain peaking**

HRM3  
 0.1 Hz BC filtering with 0.1 or 0.2 dB gain peaking  
 with noise generation in all filters  
 8 ns phase measurement granularity  
 Sync interval = 0.125 s  
 Pdelay interval = 1 s



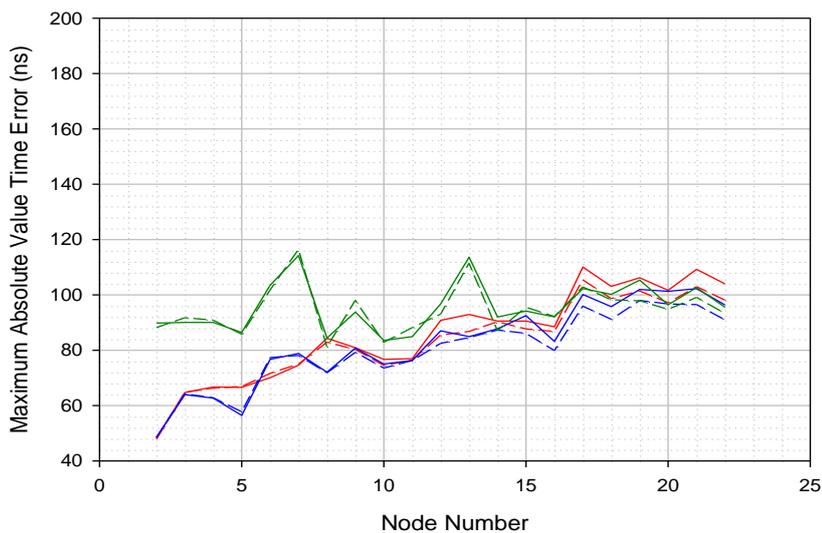
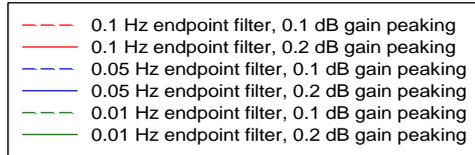
**Figure 113 – Case 1, 0.1 Hz T-BC filtering, comparison of results for 0.1 dB and 0.2 dB gain peaking**

HRM3  
 no BC filtering  
 with noise generation in all filters  
 8 ns phase measurement granularity  
 Sync interval = 1 s  
 Pdelay interval = 8 s



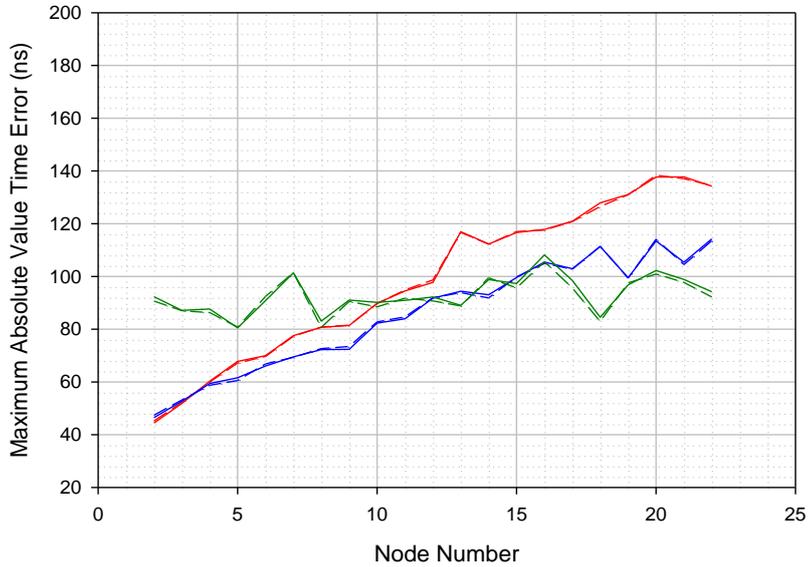
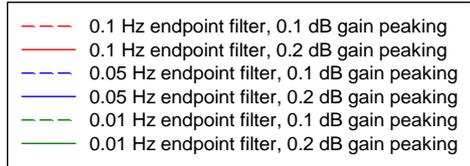
**Figure 114 – Case 2, no T-BC filtering, comparison of results for 0.1 dB and 0.2 dB gain peaking**

HRM3  
 0.1 Hz BC filtering with 0.1 or 0.2 dB gain peaking  
 with noise generation in all filters  
 8 ns phase measurement granularity  
 Sync interval = 1 s  
 Pdelay interval = 8 s



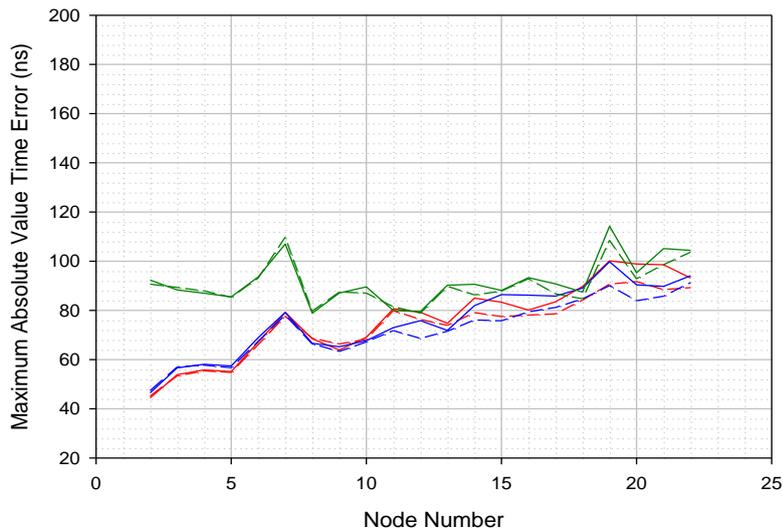
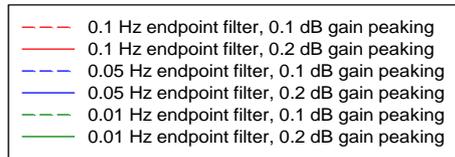
**Figure 115 – Case 2, 0.1 Hz T-BC filtering, comparison of results for 0.1 dB and 0.2 dB gain peaking**

HRM3  
 no BC filtering  
 with noise generation in all filters  
 40 ns phase measurement granularity  
 Sync interval = 0.125 s  
 Pdelay interval = 1 s



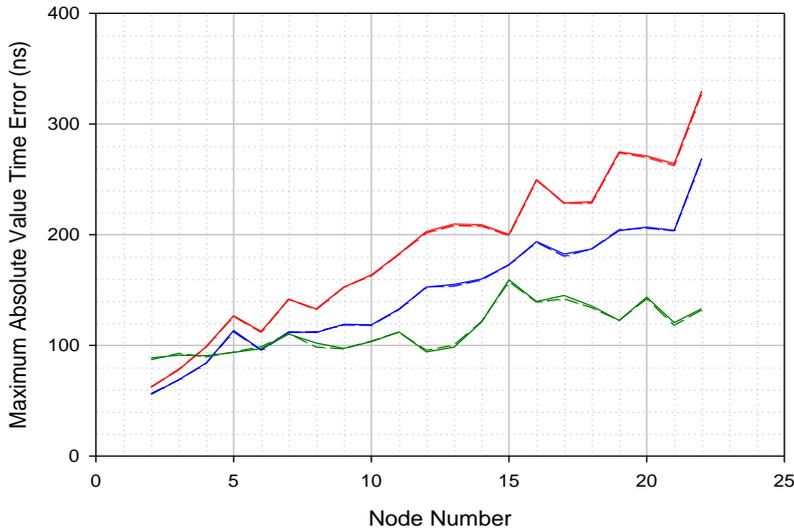
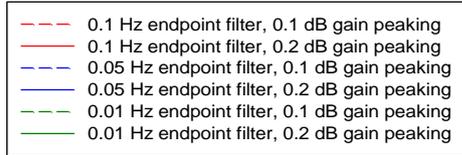
**Figure 116 – Case 3, no T-BC filtering, comparison of results for 0.1 dB and 0.2 dB gain peaking**

HRM3  
 0.1 Hz BC filtering with 0.1 or 0.2 dB gain peaking  
 with noise generation in all filters  
 40 ns phase measurement granularity  
 Sync interval = 0.125 s  
 Pdelay interval = 1 s



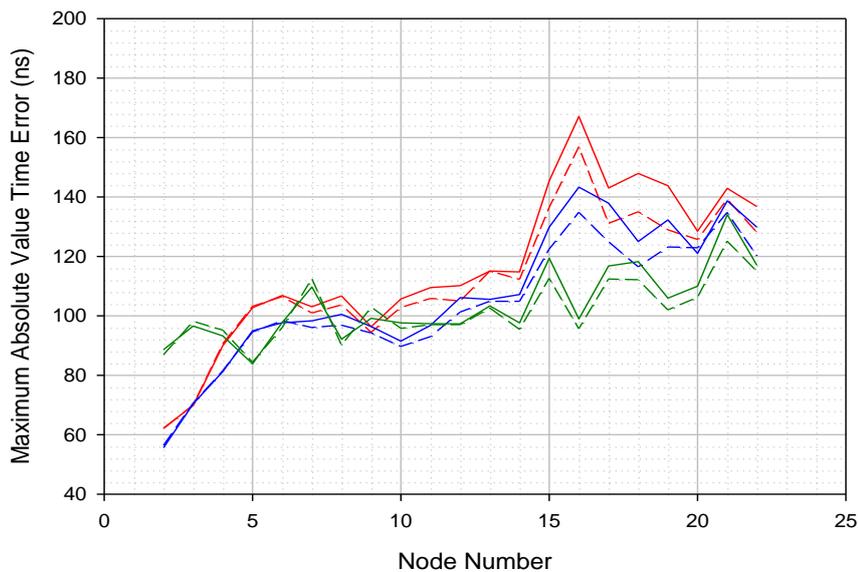
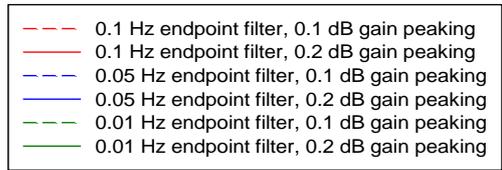
**Figure 117 – Case 3, 0.1 Hz T-BC filtering, comparison of results for 0.1 dB and 0.2 dB gain peaking**

HRM3  
 no BC filtering  
 with noise generation in all filters  
 40 ns phase measurement granularity  
 Sync interval = 1 s  
 Pdelay interval = 8 s



**Figure 118 – Case 4, no T-BC filtering, comparison of results for 0.1 dB and 0.2 dB gain peaking**

HRM3  
 0.1 Hz BC filtering with 0.1 or 0.2 dB gain peaking  
 with noise generation in all filters  
 40 ns phase measurement granularity  
 Sync interval = 1 s  
 Pdelay interval = 8 s



**Figure 119 – Case 4, 0.1 Hz T-BC filtering, comparison of results for 0.1 dB and 0.2 dB gain peaking**

#### 12.1.2.4 Summary and conclusions for HRM3 single-replication simulation cases

This clause has presented simulation results for transport of time over the HRM3 chain of T-BCs, for several endpoint filter bandwidths, possible filtering at the T-BCs, and noise generation due to SyncE modelled at all nodes. The cases simulated included several endpoint filter bandwidths, and sub-cases with and without filtering at the T-BCs.

The results with endpoint filtering but no T-BC filtering indicate that there are two competing effects of making the endpoint filter bandwidth narrower:

- a) Noise generation increases due to the smaller high-pass filter corner frequency, causing increased time error, and
- b) The narrower endpoint filter bandwidth results in narrower-bandwidth low-pass filtering of the time error, causing it to decrease.

For 8 ns timestamp granularity and smaller PTP message intervals (i.e., 0.125 s Sync interval and 1 s Pdelay interval), the unfiltered time error accumulation is relatively small. If the endpoint filter bandwidth is decreased from 0.1 Hz to 0.05 Hz, the decrease in time error accumulation due to the filtering is roughly the same as the increase due to the noise generation, and the net effect is roughly the same overall time error accumulation for both filter bandwidths (Figure 100). Further narrowing the bandwidth to 0.01 Hz results in larger time error accumulation, due to the increased noise generation. For 40 ns timestamp granularity and/or larger PTP message intervals (i.e., 1 s Sync interval and 8 s Pdelay interval), the maximum absolute value time error increases for a smaller number of hops and decreases for a larger number of hops as the endpoint filter bandwidth decreases.

It is concluded that narrowing the endpoint filter bandwidth does not necessarily result in improved performance. Whether or not the performance improves depends on the timestamp granularity and PTP message intervals.

The results with 0.1 Hz T-BC filtering also show two competing effects of adding the T-BC filters:

- a) The narrow-bandwidth filtering at each node reduces the time error accumulation up to that node.
- b) Noise generation of the T-BC filter tends to increase the time error. However, this is filtered at the next node (and the noise generation of the last T-BC is filtered at the endpoint filter).

Therefore, it is expected that T-BC filtering will have a larger benefit in cases where the time error accumulation is larger, e.g., (i) larger granularity, or (ii) larger PTP message intervals. As with endpoint filtering, there may little benefit of adding 0.1 Hz T-BC filtering for 8 ns timestamp granularity and smaller PTP message intervals (i.e., 0.125 s Sync interval and 1 s Pdelay interval). However, there may be benefit for 40 ns timestamp granularity and/or larger PTP message intervals (i.e., 1 s Sync interval and 8 s Pdelay interval).

The results with 0.1 Hz T-BC filtering also show that if this filtering is present, there is no benefit (and performance may actually be worse) in making the endpoint filter bandwidth narrower than 0.1 Hz, i.e., the endpoint and T-BC filter bandwidths can be the same. This conclusion will be relevant if the analysis of SyncE rearrangements indicates that there is benefit of T-BC filtering for these cases.

Finally, the comparison of results for 0.2 dB and 0.1 dB filter gain peaking indicate that the difference in maximum absolute value time error accumulation is small, and in many cases negligible.

## 12.2 Simulation results for HRM3 cases based on single replications of simulations

### 12.2.1 Description of simulation cases

The cases simulated here are all based on cases 1 and 3 of clause 12.1. A total of 24 sub-cases (with 0.1 dB gain peaking) were simulated there for cases 1 to 4. Twelve of those sub-cases were for cases 1 – 3. Here, 8 of those 12 sub-cases were simulated. It will be recalled that cases 1 and 3 of clause 12.1

have 0.125 s Sync interval and 1 s Pdelay interval. Case 1 has 8 ns timestamp granularity, and case 3 has 40 ns timestamp granularity.

Table 31 summarizes the parameters and assumptions common to all simulation sub-cases described here. The table is the same as Table 27 of clause 12.1.1, except that now the Sync and Pdelay intervals are included in the table (rather than in an additional table with parameters specific to each sub-case) because they are the same in all the sub-cases considered here.

**Table 31 – Model parameters and assumptions common to all simulation cases**

<b>Parameter</b>	<b>Value</b>
Hypothetical Reference Model	HRM3 of Appendix III of [ITU-T G.8271]
SyncE phase noise accumulation	Results of clause 8.1.4.2
SyncE PRC frequency accuracy	$\pm 10^{-11}$
Model for sending of Sync and Pdelay_Req messages	Model based on gamma distribution, as described in clause 11.2 above, such that actual message intervals are within $\pm 30\%$ of mean intervals with 90% confidence (Note that sending of Sync message on master port of T-BC is asynchronous with receiving of Sync message on slave port of same T-BC)
Actual link propagation time	0.1 ms (total for each link, including transmit and receive PHYs; based on 20 km link and approximately 5 ns/m group delay); taken to be the same in both directions
Link and PHY asymmetries	0 (i.e., no link or PHY asymmetry)
Assumption for Grandmaster time	Grandmaster is assumed perfect; time errors computed relative to T-GM
Pdelay turnaround time	10 ms
Simulation time	11,000 s (first 1000 s removed when computing peak or peak-to-peak values, to remove any initial transient)
Endpoint filter 3 dB bandwidth	0.05 Hz (note that results are also obtained for the case of no endpoint filtering)
Endpoint filter gain peaking	0.1 dB
T-BC filtering assumption	When present, 0.1 Hz 3 dB bandwidth with 0.1 dB gain peaking
One-step/two-step behaviour	T-BC and T-TSC models based on one-step behaviour
Timestamp granularity	40nsec or 8nsec
Noise generation in each filter (both endpoint filters and, if included, T-BC filters)	Modelled as described in clause 9.3.3
Sync interval	0.125 s
Pdelay interval	1 s

Table 32 summarizes all the simulation cases considered here. They are designated (a) – (h), and all correspond to cases 1 or 3 of clause 12.1. As in clause 12.1, the simulations are for HRM3 of Appendix III of [ITU-T G.8271] (see clause 7). This HRM consists of a T-GM, followed by 20 T-BCs, followed by a T-TSC. The number of hops therefore ranges from 1 to 21. Also as in clause 12.1, the nodes are numbered from 1 to 22, with node 1 being the grandmaster (T-GM), nodes 2 to 21 BCs, and node 22 the final T-TSC. Results are given for nodes 2 (the first T-BC after the T-GM) through 22 (the time error at node 1 is zero because the T-GM is assumed perfect).

**Table 32 – Summary of simulation cases**

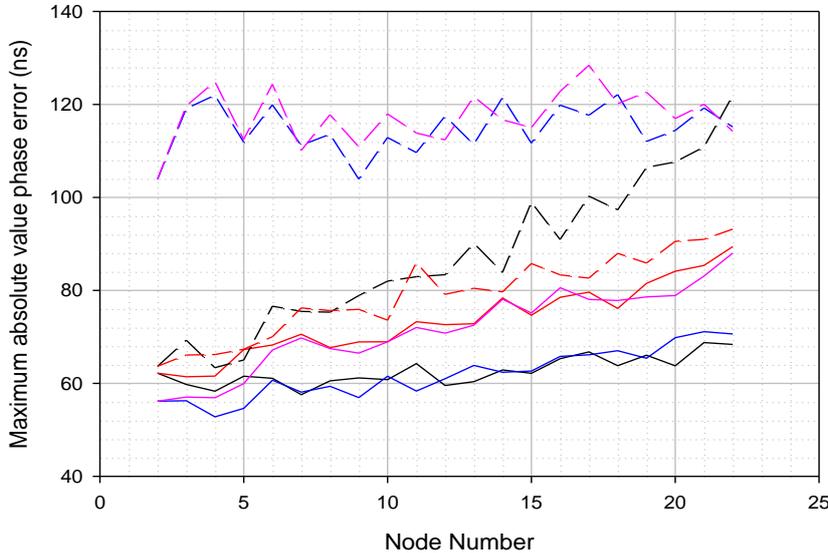
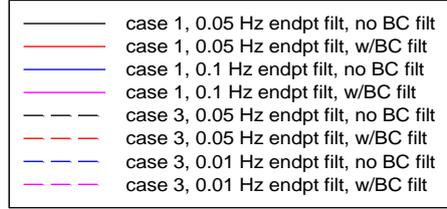
Case designation here	Table 28 case number	Timestamp granularity (ns)	Endpoint filter bandwidth (Hz)	0.1 Hz T-BC filter present
a	1	8	0.05	No
b	1	8	0.05	Yes
c	1	8	0.1	No
d	1	8	0.1	Yes
e	3	40	0.05	No
f	3	40	0.05	Yes
g	3	40	0.01	No
h	3	40	0.01	Yes

As discussed in clause 12.1, link and PHY asymmetry are not considered in these cases. The simulator includes models for these asymmetries in that the total actual link delays specified as inputs to the simulator need not be the same in both directions. As in clause 12.1, the main interest is in the effect of asymmetry on the worst-case. In any case, in this clause results are compared for different timestamp granularity, endpoint filter bandwidth, and T-BC filter present or not present. Asymmetry does not impact this comparison.

### 12.2.2 Simulation results

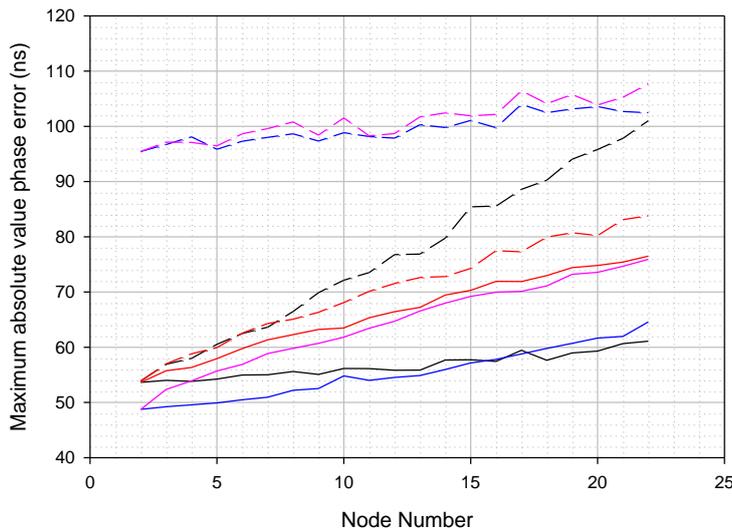
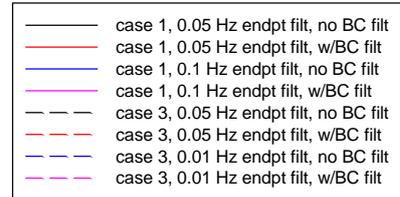
Figures 1 and 2 present results for the maximum absolute value of time error. A sample is computed for each replication as the maximum of the absolute value of time error for 1000 – 11000 s. Figure 120 shows the maximum over the 300 replications (300 samples), for each node. Figure 121 shows a point estimate of the 0.95 quantile, over the 300 replications. As indicated previously, the point estimate is taken as the 285<sup>th</sup> smallest sample, after the samples have been placed in ascending order.

Case a - h, maximum at each node over 300 replications  
 Noise generation modeled in all filters  
 0.1 dB gain peaking in all filters  
 0.1 Hz BC filtering, when BC filtering is present  
 0.125 s Sync interval, 1 s Pdelay interval  
 case 1: 8 ns timestamp granularity  
 case 3: 40 ns timestamp granularity



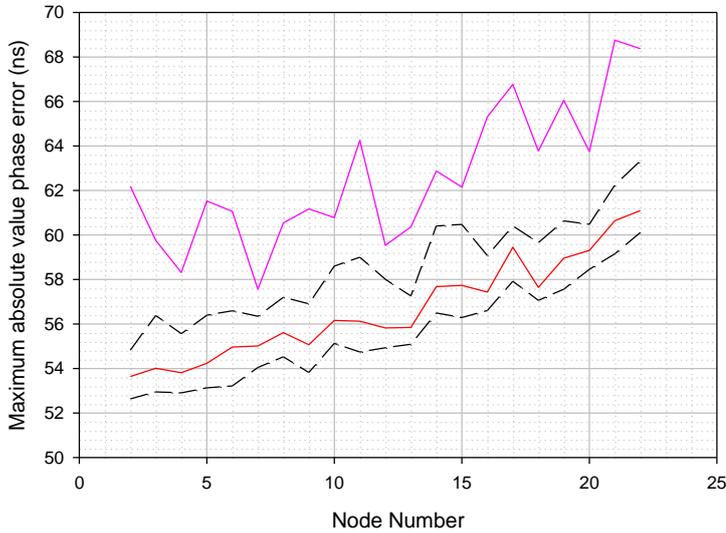
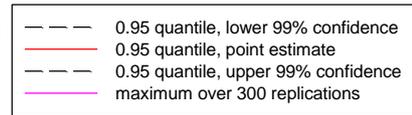
**Figure 120 – Cases a – h, maximum absolute value time error at each node, maximum over 300 replications**

Case a - h, 0.95 quantile point estimate at each node over 300 replications  
 Noise generation modeled in all filters  
 0.1 dB gain peaking in all filters  
 0.1 Hz BC filtering, when BC filtering is present  
 0.125 s Sync interval, 1 s Pdelay interval  
 case 1: 8 ns timestamp granularity  
 case 3: 40 ns timestamp granularity



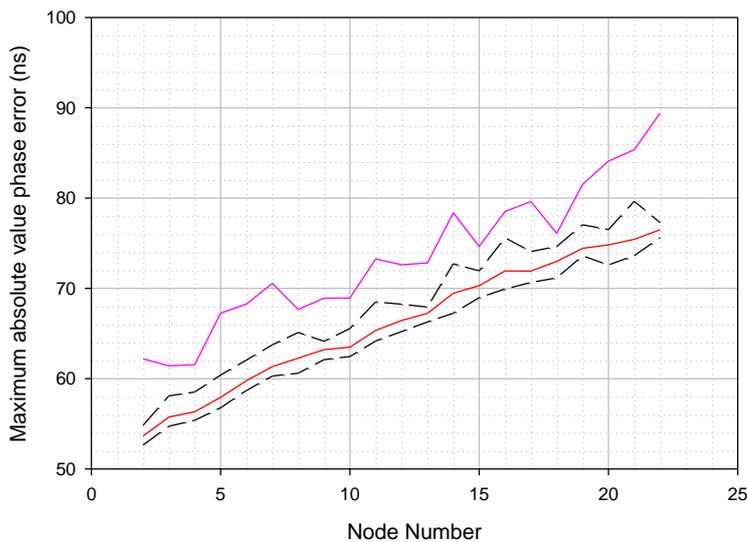
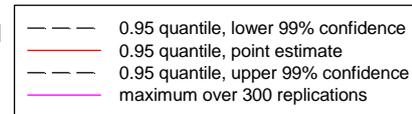
**Figure 121 – Cases a to h, maximum absolute value time error at each node, 0.95-quantile point estimate over 300 replications**

Case a: results for 300 replications  
 Maximum; 0.95 quantile point estimate and 99% confidence interval  
 Noise generation modeled in all filters  
 0.1 dB gain peaking in all filters  
 No BC filtering  
 0.05 Hz endpoint filtering  
 0.125 s Sync interval, 1 s Pdelay interval  
 case 1: 8 ns timestamp granularity



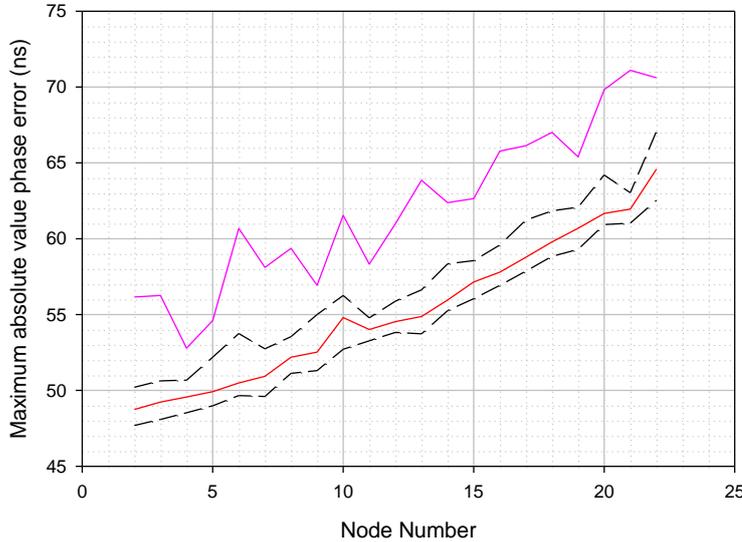
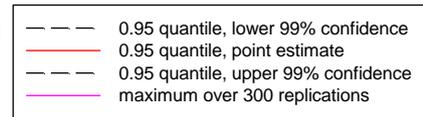
**Figure 122 – Case a, maximum absolute value time error, maximum over 300 replications and 0.95-quantile point estimate and 99% confidence interval**

Case b: results for 300 replications  
 Maximum; 0.95 quantile point estimate and 99% confidence interval  
 Noise generation modeled in all filters  
 0.1 dB gain peaking in all filters  
 0.1 Hz BC filtering  
 0.05 Hz endpoint filtering  
 0.125 s Sync interval, 1 s Pdelay interval  
 Case 1: 8 ns timestamp granularity



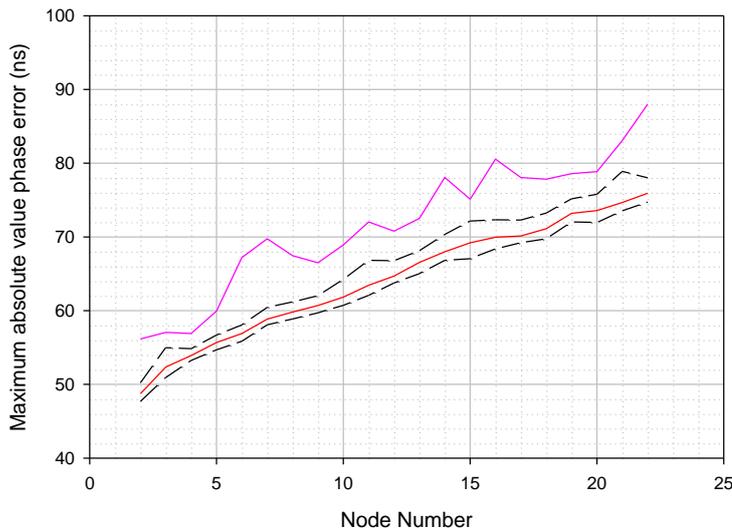
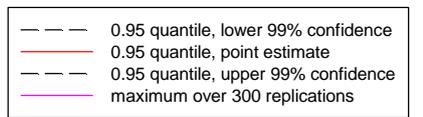
**Figure 123 – Case b, maximum absolute value time error, maximum over 300 replications and 0.95-quantile point estimate and 99% confidence interval**

Case c: results for 300 replications  
 Maximum; 0.95 quantile point estimate and 99% confidence interval  
 Noise generation modeled in all filters  
 0.1 dB gain peaking in all filters  
 No BC filtering  
 0.1 Hz endpoint filtering  
 0.125 s Sync interval, 1 s Pdelay interval  
 Case 1: 8 ns timestamp granularity



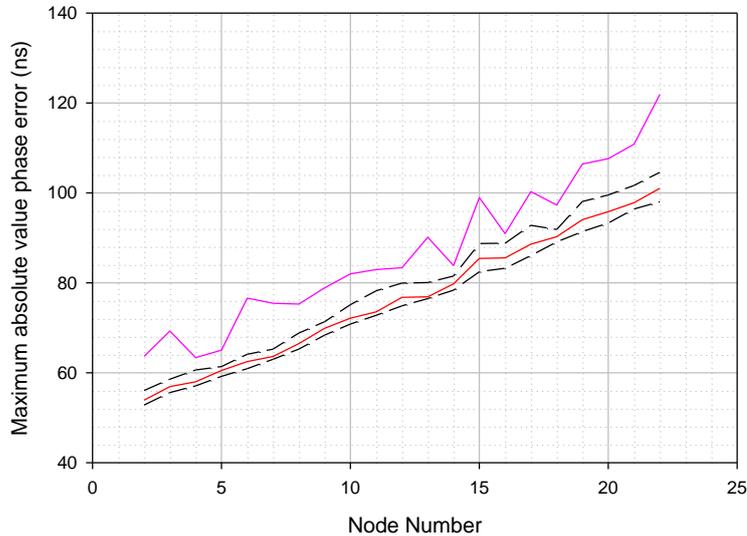
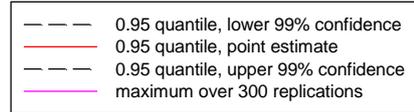
**Figure 124 – Case c, maximum absolute value time error, maximum over 300 replications and 0.95-quantile point estimate and 99% confidence interval**

Case d: results for 300 replications  
 Maximum; 0.95 quantile point estimate and 99% confidence interval  
 Noise generation modeled in all filters  
 0.1 dB gain peaking in all filters  
 0.1 Hz BC filtering  
 0.1 Hz endpoint filtering  
 0.125 s Sync interval, 1 s Pdelay interval  
 Case 1: 8 ns timestamp granularity



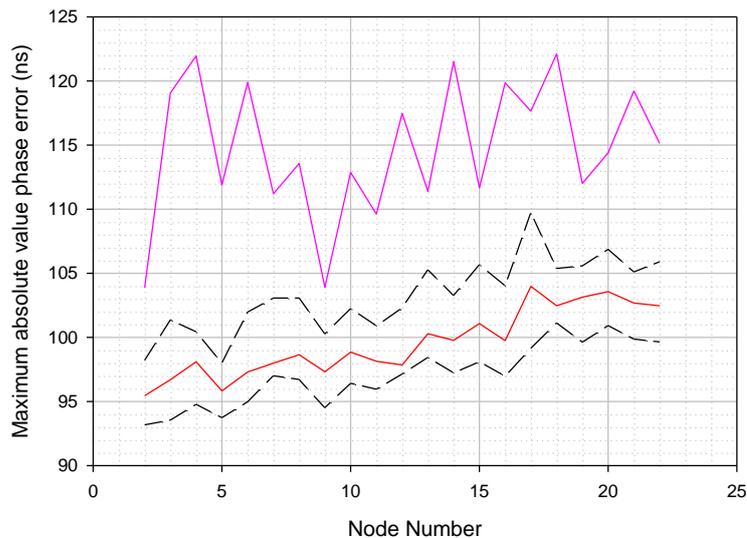
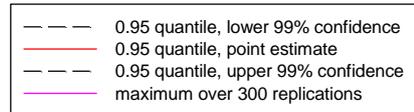
**Figure 125 – Case d, maximum absolute value time error, maximum over 300 replications and 0.95-quantile point estimate and 99% confidence interval**

Case e: results for 300 replications  
 Maximum; 0.95 quantile point estimate and 99% confidence interval  
 Noise generation modeled in all filters  
 0.1 dB gain peaking in all filters  
 No BC filtering  
 0.05 Hz endpoint filtering  
 0.125 s Sync interval, 1 s Pdelay interval  
 Case 3: 40 ns timestamp granularity



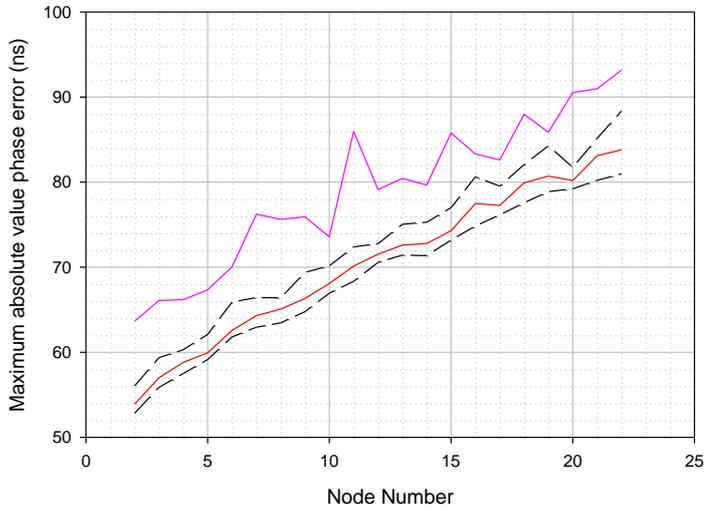
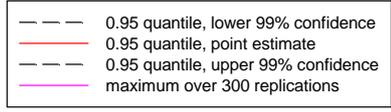
**Figure 126 – Case e, maximum absolute value time error, maximum over 300 replications and 0.95-quantile point estimate and 99% confidence interval**

Case f: results for 300 replications  
 Maximum; 0.95 quantile point estimate and 99% confidence interval  
 Noise generation modeled in all filters  
 0.1 dB gain peaking in all filters  
 No BC filtering  
 0.01 Hz endpoint filtering  
 0.125 s Sync interval, 1 s Pdelay interval  
 40 ns timestamp granularity



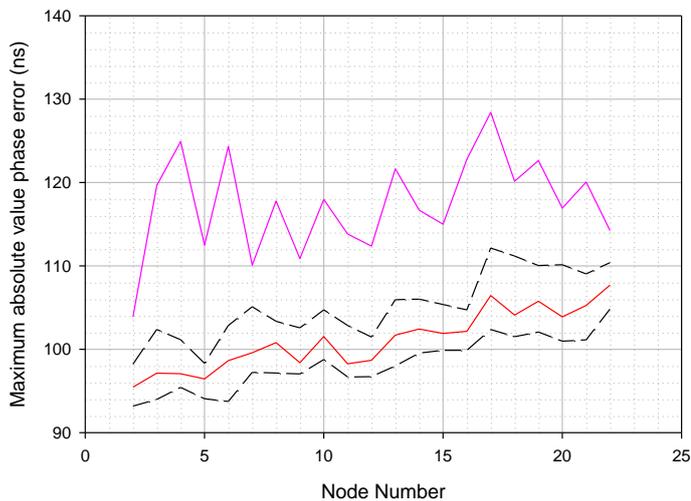
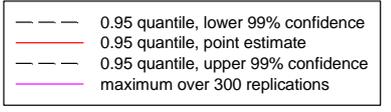
**Figure 127 – Case f, maximum absolute value time error, maximum over 300 replications and 0.95-quantile point estimate and 99% confidence interval**

Case g: results for 300 replications  
 Maximum; 0.95 quantile point estimate and 99% confidence interval  
 Noise generation modeled in all filters  
 0.1 dB gain peaking in all filters  
 0.1 Hz BC filtering  
 0.05 Hz endpoint filtering  
 0.125 s Sync interval, 1 s Pdelay interval  
 Case 3: 40 ns timestamp granularity



**Figure 128 – Case g, maximum absolute value time error, maximum over 300 replications and 0.95-quantile point estimate and 99% confidence interval**

Case h: results for 300 replications  
 Maximum; 0.95 quantile point estimate and 99% confidence interval  
 Noise generation modeled in all filters  
 0.1 dB gain peaking in all filters  
 0.1 Hz BC filtering  
 0.01 Hz endpoint filtering  
 0.125 s Sync interval, 1 s Pdelay interval  
 Case 3: 40 ns timestamp granularity



**Figure 129 – Case h, maximum absolute value time error, maximum over 300 replications and 0.95-quantile point estimate and 99% confidence interval**

The case a, b, c and d results in Figures 120 and 121 are given by the solid lines in the plots. These correspond to the single replication results given by the red and blue curves in Figure 108 of clause 12.1. The blue dashed and solid curves (in Figure 108 of clause 12.1) are single-replication results for case a (solid black curve in Figures 120 and 121) and b (solid red curve in Figures 120 and 121), respectively. The red dashed and solid curves (in Figure 108 of clause 12.1) are single-replication results for case c (solid blue curve in Figures 120 and 121) and d (solid pink curve in Figures 120 and 121), respectively.

The case a, b, c and d results in Figure 120 and Figure 121 are consistent with the results in Figure 108 of clause 12.1. The results with T-BC filtering (case b and d, solid red and pink curves here) show relatively little difference in magnitude, as do the solid blue and red curves in Figure 108 of clause 12.1. The results given in Figure 120 are somewhat larger than those of Figure 121, since the former are maximum values over the 300 replications, while the latter are point estimates for the 0.95-quantile (and are the 285<sup>th</sup> smallest rather than the 300<sup>th</sup> smallest (i.e., largest) values). For example, the former are approximately 88 ns at node 22, while the latter are approximately 76 ns. Similarly, the results given in Figure 121 are somewhat above the results in Figure 108 of clause 12.1, as the latter are for a single replication. For example, the former are approximately 76 ns at node 22, while the latter are approximately 65 to 70 ns.

Likewise, the results without T-BC filtering (case a and c solid black and blue curves here) show relatively little difference in magnitude, as do the dashed blue and red curves in Figure 108 of clause 12.1. As for the cases with T-BC filtering, the results given in Figure 120 are somewhat larger than those in Figure 121, since the former are maximum values over the 300 replications, while the latter are point estimates for the 0.95-quantile). For example, the former are approximately 65 to 70 ns at node 22, while the latter are approximately 60 to 65 ns. Similarly, the results given in Figure 121 are somewhat above the results in Figure 108 of clause 12.1, as the latter are for a single replication.

The results for cases e, f, g and h (dashed lines in Figures 120 and 121) clearly show that the phase error is larger for 0.01 Hz endpoint filter (cases g and h, blue and pink dashed lines) than for 0.05 Hz endpoint filtering (cases e and f, black and red dashed lines), both with and without T-BC filtering. This is consistent with the results of Figure 110 of clause 12.1 (the blue dashed and solid curves of Figure 110 of clause 12.1 correspond to cases e (black dashed curve of Figures 120 and 121) and f (red dashed curve of Figures 120 and 121 here), respectively). The green dashed and solid curves of Figure 110 of clause 12.1 correspond to cases g (blue dashed curve of Figures 120 and 121) and h (pink dashed curve of Figures 120 and 121), respectively). Note that the increased noise generation for 0.01 Hz endpoint filter is due to the noise generation and the lower high-pass filter corner frequency (see clause 12.1). As for cases a, b, c and d, the maximum values in Figure 120 are larger than the 0.95-quantile point estimates of Figure 121. The point estimates are, in turn, usually larger than the corresponding values in Figure 110 of 12.1; however, this is not always true. For example, the point estimate for case e, at node 22 (Figure 121, black dashed curve) is approximately 101 ns. The corresponding result in Figure 110 of clause 12.1 (blue dashed curve) is approximately 114 ns, which is larger. The maximum result of Figure 120 (black dashed curve) is 120 ns, which exceeds both the 0.95-quantile point estimate and the result of 12.1. This is expected, since the results given in Figure 120 are the maximum over 300 replications and use the same initial random number state for the first replication as the simulations in clause 12.1 (consequently, the maximum over the 300 replications simulated here cannot be less than the corresponding result in clause 12.1).

In summary, the results of Figures 120 and 121 are consistent with the conclusions of clause 12.1 that:

- a) making the endpoint filter bandwidth less than 0.05 Hz, e.g., 0.01 Hz, can increase the time error accumulation, due to noise generation and the lower high-pass filter corner frequency; and

- b) if 0.1 Hz endpoint filtering is present, there is no benefit in making the endpoint filter bandwidth narrower than 0.1 Hz (and making the endpoint filter bandwidth much narrower, e.g., 0.01 Hz, can result in worse performance due to noise generation.

Figures 122 to 129 show 99% confidence intervals for the 0.95-quantile of maximum absolute value time error, for cases a to h, respectively, along with the maximum value over the 300 replications, as a function of node number. The variation in point estimate with node number is much closer to a linear trend than the variation in maximum value with point estimate.

### 12.2.3 Summary and conclusions for HRM3 multiple-replication simulation cases

This clause has presented initial simulation results based on multiple replications for transport of time over the HRM3 chain of T-BCs. The cases simulated were a subset of the sub-cases of clause 12.1 (single-replication cases) based on cases 1 and 3. These cases have 0.125 s Sync interval, 1 s Pdelay interval, and either 8 ns (case 1) or 40 ns (case 3) timestamp granularity. Eight of the sub-cases were simulated, which covered several endpoint filter bandwidths with and without 0.1 Hz T-BC filtering present.

The results obtained here are consistent with the results of clause 12.1, i.e.:

- a) If the endpoint filter bandwidth is made narrower, the time error accumulation can increase due to noise generation and the decrease in the high-pass filter corner frequency (this was observed in cases where the endpoint filter bandwidth was decreased from 0.05 Hz to 0.01 Hz, consistent with the results of clause 12.1.
- b) If 0.1 Hz T-BC filtering is present, there is no benefit in making the endpoint filter bandwidth narrower than 0.1 Hz. Making the endpoint filter bandwidth much narrower, e.g., 0.01 Hz, can result in worse performance due to noise generation.

The multiple replication results for maximum absolute value time error obtained here were usually (but not always) somewhat larger than the results obtained for corresponding sub-cases in clause 12.1 based on single replications. Specifically, the 0.95-quantile point estimates and confidence intervals were usually somewhat larger than the corresponding single replication results of clause 12.1. This was expected, as the 99% confidence intervals are bounded by the 275<sup>th</sup> and 294<sup>th</sup> smallest values of the 300 runs, for each case; usually they will be larger, but there can be an exception if a single replication sub-case of clause 12.1, which corresponds to the first replication for the corresponding case here, happens to give results that are between the 295<sup>th</sup> and 300<sup>th</sup> smallest values for the 300 replications. The results of 12.1 were never larger than the corresponding maximum results (over the 300 replications) obtained here.

## 12.3 Simulation results for HRM3 cases with timestamping relative to the corrected or uncorrected clock in PTP, based on single replications of simulations

### 12.3.1 Introduction

There can be a potential source of time error accumulation in a chain of boundary clocks that is not present in the simulation model that is based on the peer-to-peer delay mechanism with timestamping relative to the uncorrected clock (i.e., model B of clause 9). The error can arise when Sync and either Pdelay\_Req/Pdelay\_Resp or Delay\_Req messages are timestamped relative to the corrected clock, i.e., relative to the T-BC's estimate of the grandmaster (T-GM) time. Specifically, a PTP slave port computes mean propagation delay and time offset using four timestamps. The first timestamp, T1, represents the estimate of the T-GM time at the T-BC upstream of the slave port of the T-BC in question when a Sync message is sent. The value of T1 is given by the sum of the originTimestamp and correction field of the Sync message in the case of a one-step clock (or the sum of the preciseOriginTimestamp and correctionField of the associated Follow\_Up message in the case of a two-step clock). The second timestamp, T2, represents the time of arrival of the Sync message at the slave port of the T-BC in question (i.e., the <syncEventIngressTimestamp>). The third timestamp, T3, represents the time of transmission of a Delay\_Req message from the slave port of the T-BC in

question to the master port of the upstream T-BC. The fourth timestamp, T4, represents the time of arrival of the Delay\_Req message at the master port of the upstream T-BC. T4 is conveyed to the slave port in the associated Delay\_Resp message, and the slave port can use T1, T2, T3 and T4 to compute both the mean propagation delay and the time offset.

An error in time offset can arise if (a) the corrected clock at the master is adjusted in between the timestamps T1 and T4 (i.e., in between the sending of Sync and receipt of Delay\_Req), or (b) the corrected clock at the slave is adjusted in between the timestamps T2 and T3 (i.e., in between the receipt of Sync and the sending of Delay\_Req).

One way to avoid this error is to timestamp relative to the uncorrected clock. In that case, the error does not arise because the clock is not corrected. In addition, if timestamping is done relative to the uncorrected clock, the peer-to-peer delay mechanism must be used. In the Delay Request-Response mechanism, the timestamp T1 is always relative to the corrected clock, i.e., it represents the estimate of the T-GM time when the Sync message is transmitted by the upstream node. This means that at least the timestamp T4 must also be relative to the corrected clock, because T1 and T4 must be relative to the same timebase. In this case, the error can arise if there is a corrected clock adjustment at the upstream node in between T1 and T4 (e.g., due to the arrival of a Sync message from the node upstream of that node). However, if the peer-to-peer mechanism is used, the Pdelay message timestamps T1 (transmission of Pdelay\_Req by the requestor), T2 (receipt of Pdelay\_Req by the responder), T3 (transmission of Pdelay\_Resp by the responder), and T4 (receipt of Pdelay\_Resp by the requestor) can all be relative to the uncorrected clock. It should be noted that the Sync message still carries the originTimestamp and correctionField, which together carry the upstream T-BC's estimate of the T-GM time, but this is not needed for the mean propagation delay measurement.

The magnitude of the error could be significantly reduced by filtering (e.g., PLL filtering) in the T-BC. Such filtering would cause corrected clock time adjustments to be gradual rather than instantaneous, and the magnitude of such adjustments during the time interval between T2 and T3 or between T1 and T4 would be smaller. In response to this, it should be noted that while filtering might reduce the error, it would not eliminate it.

This clause presents simulation results to compare the performance of timestamping relative to the corrected and uncorrected clocks. Simulation results are provided for corresponding cases using (a) Pdelay mechanism and timestamping relative to the uncorrected clock, (b) Pdelay mechanism and timestamping relative to the corrected clock, (c) Delay Request/Response mechanism and timestamping relative to the corrected clock. For each case, examples with and without T-BC filtering are considered. The results indicate that, for the Sync, Pdelay, and Delay\_Req intervals, and Pdelay turnaround time considered, all three cases give similar time error performance if 0.1 Hz T-BC filtering is present in the BCs. However, the performance for the Delay Request/Response mechanism is considerably worse if T-BC filtering is not used.

### **12.3.2 Description of simulation cases**

The simulation cases are all based on case 1 of clause 12.1 This case uses HRM3. The timestamp granularity is 8 ns, the mean Sync interval is 0.125 s, the mean Pdelay or Delay\_Req interval is 1 s, and the Pdelay or Delay Request/Response turnaround time is 10 ms. T-BC and endpoint filters, when present, have 3 dB bandwidth of 0.1 Hz and 0.1 dB gain peaking. The full set of simulation parameters are presented in detail in Appendix I of the present contribution.

As indicated in clause 12.3.1, we considered three different sub-cases, for each case, regarding the delay measurement mechanism and timestamping relative to the corrected or uncorrected clock: (a) Pdelay mechanism and timestamping relative to the uncorrected clock, (b) Pdelay mechanism and timestamping relative to the corrected clock, (c) Delay Request/Response mechanism and timestamping relative to the corrected clock.

Three main cases were considered, and are summarized in Table 33. In case 1, the initial time offset at each T-BC to a different non-zero value are clearly set. The purpose of this case was to see if the

convergence time to steady-state would be different for timestamping relative to the corrected clock versus uncorrected clock. It was initially thought that it might possibly be different because initial timestamps relative to the corrected clocks would be in error by the respective initial phase offsets. However, the results showed that any effect on the convergence time is small compared to the effect of 0.1 Hz filtering. In view of this, the discussion of case 1 (including the specific initial time offset assumed at each node) is given in Appendix I.

**Table 33 – Summary of cases simulated**

Case	SyncE noise	T-BC filtering		Non-zero initial time offsets
1	No	No		Yes
2	No	Results obtained both with and without T-BC filtering		No
3	Yes			No

For each case/sub-case, a single replication was simulated. The simulation time for cases 2 and 3 was 12500 s. Maximum absolute value time error for a simulation run was computed after discarding the first 1000 s (to eliminate any initial transient). The same initial random number generator state and SyncE phase histories (in cases that model SyncE noise) was used for all simulation runs, to ensure that results for different sub-cases (i.e., different assumptions on delay measurement mechanism and timestamping relative to the corrected versus uncorrected clock) are comparable.

Table 34 summarizes the parameters and assumptions common to all the simulation cases.

**Table 34 – Model parameters and assumptions common to all simulation cases**

Parameter	Value
Hypothetical reference model	HRM3 Appendix III of [ITU-T G.8271.1]
SyncE phase noise accumulation	Results of clause 8.1.4.2
SyncE PRC frequency accuracy	$\pm 10^{-11}$
Model for sending of Sync and Pdelay_Req messages	Model based on gamma distribution, as described in clause 11.2, such that actual message intervals are within $\pm 30\%$ of mean intervals with 90% confidence, but do not exceed twice the mean interval.
Actual link propagation time	0.1 ms (total for each link, including transmit and receive PHYs; based on 20 km link and approximately 5 ns/m group delay); taken to be the same in both directions
Link and PHY asymmetries	0 (i.e., no link or PHY asymmetry)
Assumption for Grandmaster time	Grandmaster is assumed perfect; time errors computed relative to T-GM
Pdelay turnaround time	10 ms
Simulation time	Case 1: 2100 s Cases 2 and 3: 12,500 s (first 1000 s removed when computing peak or peak-to-peak values, to remove any initial transient)
Endpoint filter 3 dB bandwidth	0.1 Hz (note that results are also obtained for the case of no endpoint filtering for selected cases)
Endpoint filter gain peaking	0.1 dB (corresponds to damping ratio of 4.3138)

**Table 34 – Model parameters and assumptions common to all simulation cases**

Parameter	Value
T-BC filtering assumption	When present, 0.1 Hz 3 dB bandwidth with 0.1 dB gain peaking
One-step/two-step behavior	T-BC and T-TSC models based on one-step behavior
Noise generation in each filter (both endpoint filters and, if included, T-BC filters)	Modelled as described in clause 9.3.3, where now the SyncE phase error that is high-pass filtered includes both phase noise accumulation and phase error due to rearrangements

The simulation cases are described in Table 34. For case 1, the initial time offset at time zero at each T-BC and the final T-TSC was assumed to be non-zero, to see if there is any difference in time to reach steady state for the three sub-cases (timestamping relative to the corrected clock using either the Pdelay or Delay Request/Response mechanism, timestamping relative to the uncorrected clock using the Pdelay mechanism). The initial time offset at each node is given in Table 35.

**Table 35 – Initial time offsets for case 1**

Initial time offsets for Case 1					
Node	Initial time offset (ns)	Node	Initial Time Offset (ns)	Node	Initial Time Offset (ns)
1	0	9	896000	17	6259
2	4000	10	-536000	18	3389
3	6890	11	336000	19	6050
4	6000	12	226000	20	6900
5	6000	13	336000	21	7200
6	333000	14	-26000	22	3000
7	46000	15	-6000		
8	776000	16	1090		

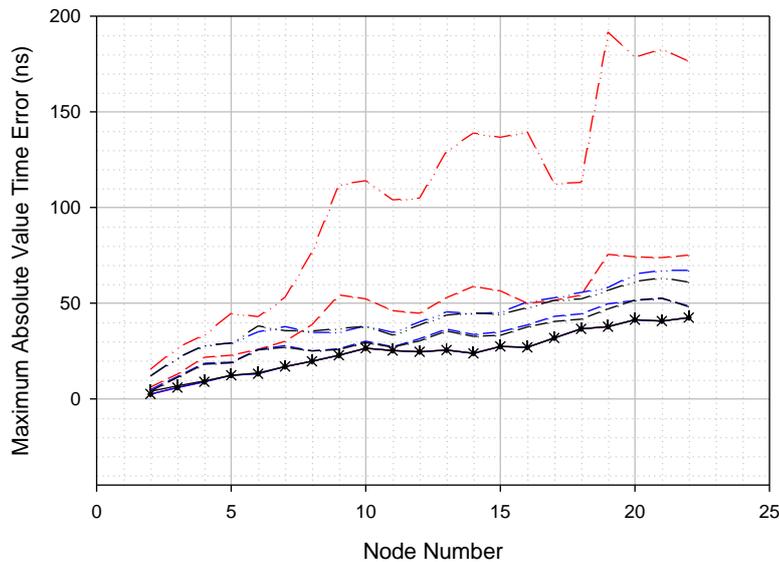
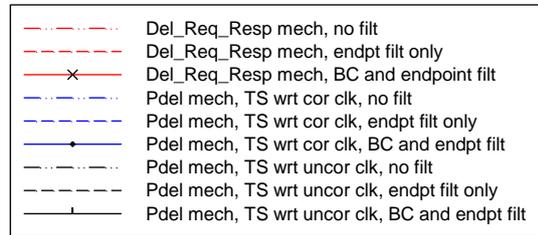
### 12.3.3 Simulation results

The results for maximum absolute value time error for case 2 (no SyncE phase noise) are shown in Figure 130. The three solid lines (which have the ×, •, and | symbols) are the results for the three sub-cases with both endpoint and T-BC filtering. These results are practically identical, and also are better in terms of performance than any of the results without endpoint or T-BC filtering. In contrast, if only endpoint filtering is used, or if there is neither endpoint nor T-BC filtering, the performance of the sub-cases that use the Pdelay mechanism with timestamping relative to the corrected or uncorrected clock is similar. However, the performance of the sub-cases that use the Delay Request/Response mechanism is significantly worse if there is no T-BC filtering.

The results are explained as follows. If there is no T-BC filtering, the corrected clock adjustments at the BCs is instantaneous on receipt of a Sync message. As explained in the introduction, this results in a time error if a Sync message arrives and causes a correction between the timestamps T1 and T4 or between the timestamps T2 and T3. For the Delay Request/Response mechanism, the sending of Sync by a master port at one T-BC and the sending of Delay\_Req by the downstream slave port are independent processes. The time between T1 and T4 or between T2 and T3 can be on the order of a Sync interval, and there is ample opportunity for a Sync message to arrive from upstream between such pairs of event. However, for the Pdelay mechanism, a Sync message has to arrive in an interval on the order of the Pdelay turnaround time for the larger time error obtained with the Delay Request/Response mechanism to be produced. This is a much lower probability event, because the

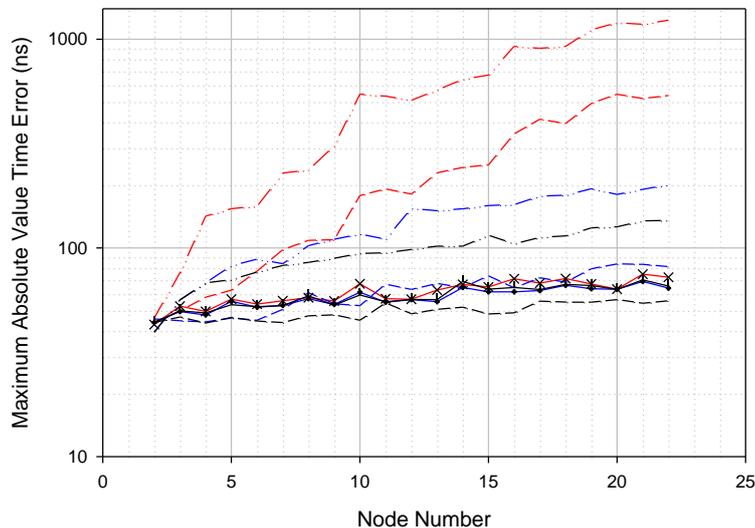
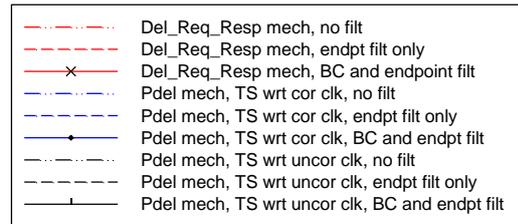
Pdelay turnaround time is 10 ms, i.e.,  $10/125 = 0.08$  of a Sync interval. If there is T-BC filtering, the results for the Delay Request/Response mechanism are significantly better because the filtering prevents the corrected clock from changing rapidly over a Sync interval. This is because the 0.1 Hz clock bandwidth corresponds to a time constant of  $1/[2\pi(0.1 \text{ Hz})] = 1.27 \text{ s}$ , which is approximately 10 times as long as the 0.125 s mean Sync interval.

HRM3  
 0.1 dB gain peaking for endpoint and BC filters  
 0.1 Hz BC filtering when present  
 0.1 Hz endpoint filtering when present  
 No SyncE noise (or SyncE noise generation)  
 8 ns phase measurement granularity  
 Sync interval = 0.125 s  
 Pdelay interval = 1 s  
 10 ms turnaround time



**Figure 130 – Maximum absolute value time error results for case 2 sub-cases**

HRM3  
 0.1 dB gain peaking for endpoint and BC filters  
 0.1 Hz BC filtering when present  
 0.1 Hz endpoint filtering when present  
 No SyncE noise (or SyncE noise generation)  
 8 ns phase measurement granularity  
 Sync interval = 0.125 s  
 Pdelay interval = 1 s  
 10 ms turnaround time



**Figure 131 – Maximum absolute value time error results for case 3 sub-cases**

The results for the maximum absolute value time error for case 3 (with SyncE phase noise) are shown in Figure 131 (note that the vertical scale is a log scale; this was done so that the details of the smaller results would be easily visible). The results for the three sub-cases with both endpoint and T-BC filtering, while not identical, are very similar (maximum absolute value time error between 60 and 80 ns). The performance of these three sub-cases (i.e., the solid line red, blue, and black curves) is also similar to the performance of cases with only endpoint filtering that use the Pdelay mechanism, with timestamping relative to either the corrected or uncorrected clock (the dashed blue and black curves, respectively). In fact, the performance of the Pdelay mechanism with only endpoint filtering and timestamping relative to the uncorrected clock is slightly better, and is the best of all the sub-cases (maximum absolute value time error between 50 and 60 ns). The Pdelay cases with no filtering at all have maximum absolute value time error of approximately 130 ns for timestamping relative to the uncorrected clock, and 200 ns for timestamping relative to the corrected clock. Finally, the cases using the Delay Request/Response mechanism have the maximum absolute value time error of approximately 520 ns if there is only endpoint filtering, and 1100 ns if there is no filtering at all.

Detailed time history results for selected nodes and sub-cases of cases 2 and 3, as well as case 1 results, are given in clause 12.3.3.1.

### 12.3.3.1 Results for time history of time error at selected BCs for selected cases/sub-cases

This clause shows the results for the time history of time error, for selected nodes, for cases 1 to 3. The results for each case are in a separate subclause.

#### 12.3.3.1.1 Results for case 1

It can be recalled from Table 33 that in case 1 there is no SyncE noise and no T-BC filtering. The results for both with and without endpoint filtering were looked at. Initial time offset at each node was non-zero, as indicated in Table 35.

Figure 132 shows the results for the Delay Request/Response mechanism, for the unfiltered time error at node 2, i.e., the first T-BC after the T-GM. There is a sizeable initial transient, but it is over in the first 10 s. Analogous results are shown for the Pdelay mechanism with timestamping relative to the corrected clock in Figure 142, and for the Pdelay mechanism with timestamping relative to the uncorrected clock in Figure 152. The two sets of results for the Pdelay mechanism also show a sizeable initial transient that is over in the first 10 s. In addition, the Pdelay results are very similar to each other. This is due to the error described in clause 12.3.1 which has very little effect when the Pdelay mechanism is used due to the small Pdelay turnaround time, while with the Delay Request/Response mechanism the time between the sending of Delay\_Req and receipt of the most recent Sync can be as large as one Sync interval. The time error is somewhat larger in the Delay Request Response case than in the Pdelay cases.

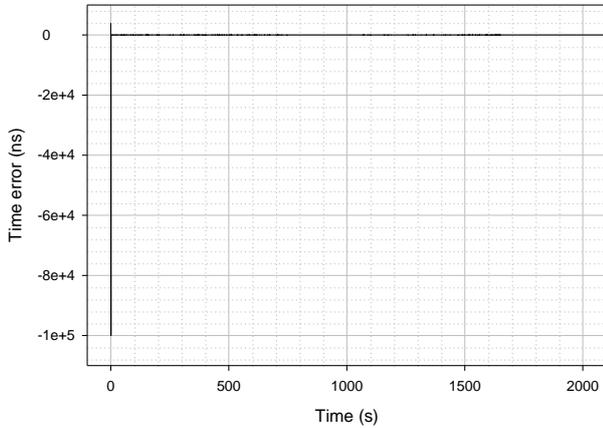
Figures 132 and 133 show the results for the Delay Request/Response mechanism, for time error at node 2 with 0.1 Hz endpoint filtering. The duration of the transient is increased due to the filtering. The exact duration depends on how close the steady-state error considered must be in order to declare that the transient is over. If it must be within the approximate steady-state error bounds, then the transient duration is approximately 500 s. Figures 143 and 144, show analogous results for the Pdelay mechanism with timestamping relative to the corrected clock, and Figures 153 and 154 for timestamping relative to the uncorrected clock. The transient duration is approximately 500 s in these cases also. The steady-state performance is similar for the two Pdelay cases, and worse for the Delay Request/Response case.

Figures 135 and 136 show the mean propagation delay upstream of node 2, for the Delay Request/Response mechanism. Figures 145 and 146 show similar results for the Pdelay mechanism with timestamping relative to the corrected clock, and Figures 155 and 156 for the Pdelay mechanism with timestamping relative to the uncorrected clock. Since the actual propagation delay is  $10^5$  ns (i.e., 0.1 ms, see Table 34), while the measured propagation delay is initialized to 0 (i.e., it is assumed to be 0 if it is needed in a computation before the first measurement is made), there is a large initial transient in all three cases. However, in all three cases the transient is over in 10 s. The error in measured propagation delay, i.e., the difference between the measured delay and actual delay, is comparable for the Pdelay cases and slightly larger for the Delay Request/Response case.

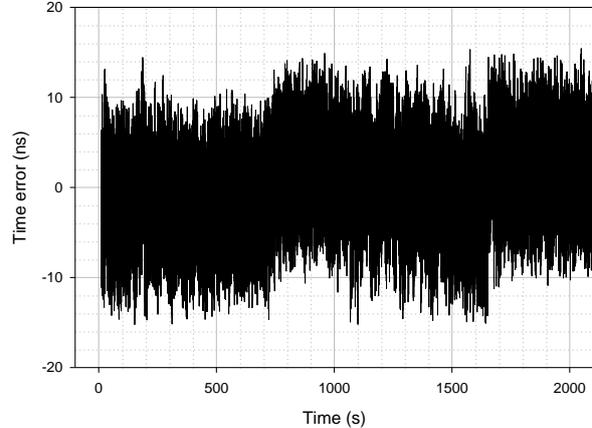
Figures 138 and 139 show filtered and unfiltered time error results for the Delay Request/Response mechanism for node 22 (i.e., T-TSC that follows the final T-BC). Analogous results for the Pdelay mechanism with timestamping relative to the corrected clock are shown in Figures 147 to 149, and for the Pdelay mechanism with timestamping relative to the uncorrected clock in Figures 157 to 159. The results are qualitatively similar to the results for node 2. The corresponding Pdelay results are very similar, and the corresponding Delay Request/Response results are somewhat worse. There is an initial transient that is over in approximately 500 s in all the cases.

The results for mean propagation delay for the link upstream of node 22 are shown in Figures 140 and 141 for the Delay Request/Response mechanism, Figures 150 and 151 for the Pdelay mechanism with timestamping relative to the corrected clock, and Figures 160 and 161 for the Pdelay mechanism with timestamping relative to the uncorrected clock. The results are similar to the results for the link upstream of node 2. Steady-state is reached within the first 10 s. The error in the measured propagation delay is similar for the two Pdelay mechanism cases, and somewhat worse in the Delay Request/Response case.

Node 2 (1st BC after GM)  
 Delay\_Req/Resp mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity

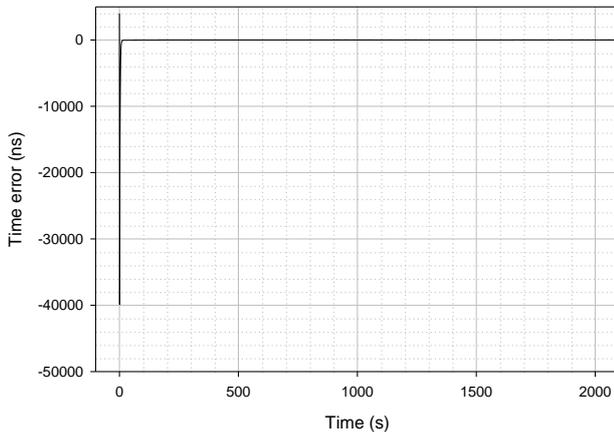


Node 2 (1st BC after GM)  
 Delay\_Req/Resp mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 Initial transient removed

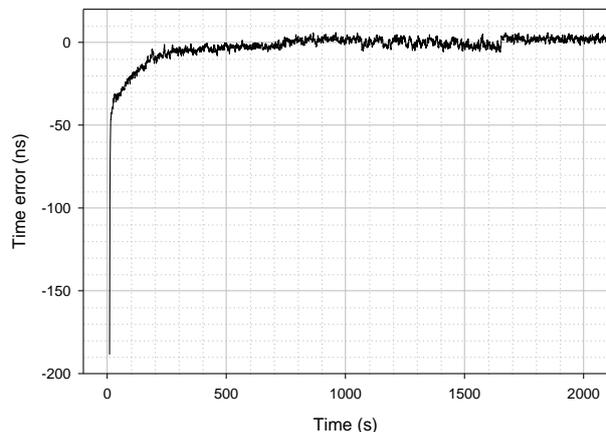


**Figure 132 – Time error for case 1, node 2, Delay Request/Response mechanism, no T-BC or endpoint filtering**

Node 2 (1st BC after GM)  
 Delay\_Req/Resp mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 0.1 Hz endpoint filtering (no SyncE noise generation)  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity

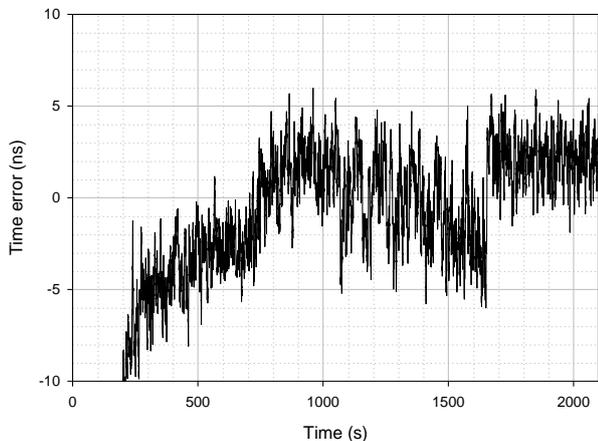


Node 2 (1st BC after GM)  
 Delay\_Req/Resp mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 0.1 Hz endpoint filtering (no SyncE noise generation)  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 first 10 s removed

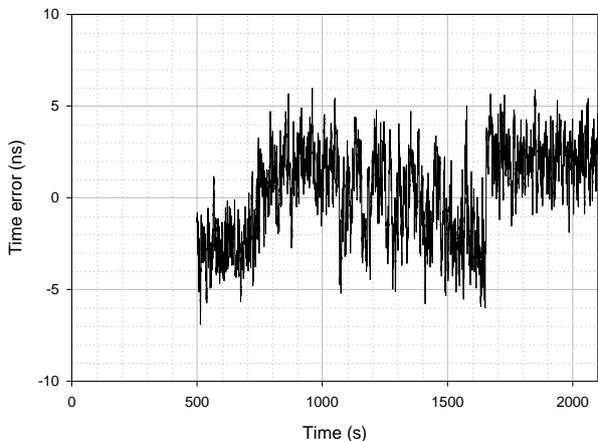


**Figure 133 – Time error for case 1, node 2, Delay Request/Response mechanism, no T-BC filtering, 0.1 Hz endpoint filtering**

Node 2 (1st BC after GM)  
 Delay\_Req/Resp mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 0.1 Hz endpoint filtering (no SyncE noise generation)  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 first 200 s removed

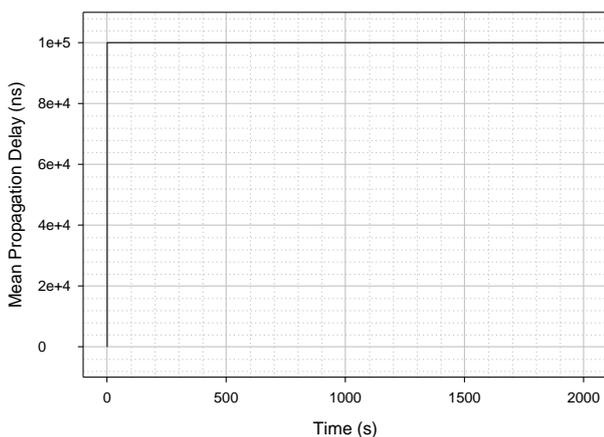


Node 2 (1st BC after GM)  
 Delay\_Req/Resp mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 0.1 Hz endpoint filtering (no SyncE noise generation)  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 first 500 s removed

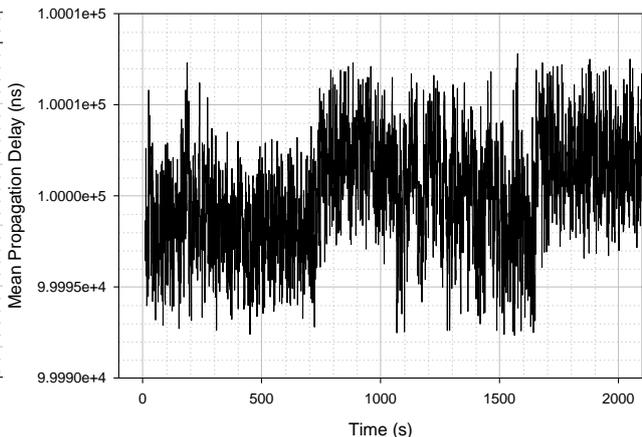


**Figure 134 – Time error for case 1, node 2, Delay Request/Response mechanism, no T-BC filtering, 0.1 Hz endpoint filtering. Details of 300 – 2100 s and 500 – 2100 s**

Node 2 (1st BC after GM)  
 Upstream Mean Propagation Delay  
 Delay\_Req/Resp mechanism (timestamp relative to corrected c  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity

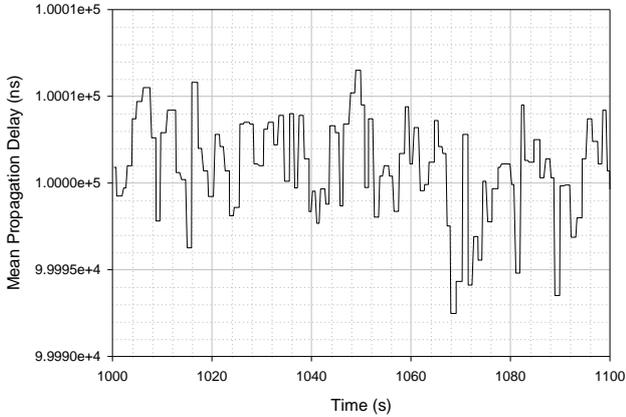


Node 2 (1st BC after GM)  
 Upstream Mean Propagation Delay  
 Delay\_Req/Resp mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 first 10 s removed

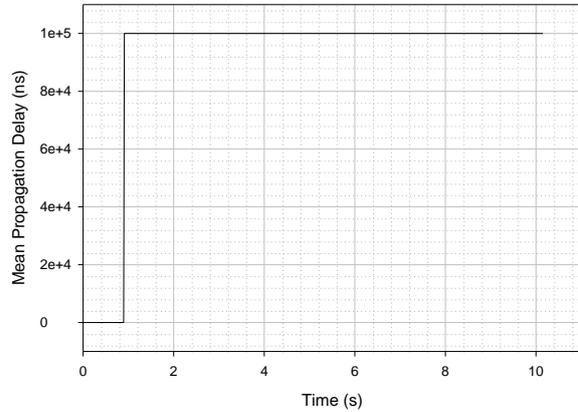


**Figure 135 – Mean propagation delay for case 1, node 2, Delay Request/Response mechanism, no T-BC or endpoint filtering, 0.1 Hz endpoint filtering**

Node 2 (1st BC after GM)  
 Upstream Mean Propagation Delay  
 Delay\_Req/Resp mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 detail of 1000 - 1100 s

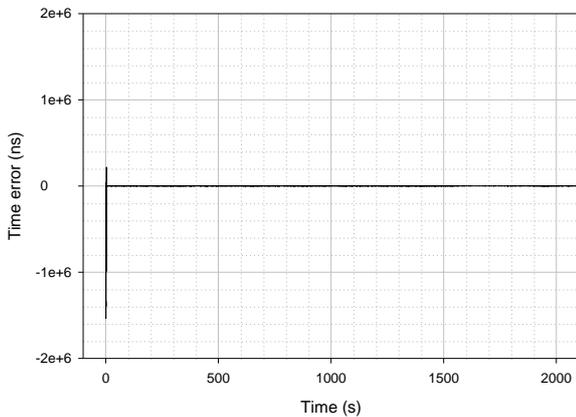


Node 2 (1st BC after GM)  
 Upstream Mean Propagation Delay  
 Delay\_Req/Resp mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 detail of initial transient (0 -10 s)

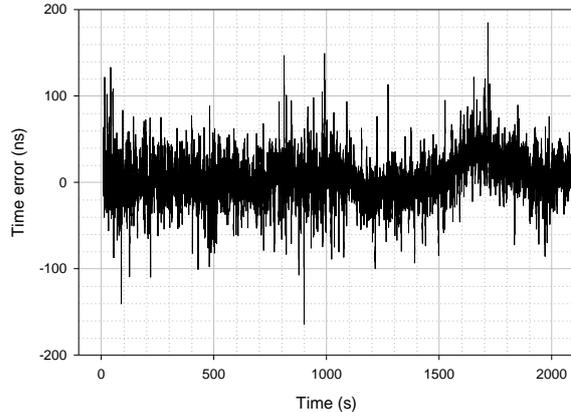


**Figure 136 – Mean propagation delay for case 1, node 2, Delay Request/Response mechanism, no T-BC or endpoint filtering, 0.1 Hz endpoint filtering. Detail of 1000 – 1100 s and 0 – 10 s**

Node 22 (OC that follows final BC)  
 Delay\_Req/Resp mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity

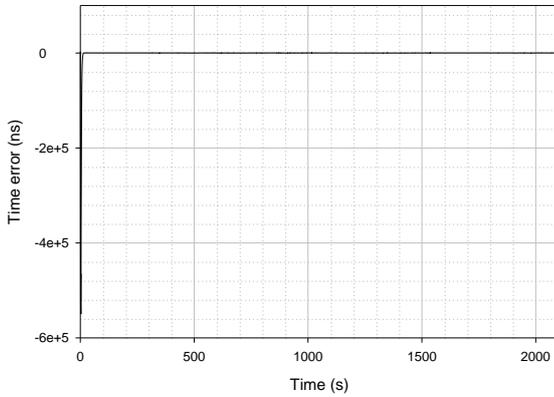


Node 22 (OC that follows final BC)  
 Delay\_Req/Resp mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 Initial transient removed

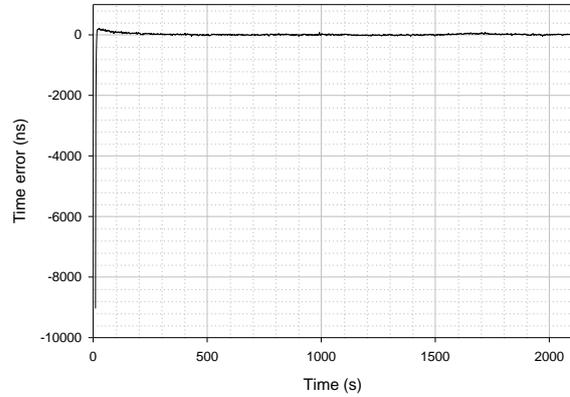


**Figure 137 – Time error for case 1, node 22, Delay Request/Response mechanism, no T-BC or endpoint filtering**

Node 22 (OC that follows final BC)  
 Delay\_Req/Resp mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 0.1 Hz endpoint filtering (no SyncE noise generation)  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity

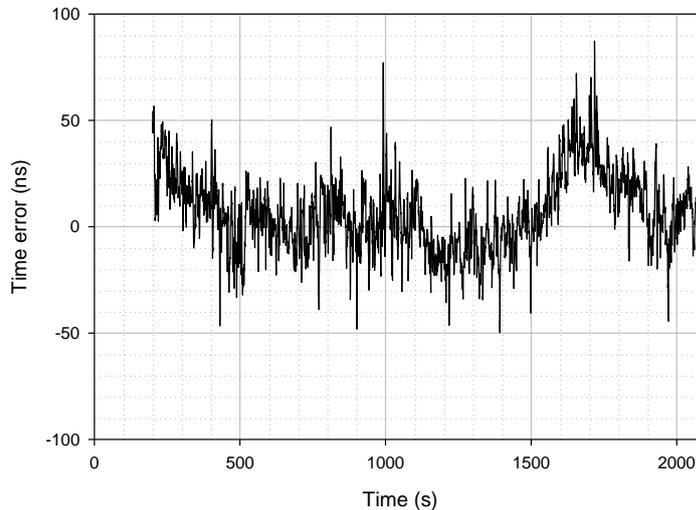


Node 22 (OC that follows final BC)  
 Delay\_Req/Resp mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 0.1 Hz endpoint filtering (no SyncE noise generation)  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 first 10 s removed



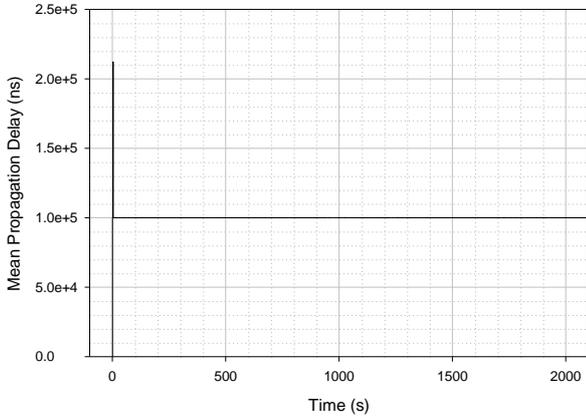
**Figure 138 – Time error for case 1, node 22, Delay Request/Response mechanism, no T-BC filtering, 0.1 Hz endpoint filtering**

Node 22 (OC that follows final BC)  
 Delay\_Req/Resp mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 0.1 Hz endpoint filtering (no SyncE noise generation)  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 first 200 s removed

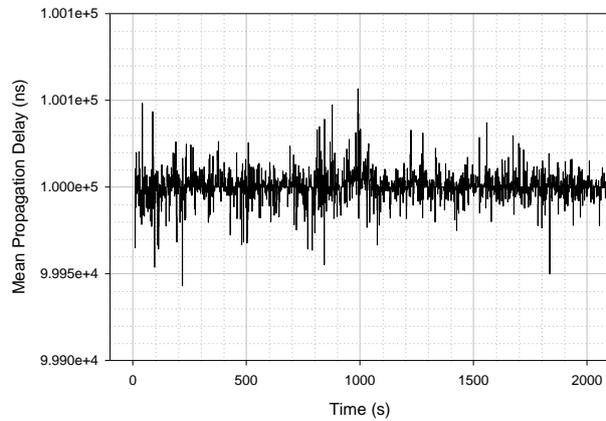


**Figure 139 – Time error for case 1, node 2, Delay Request/Response mechanism, no T-BC filtering, 0.1 Hz endpoint filtering. Details of 200 – 2100 s**

Node 22 (OC that follows final BC)  
 Upstream Mean Propagation Delay  
 Delay\_Req/Resp mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity

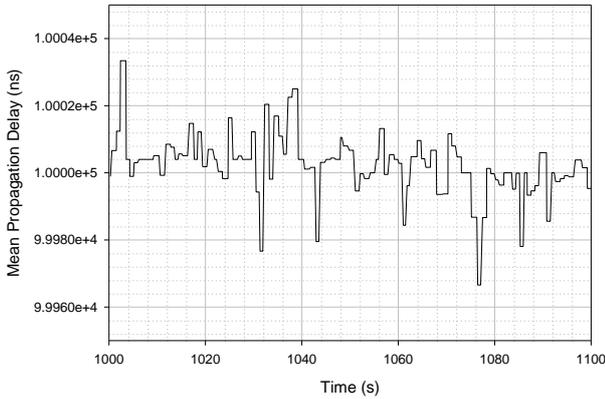


Node 22 (OC that follows final BC)  
 Upstream Mean Propagation Delay  
 Delay\_Req/Resp mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 first 10 s removed

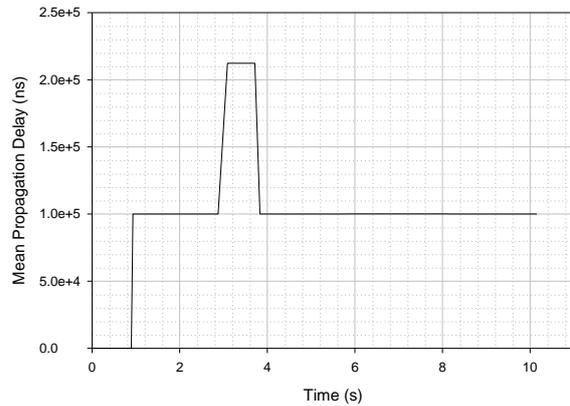


**Figure 140 – Mean propagation delay for case 1, node 22, Delay Request/Response mechanism, no T-BC or endpoint filtering**

Node 22 (OC that follows final BC)  
 Upstream Mean Propagation Delay  
 Delay\_Req/Resp mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 detail of 1000 - 1100 s

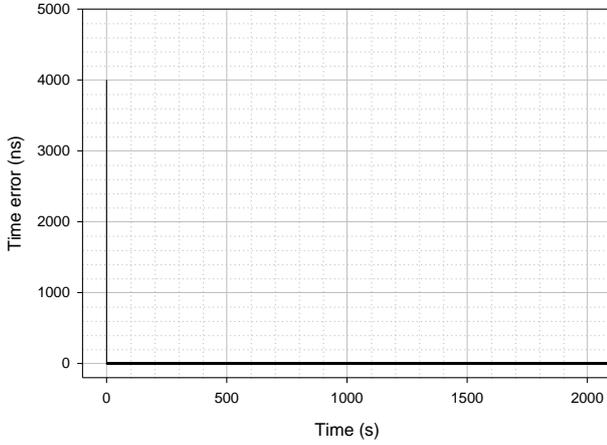


Node 22 (OC that follows final BC)  
 Upstream Mean Propagation Delay  
 Delay\_Req/Resp mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 detail of initial transient (0 - 10 s)

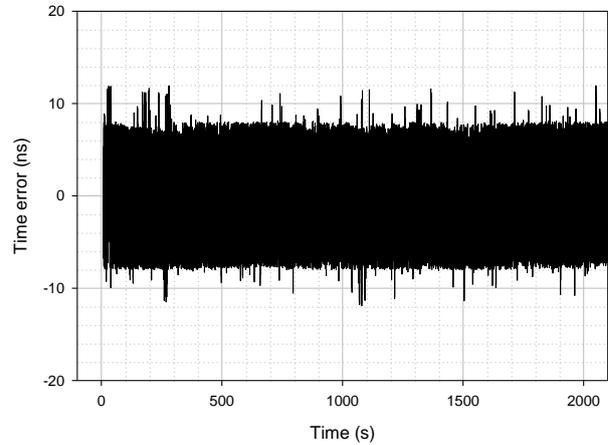


**Figure 141 – Mean propagation delay for case 1, node 2, Delay Request/Response mechanism, no T-BC or endpoint filtering. Detail of 1000 – 1100 s and 0 – 10 s**

Node 2 (1st BC after GM)  
 Pdelay mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity

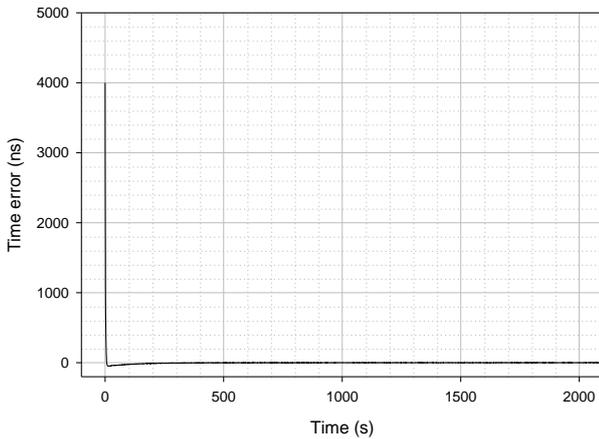


Node 2 (1st BC after GM)  
 Pdelay mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 Initial transient removed

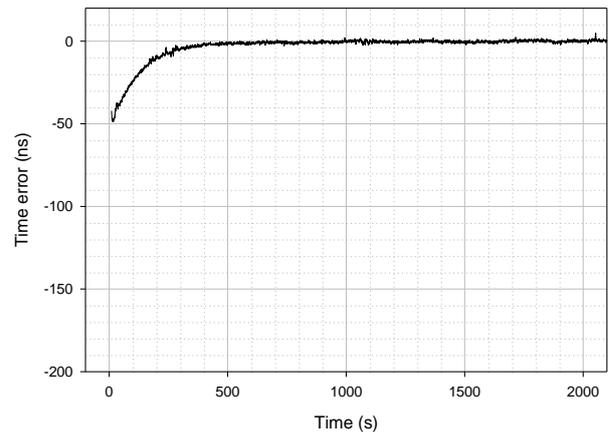


**Figure 142 – Time error for case 1, node 2, Pdelay mechanism with timestamping with respect to corrected clock, no T-BC or endpoint filtering**

Node 2 (1st BC after GM)  
 Pdelay mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 0.1 Hz endpoint filtering (no SyncE noise generation)  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity

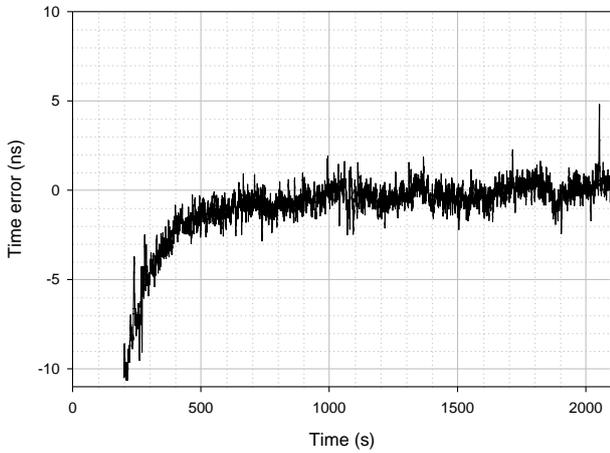


Node 2 (1st BC after GM)  
 Pdelay mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 0.1 Hz endpoint filtering (no SyncE noise generation)  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 first 10 s removed

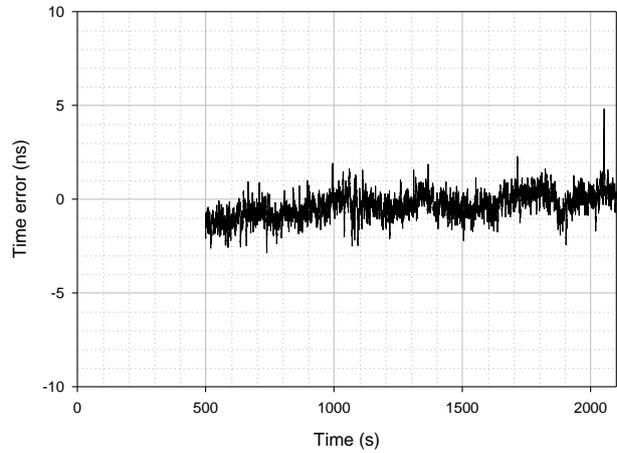


**Figure 143 – Time error for case 1, node 2, Pdelay mechanism with timestamping with respect to corrected clock, no T-BC filtering and 0.1 Hz endpoint filtering**

Node 2 (1st BC after GM)  
 Pdelay mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 0.1 Hz endpoint filtering (no SyncE noise generation)  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 first 200 s removed

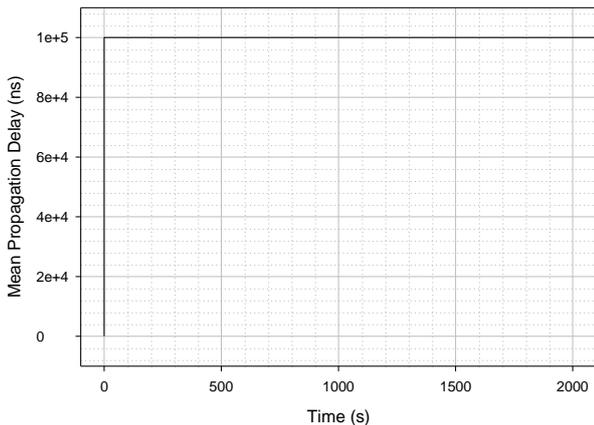


Node 2 (1st BC after GM)  
 Pdelay mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 0.1 Hz endpoint filtering (no SyncE noise generation)  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 first 500 s removed

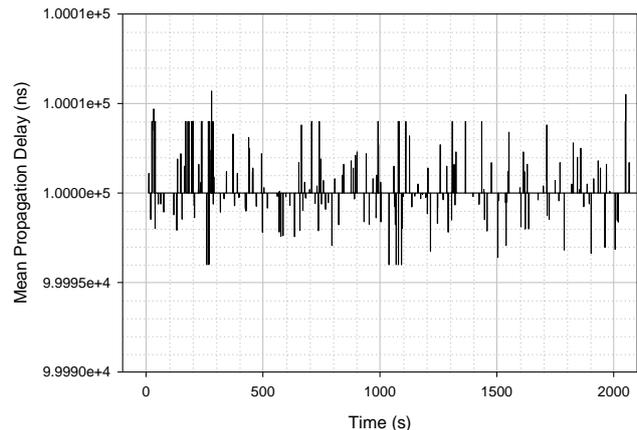


**Figure 144 – Time error for case 1, node 2, Pdelay mechanism with timestamping with respect to corrected clock, no T-BC filtering and 0.1 Hz endpoint filtering. Detail of 200 – 2100 s and 500 – 2100 s**

Node 2 (1st BC after GM)  
 Upstream Mean Propagation Delay  
 Pdelay mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity

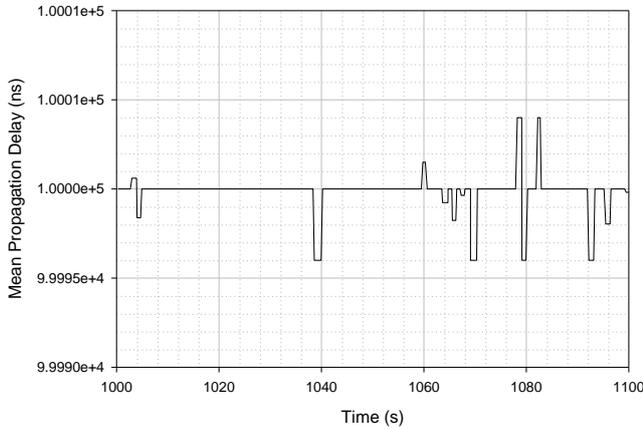


Node 2 (1st BC after GM)  
 Upstream Mean Propagation Delay  
 Pdelay mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 first 10 s removed

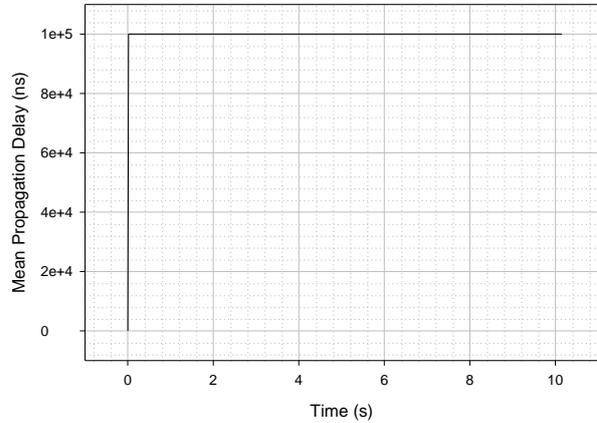


**Figure 145 – Mean propagation delay for case 1, node 2, Pdelay mechanism with timestamping with respect to corrected clock, no T-BC or endpoint filtering**

Node 2 (1st BC after GM)  
 Upstream Mean Propagation Delay  
 Pdelay mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 detail of 1000 - 1100 s

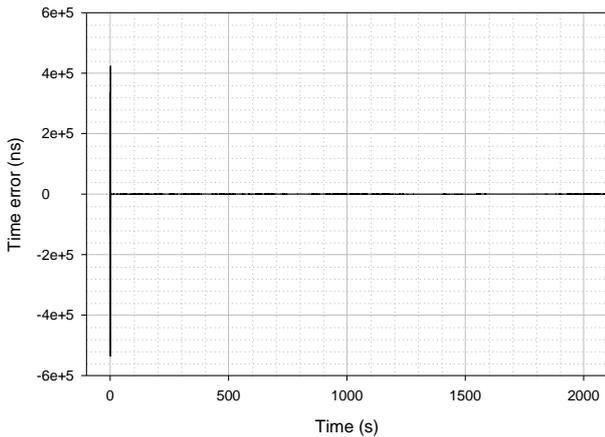


Node 2 (1st BC after GM)  
 Upstream Mean Propagation Delay  
 Pdelay mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 detail of initial transient (0 - 10 s)

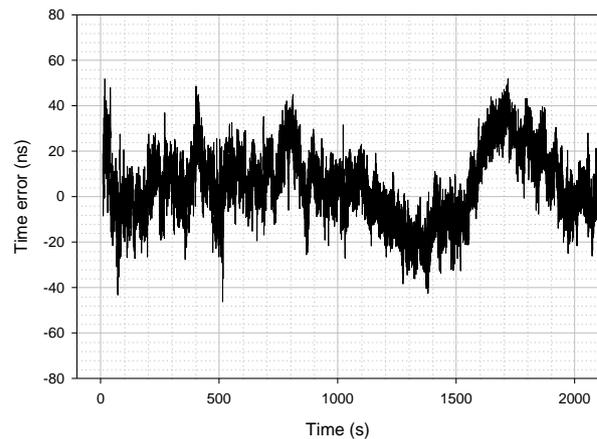


**Figure 146 – Mean propagation delay for case 1, node 2, Pdelay mechanism with timestamping with respect to corrected clock, no T-BC or endpoint filtering. Detail of 1000 – 1100 s and 0 – 10 s**

Node 22 (OC that follows final BC)  
 Pdelay mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity



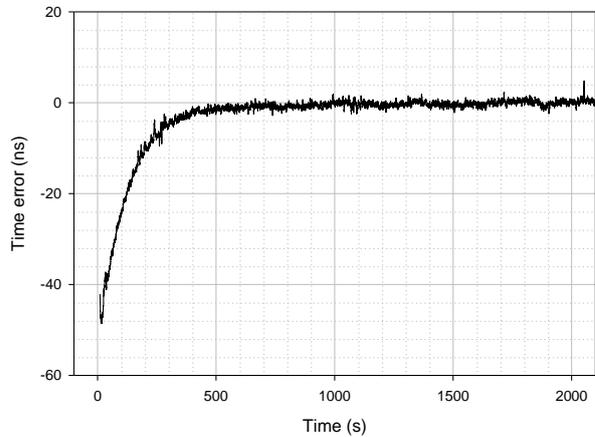
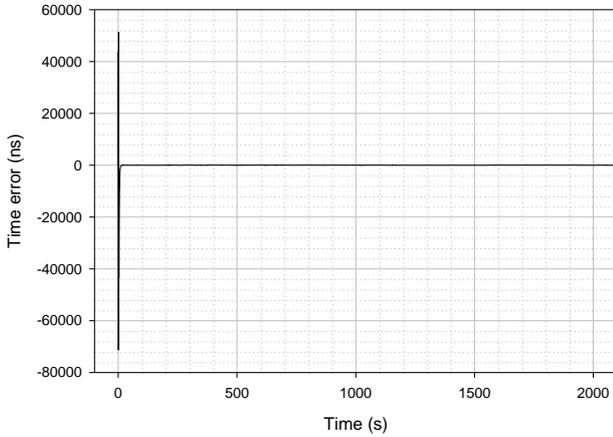
Node 22 (OC that follows final BC)  
 Pdelay mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 Initial transient removed



**Figure 147 – Time error for case 1, node 22, Pdelay mechanism with timestamping with respect to corrected clock, no T-BC or endpoint filtering**

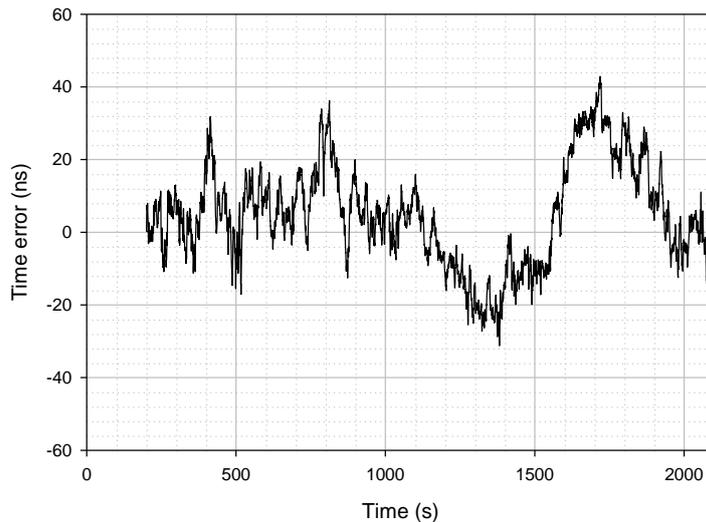
Node 22 (OC that follows final BC)  
 Pdelay mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 0.1 Hz endpoint filtering (no SyncE noise generation)  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity

Node 22 (OC that follows final BC)  
 Pdelay mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 0.1 Hz endpoint filtering (no SyncE noise generation)  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 first 10 s removed



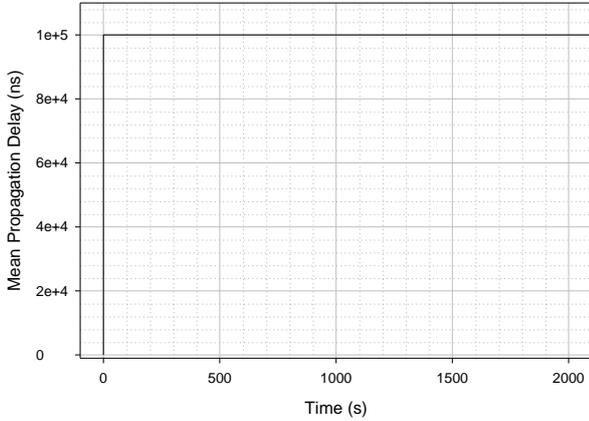
**Figure 148 – Time error for case 1, node 22, Pdelay mechanism with timestamping with respect to corrected clock, no T-BC filtering and 0.1 Hz endpoint filtering**

Node 22 (OC that follows final BC)  
 Pdelay mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 0.1 Hz endpoint filtering (no SyncE noise generation)  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 first 200 s removed

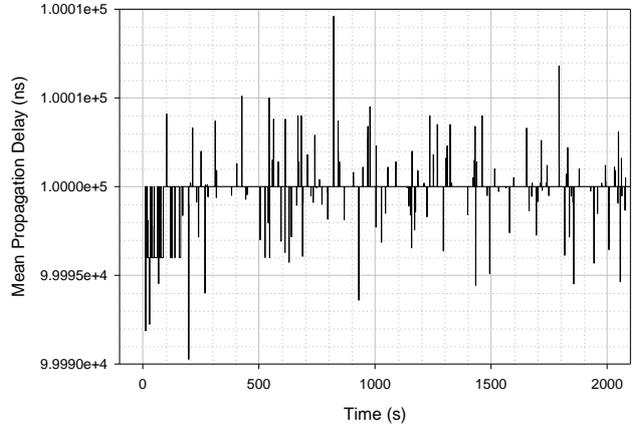


**Figure 149 – Time error for case 1, node 22, Pdelay mechanism with timestamping with respect to corrected clock, no T-BC filtering, 0.1 Hz endpoint filtering. Details of 200 – 2100 s**

Node 22 (OC that follows final BC)  
 Upstream Mean Propagation Delay  
 Pdelay mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity

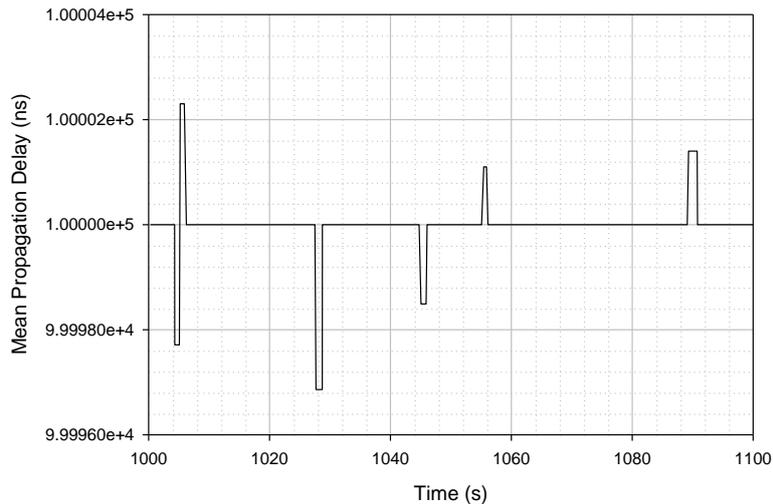


Node 22 (OC that follows final BC)  
 Upstream Mean Propagation Delay  
 Pdelay mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 first 10 s removed



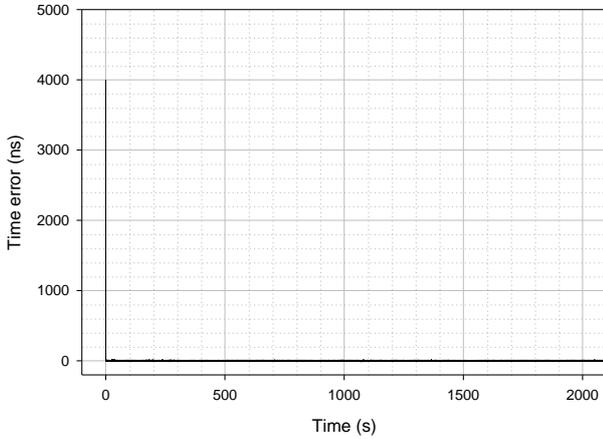
**Figure 150 – Mean propagation delay for case 1, node 22, Pdelay mechanism with timestamping with respect to corrected clock, no T-BC or endpoint filtering**

Node 22 (OC that follows final BC)  
 Upstream Mean Propagation Delay  
 Pdelay mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 detail of 1000 - 1100 s

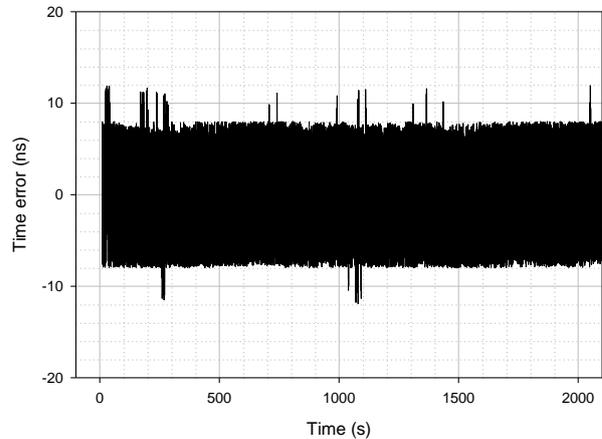


**Figure 151 – Mean propagation delay for case 1, node 22, Pdelay mechanism with timestamping with respect to corrected clock, no T-BC or endpoint filtering, detail of 1000 – 1100 s**

Node 2 (1st BC after GM)  
 Pdelay mechanism (timestamp relative to uncorrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity

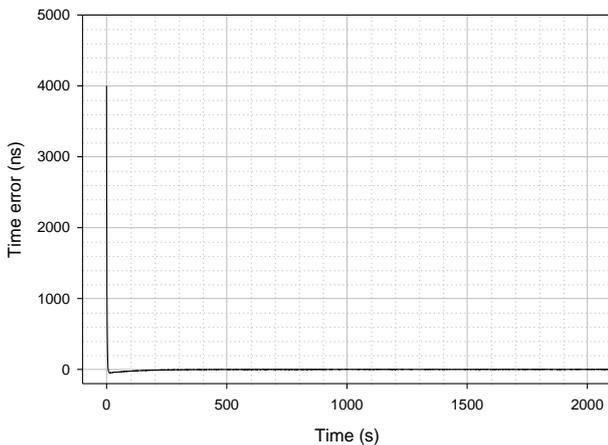


Node 2 (1st BC after GM)  
 Pdelay mechanism (timestamp relative to uncorrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 Initial transient removed

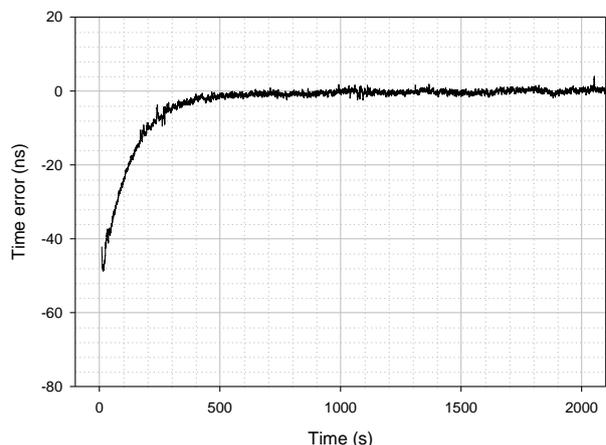


**Figure 152 – Time error for case 1, node 2, Pdelay mechanism with timestamping with respect to uncorrected clock, no T-BC or endpoint filtering**

Node 2 (1st BC after GM)  
 Pdelay mechanism (timestamp relative to uncorrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 0.1 Hz endpoint filtering (no SyncE noise generation)  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity

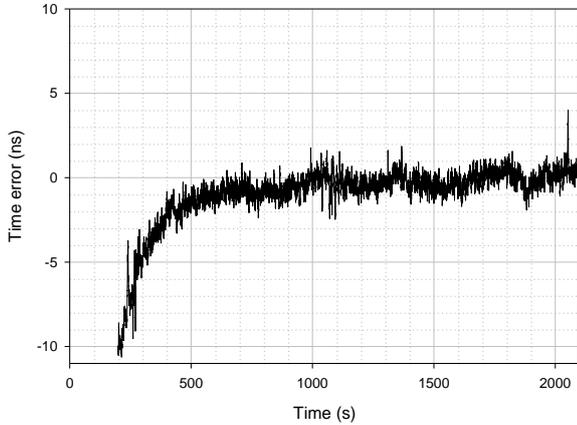


Node 2 (1st BC after GM)  
 Pdelay mechanism (timestamp relative to uncorrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 0.1 Hz endpoint filtering (no SyncE noise generation)  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 first 10 s removed

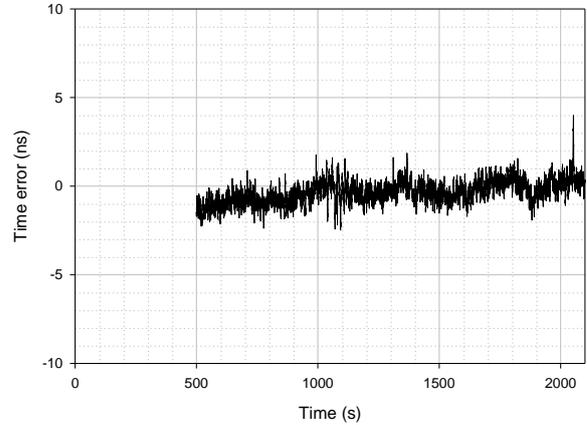


**Figure 153 – Time error for case 1, node 2, Pdelay mechanism with timestamping with respect to uncorrected clock, no T-BC filtering, 0.1 Hz endpoint filtering**

Node 2 (1st BC after GM)  
 Pdelay mechanism (timestamp relative to uncorrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 0.1 Hz endpoint filtering (no SyncE noise generation)  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 first 200 s removed

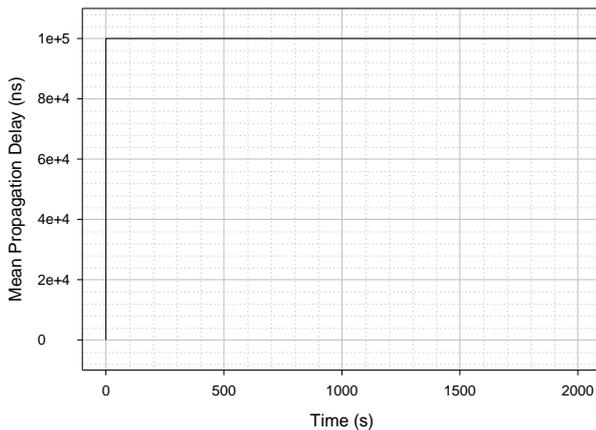


Node 2 (1st BC after GM)  
 Pdelay mechanism (timestamp relative to uncorrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 0.1 Hz endpoint filtering (no SyncE noise generation)  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 first 500 s removed

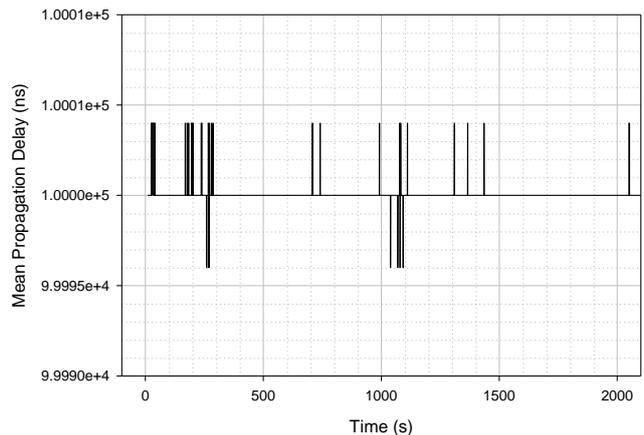


**Figure 154 – Time error for case 1, node 2, Pdelay mechanism with timestamping with respect to uncorrected clock, no T-BC filtering, 0.1 Hz endpoint filtering. Detail of 200 – 2100 s and 500 – 2100 s**

Node 2 (1st BC after GM)  
 Upstream Mean Propagation Delay  
 Pdelay mechanism (timestamp relative to uncorrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity

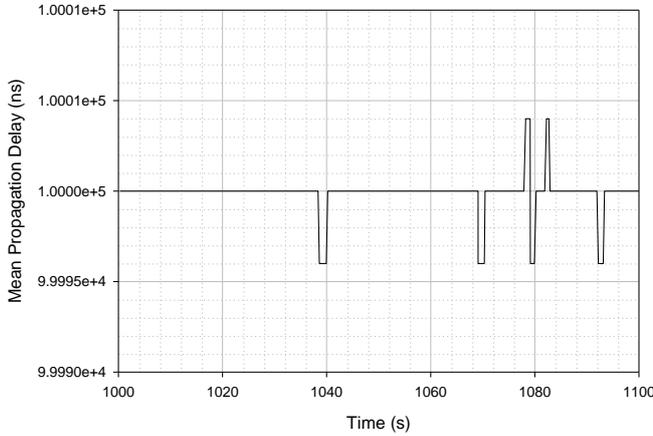


Node 2 (1st BC after GM)  
 Upstream Mean Propagation Delay  
 Pdelay mechanism (timestamp relative to uncorrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 first 10 s removed

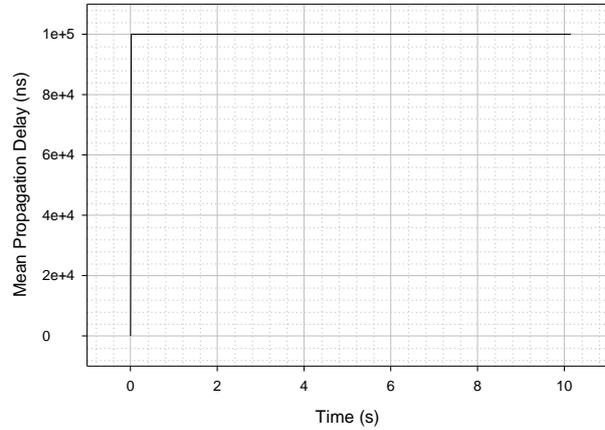


**Figure 155 – Mean propagation delay for case 1, node 2, Pdelay mechanism with timestamping with respect to uncorrected clock, no T-BC or endpoint filtering**

Node 2 (1st BC after GM)  
 Upstream Mean Propagation Delay  
 Pdelay mechanism (timestamp relative to uncorrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 detail of 1000 - 1100 s

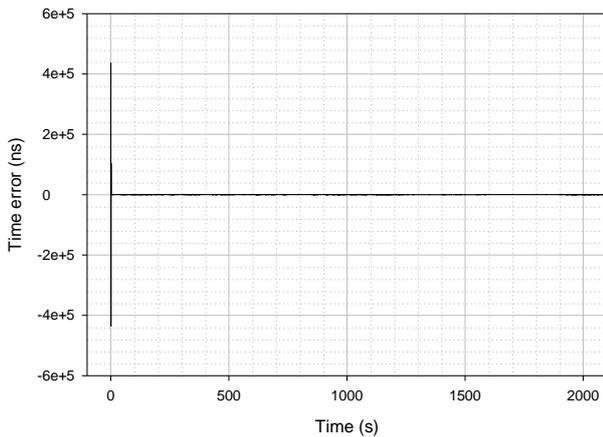


Node 2 (1st BC after GM)  
 Upstream Mean Propagation Delay  
 Pdelay mechanism (timestamp relative to uncorrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 detail of initial transient (0 -10 s)

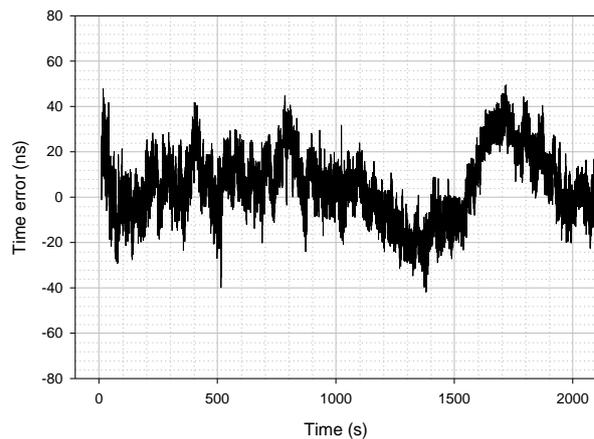


**Figure 156 – Mean propagation delay for case 1, node 2, Pdelay mechanism with timestamping with respect to uncorrected clock, no T-BC or endpoint filtering. Detail of 1000 – 1100 s and 0 – 10 s**

Node 22 (OC that follows final BC)  
 Pdelay mechanism (timestamp relative to uncorrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity

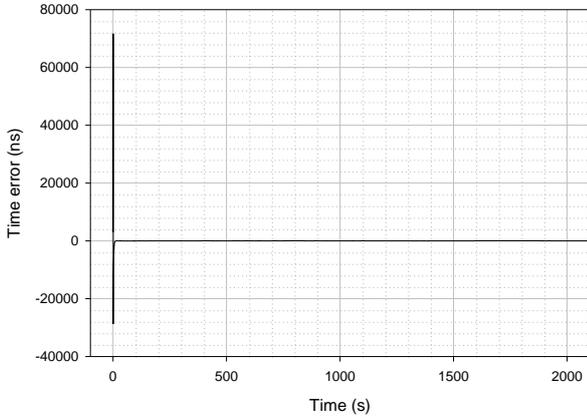


Node 22 (OC that follows final BC)  
 Pdelay mechanism (timestamp relative to uncorrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 Initial transient removed

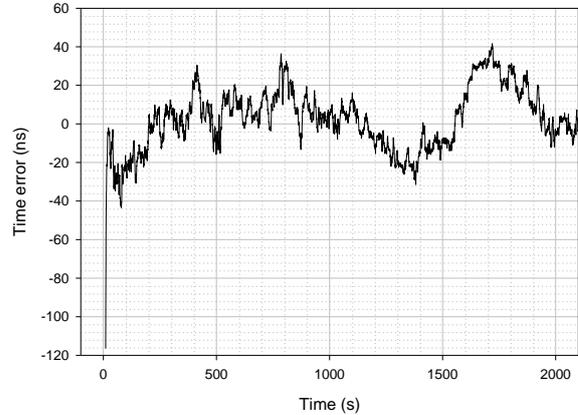


**Figure 157 – Time error for case 1, node 22, Pdelay mechanism with timestamping with respect to uncorrected clock, no T-BC or endpoint filtering**

Node 22 (OC that follows final BC)  
 Pdelay mechanism (timestamp relative to uncorrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 0.1 Hz endpoint filtering (no SyncE noise generation)  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity

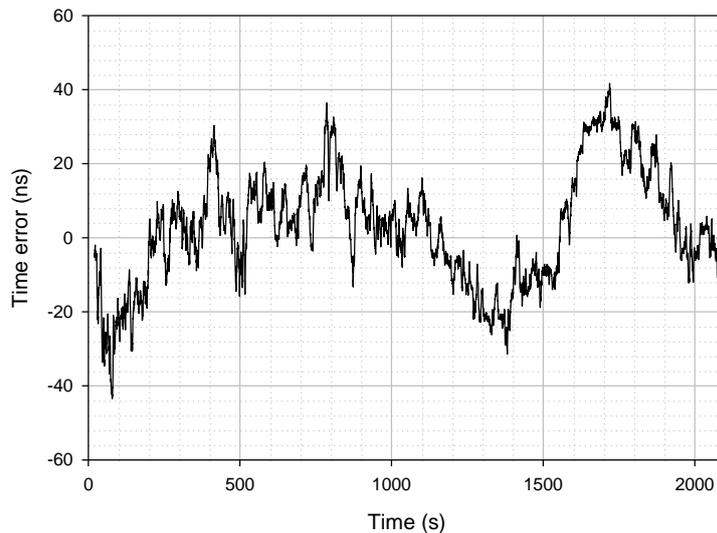


Node 22 (OC that follows final BC)  
 Pdelay mechanism (timestamp relative to uncorrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 0.1 Hz endpoint filtering (no SyncE noise generation)  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 first 10 s removed



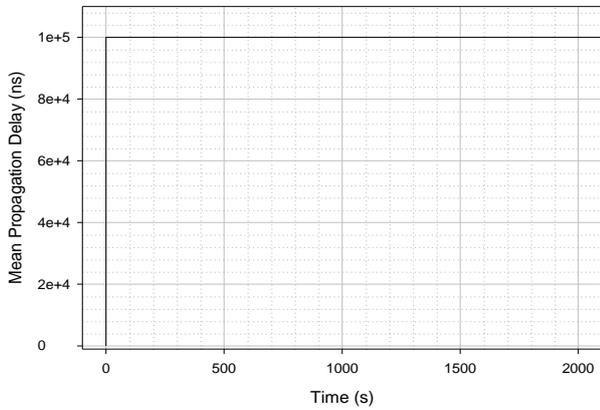
**Figure 158 – Time error for case 1, node 22, Pdelay mechanism with timestamping with respect to uncorrected clock, no T-BC filtering and 0.1 Hz endpoint filtering**

Node 22 (OC that follows final BC)  
 Pdelay mechanism (timestamp relative to uncorrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 0.1 Hz endpoint filtering (no SyncE noise generation)  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 first 20 s removed

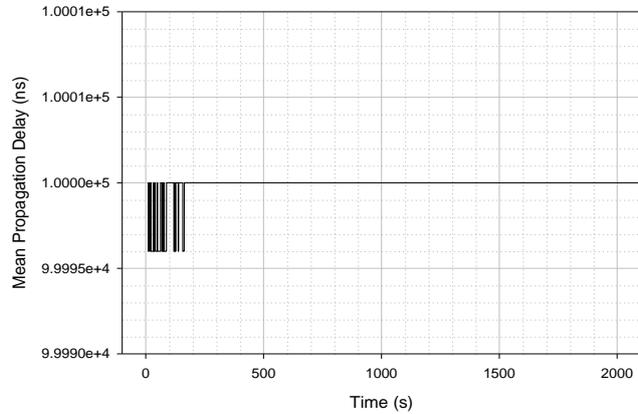


**Figure 159 – Time error for case 1, node 22, Pdelay mechanism with timestamping with respect to uncorrected clock, no T-BC filtering, 0.1 Hz endpoint filtering. Details of 200 – 2100 s**

Node 22 (OC that follows final BC)  
 Upstream Mean Propagation Delay  
 Pdelay mechanism (timestamp relative to uncorrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity

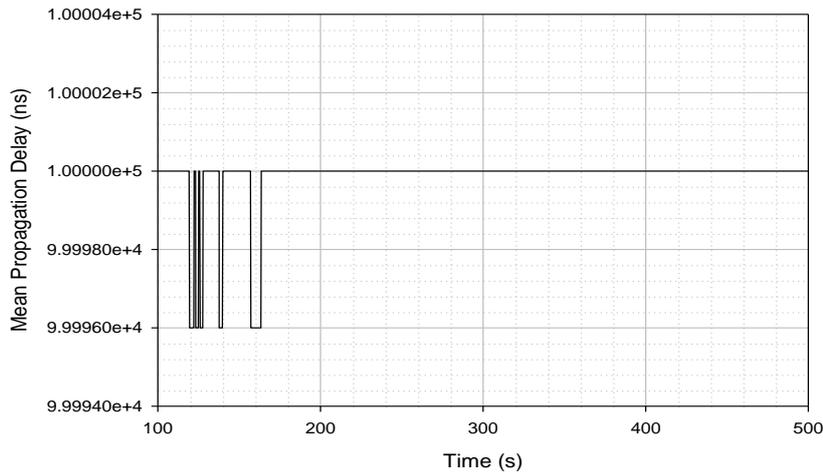


Node 22 (OC that follows final BC)  
 Upstream Mean Propagation Delay  
 Pdelay mechanism (timestamp relative to uncorrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 first 10 s removed



**Figure 160 – Mean propagation delay for case 1, node 22, Pdelay mechanism with timestamping with respect to uncorrected clock, no T-BC or endpoint filtering**

Node 22 (OC that follows final BC)  
 Upstream Mean Propagation Delay  
 Pdelay mechanism (timestamp relative to uncorrected clock)  
 No SyncE noise  
 Initial nonzero phase offset at each node  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 detail of 100 - 500 s



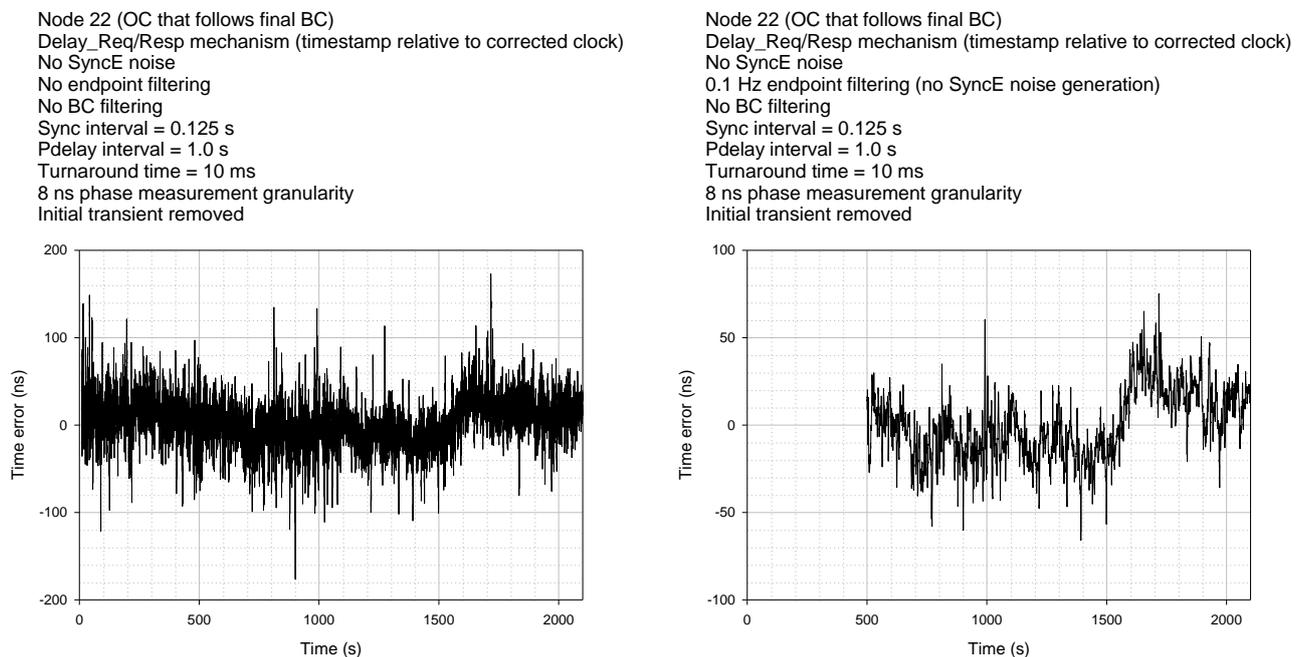
**Figure 161. Mean propagation delay for case 1, node 22, Pdelay mechanism with timestamping with respect to uncorrected clock, no T-BC or endpoint filtering, detail of 1000 – 1100 s**

### 12.3.3.1.2 Results for case 2

Recall from Table 33 that in case 2 sub-cases both with and without T-BC filtering were looked at. Initial time offset at each node 0.

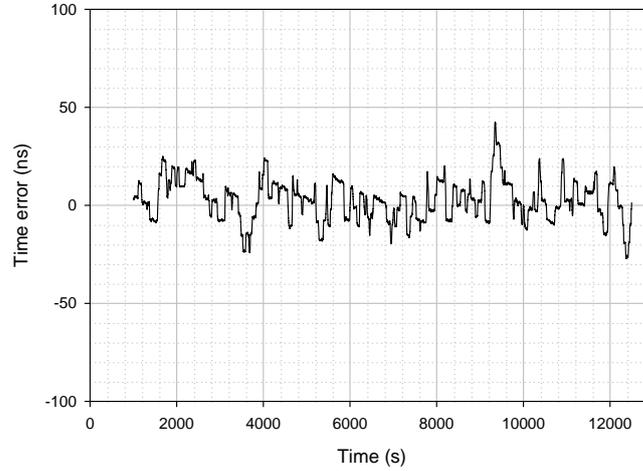
Figure 162 and Figure 163 show the results for time error for node 22 (T-TSC that follows final T-BC), for the Delay Request/Response mechanism. The three plots show results for no filtering, only 0.1 Hz endpoint filtering, and 0.1 Hz T-BC and endpoint filtering. Analogous results are shown in Figures 164 and 165 for the Pdelay mechanism with timestamping relative to the corrected clock, and in Figures 166 and 167 for the Pdelay mechanism with timestamping relative to the uncorrected clock.

As in the sub-cases of case 1, the corresponding cases for the Pdelay mechanism with timestamping relative to the corrected or uncorrected clock give very similar results, i.e., Figure 164 is very similar to Figure 166, and Figure 165 is very similar to Figure 167. In addition, the results for the Delay Request/Response mechanism with both T-BC and endpoint filtering is very similar to the corresponding Pdelay cases, i.e., Figure 163 is very similar to Figure 165 and Figure 167. However, the results for the Delay Request/Response mechanism with either no filtering or only endpoint filtering is somewhat worse than the corresponding Pdelay cases, i.e., the results given in Figure 162 are somewhat worse than those given in Figures 134 and 166.



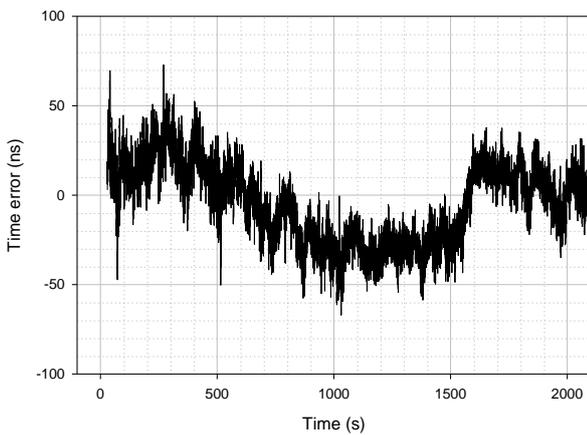
**Figure 162 – Time error for case 2, node 22, Delay Request/Response mechanism, no T-BC filtering. No endpoint filtering (left plot), 0.1 Hz endpoint filtering (right plot)**

Node 22 (OC that follows final BC)  
 Delay\_Req/Resp mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 0.1 Hz endpoint filtering (no SyncE noise generation)  
 0.1 Hz BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 Initial transient removed

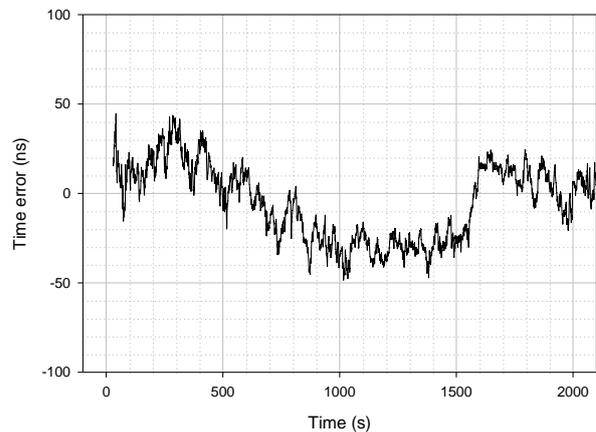


**Figure 163 – Time error for case 2, node 22, Delay Request/Response mechanism, 0.1 Hz T-BC and endpoint filtering**

Node 22 (OC that follows final BC)  
 Pdelay mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 Initial transient removed

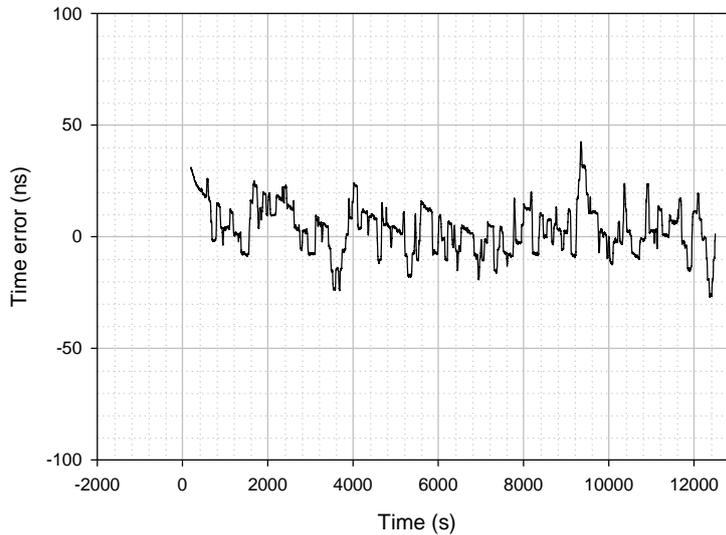


Node 22 (OC that follows final BC)  
 Pdelay mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 0.1 Hz endpoint filtering (no SyncE noise generation)  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 Initial transient removed



**Figure 164 – Time error for case 2, node 22, Pdelay mechanism with timestamping with respect to corrected clock, no T-BC filtering. No endpoint filtering (left plot), 0.1 Hz endpoint filtering (right plot)**

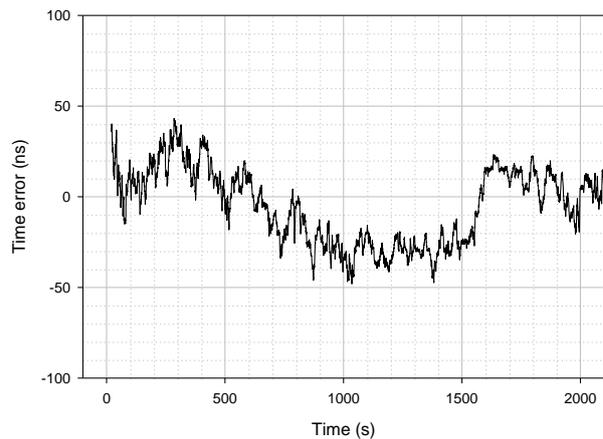
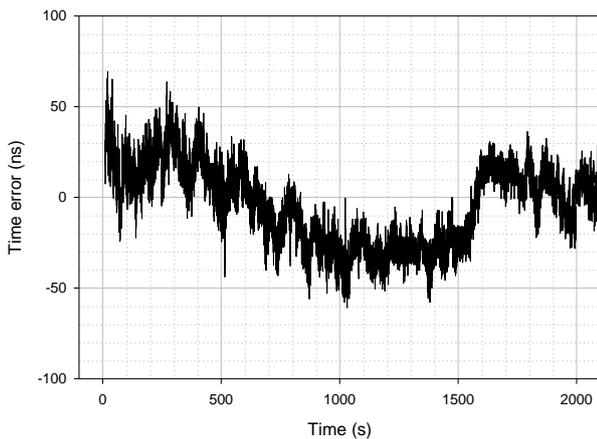
Node 22 (OC that follows final BC)  
 Pdelay mechanism (timestamp relative to corrected clock)  
 No SyncE noise  
 0.1 Hz endpoint filtering (no SyncE noise generation)  
 0.1 Hz BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 Initial transient removed



**Figure 165 – Time error for case 2, node 22, Pdelay mechanism with timestamping with respect to corrected clock, no T-BC filtering. 0.1 Hz T-BC and endpoint filtering**

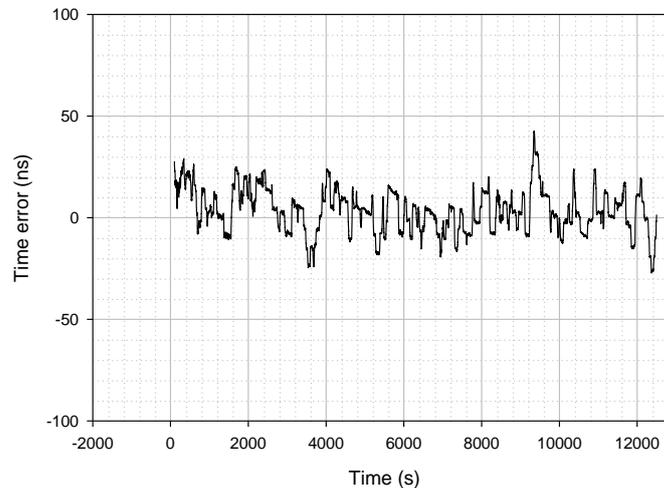
Node 22 (OC that follows final BC)  
 Pdelay mechanism (timestamp relative to uncorrected clock)  
 No SyncE noise  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 Initial transient removed

Node 22 (OC that follows final BC)  
 Pdelay mechanism (timestamp relative to uncorrected clock)  
 No SyncE noise  
 0.1 Hz endpoint filtering (no SyncE noise generation)  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 Initial transient removed



**Figure 166 – Time error for case 2, node 22, Pdelay mechanism with timestamping with respect to uncorrected clock, no T-BC filtering. No endpoint filtering (left plot), 0.1 Hz endpoint filtering (right plot)**

Node 22 (OC that follows final BC)  
 Pdelay mechanism (timestamp relative to uncorrected clock)  
 No SyncE noise  
 0.1 Hz endpoint filtering (no SyncE noise generation)  
 0.1 Hz BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 Initial transient removed



**Figure 167 – Time error for case 2, node 22, Pdelay mechanism with timestamping with respect to uncorrected clock, no T-BC filtering. 0.1 Hz T-BC and endpoint filtering**

### 12.3.3.1.3 Results for case 3

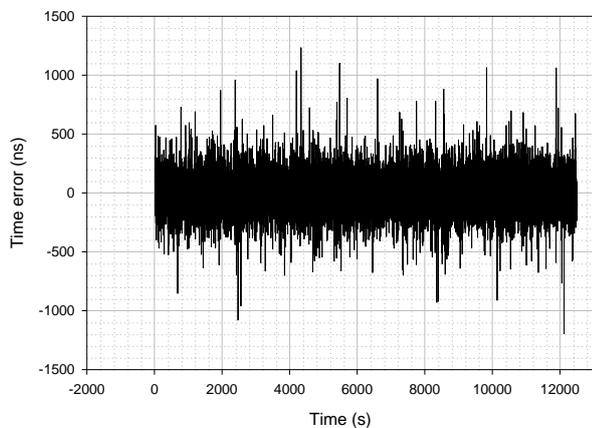
It can be recalled from Table 33 that for case 2, the sub-cases both with and without T-BC filtering were looked at. Initial time offset at each node 0. Case 3 differs from case 2 in that case 3 has SyncE phase noise while case 2 does not have SyncE phase noise.

Figure 168 and Figure 169 show the results for time error for node 22 (T-TSC that follows the final T-BC), for the Delay Request/Response mechanism. The three plots show the results for no filtering, only 0.1 Hz endpoint filtering, and 0.1 Hz T-BC and endpoint filtering. Analogous results are shown in Figures 170 and 171 for the Pdelay mechanism with timestamping relative to the corrected clock, and in Figures 172 and 173 for the Pdelay mechanism with timestamping relative to the uncorrected clock.

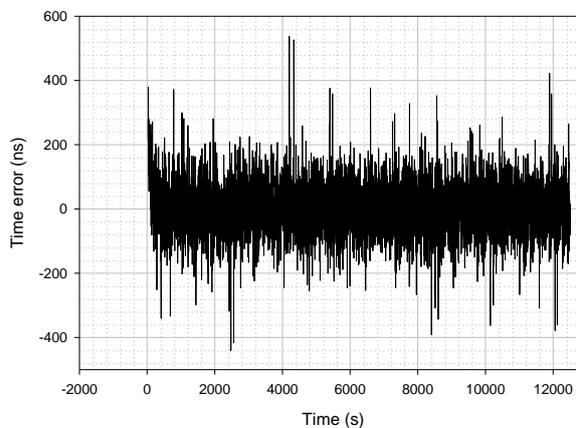
The corresponding cases for the Pdelay mechanism with timestamping relative to the corrected or uncorrected clock give similar results, though the results with no filtering are only endpoint filtering are not as similar as in case 2 (for no SyncE noise). Figure 170 is somewhat similar to Figure 172, though the results for timestamping with respect to the corrected clock (Figure 170) are somewhat worse. Nevertheless, the results with both endpoint and T-BC filtering (Figure 171 and Figure 173) are very similar.

The results for the Delay Request/Response mechanism with either no filtering or only endpoint filtering are much worse than the results for the corresponding Pdelay cases. The former show peak-to-peak time error of approximately 2400 ns with no filtering and 870 ns with only endpoint filtering (Figure 168). The corresponding results for the latter are approximately 380 ns with no filtering and 160 ns with endpoint filtering only (Figures 170 and 172). The results for the Delay Request/Response mechanism with T-BC and endpoint filtering are very similar to the corresponding results for the Pdelay cases. All the results show peak-to-peak time error of approximately 130 – 140 ns (Figures 169, 171, and 173).

Node 22 (OC that follows final BC)  
 Delay\_Req/Resp mechanism (timestamp relative to corrected clock)  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 Initial transient removed

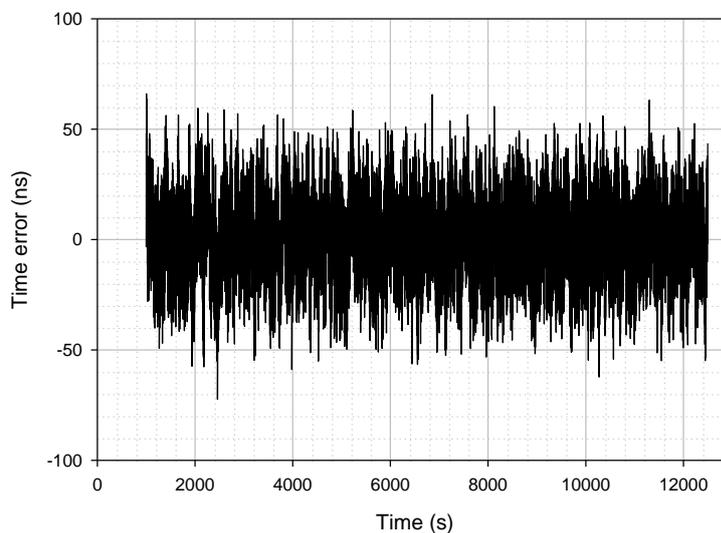


Node 22 (OC that follows final BC)  
 Delay\_Req/Resp mechanism (timestamp relative to corrected clock)  
 0.1 Hz endpoint filtering with noise generation  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 Initial transient removed



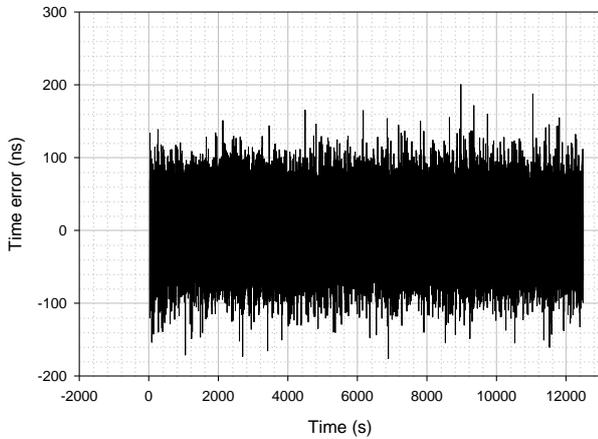
**Figure 168 – Time error for case 3, node 22, Delay Request/Response mechanism, no T-BC filtering. No endpoint filtering (left plot), 0.1 Hz endpoint filtering (right plot)**

Node 22 (OC that follows final BC)  
 Delay\_Req/Resp mechanism (timestamp relative to corrected clock)  
 0.1 Hz endpoint filtering with noise generation  
 0.1 Hz BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 Initial transient removed

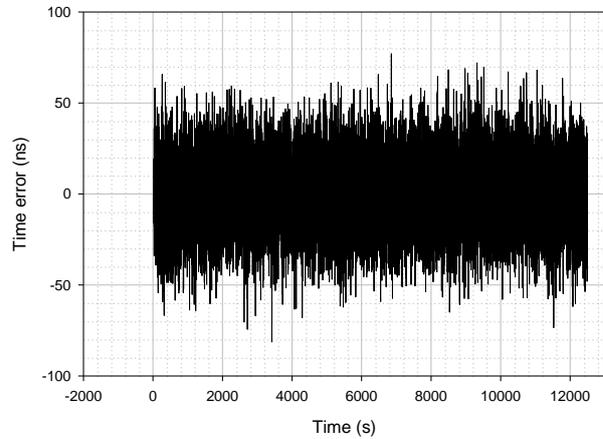


**Figure 169 – Time error for case 3, node 22, Delay Request/Response mechanism, 0.1 Hz T-BC and endpoint filtering**

Node 22 (OC that follows final BC)  
 Pdelay mechanism (timestamp relative to corrected clock)  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 Initial transient removed

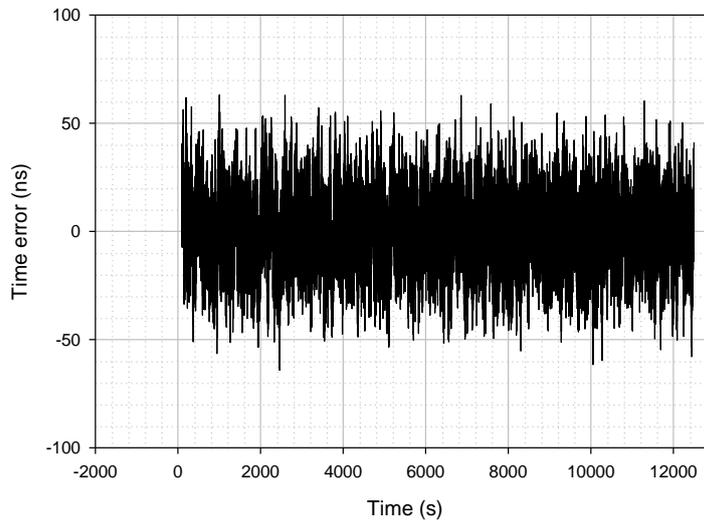


Node 22 (OC that follows final BC)  
 Pdelay mechanism (timestamp relative to corrected clock)  
 0.1 Hz endpoint filtering with noise generation  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 Initial transient removed



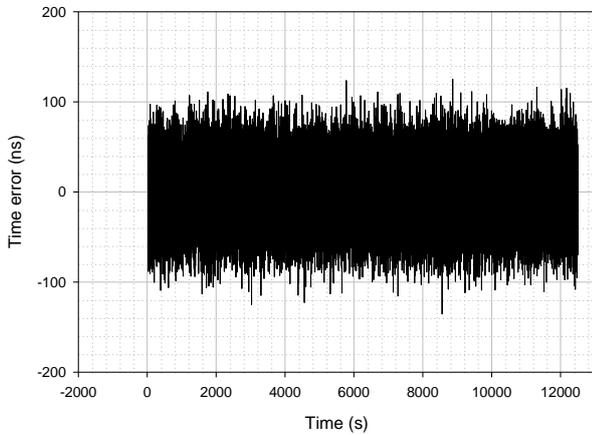
**Figure 170 – Time error for case 3, node 22, Pdelay mechanism with timestamping with respect to corrected clock, no T-BC filtering. No endpoint filtering (left plot), 0.1 Hz endpoint filtering (right plot)**

Node 22 (OC that follows final BC)  
 Pdelay mechanism (timestamp relative to corrected clock)  
 0.1 Hz endpoint filtering with noise generation  
 0.1 Hz BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 Initial transient removed

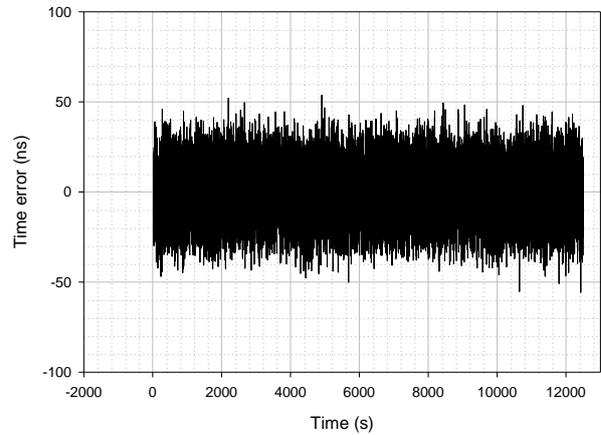


**Figure 171 – Time error for case 3, node 22, Pdelay mechanism with timestamping with respect to corrected clock, no T-BC filtering. 0.1 Hz T-BC and endpoint filtering**

Node 22 (OC that follows final BC)  
 Pdelay mechanism (timestamp relative to uncorrected clock)  
 No endpoint filtering  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 Initial transient removed

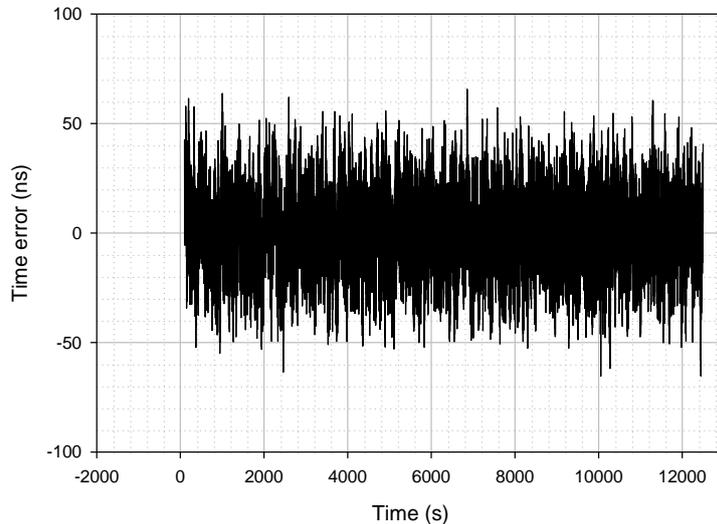


Node 22 (OC that follows final BC)  
 Pdelay mechanism (timestamp relative to uncorrected clock)  
 0.1 Hz endpoint filtering with noise generation  
 No BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 Initial transient removed



**Figure 172 – Time error for case 3, node 22, Pdelay mechanism with timestamping with respect to uncorrected clock, no T-BC filtering. No endpoint filtering (left plot), 0.1 Hz endpoint filtering (right plot)**

Node 22 (OC that follows final BC)  
 Pdelay mechanism (timestamp relative to uncorrected clock)  
 0.1 Hz endpoint filtering with noise generation  
 0.1 Hz BC filtering  
 Sync interval = 0.125 s  
 Pdelay interval = 1.0 s  
 Turnaround time = 10 ms  
 8 ns phase measurement granularity  
 Initial transient removed



**Figure 173 – Time error for case 3, node 22, Pdelay mechanism with timestamping with respect to uncorrected clock, no T-BC filtering. 0.1 Hz T-BC and endpoint filtering**

### 12.3.4 Summary and conclusions for HRM3 cases with timestamping relative to the corrected or uncorrected clock in PTP

The following conclusions may be drawn from the results in clause 12.3.3 (note that all the results are based on 0.125 s Sync interval, 1 s Pdelay interval, and 10 ms Pdelay or Delay Request/Response turnaround time):

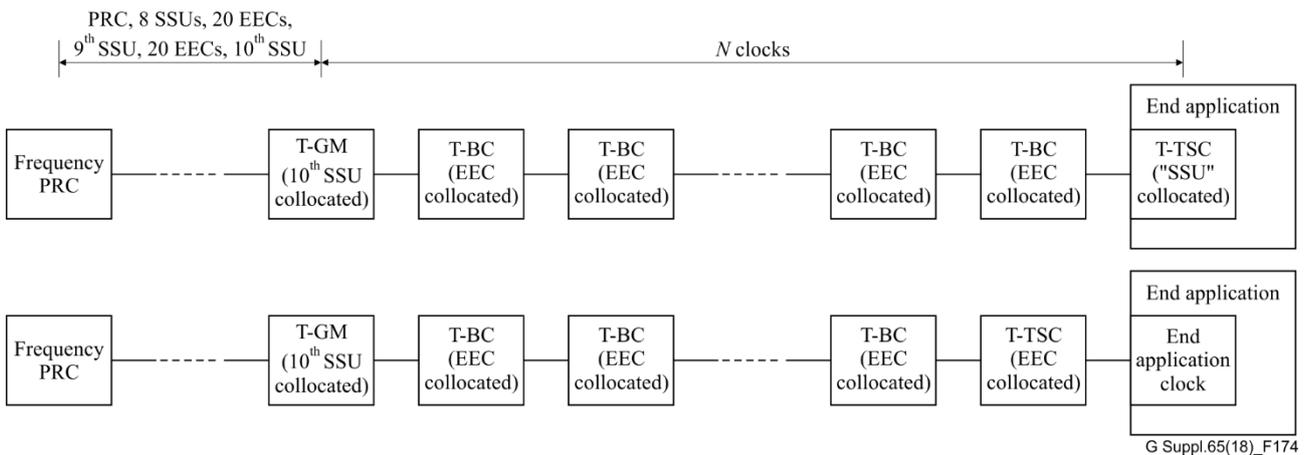
- a) If 0.1 Hz T-BC and endpoint filtering are used, the performance of the Delay Request/Response mechanism, the Pdelay mechanism with timestamping relative to the corrected clock, and the Pdelay mechanism with timestamping relative to the uncorrected clock, are very similar. This is true for both with and without SyncE noise.
- b) If there is no T-BC filtering, performance is somewhat better for the Pdelay mechanism with timestamping relative to the uncorrected clock than relative to the corrected clock. However, both are significantly better than performance of the Delay Request/Response mechanism with no T-BC filtering. This is true for both with and without SyncE noise. Consistent with previous contributions, the results for the Pdelay mechanism with timestamping relative to the uncorrected clock are slightly better without T-BC filtering than with T-BC filtering, due to noise generation in the T-BC filter in the latter case.

The results in clause 12.3.3.1.1, for case 1, indicate that the effect of initial time errors on the time to converge to steady state is small compared to the effect of 0.1 Hz endpoint filtering.

## 12.4 Simulation results for HRM2 cases based on single and multiple replications of simulations

### 12.4.1 Review of HRM2

HRM2 is shown in Figure 174, which is copied from Figure II.3 of of [ITU-T G.8271.1]. It consists of a time T-GM, followed by  $N - 1$  BCs, followed by a T-TSC. In addition, an EEC or SSU of a single SyncE reference chain is collocated with the time T-GM, each T-BC, and the endpoint T-TSC. [ITU-T G.8271.1] describes both the case where the end application clock is co-located with the final T-TSC and the case where the end application clock is separate from the T-TSC. This distinction is not important for the simulation cases described here, because the end-application clock is not considered.



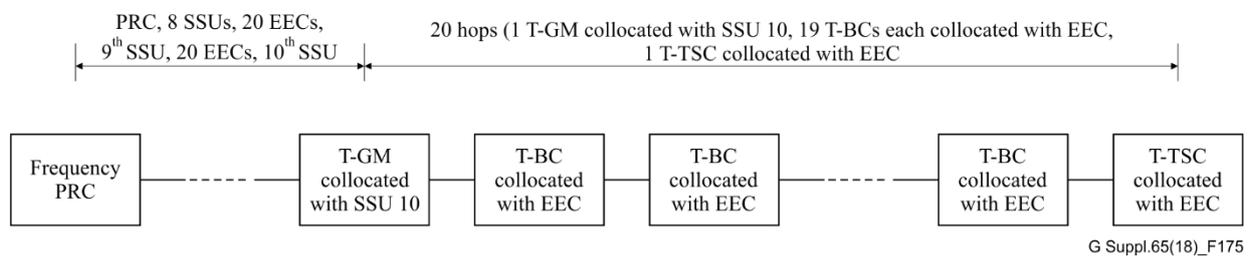
**Figure 174 – Schematic of HRM2**

In HRM3, there are 20 T-BCs (i.e.,  $N = 21$ ), which means there are 22 nodes and 21 hops in the PTP chain. In addition, each SyncE chain is an ITU-T G.803 reference chain with only 18 final EECs instead of 20. That is, it consists of a PRC, followed by 8 SSUs, followed by 20 EECs, followed by a ninth SSU, followed by 20 EECs, followed by a 10<sup>th</sup> SSU, followed by 18 EECs. The reason the final chain of EECs has 18 nodes rather than the maximum of 20 allowed by ITU-T G.803 is that it

was desirable to use, for simulation purposes, the same SyncE reference chain that was used for OTN (see Appendix VII of [ITU-T G.8251]). This would enable the SyncE noise accumulation simulation results obtained for OTN simulations to be re-used.

For HRM2, an initial thought might be to use the same number of BCs as in HRM3, i.e., have 22 PTP nodes (a T-GM, 20 BCs, and an T-TSC). However, it is also desirable to have the time T-GM collocated with an SSU. Finally, the most conservative case is to have the PTP nodes collocated with SyncE nodes at the end of the ITU-T G.803 reference chain, as the SyncE noise accumulation is largest here. Since the ITU-T G.803 reference chain allows an SSU to be followed by at most 20 EECs, these considerations are not consistent. That is, there is no adherence to the ITU-T G.803 reference chain, there can be no 20 BCs, no time T-GM collocated with an SSU, and no PTP nodes collocated with SyncE nodes at the end of the SyncE reference chain, all simultaneously. In a scenario where the chosen number of BCs is 20,+ and a full ITU-T G.803 reference chain with 20 EECs at the end is used (rather than 18 as used in HRM3 and for OTN), and the PTP nodes with SyncE nodes at the end of the reference chain are collocated, then the result would be that the 20 EECs at the end of the SyncE reference chain are collocated with the T-TSC and the final 19 T-BCs in the PTP chain. However, the first T-BC in the PTP chain is collocated with the tenth SSU, and the time T-GM is collocated with the EEC that precedes the tenth SSU. The problem is that there are 21 PTP nodes following the T-GM, but at most 20 consecutive EECs in the SyncE reference chain.

Two possible ways to resolve this difficulty are to (a) add a 21<sup>st</sup> EEC to the end of the ITU-TG.803 reference chain, or (b) reduce the number of T-BCs in the PTP chain by one (we could also choose to collocate the T-GM with an EEC and the first T-BC with an SSU, but it seems more desirable to better collocate time clocks with better frequency clocks). Since the ITU-T G.803 reference model has existed for many years, it was decided not to change this model, but instead to reduce the number of T-BCs by 1, to 19. The resulting HRM2 is shown in Figure 175.



**Figure 175 – HRM2, with selections made for numbers of SyncE and PTP nodes of each type and which SyncE nodes are collocated with which PTP nodes**

HRM2 is used for the simulations of this Supplement. Note that this does not affect HRM2 in [ITU-T G.8271.1].

#### 12.4.2 Description of simulation cases

The parameters and assumptions for the simulation cases of the present contribution are chosen to be as consistent as possible with the parameters and assumptions for the cases for HRM3 of clause 12.1, with differences specifically related to the fact that the HRMs are different. There are two main differences:

- a) Since the cases here are based on HRM2 rather than HRM3, there is only a single SyncE reference chain instead of one chain per PTP node
- b) There are 19 BCs instead of 20 (i.e., one less PTP node compared to HRM3)

Item (a) means that, for each simulation run, the noise accumulation results is used for the 10<sup>th</sup> SSU and final 20 EECs of a single SyncE noise accumulation simulation run, rather than the results for the 18<sup>th</sup> EEC following the 10<sup>th</sup> SSU from each of 21 SyncE noise accumulation simulation runs (for

HRM3; note that SyncE noise accumulation results were not needed at the T-GM for the HRM3 simulations).

Consistent with the HRM3 simulations, it is assumed that the T-GM is independent of the SyncE reference chain, i.e., its frequency is not traceable to the collocated SSU. This means that the accumulated EEC noise is not needed at the time T-GM (as was the case in HRM3) and that the EEC phase error at each T-BC and the final T-TSC is the full EEC phase error accumulation relative to the T-GM (plus the effect of the frequency offset of the PRC, though this effect is very small). It might have been assumed, instead, that the time T-GM frequency is the same as that of the collocated SSU. However, if it was required for the time to be traceable to TAI, this would mean the PRC frequency would also have to be traceable to the TAI frequency. In this case, the SyncE phase error at each T-BC and the final T-TSC would be the noise accumulation relative to the 10<sup>th</sup> SSU, and the PRC frequency offset would be zero (rather than in the range  $\pm 10^{-11}$ ). The resulting time error accumulation would be less than that under the current assumptions. The current assumptions are therefore conservative.

Table 36 summarizes the parameters and assumptions common to all simulation cases described here. The same random number generator initial state is used here as in the cases of clause 12.1. Since the only change here that changes the number of pseudo-random samples used is the number of PTP nodes, this means that the initial frequency offsets of nodes 2 to 21 here will be the same as for nodes 2 to 21 in the HRM3 cases of 13-1 for single-replication simulations. For multiple replication simulations, the frequency offsets will not be the same as for the HRM3 cases in replications after the first one due to the different number of PTP nodes. However, the results obtained here will differ from those for the corresponding HRM3 cases in any event due to the different SyncE phase noise inputs.

**Table 36 – Model parameters and assumptions common to all simulation cases**

Parameter	Value
Hypothetical Reference Model	HRM2 of Figure 175
SyncE phase noise accumulation	Results of clause 8.1.4.2
SyncE PRC frequency accuracy	$\pm 10^{-11}$
Model for sending of Sync and Pdelay_Req messages	Model based on gamma distribution, as described in clause 11.2, such that actual message intervals are within $\pm 30\%$ of mean intervals with 90% confidence, but do not exceed twice the mean interval.
Actual link propagation time	0.1 ms (total for each link, including transmit and receive PHYs; based on 20 km link and approximately 5 ns/m group delay); taken to be the same in both directions
Link and PHY asymmetries	0 (i.e., no link or PHY asymmetry)
Assumption for Grandmaster time	Grandmaster is assumed perfect and independent of the collocated SSU; time errors computed relative to T-GM
Pdelay turnaround time	10 ms
Simulation time	11,000 s (first 1000 s removed when computing peak or peak-to-peak values, to remove any initial transient)
Endpoint filter gain peaking	0.1 dB (corresponds to damping ratio of 4.3138)
T-BC filtering assumption	When present, 0.1 Hz 3 dB bandwidth with 0.1 dB gain peaking
One-step/two-step behaviour	T-BC and T-TSC models based on one-step behaviour
Noise generation in each filter (both endpoint filters and, if included, T-BC filters)	Modelled as described in clause 9.3.3, where now the SyncE phase error that is high-pass filtered includes both phase noise accumulation and phase error due to rearrangements

Table 37 summarizes all the simulation cases, using the case numbering of clause 12.1.

Link and PHY asymmetry are not considered in these cases. As for the HRM3 cases, the main interest is the effect of asymmetry on the worst-case scenario.

**Table 37 – Summary of simulation cases**

Case number (numbering scheme of 12.1)	Timestamp granularity (ns)	Sync Interval; Pdelay Interval (s)	0.1 Hz T-BC filter present	Endpoint filter bandwidth (Hz)
1	8	0.125; 1	No	0.01
	8	0.125; 1	No	0.05
	8	0.125; 1	No	0.1
	8	0.125; 1	Yes	0.01
	8	0.125; 1	Yes	0.05
	8	0.125; 1	Yes	0.1
2	8	1;8	No	0.01
	8	1;8	No	0.05
	8	1;8	No	0.1
	8	1;8	Yes	0.01
	8	1;8	Yes	0.05
	8	1;8	Yes	0.1
3	40	0.125; 1	No	0.01
	40	0.125; 1	No	0.05
	40	0.125; 1	No	0.1
	40	0.125; 1	Yes	0.01
	40	0.125; 1	Yes	0.05
	40	0.125; 1	Yes	0.1
4	40	1;8	No	0.01
	40	1;8	No	0.05
	40	1;8	No	0.1
	40	1;8	Yes	0.01
	40	1;8	Yes	0.05
	40	1;8	Yes	0.1

### 12.4.3 Simulation results for single replications of cases

This clause presents results for single replications of the simulation cases of clause 12.4.2, using the same format as in clause 12.1 for corresponding cases for HRM3. The following comparisons are made among the maximum absolute value time error results for the respective sub-cases of cases 1 to 4 of Table 37:

- a) For each of the cases in 1 to 4, without T-BC filtering and with 0.1 dB gain peaking, the results are compared for the three endpoint filter bandwidths, and
- b) For each of the cases in 1 to 4, with 0.1 dB gain peaking, the results are compared with and without T-BC filtering for each of the three endpoint filter bandwidths.

### 12.4.3.1 Results for cases without T-BC filtering

Figures 176 to 179 show the results for the maximum absolute value of filtered time error, as a function of T-BC/T-TSC node number, for cases 1 to 4 with no T-BC filtering. For each case, results for no endpoint filtering and endpoint filter bandwidths of 0.1 Hz, 0.05 Hz, and 0.01 Hz are compared. Since time error is computed relative to the T-GM, the time error at the T-GM is zero. Therefore, the time error at the T-GM is not shown, and the results begin at node 2. Table 38 gives the numerical results for maximum absolute value time error for the same cases and sub-cases as Figures 4-1 to 4-4, for the first T-BC (node 2), a T-BC near the middle of the chain (node 12), and the T-TSC at the end of the chain (node 21). The results in Table 38 are contained in Figures 4-1 to 4-4, though it may be difficult to obtain the same resolution visually from the figures.

The results for filtered time error are qualitatively similar to the results for HRM3 in clause 12.1. They indicate that there are two competing effects of making the endpoint filter bandwidth narrower:

- a) Noise generation increases due to the smaller high-pass filter corner frequency, causing increased time error, and
- b) The narrower endpoint filter bandwidth results in narrower-bandwidth low-pass filtering of the time error, causing it to decrease.

In case 1 (Figure 176), there is less overall time error accumulation due to the smaller granularity and PTP message interval (compared to the other cases). As a result, when the endpoint filter bandwidth is decreased to 0.01 Hz the increase in noise generation exceeds the decrease in the time error accumulation due to the filtering. The result is a larger time error accumulation compared to the sub-cases with 0.05 Hz and 0.1 Hz endpoint filter bandwidths. This is the same behaviour as was observed for HRM3 (Figure 100 of clause 12.1). However, the magnitude of the filtered phase error for HRM2 is smaller for nodes closer to the beginning of the HRM compared to HRM3, but increases more with increasing hop number. This is because, unlike in HRM3, the SyncE noise accumulation in HRM2 increases with increasing node number as the SyncE reference chain follows the chain of T-BCs.

In case 2 (Figure 177), the results are similar to case 1, though the filtered time error accumulates to a larger value due to the larger message interval. The filtered time error is still larger for the 0.01 Hz filter bandwidth (compared to 0.05 Hz and 0.1 Hz) due to noise generation, but unlike in HRM3 the time error for 0.01 Hz bandwidth is larger for all numbers of hops.

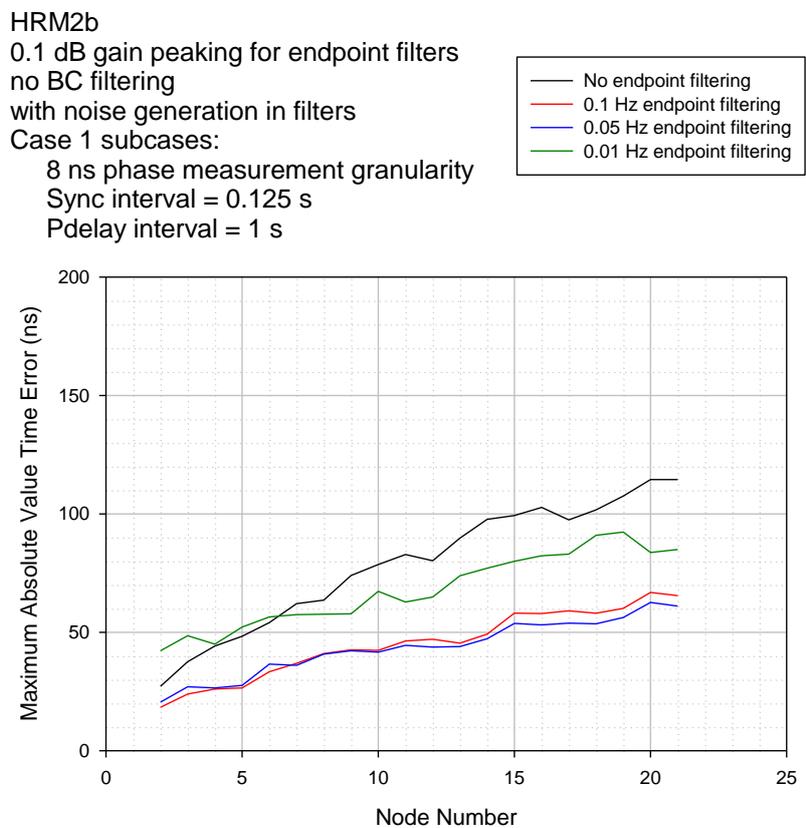
The behaviour in case 3 (Figure 178) is similar to HRM3 (Figure 102 of clause 12.1). The filtered time error results are slightly less than those for HRM3, with the difference being larger for a smaller number of hops. However, overall the HRM2 and HRM3 results are closer in case 3 than in case 1, because the timestamp granularity is much larger (40 ns instead of 8 ns) compared to the SyncE noise. The timestamp granularity is the same in both HRM3 and HRM2. In addition, now the increase noise generation due to decreased bandwidth is less than the decrease in time error accumulation due to filtering.

The behaviour in case 4 (Figure 179) is similar to HRM3 (Figure 103 of clause 12.1). The increase in both timestamp granularity and message interval result in a somewhat more rapid time error accumulation compared to cases 2 and 3. As in case 3, the increase noise generation due to decreased bandwidth is less than the decrease in time error accumulation due to filtering.

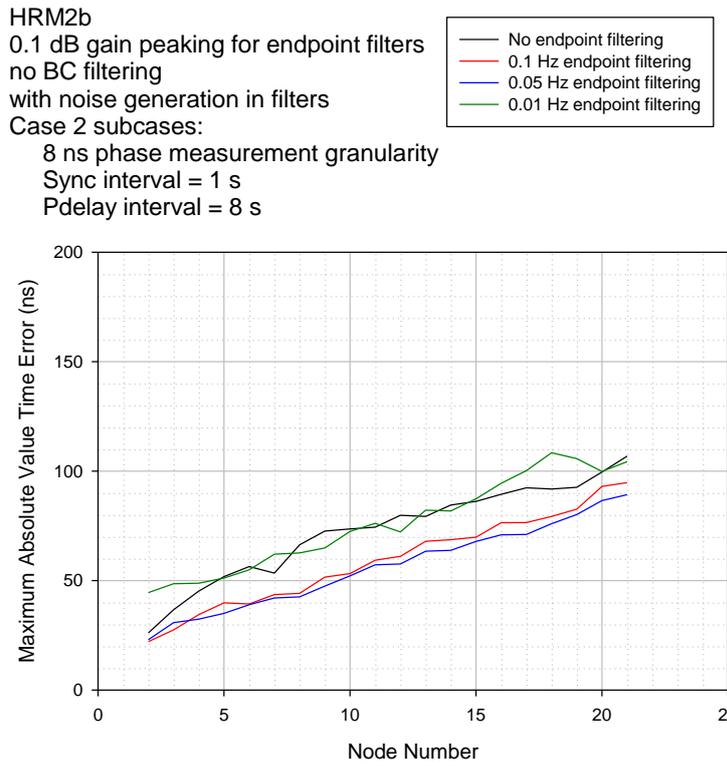
The unfiltered time error accumulation results apparently have much less dependence on the message interval compared to HRM3, for 8 ns timestamp granularity. For example, the unfiltered time error in both cases 1 and 2 increases from approximately 30 ns to just over 100 ns in HRM2. For HRM3, the unfiltered time error increases from approximately 50 ns to 140 ns for case 1 and from approximately 50 ns to 250 ns for case 2. For 40 ns timestamp granularity, the unfiltered time error accumulation results have little dependence on the message interval for both HRM2 and HRM3. The unfiltered time error in both cases 3 and 4 increases from approximately 60 ns to around 350 ns in

HRM2. For HRM3, the unfiltered time error increases from approximately 80 ns to 380 ns for case 1 and from approximately 90 ns to 400 ns for case 2.

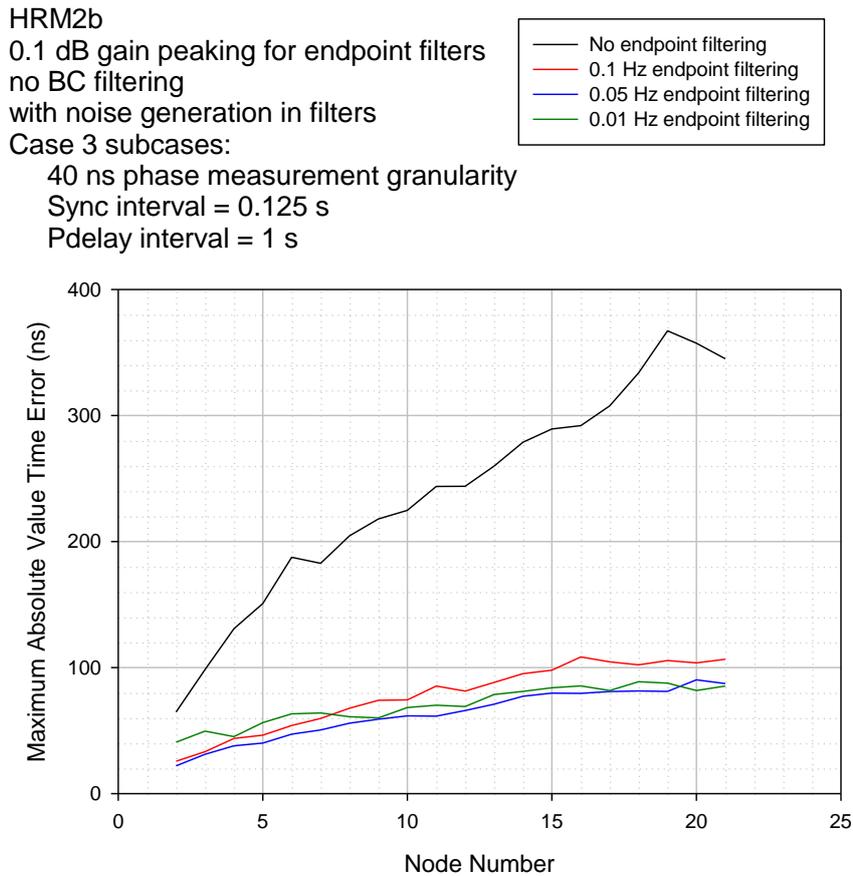
The multiple replication results in clause 5 (see Figures 184 and 185) will show that there is almost no dependence of the unfiltered time error on message interval for cases 1 and 2 (8 ns timestamp granularity), and a small decrease in unfiltered time error when the message interval increases for cases 3 and 4, i.e., 40 ns timestamp granularity. This latter result, for cases 3 and 4, was unexpected, but is due to the fact that, for 40 ns timestamp granularity, the timestamp granularity is appreciable compared to the SyncE phase noise. The time error due to timestamp granularity decreases with increasing message interval (for the same simulation time) because, with larger message interval, there are fewer timestamps taken (i.e., fewer sampling intervals in a given time) and it is less likely that the worst-case combination of timestamp granularity errors will occur. The appendix contains a frequency domain analysis that shows this result. The effect is visible in HRM2 because the accumulated error due to SyncE phase noise is smaller (as the accumulated SyncE noise at the BCs closer to the beginning of the HRM is smaller; this will be seen in the appendix).



**Figure 176 – Maximum absolute value of filtered time error, case 1, no T-BC filtering**

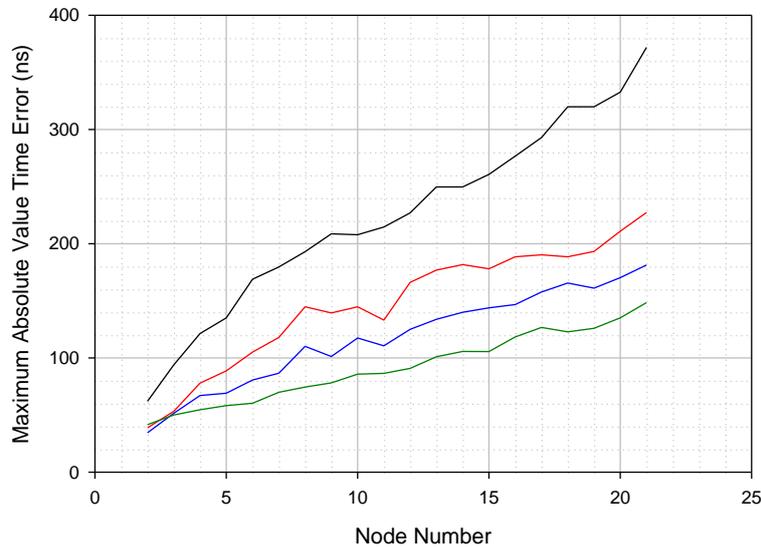


**Figure 177 – Maximum absolute value of filtered time error, case 2, no T-BC filtering**



**Figure 178 – Maximum absolute value of filtered time error, case 3, no T-BC filtering**

HRM2b  
 0.1 dB gain peaking for endpoint filters  
 no BC filtering  
 with noise generation in filters  
 Case 4 subcases:  
 40 ns phase measurement granularity  
 Sync interval = 1 s  
 Pdelay interval = 8 s



**Figure 179 – Maximum absolute value of filtered time error, case 3, no T-BC filtering**

**Table 38 – Results for maximum absolute value of filtered time error, cases 1 to 4 with no T-BC filtering, for nodes 2, 12 and 21**

Case/Node	Maximum absolute value of time error (ns)			
	No endpoint filtering	0.1 Hz endpoint filter	0.05 Hz endpoint filter	0.01 Hz endpoint filter
Case 1, Node 2	27.1	18.3	20.5	42.2
Case 1, Node 12	89.7	45.4	44.0	73.8
Case 1, Node 21	114.5	65.4	61.0	85.0
Case 2, Node 2	26.1	22.1	23.0	44.5
Case 2, Node 12	79.3	67.9	63.4	82.2
Case 2, Node 21	106.9	94.7	89.2	104.3
Case 3, Node 2	64.7	25.6	22.0	40.7
Case 3, Node 12	259.9	88.2	70.9	78.6
Case 3, Node 21	344.9	106.6	87.3	85.3
Case 4, Node 2	62.0	38.8	34.4	41.6
Case 4, Node 12	249.6	176.8	133.8	101.0
Case 4, Node 21	371.8	227.3	181.3	148.4

### 12.4.3.2 Results for cases with T-BC filtering

Figures 180 to 183 show the results for the maximum absolute value of filtered time error, as a function of T-BC/T-TSC node number, for cases 1 – 4 with and without T-BC filtering. For each case, results with and without T-BC filtering are compared for endpoint filter bandwidths of 0.1 Hz, 0.05 Hz, and 0.01 Hz. As in clause 13.4.3.3.1, the time error is computed relative to the T-GM. The

time error at the T-GM is zero and is not shown, and the results begin at node 2. Table 39 gives the numerical results for the maximum absolute value time error for the same cases and sub-cases as Figures 9 to 12, for the first T-BC (node 2), a T-BC near the middle of the chain (node 12), and the T-TSC at the end of the chain (node 21). The results in Table 39 are contained in Figures 1 to 4, though it may be difficult to obtain the same resolution visually from the figures.

As was the case for the corresponding HRM3 cases (see clause 12.1), the results indicate that there are two competing effects of adding T-BC filtering:

- a) The narrow-bandwidth filtering at each node reduces the time error accumulation up to that node.
- b) Noise generation of the T-BC filter tends to increase the time error. However, this is filtered at the next node (and the noise generation of the last T-BC is filtered at the endpoint filter). Note that the noise generation of the T-BC filter is statistically the same as that of the 0.1 Hz endpoint filter (and therefore statistically less than that of the 0.05 Hz and 0.01 Hz endpoint filters).

Therefore, it is expected that T-BC filtering will have a larger benefit in cases where the time error accumulation is larger, e.g., (i) larger granularity, or (ii) larger PTP message intervals.

In all four cases, the time error accumulation with and without T-BC filtering is the same at node 2 (as expected, because T-BC filtering does not affect the time error accumulation at the first T-BC). In cases 1 and 2, the time error accumulation with T-BC filtering increases faster than the time error accumulation without T-BC filtering, for all three endpoint filter bandwidths. In case 3, the time error accumulation with T-BC filtering also increases faster than the time error accumulation without T-BC filtering, although for 0.1 Hz endpoint filter bandwidth the difference between the cases with and without T-BC filtering is not large. In case 4, the time error accumulation with T-BC filtering increases faster than without T-BC filtering for 0.01 Hz bandwidth, at about the same rate for 0.05 Hz bandwidth, and more slowly for 0.1 Hz bandwidth. Compared to HRM3 (Figures 108 to 111 of clause 12.1) the difference in time error accumulation between cases with and without T-BC filtering is generally larger here, though the difference is less for case 4 and 0.05 Hz endpoint filter bandwidth. However, adding T-BC filtering tends to increase the time error accumulation, except for case 4 and larger filter bandwidths. It should be noted that the error due to timestamp granularity is larger (compared to the error due to SyncE noise) in cases 3 and 4 compared to cases 1 and 2. In addition, in HRM2 the SyncE noise is smaller at the earlier nodes. This means that in case 1 the SyncE noise has a lower value at node 2, increases progressively with each node, and gives rise to noise generation. At each successive node the SyncE noise from the previous node is filtered by the T-BC filter, but the noise generation in the T-BC filter at this node is larger than at the previous node. If the other noise sources (e.g., timestamp granularity) are smaller, the result is that the time error increases faster with T-BC filtering than without T-BC filtering. As other noise sources become more appreciable (e.g., in cases 3 and 4), this effect of the SyncE noise is less evident.

Also, in cases 1 – 3 the time error accumulation is largest for 0.01 Hz endpoint filtering, both with and without T-BC filtering. This is due to the larger noise generation in the 0.01 Hz endpoint filter, compared to 0.05 Hz and 0.1 Hz. In case 4 the time error accumulation is largest for 0.1 Hz endpoint filtering. Here the effect of the 40 ns timestamp granularity outweighs the effect of the SyncE noise.

Finally, as for HRM3, all four cases show very little difference in the results for 0.1 Hz endpoint filtering and 0.05 Hz endpoint filter, when T-BC filtering is present. Furthermore, as for HRM3, the results for 0.01 Hz endpoint filtering, when T-BC filtering is present, are either worse than the results for 0.05 Hz and 0.1 Hz endpoint filtering or are of the same order as the results for 0.05 Hz and 0.1 Hz endpoint filtering (e.g., in cases 1 – 3 the 0.01 Hz endpoint filter results are worse than the 0.05 Hz and 0.1 Hz results for 1 – 20 hops. In case 4 the former are of the same order as the latter). This means that, as for HRM3, if T-BC filtering is present there is no benefit to making the endpoint filter bandwidth narrower than 0.1 Hz, and the performance will likely be worse if the bandwidth is narrowed to 0.01 Hz. This conclusion is favourable because it means that, in the case where T-BC

filtering is present, (a) the endpoint filter bandwidth need not be too narrow, and (b) the endpoint filter bandwidth can have the same value as the T-BC filter bandwidth (i.e., the two clocks need not have different bandwidth requirements). This result was obtained for HRM3.

In conclusion, for cases with T-BC filtering:

- a) In cases 1 to 3, adding T-BC filtering increases the maximum time error accumulation.
- b) In case 4, there may be benefit in adding T-BC filtering for larger endpoint filter bandwidths, e.g., 0.1 Hz (but not for smaller filter bandwidths, e.g., 0.01 Hz).
- c) If 0.1 Hz T-BC filtering is present, there is no benefit to making the endpoint filter bandwidth narrower than 0.1 Hz, i.e., the endpoint and T-BC filter bandwidths can be the same.

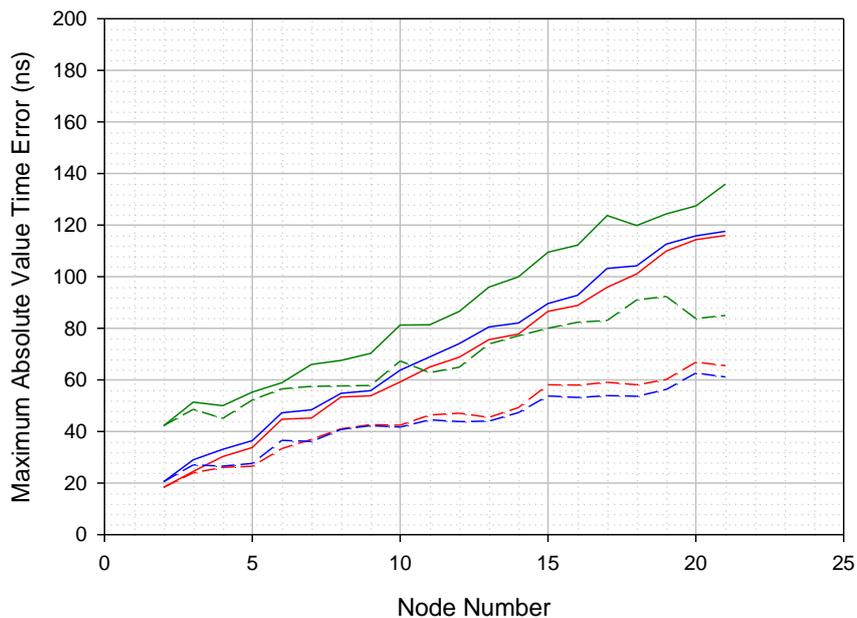
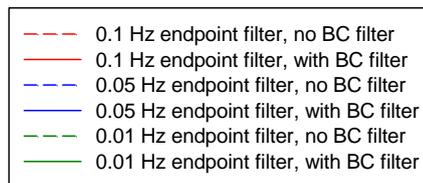
As indicated above, HRM2 cases with SyncE rearrangements have not yet been analysed. There may be benefit of T-BC filtering for these cases if the T-BC filter time constant is of the order of the Sync interval or larger. In this case, conclusion (c) above is relevant.

**HRM2b**

0.1 dB gain peaking for endpoint and BC filters  
 0.1 Hz BC filter bandwidth  
 with noise generation in all filters

Case 1 subcases:

8 ns phase measurement granularity  
 Sync interval = 0.125 s  
 Pdelay interval = 1 s



**Figure 180 – Case 1, comparison of results with and without T-BC filtering**

HRM2b  
 0.1 dB gain peaking for endpoint and BC filters  
 0.1 Hz BC filter bandwidth  
 with noise generation in filters  
 Case 2 subcases:  
 8 ns phase measurement granularity  
 Sync interval = 1 s  
 Pdelay interval = 8 s

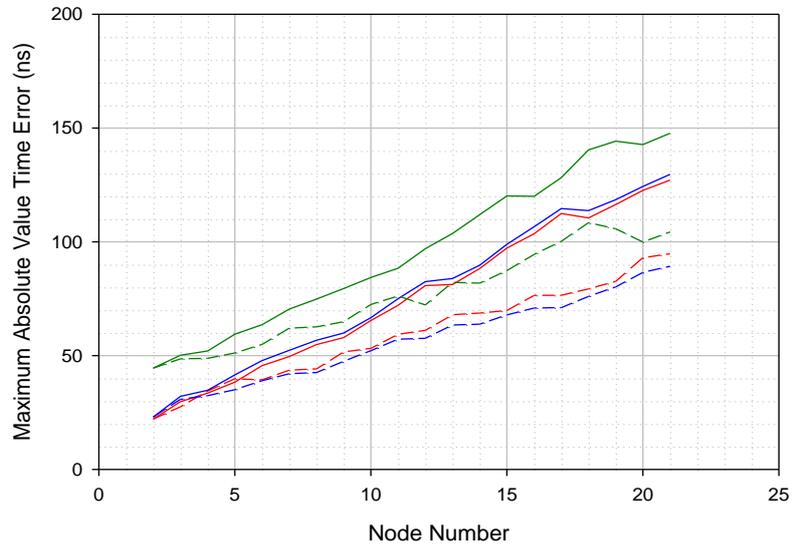
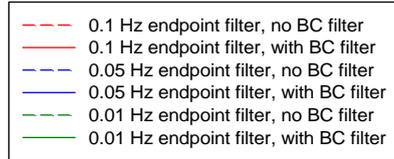


Figure 181 – Case 2, comparison of results with and without T-BC filtering

HRM2b  
 0.1 dB gain peaking for endpoint and BC filters  
 0.1 Hz BC filter bandwidth  
 with noise generation in all filters  
 Case 3 subcases:  
 40 ns phase measurement granularity  
 Sync interval = 0.125 s  
 Pdelay interval = 1 s

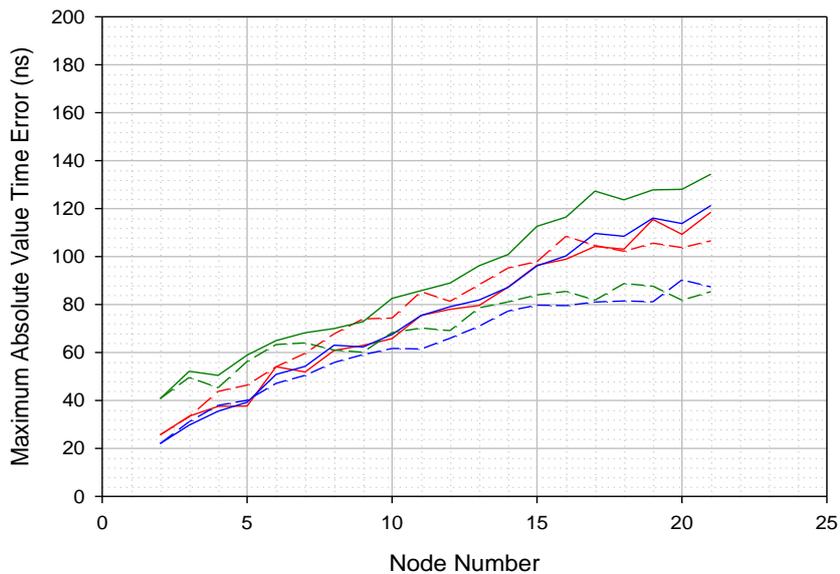
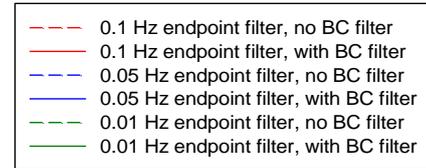


Figure 182 – Case 3, comparison of results with and without T-BC filtering

HRM2b  
 0.1 dB gain peaking for endpoint and BC filters  
 0.1 Hz BC filter bandwidth  
 with noise generation in all filters  
 Case 4 subcases:  
 40 ns phase measurement granularity  
 Sync interval = 1 s  
 Pdelay interval = 8 s

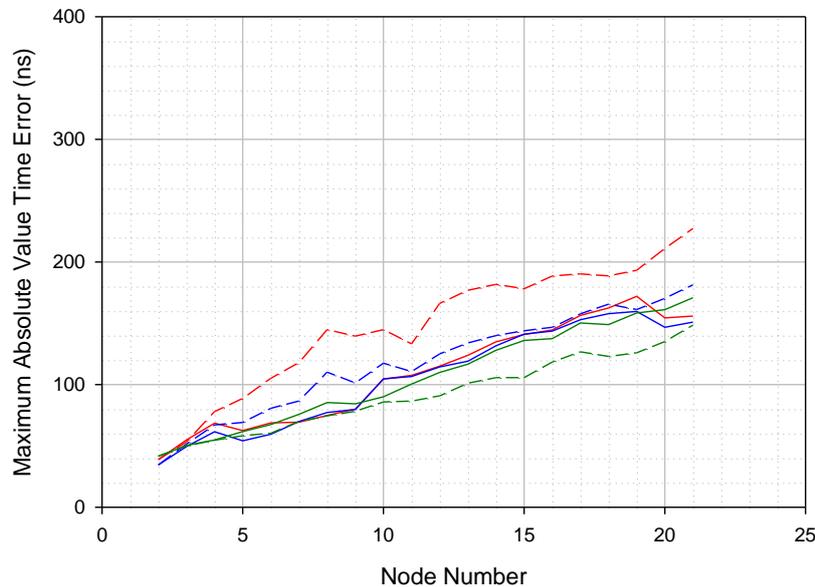
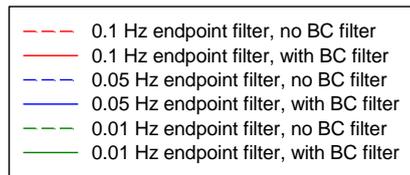


Figure 183 – Case 4, comparison of results with and without T-BC filtering

Table 39 – Results for maximum absolute value of filtered time error, cases 1 to 4, with and without T-BC filtering for the respective endpoint filter bandwidths, for nodes 2, 12 and 21

Case/Node	Maximum absolute value of time error (ns)					
	0.1 Hz endpoint filter		0.05 Hz endpoint filter		0.01 Hz endpoint filter	
	No T-BC filter	0.1 Hz T-BC filter	No T-BC filter	0.1 Hz T-BC filter	No T-BC filter	0.1 Hz T-BC filter
Case 1, Node 2	18.3	18.3	20.5	20.5	42.2	42.2
Case 1, Node 12	45.4	75.6	44.0	80.5	73.8	95.9
Case 1, Node 21	65.4	116.0	61.0	117.5	85.0	135.9
Case 2, Node 2	22.1	22.1	23.0	23.0	44.5	44.5
Case 2, Node 12	67.9	81.2	63.4	83.9	82.2	103.6
Case 2, Node 21	94.7	127.1	89.2	129.6	104.3	147.7
Case 3, Node 2	25.6	25.6	22.0	22.0	40.7	40.7
Case 3, Node 12	88.2	79.5	70.9	81.9	78.6	96.1
Case 3, Node 21	106.6	118.5	87.3	121.3	85.3	134.4
Case 4, Node 2	38.8	38.8	34.4	34.4	41.6	41.6
Case 4, Node 12	176.8	123.8	133.8	119.0	101.0	116.7
Case 4, Node 21	227.3	155.9	181.3	150.9	148.4	170.8

### 12.4.3.3 Simulation results for multiple replications of selected cases of clause 12.4.3.2

This clause presents the results for 300 multiple replications of simulations, for a subset of the 24 sub-cases (of cases 1 to 4) described in clauses 12.4.3.1 and 12.4.3.2. The results include the following:

- a) Comparison of cases 1 and 2, and of cases 3 and 4, for sub-cases with no T-BC filtering and no endpoint filtering. These results support the conclusion drawn from the single-replication results in clause 13.4.3.3.1 that the unfiltered time-error accumulation has much less dependence on the message interval compared to corresponding HRM3 cases.
- b) The same cases (a) to (h) of clause 12.2 (Table 32), but with 2 additional cases (labelled (i) and (j)) added (see Table 40 below, copied, from clause 12.1 with the two new cases added).

As in previous simulation studies, 300 replications enables 99% confidence intervals for the 0.95 quantile of maximum absolute value time error to be obtained. For each of the 300 runs, the maximum absolute value of the time error is computed at each node (except the T-GM (node 1) because time error is computed relative to the T-GM and, therefore, the T-GM time error is zero), taken over 10000 s of simulated time (the initial 1000 s of the 11000 s simulation was discarded to ensure that any initial transient was removed). For each node, the 300 samples of maximum absolute value time error were placed in ascending order. An estimate of a 99% confidence interval for the 0.95 quantile is given by the interval between the 275<sup>th</sup> and 294<sup>th</sup> smallest samples, and a point estimate of the 0.95 quantile is given by the 285<sup>th</sup> smallest sample. Some of the results that follow also show the largest (i.e., 300<sup>th</sup> smallest) sample.

**Table 40 – Simulation cases (a) to (h) of Table 32, plus new cases (i) and (j)**

Case designation here	Table 37 case number	Timestamp granularity (ns)	Endpoint filter bandwidth (Hz)	0.1 Hz T-BC filter present
a	1	8	0.05	No
b	1	8	0.05	Yes
c	1	8	0.1	No
d	1	8	0.1	Yes
e	3	40	0.05	No
f	3	40	0.05	Yes
g	3	40	0.01	No
h	3	40	0.01	Yes
i	1	8	0.01	No
j	3	40	0.1	Yes

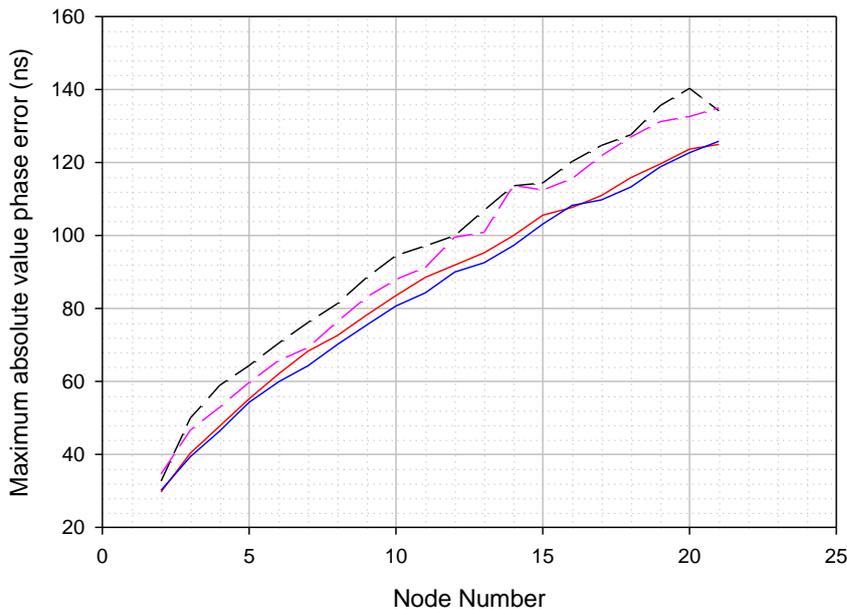
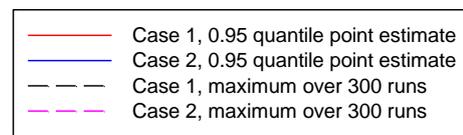
#### 12.4.3.3.1 Comparison of sub-cases with no T-BC filtering and no endpoint filtering

Figure 184 compares the maximum absolute value time error for cases 1 and 2, with no T-BC filtering and no endpoint filtering. The results are shown for the 0.95 quantile point estimate and the maximum over 300 runs, at each T-BC and the final T-TSC (nodes 2 to 21). Cases 1 and 2 differ in that the former has 0.125 s Sync interval and 1 s Pdelay interval, while the latter has 1 s Sync interval and 8 s Pdelay interval. The results show that the maximum absolute value time error is very similar in both cases, increasing from approximately 30 ns at node 2 to 135 ns at node 21. A similar result was obtained in clause 13.4.3.3.1 from single-replications of cases 1 and 2 with no filtering (see the solid black curves of Figures 176 and 177). As indicated in clause 3, a frequency-domain analysis of these results is provided in the Appendix of [b-Garner04].

Figure 185 shows the same comparison for cases 3 and 4 (i.e., no T-BC and no endpoint filtering). The difference between cases 3 and 4, and cases 1 and 2, is that the former have 40 ns timestamp

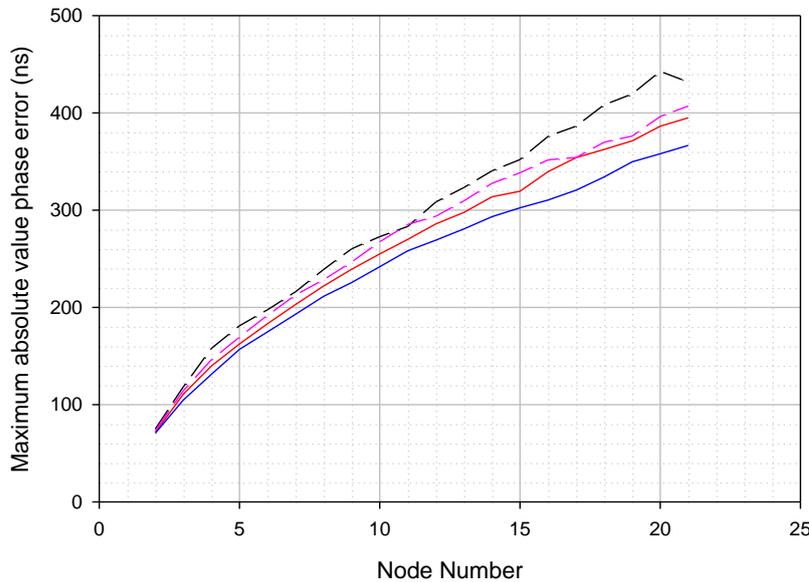
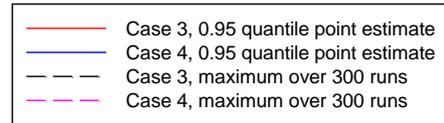
granularity and the latter have 8 ns timestamp granularity. For these cases, the maximum absolute value time error is slightly smaller in case 4 than in case 3, i.e., the maximum absolute value time error decreases slightly when the message interval is increased (from 0.125 s Sync interval and 1 s Pdelay interval to 1 s Sync interval and 8 s Pdelay interval). As briefly described in clause 13.4.3.3.1, the time error due to timestamp granularity is larger in cases 3 and 4 compared to cases 1 and 2, relative to the SyncE phase noise because the timestamp granularity is 40 ns for cases 3 and 4 and 8 ns for cases 1 and 2. The time error due to timestamp granularity decreases with increasing message interval (for the same simulation time) because, with larger message interval, there are fewer timestamps taken (i.e., fewer sampling intervals in a given time) and it is less likely that the worst-case combination of timestamp granularity errors will occur. The appendix contains a frequency domain analysis that shows this result. The effect is visible in HRM2 because the accumulated error due to SyncE phase noise is smaller (as the accumulated SyncE noise at the BCs closer to the beginning of the HRM is smaller (see the appendix of [b-Garner04]).

Cases 1 and 2: results for 300 replications  
 0.95 quantile point estimate and maximum  
 Noise generation modeled in all filters  
 0.1 dB gain peaking in all filters  
 No BC filtering  
 No endpoint filtering  
 Case 1: 0.125 s Sync interval, 1 s Pdelay interval  
 Case 2: 1 s Sync interval, 8 s Pdelay interval  
 8 ns timestamp granularity in both cases



**Figure 184 – Comparison of results for cases 1 and 2, with no T-BC filtering and no endpoint filtering**

Cases 3 and 4: results for 300 replications  
 0.95 quantile point estimate and maximum  
 Noise generation modeled in all filters  
 0.1 dB gain peaking in all filters  
 No BC filtering  
 No endpoint filtering  
 Case 3: 0.125 s Sync interval, 1 s Pdelay interval  
 Case 4: 1 s Sync interval, 8 s Pdelay interval  
 40 ns timestamp granularity in both cases



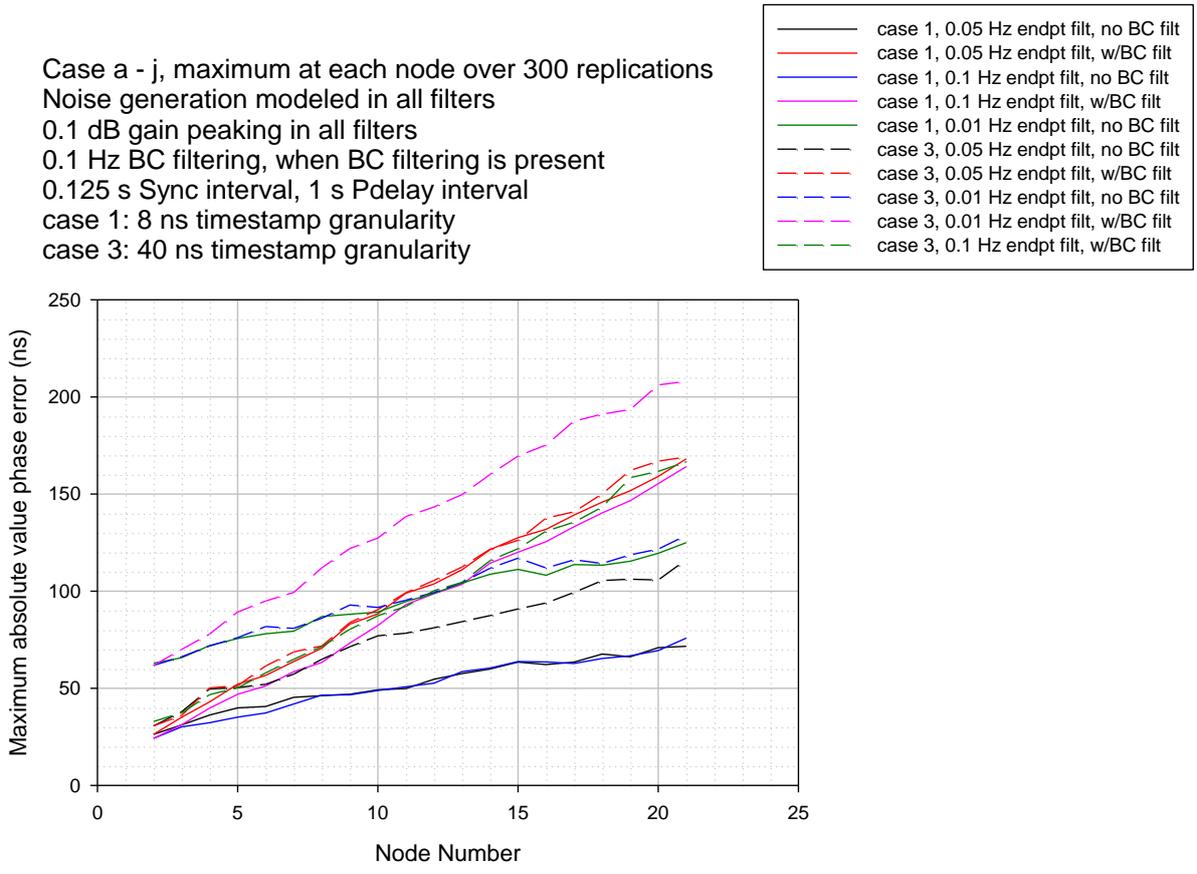
**Figure 185 – Comparison of results for cases 3 and 4, with no T-BC filtering and no endpoint filtering.**

#### 12.4.3.3.2 Results for simulation cases corresponding to cases (a) to (h) of clause 12.1

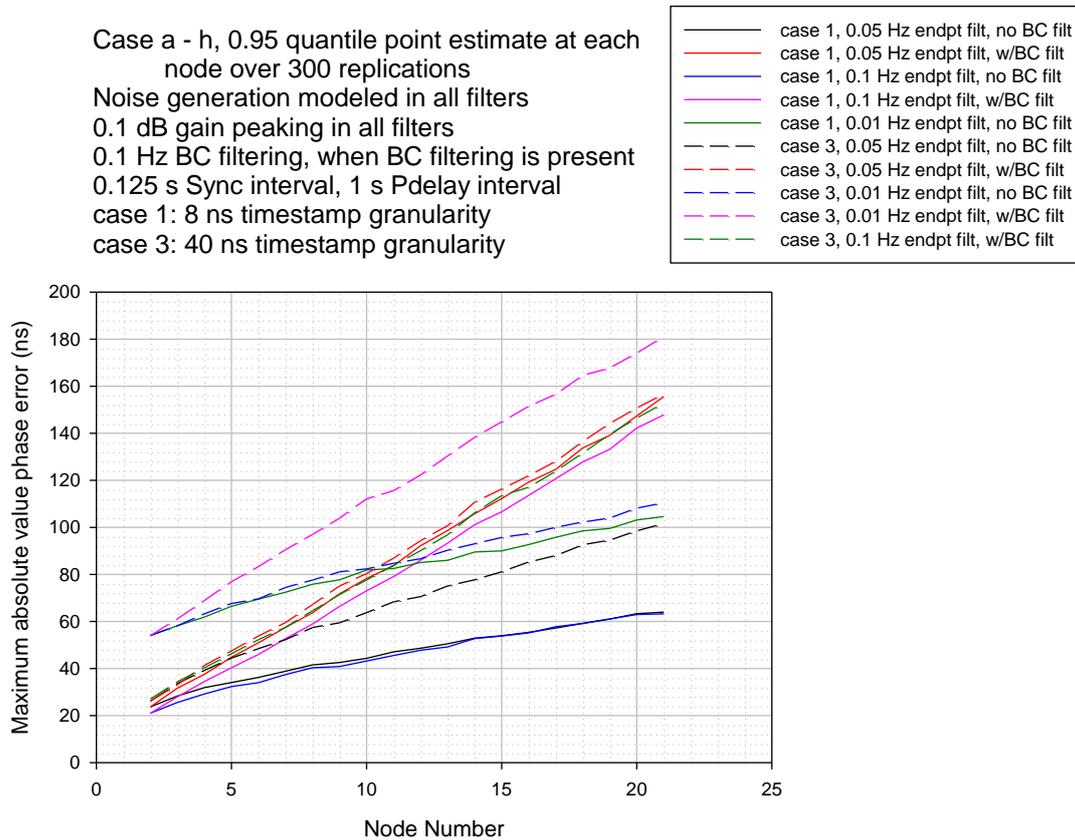
Figure 186 shows the results for cases (a) to (h) of Table 32 for the maximum over 300 runs of the maximum absolute value time error at each node. Figure 187 shows the results for cases (a) to (h) for the point estimate of the 0.95 quantile of maximum absolute value time error, based on 300 runs.

The maximum results are slightly larger than the 0.95 quantile results, as expected. Both sets of results are consistent with the single-replication results in clauses 12.4.2 and 12.4.3, though somewhat larger (as expected, due to being based on multiple runs). For example, the results for case 1 with T-BC filtering and 0.05 Hz and 0.1 Hz endpoint filters (solid red and solid purple curves, respectively, in Figures 186 and 187) increase monotonically from node 2 to node 21, with the results for 0.05 Hz endpoint filter slightly larger than the results for the 0.1 Hz endpoint filter due to the noise generation. The same results were obtained for single replications; see the solid blue (0.05 Hz) and solid red (0.1 Hz) curves in Figure 180. The corresponding results for case 1 with no T-BC filtering (solid blue (0.1 Hz endpoint filter) and solid black (0.05 Hz endpoint filter) curves here) have maximum absolute value time errors that are almost the same. Similar results were obtained for single replications (see the dashed blue (0.05 Hz) and dashed red (0.1 Hz) curves in Figure 180). Likewise, the results for case 3 with T-BC filtering and 0.05 Hz and 0.01 Hz endpoint filters (dashed red and dashed purple curves, respectively, in Figures 186 and 187) increase monotonically from node 2 to node 21, with the results for 0.01 Hz endpoint filter larger than the results for the 0.05 Hz endpoint filter due to the noise generation. The same results were obtained for single replications; see the solid blue (0.05 Hz) and solid green (0.01 Hz) curves in Figure 182. The corresponding results for case 3 with no T-BC filtering (dashed blue (0.01 Hz endpoint filter) and dashed black (0.05 Hz endpoint filter) curves here) show the former above the latter, due to the noise generation. Similar results were obtained for single replications (see the dashed blue (0.05 Hz) and dashed green (0.01 Hz) curves in Figure 182).

The consistency of the multiple replication results here with the results of clause 12.4.3.2 provides additional support for the conclusions (a) to (c) of clause 12.4.3.2.



**Figure 186 – Cases a to h, maximum absolute value time error at each node, maximum over 300 replications**



**Figure 187 – Cases a to h, maximum absolute value time error at each node, 0.95-quantile point estimate over 300 replications**

### 12.4.3.3.3 Summary and conclusions for HRM2 simulation cases

Clause 12.4 provided simulation results for the transport of time over a chain of boundary clocks based on HRM2. In this HRM, the SyncE reference chain that provides frequency follows the chain of BCs (as shown in Figure 175). The final SSU of the SyncE reference chain is collocated with the T-GM, and an EEC is collocated with each T-BC along with the final T-TSC. The results were presented for single replications of the same cases previously considered for HRM3 (see clause 12.1), and then results based on 300 replications were presented for a subset of these cases. Cases with and without 0.1 Hz T-BC filtering were considered.

As with HRM3, the results indicated that there are two competing effects of making the endpoint filter bandwidth narrower:

- a) Noise generation increases due to the smaller high-pass filter corner frequency (seen by the noise generation), causing increased time error, and
- b) The narrower endpoint filter bandwidth results in narrower-bandwidth low-pass filtering of the time error, causing it to decrease.

Also as with HRM3, it was found that adding T-BC filtering can increase the maximum time error accumulation. The only case where T-BC filtering had definite benefit in terms of decreasing the maximum time error accumulation was for longer message interval, larger timestamp granularity, and wider endpoint filter bandwidth, all taken together (i.e., case 4 with 0.1 Hz endpoint filter bandwidth). Note, however, that HRM2 cases with SyncE rearrangements have not yet been analysed. There might be some benefit of T-BC filtering in these cases.

Also as with HRM3, it was found that if 0.1 Hz T-BC filtering is present, there is no benefit to making the endpoint filter bandwidth narrower than 0.1 Hz, i.e., the endpoint and T-BC filter bandwidths can

be the same. This conclusion is beneficial, because it means that it would not be necessary to have two different filter bandwidths, one for the endpoint filter and one for the T-BC filter.

Finally, it was found that if no filtering is present (i.e., neither T-BC nor endpoint filtering), the maximum absolute value time error can decrease with increasing message interval if the dominant noise source is the timestamp granularity (i.e., quantization noise). This is because the band-pass bandwidth of the frequency response between the quantization noise and the output decreases with increasing sampling interval. This effect is seen in HRM2 with 40 ns timestamp granularity (i.e., cases 3 and 4). The effect is not large. It is not seen with 8 ns timestamp granularity in HRM2. In that case the effect of the quantization noise is smaller compared to the effect of SyncE phase noise. The effect is also not seen in any of the HRM3 cases. In those cases, the SyncE noise is apparently large enough to mask the effect.

## Appendix I

### Sub-Nyquist artefacts and sampling moiré effects when measuring PTP to PTP and PTP to 1 PPS noise transfer

#### I.1 Introduction

When verifying the PTP to PTP and PTP to 1 PPS noise transfer requirements in [ITU-T G.8273.2], there is a possibility of aliasing if the frequency of the applied sinusoid exceeds the Nyquist frequency corresponding to the PTP Sync rate (i.e., 8 Hz) or the 1 PPS rate (i.e., 1 Hz). Notes in clauses 7.3 and C.2.3 of [ITU-T G.8273.2] indicate the possibility of this aliasing and also briefly mention that, even if the Nyquist condition is satisfied, apparent low-frequency artefacts can be observed in the sampled signal. Specifically, if the frequency of the sinusoid is  $f$ , one or more low-frequency envelopes can be observed in the sampled signal if  $f$  is very close to  $(m/n)f_s$ , where  $f_s$  is the sampling frequency and both  $m$  and  $n$  are integers with  $m < n$  (in the case where  $m = 1$  and  $n = 2$ ,  $f$  must also be strictly less than  $f_s/2$  for the Nyquist condition to be met). These artefacts are referred to as *Sub-Nyquist artefacts* or as *sampling moiré effects*. However, low-frequency components are not actually present; this can be shown by computing the discrete Fourier transform (DFT) of the sampled signal.

The theory of sub-Nyquist artefacts, with a detailed description and many examples, is given in [b-Amidoror]. This appendix very briefly summarizes the theory, and provides examples relevant to measurement of noise transfer in [ITU-T G.8273.2].

#### I.2 Description of sub-Nyquist artefacts and examples

Consider the sinusoidal (i.e., cosine) signal with unit amplitude, zero phase relative to time zero, and frequency  $f$ . This signal is given by

$$y(t) = \cos 2\pi ft. \quad (\text{I-1})$$

Let the signal be sampled at sampling rate  $f_s$ . The sampling interval,  $T$ , is  $1/f_s$ . Let  $k$  denote the successive sampling instants. Then the sampled signal is given by

$$y_k = y(kT) = \cos 2\pi fkT \quad (\text{I-2})$$

Now let the frequency of the signal be chosen as

$$f = \frac{m}{n} f_s + \varepsilon, \quad (\text{I-3})$$

where  $m$  and  $n$  are positive integers, and  $\varepsilon \ll f$ . Assuming that the fraction  $m/n$  is reduced, but otherwise place no restriction on  $m$  and  $n$ . Substituting this into equation (I-2) gives

$$y_k = \cos 2\pi \left( \frac{m}{n} f_s + \varepsilon \right) kT = \cos \left( 2\pi \frac{mk}{n} + 2\pi \varepsilon kT \right). \quad (\text{I-4})$$

Since  $\varepsilon \ll f$ , we have  $\varepsilon T \ll fT$ , and therefore  $\varepsilon T \ll f_s(m/n)T$ , or

$$\varepsilon T \ll \frac{m}{n}. \quad (\text{I-5})$$

It is seen from equation (I-4) that every  $n$  samples of the cosine function advance through approximately  $m$  cycles. However, due to the second term of the argument of the cosine, i.e.,  $2\pi \varepsilon kT$ , the samples advance slowly through each cycle.

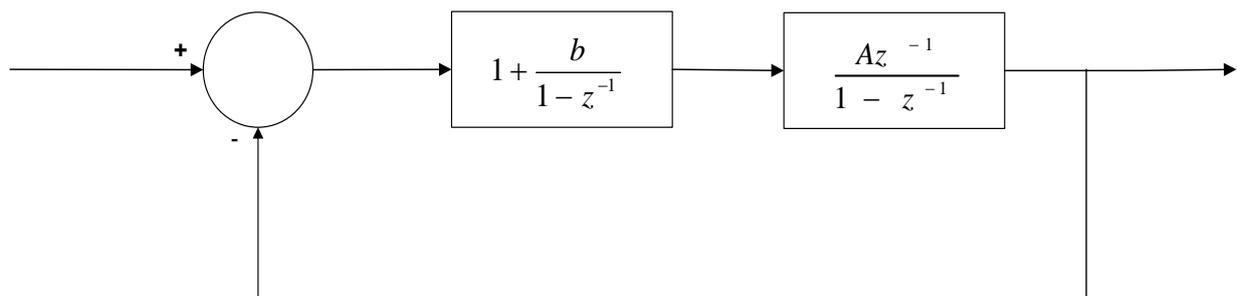
The next two cases that must be consider are,  $m/n \geq 1/2$ , and  $m/n < 1/2$ . In the former case, the sampling frequency is less than twice the frequency of the cosine, and aliasing occurs. In the latter case, the sampling frequency is less than twice the frequency of the cosine, and there is no aliasing. This is the

case of interest in this appendix, and will be the focus from here on. The frequency  $f_s/2$  is the Nyquist frequency, denoted as  $f_{Nyquist}$ .

As a result of this slow advance, the sampled signal, for the case of no aliasing, appears as a cosine function whose amplitude is slowly modulated by one or more low-frequency envelopes. It might be thought that these low frequencies would appear in the frequency spectrum of the sampled signal, e.g., if the DFT is computed. However, the frequency of the underlying signal in this case is less than  $f_{Nyquist}$ , and there is no aliasing. If the DFT is computed, the low frequencies associated with the envelopes are not present in the spectrum. The low-frequency envelopes are referred to as sub-Nyquist artefacts or as *sampling moiré effects*.

Some examples of this are given below. In [b-Amidoror], a number of examples are provided and it also discusses the general case where the signal is periodic but not necessarily sinusoidal, and proves that the sub-Nyquist artefacts can occur in this case. [b-Amidoror] also indicates that, since the Nyquist condition is satisfied in this case, ideally the signal could be completely reconstructed from the sampled signal (even though the low-frequency envelope(s) is (are) present). However, [b-Amidoror] then indicates that this ideal reconstruction assumes that sinc() functions are used, and points out that in practice non-ideal interpolation would be used. In such non-ideal cases, the artefacts might remain after reconstruction. Finally, [b-Amidoror] points out that if the signal is processed in a non-linear manner, e.g., if envelope detection is performed, the artefacts could appear after the processing and their frequencies would then actually be present.

In the noise transfer measurement, reconstructing the analogue signal from the samples is not of interest. Rather, the point of interest is in knowing what happens when the sampled signal is filtered with a low-pass digital filter that represents a digital PLL (i.e., a T-BC or T-TSC). The two examples below starts with a continuous time cosine signal, sampled according to equation (I-3), focusing on case  $m=1$ . DFT is computed to confirm that the low frequencies are not present in the spectrum. The sampled signal is then filtered with a second-order, low-pass filter with 0.1 Hz bandwidth and 0.1 dB gain peaking. Two different filter models are used. The first model, referred to as model 1, is the model of Appendix VIII of [ITU-T G.8251]. This model is used because it has been used in many simulation analyses in ITU-T for approximately the last 15-20 years. The second model, referred to as model 2, is the digital PLL model of [b-Rogers]. A simplified representation of this model is given in Figure I.1.



**Figure I.1 – Model 2 discrete second-order filter**

The transfer function for the model 2 filter is

$$H(z) = \frac{K(z - \alpha)}{z^2 + (K - 2)z + (1 - \alpha K)}, \quad (I-6)$$

where

$$K = (1 + b)A$$

$$\alpha = \frac{1}{1 + b} \quad (I-7)$$

Equation (I-6) above is the same as equation (3.35) of [b-Rogers]. It is shown in [b-Rogers] that if  $\omega_n$  is the undamped natural frequency,  $\zeta$  is the damping ratio, and  $T$  is the sampling interval, then  $K$  and  $\alpha$  are related to these quantities by

$$\alpha = \frac{4\zeta - \omega_n T}{4\zeta + \omega_n T}$$

$$K = \frac{\omega_n^2 T^2}{2} \left( 1 + \frac{4\zeta}{\omega_n T} \right) \quad (I-8)$$

The undamped natural frequency and damping ratio are related to the 3 dB bandwidth by equation (VIII.2-8) of [ITU-T G.8251]. The damping ratio is related to the gain peaking as described in section 3-5 of [b-Wolaver]. In any case, it may be shown that if the 3 dB bandwidth is 0.1 Hz and the gain peaking is 0.1 dB, then  $K = 0.99948$  and  $\alpha = 3.8761 \times 10^{-2}$ .

It also is shown in [b-Rogers] that, in order for model 2 to be stable, the ratio of sampling rate to 3 dB bandwidth must be approximately greater than  $\pi$  (and, in previous ITU-T study group discussions, some have recommended that a larger ratio, e.g., 10, be used to allow margin). Model 1 has no such restriction because in model 1 the exact analytic form of the system function is used in expressing the solution to the differential equation for the filter (see Eqs. (VIII.2-10) to (VIII.2-16) of Appendix VIII of [ITU-T G.8251]). In any case, stability is not a concern for the cases here because the ratio of sampling rate to 3dB bandwidth is 16 Hz/0.1 Hz = 160.

In the following two examples, the following parameters are used for the sampling rate, 3dB bandwidth of the filter, gain peaking  $H_p$ , and related parameters:

$$f_s = 16 \text{ Hz}$$

$$T = 0.0625 \text{ s}$$

$$f_{Nyquist} = 8 \text{ Hz} \quad (I-9)$$

$$f_{3dB} = 0.1 \text{ Hz}$$

$$H_p = 0.1 \text{ dB}$$

The amplitude of the cosine signal is 1, and is taken to be dimensionless (the actual value of the input amplitude does not matter. What matters is the amplitude of the output of each filter relative to the input).

For each example, as the following is shown (plotted)

- the basic analogue (cosine) signal (note that the analogue signal cannot actually be plotted, as that would entail a continuous plot. However, it is plotted with a sampling frequency much higher than 16 Hz,
- the sampled signal, sampled at frequency  $f_s$ ,
- the DFT of the sampled signal,
- the result of filtering the signal with the model 1 filter, and
- the result of filtering the signal with the model 2 filter.

### Example 1

In this example, the frequency of the cosine signal is 4.0625 Hz, i.e.,  $m = 1$ ,  $n = 4$ , and  $\varepsilon = 0.0625$  Hz. The simulation time is 30 s. This means that approximately 122 cycles of the cosine signal are

simulated. For plotting the analogue signal, the time step in the plot is  $10\ \mu\text{s}$  (i.e.,  $1/6250$  of the sampling interval). Due to limitations in the plotting software, often the full 30 s is not plotted.

Figure I.2 shows the basic analogue input signal (cosine) for 0 to 1 s. The full 30 s is not shown because, with plotting a point every  $1\ \mu\text{s}$ , 30 s of data would exceed the capacity of the plotting software.

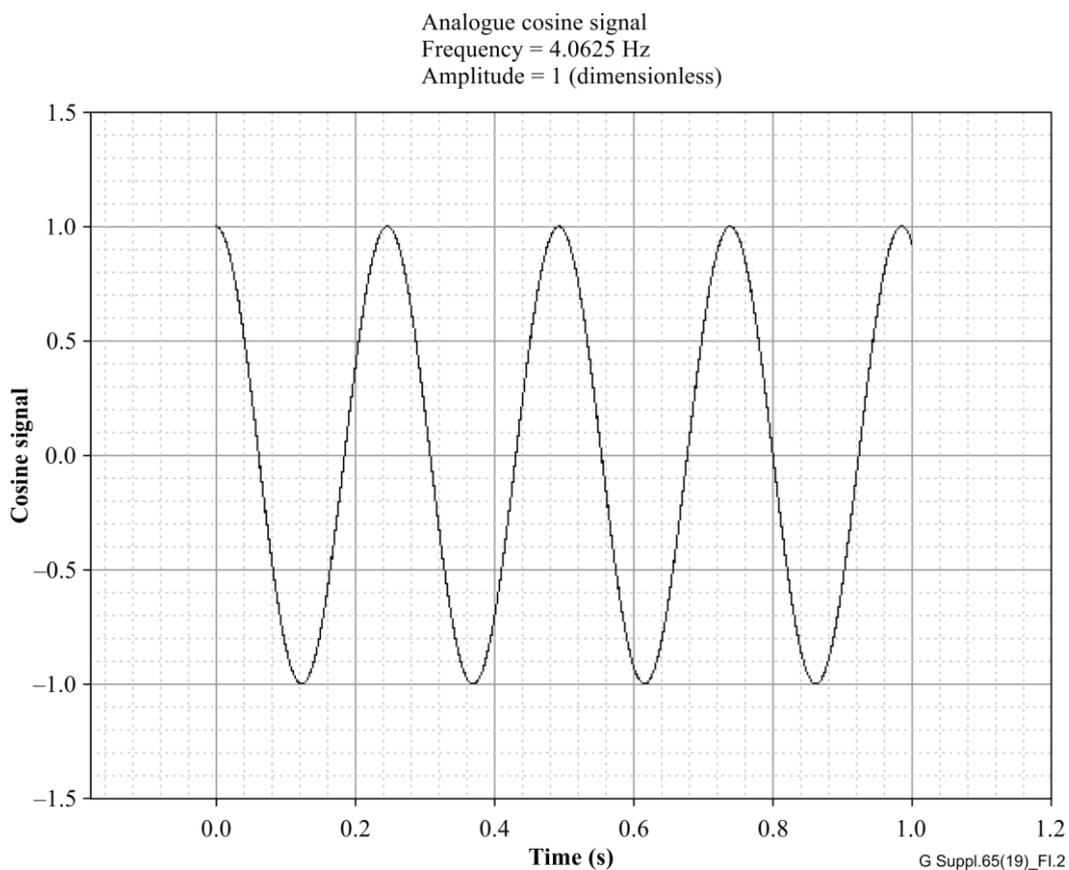
Figure I.3 shows the result of sampling the input at 4.0625 Hz. The modulating envelope is clearly present. Its peak-to-peak is approximately 0.3. Its period is approximately 4 s.

Figure I.4 shows the DFT of the sampled input. As described above (and consistent with [b-Amidor]), only the frequency of the cosine, i.e., 4.0625 Hz, is present. The frequency of the envelope, i.e., 0.25 Hz, is not present.

Figure I.5 shows the result of filtering the sampled input with filter model 1. There is an initial start-up transient that lasts approximately 5 s. After the transient has decayed, the signal looks exactly like the sampled input, except its overall magnitude is reduced by a factor of approximately 50. The amplitude is reduced from approximately 1 to 0.02. The trough of the upper envelope is reduced from approximately 0.7 to approximately 0.012 (a factor of 58.3).

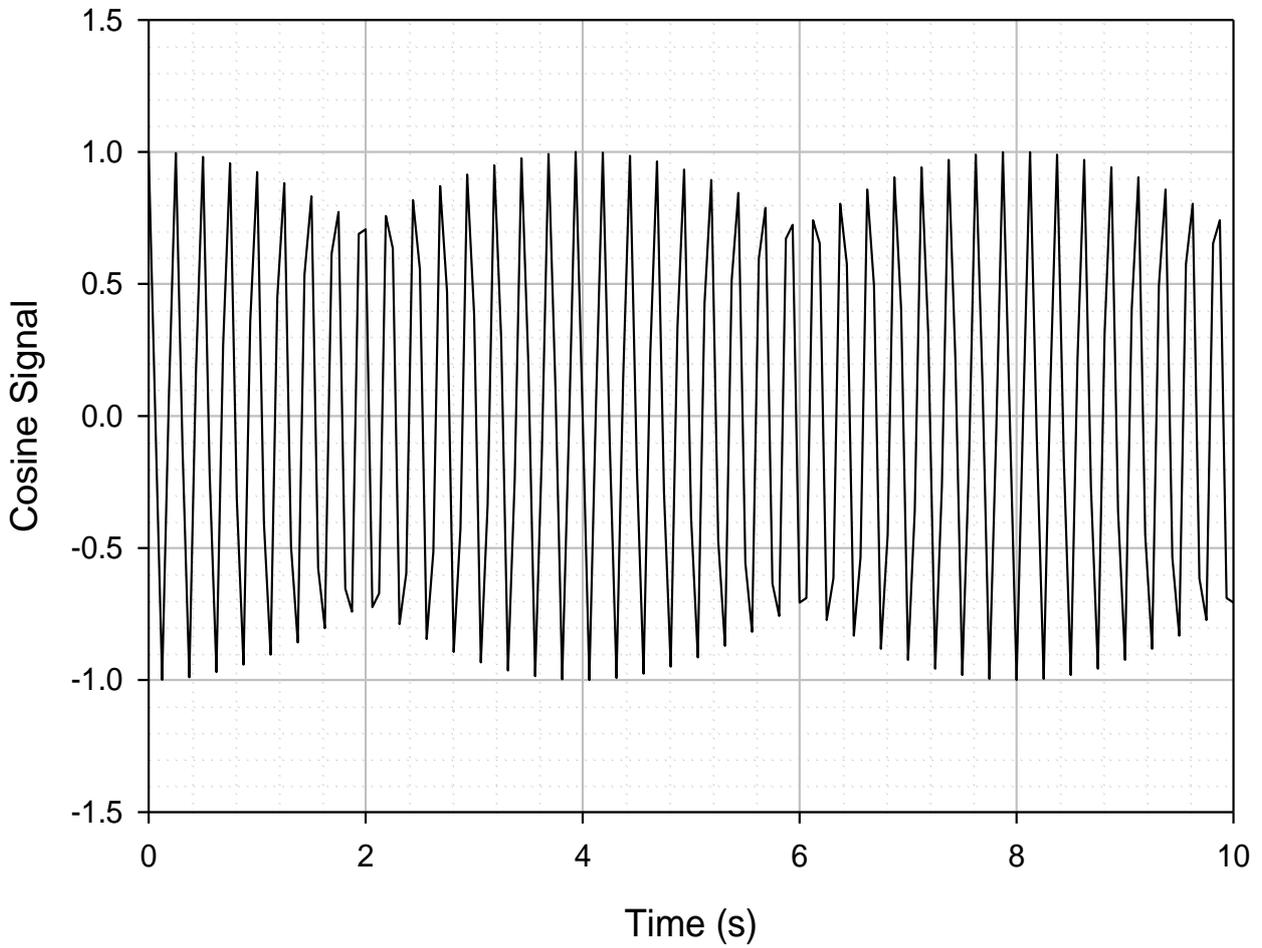
Figure I.6 shows the result of filtering the sampled input with filter model 2. There is an initial transient that lasts approximately 4 s. After the transient has decayed, the signal looks exactly like the sampled input, except its overall magnitude is reduced by a factor of approximately 36. The amplitude is reduced from approximately 1 to 0.028. The trough of the upper envelope is reduced from approximately 0.7 to approximately 0.02 (a factor of 35).

The difference between filter 1 and filter 2 outputs are due to the fact that these are digital approximations to an analogue filter whose bandwidth is 0.1 Hz.



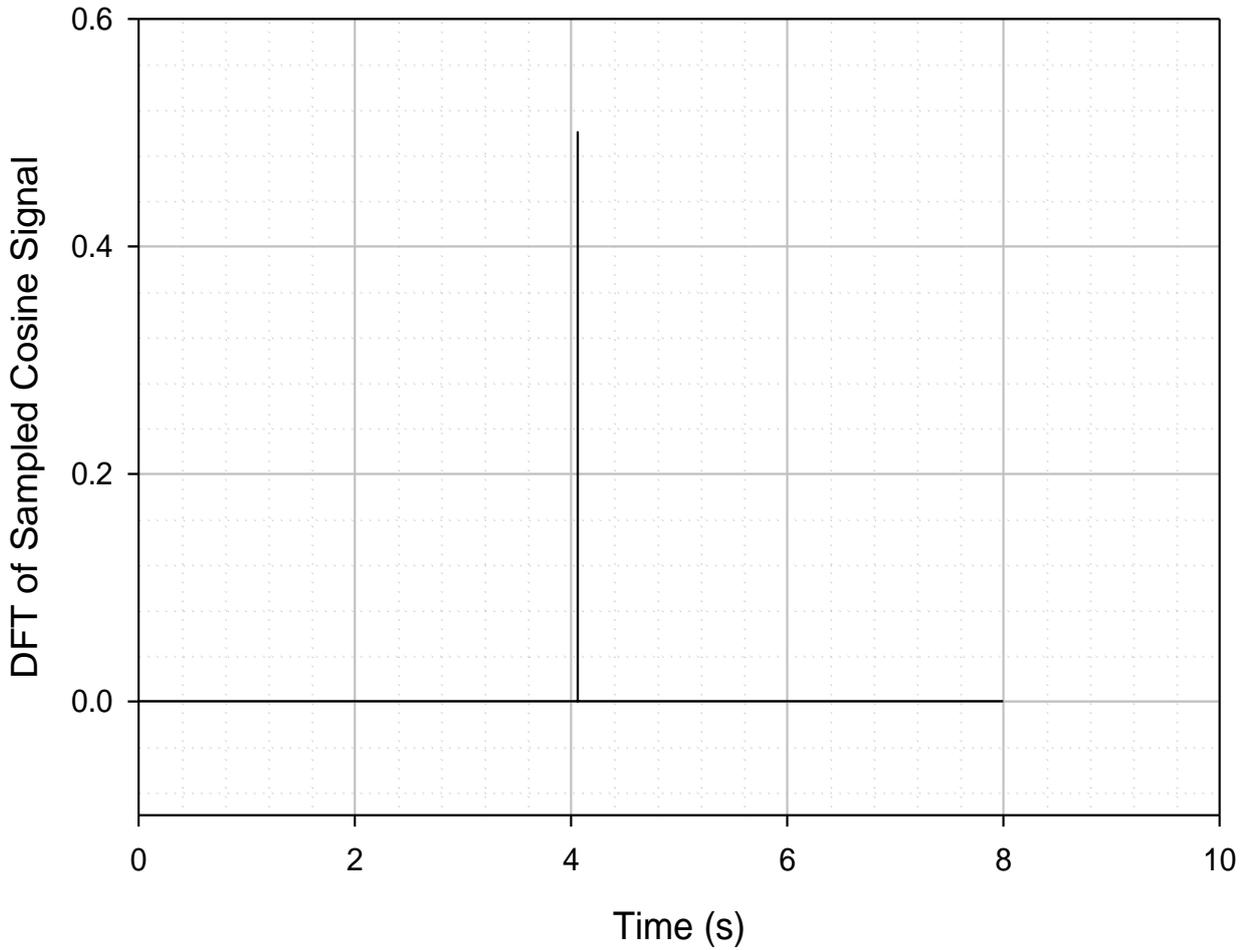
**Figure I.2 – Example 1, analogue signal before sampling**

Sampled cosine signal  
frequency = 4.0625 Hz  
Amplitude = 1 (dimensionless)  
Sampling frequency = 16 Hz



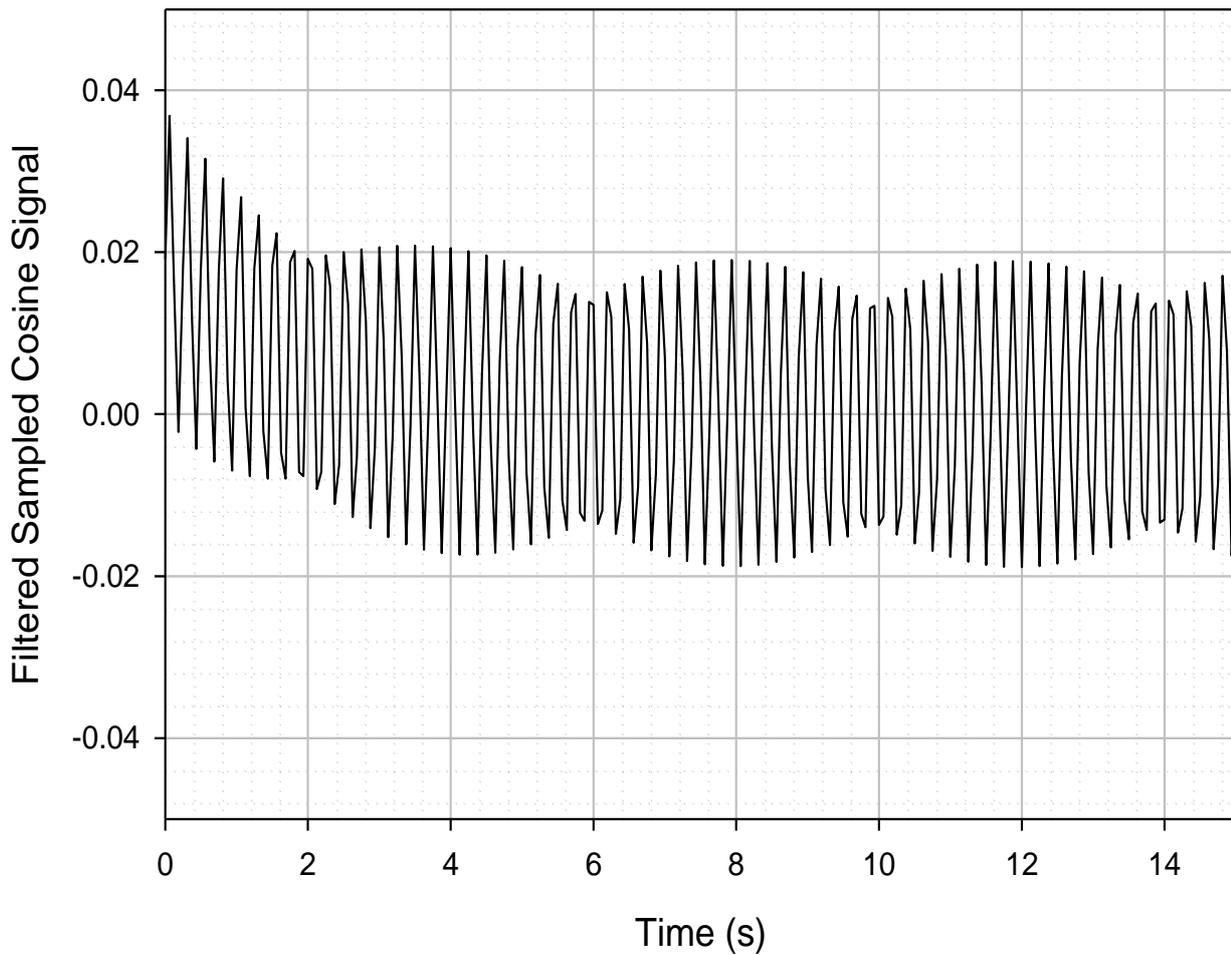
**Figure I.3 – Example 1, input signal after sampling**

Sampled cosine signal  
frequency = 4.0625 Hz  
Amplitude = 1 (dimensionless)  
Sampling frequency = 16 Hz  
Frequency Spectrum (DFT)



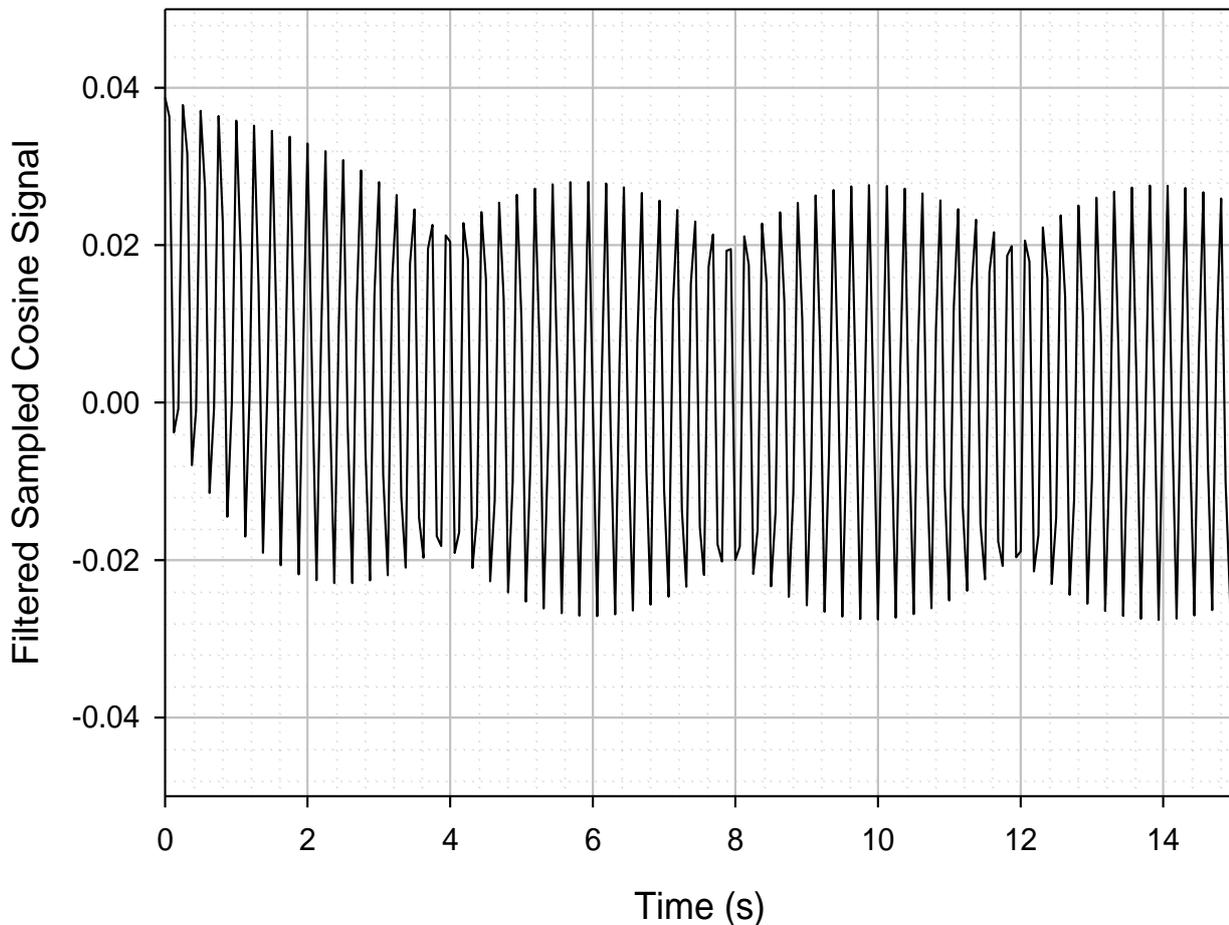
**Figure I.4 – Example 1, DFT of sampled input signal**

Filtered Sampled cosine signal  
frequency = 4.0625 Hz  
Amplitude = 1 (dimensionless)  
Sampling frequency = 16 Hz  
Filter model 1  
Bandwidth = 0.1 Hz  
Gain Peaking = 0.1 dB



**Figure I.5 – Example 1, filtered sampled input signal, filter model 1**

Filtered Sampled cosine signal  
 frequency = 4.0625 Hz  
 Amplitude = 1 (dimensionless)  
 Sampling frequency = 16 Hz  
 Filter model 2  
 Bandwidth = 0.1 Hz  
 Gain Peaking = 0.1 dB



**Figure I.6 – Example 1, filtered sampled input signal, filter model 2**

### Example 2

In this example, the frequency of the cosine signal is 4.001 Hz, i.e.,  $m = 1$ ,  $n = 4$ , and  $\varepsilon = 0.001$  Hz. The simulation time is 300 s. This means that approximately 1200 cycles of the cosine signal are simulated. For plotting the analogue signal, the time step in the plot is 100  $\mu$ s (i.e., 1/625 of the sampling interval). Due to limitations in the plotting software, often the full 300 s is not plotted.

Figure I.7 shows the basic analogue input signal (cosine) for 0 to 10 s. The full 300 s is not shown because, with plotting a point every 10  $\mu$ s, 30 s of data would exceed the capacity of the plotting software (similar to example 1).

Figure I.8 shows the result of sampling the input at 4.001 Hz. The modulating envelope is clearly present. Its peak-to-peak is approximately 0.3 as in example 1. The individual cycles of the sampled signal are not visible on the scale of the plot (the sampled signal could have been plotted over a shorter

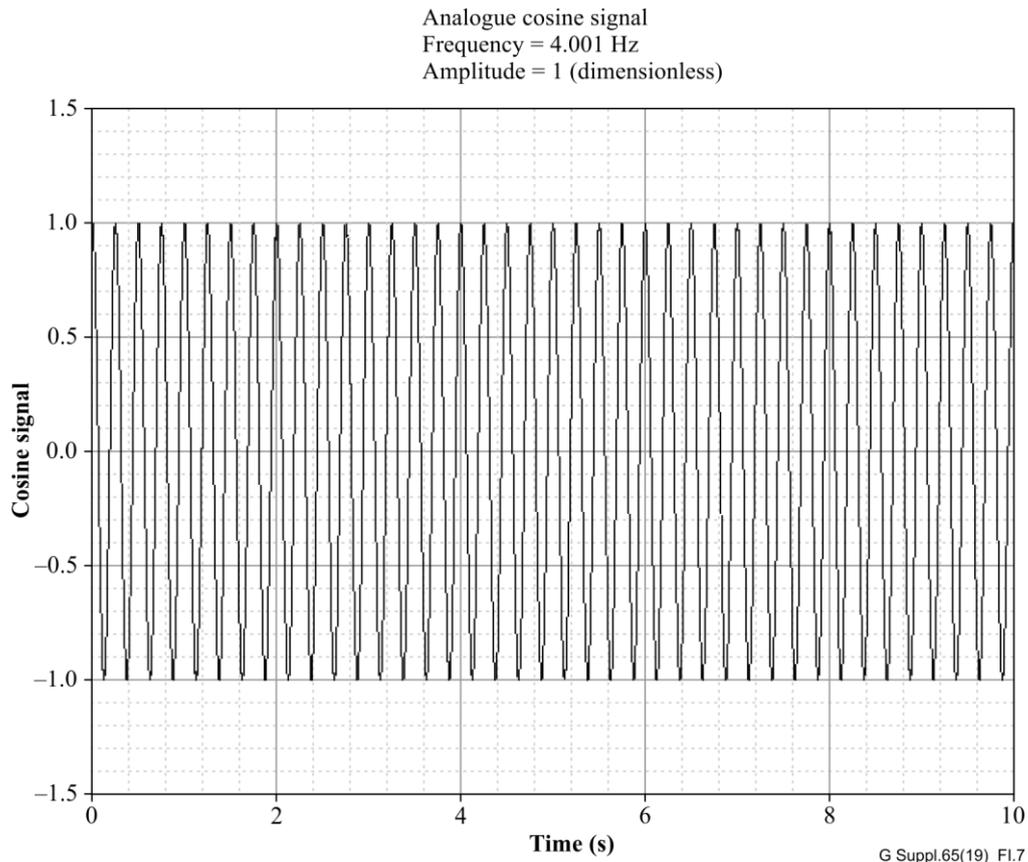
time so that the individual cycles would be seen. However, then the modulating envelope would not be visible). Nevertheless, its period is approximately 250 s, i.e. much longer than in example 1. This is due to the fact that  $\epsilon$  is much smaller than in example 1 (i.e., 0.001 Hz here, versus 0.0625 Hz in example 1). In fact, the period here is longer than the period in example 1 by a factor of  $0.0625 \text{ Hz} / 0.001 \text{ Hz} = 62.5$  (i.e.,  $250 \text{ s} / 4 \text{ s}$ ).

Figure I.9 shows the DFT of the sampled input. As described above (and consistent with [b-Amidror]), only the frequency of the cosine, i.e., 4.001 Hz, is present. The frequency of the envelope, i.e., 4 MHz, is not present.

Figure I.10 shows the result of filtering the sampled input with filter model 1. There is an initial start-up transient that lasts approximately 10 s. After the transient has decayed, the signal looks exactly like the sampled input, except its overall magnitude is reduced by a factor of approximately 50, as in example 1. The amplitude is reduced from approximately 1 to 0.02. The trough of the upper envelope is reduced from approximately 0.7 to approximately 0.012 (a factor of 58.3).

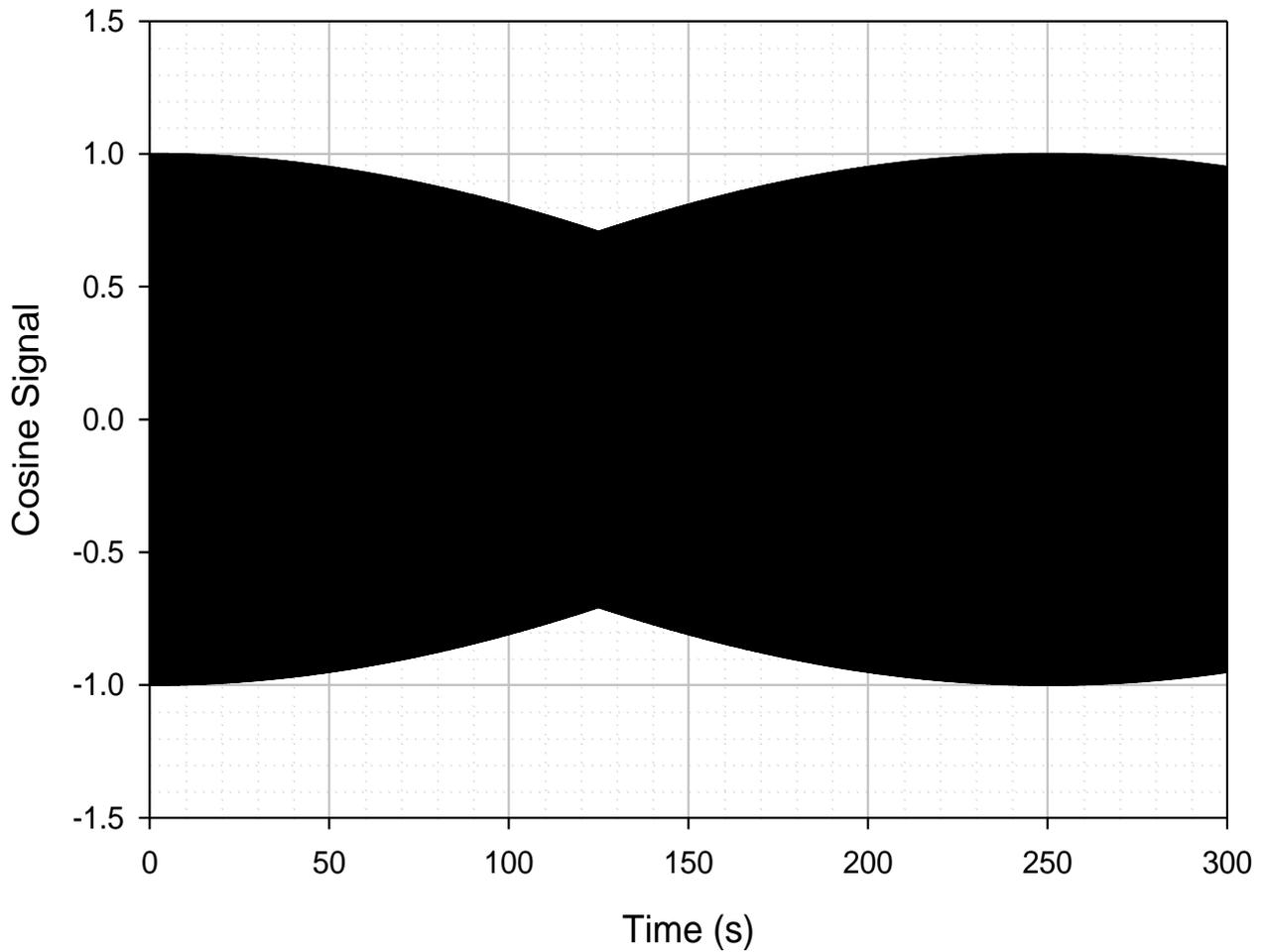
Figure I.11 shows the result of filtering the sampled input with filter model 2. There is an initial transient that lasts approximately 5-10 s. After the transient has decayed, the signal looks exactly like the sampled input, except its overall magnitude is reduced by a factor of approximately 36. The amplitude is reduced from approximately 1 to 0.028. The trough of the upper envelope is reduced from approximately 0.7 to approximately 0.02 (a factor of 35).

As in example 1, the difference between filter 1 and filter 2 output are due to the fact that these are digital approximations to an analogue filter whose bandwidth is 0.1 Hz.



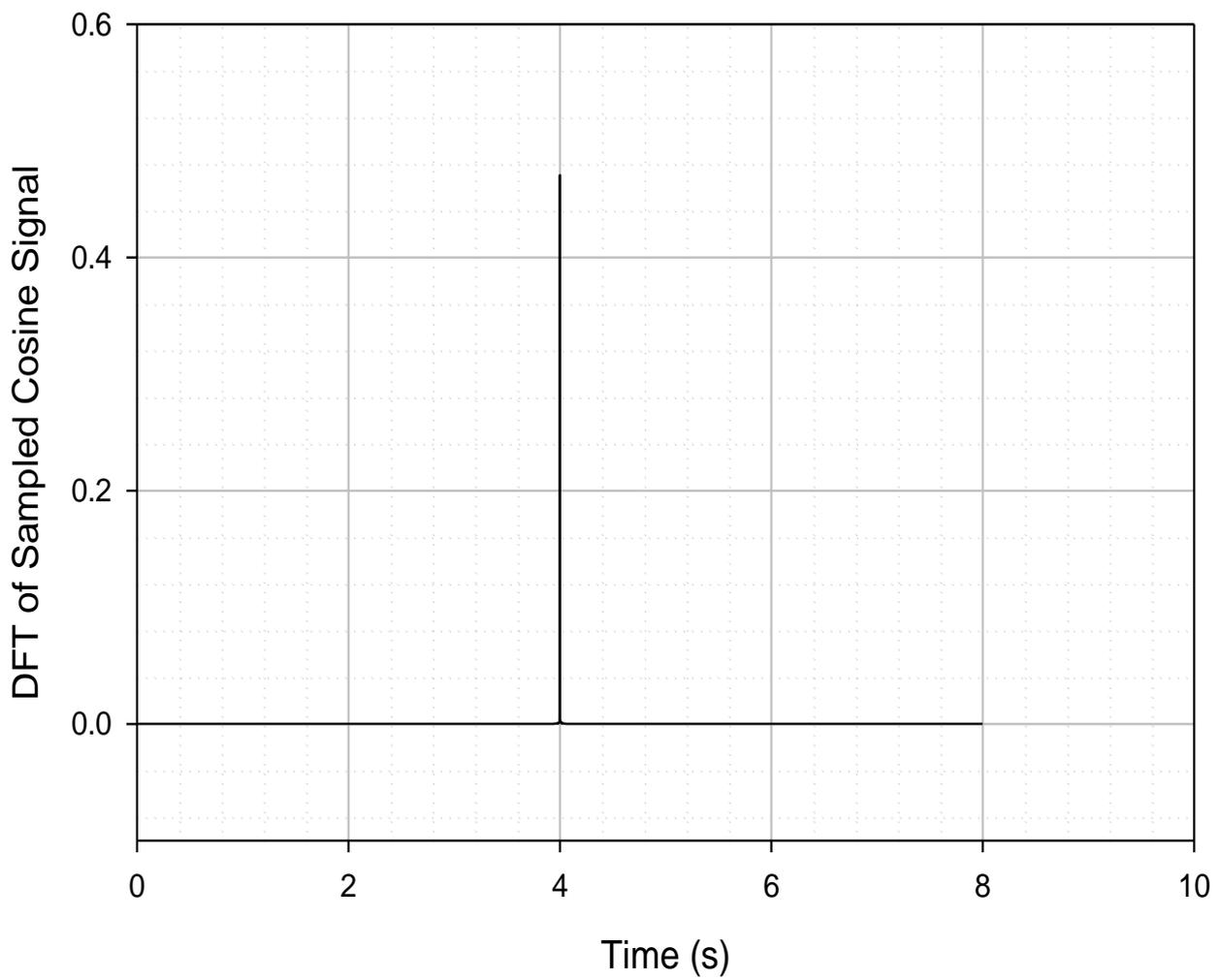
**Figure I.7 – Example 2, analogue signal before sampling**

Sampled cosine signal  
frequency = 4.001 Hz  
Amplitude = 1 (dimensionless)  
Sampling frequency = 16 Hz



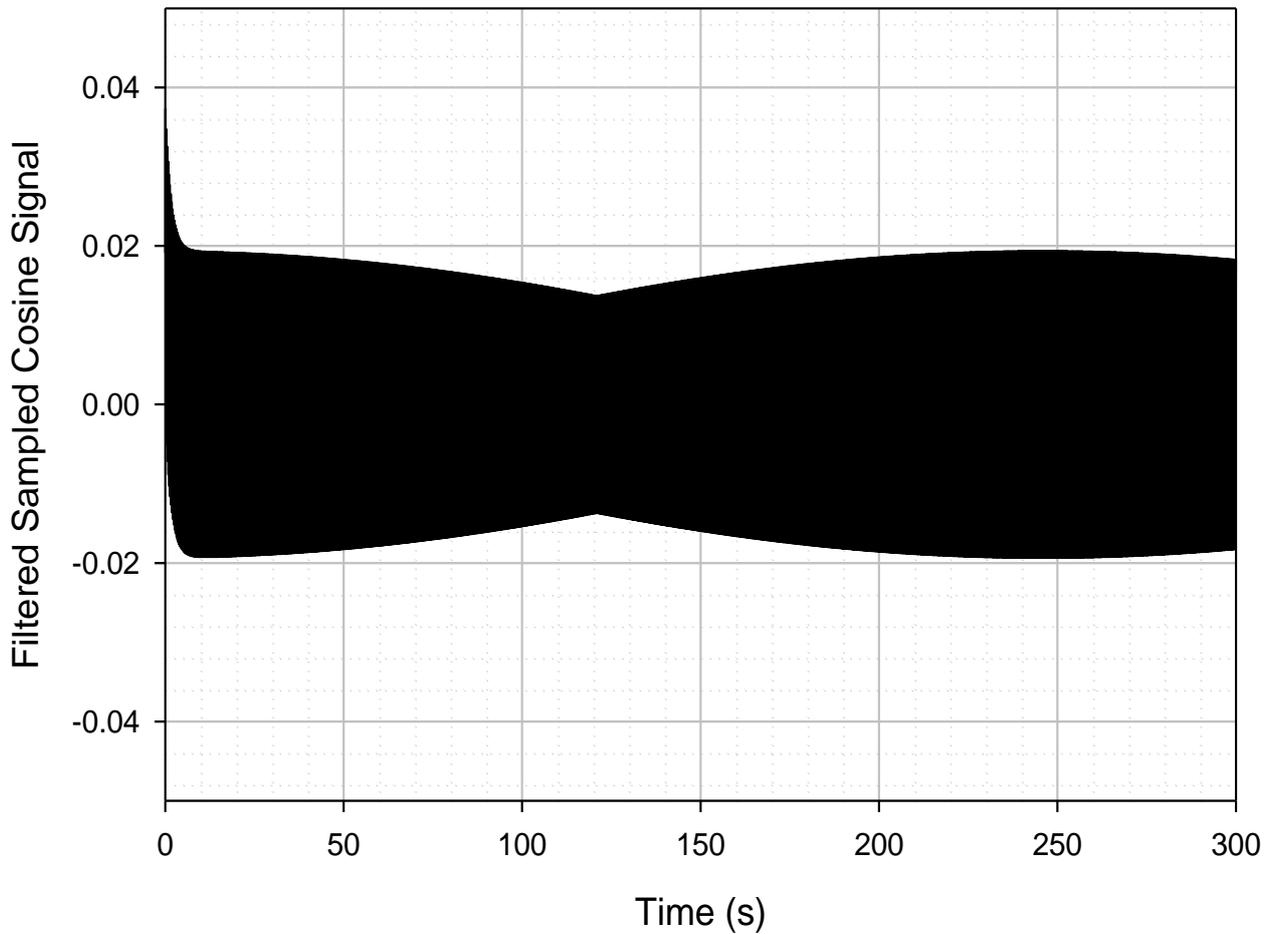
**Figure I.8 – Example 2, input signal after sampling**

Sampled cosine signal  
frequency = 4.001 Hz  
Amplitude = 1 (dimensionless)  
Sampling frequency = 16 Hz  
Frequency Spectrum (DFT)



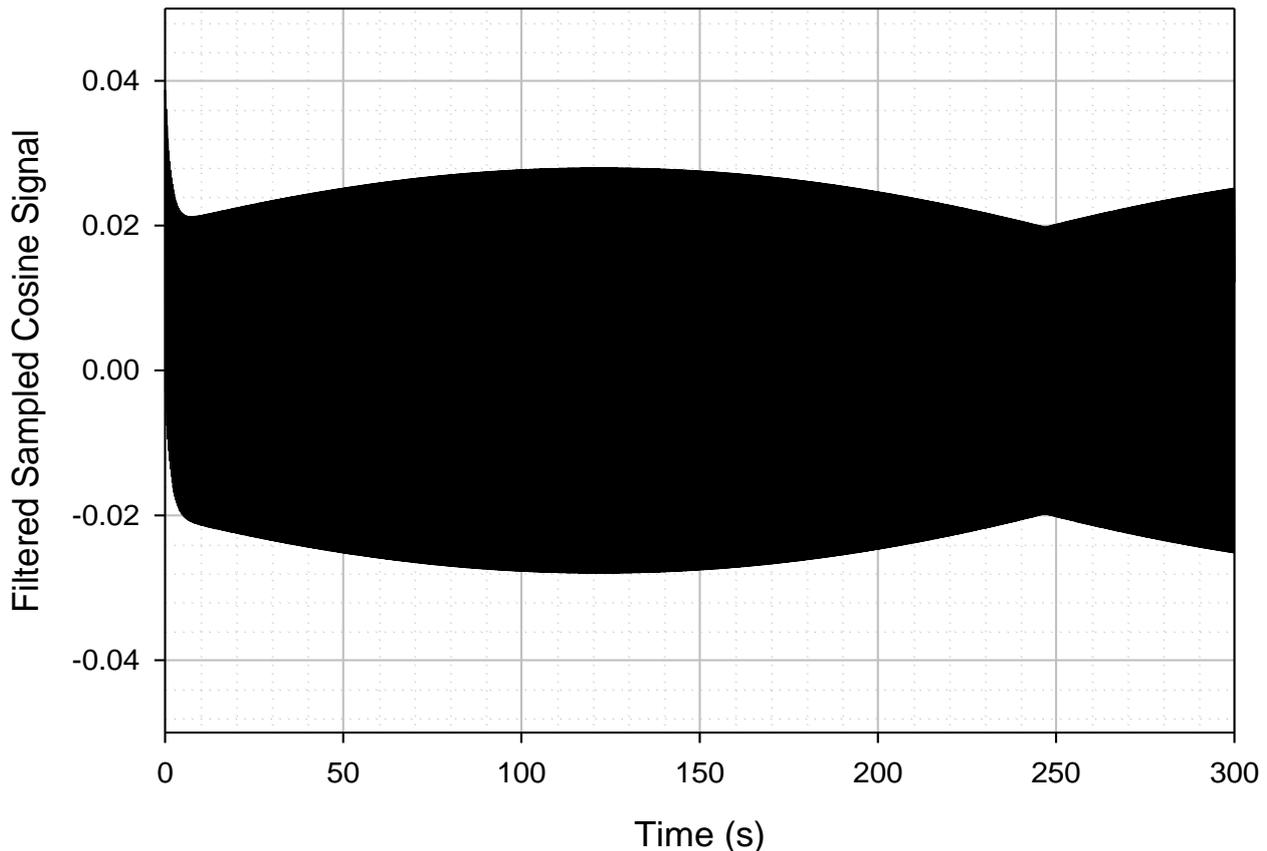
**Figure I.9 – Example 2, DFT of sampled input signal**

Filtered Sampled cosine signal  
frequency = 4.001 Hz  
Amplitude = 1 (dimensionless)  
Sampling frequency = 16 Hz  
Filter model 1  
Bandwidth = 0.1 Hz  
Gain Peaking = 0.1 dB



**Figure I.10 – Example 2, filtered sampled input signal, filter model 1**

Filtered Sampled cosine signal  
frequency = 4.001 Hz  
Amplitude = 1 (dimensionless)  
Sampling frequency = 16 Hz  
Filter model 2  
Bandwidth = 0.1 Hz  
Gain Peaking = 0.1 dB



**Figure I.11 – Example 2, filtered sampled input signal, filter model 2**

### I.3 Discussion

The results here are consistent with [b-Amidoror]. Note that [b-Amidoror] does not show examples of filtering the sampled signal that contains the sub-Nyquist artefacts with a low-pass filter. The low-pass filter does not remove the artefacts. However, the entire signal is reduced in amplitude by an amount that a signal of the same frequency, but without the artefacts, would be expected to be reduced.

Note that both filter models used here are linear. It is indicated in [b-Amidoror] that, if a sampled signal that has the artefacts undergoes any nonlinear processing, the artefacts could remain (even though the basic signal frequency would be reduced) with the frequency of the artefacts present in the DFT. This might happen if, for example, a nonlinear filter was used.

Note [b-Amidoror] also gives examples for cases where  $m > 1$ . In these cases, the sampled signal can have multiple envelopes.

## Bibliography

- [b-Abramowitz] Milton Abramowitz and Irene A. Stegun, *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*, National Bureau of Standards, Applied Mathematics Series, June 1964.
- [b-Amidror] Isaac Amidror, *Sub-Nyquist Artefacts and Sampling Moiré Effects*, Royal Society Open Science, 2015 March, 2(3): 140550, published online March 18, 2015, freely available at <http://rsos.royalsocietypublishing.org/content/2/3/140550>.
- [b-Anritsu] CIAJ (Anritsu), MTIE Upper Limit for TDEV Wander Noise, Contribution to ITU-T SG 4, Q5, February 5-14, 2003, WD17.
- [b-ANSI01] ANSI T1.105.09-1999, *American National Standard for Telecommunications – Synchronous Optical Network (SONET) Network Element Timing and Synchronization*, ATIS, 1996.
- [b-ANSI02] ANSI T1.101-1999, *American National Standard for Telecommunications – Synchronization Interface Standard*, ATIS, 1999.
- [b-Barnes01] J.A. Barnes and Stephen Jarvis, Jr. (1971), *Efficient Numerical and Analog Modeling of Flicker Noise Processes*, National Bureau of Standards, NBS Technical Note 604, June.
- [b-Barnes02] James. A. Barnes and Charles A. Greenhall (1987), *Large Sample Simulation of Flicker Noise*, 19<sup>th</sup> Annual Precise Time and Time Interval (PTTI) Applications Planning Meeting, December.
- [b-Bregni] Stefano Bregni (2002), *Synchronization of Digital Telecommunications Networks*, Wiley.
- [b-Eidson] John C. Eidson (2006), *Measurement, Control and Communication Using IEEE 1588*, Springer.
- [b-ETSI01] EN 300 462-4-1 v1.1.1 (1998-05), *Transmission and Multiplexing (TM); Generic requirements for synchronization networks; Part 4-1: Timing characteristics of slave clocks suitable for synchronization supply to Synchronous Digital Hierarchy (SDH) and Plesiochronous Digital Hierarchy (PDH) equipment*, ETSI.
- [b-ETSI02] EN 300 462-5-1 v1.1.2 (1998-05), *Transmission and Multiplexing (TM); Generic requirements for synchronization networks; Part 5-1: Timing characteristics of slave clocks suitable for operation in Synchronous Digital Hierarchy (SDH) equipment*, ETSI.
- [b-Franklin] Gene F. Franklin, J. David Powell and Michael Workman (1998), *Digital Control of Dynamic Systems*, Third Edition, Addison-Wesley.
- [b-Gardner] Floyd M. Gardner (1979), *Phase-lock Techniques*, 2nd Edition, Wiley.
- [b-Garner01] Geoffrey Garner (1995), *Noise accumulation in synchronization reference chains*, AT&T Network Systems contribution to T1 Standards Project, T1X1.3/95-044, April 4.
- [b-Garner02] Geoffrey Garner (1995), *Noise accumulation in synchronization reference chains based on wander generation models that meet MTIE and TDEV masks*, AT&T Network Systems contribution to T1 Standards Project, T1X1.3/95-051, April 4.

- [b-Garner03] Geoffrey M. Garner, Michel Ouellette, Lv Jingfei and Sebastien Jobert (2011), *Initial Simulation Results for Transport of Time using PTP and Frequency using SyncE, over the HRM3 chain of Boundary Clocks*, Huawei and France Telecom contribution to ITU-T Q13/15, Indianapolis, May 2-6, WD25.
- [b-Garner04] Geoffrey M. Garner, Lv Jingfei, Sebastien Jobert and Michel Ouellette (2011), *Initial Simulation Results for Transport of Time over the HRM2b chain of Boundary Clocks*, Huawei, France Telecom Orange, and Iometrix contribution to ITU-T SG15, Q13, Geneva, December, Com 15–C1729-E.
- [b-Law] Averill M. Law and W. David Kelton (1991), *Simulation Modeling & Analysis*, Second Edition, McGraw-Hill.
- [b-Ogata] Katsuhiko Ogata (1995), *Discrete-Time Control Systems*, Second Edition, Prentice-Hall.
- [b-Oppenheim] Alan V. Oppenheim and Ronald W. Schafer (1975), *Digital Signal Processing*, Prentice-Hall, pp. 26-30.
- [b-Papoulis] Athanasios Papoulis (1991), *Probability, Random Variables, and Stochastic Processes*, Third Edition, McGraw-Hill.
- [b-Rogers] John Rogers, Calvin Plett and Foster Dai (2006), *Integrated Circuit Design for High-Speed Frequency Synthesis*, Artech House.
- [b-Shenoi01] Kishan Shenoi (2009), *Synchronization and Timing in Telecommunications*, BookSurge Publishing.
- [b-Shenoi02] Chip Webb and Kishan Shenoi (2011), *Study of maximum absolute time error accumulation in a chain of Boundary Clocks with Physical Layer frequency assist*, Anue contribution to ITU-T Q13/15, York, 26-30 September, WD25.
- [b-Wolaver] Dan H. Wolaver (1991), *Phase-Locked Loop Circuit Design*, Prentice-Hall.





## SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series D	Tariff and accounting principles and international telecommunication/ICT economic and policy issues
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
<b>Series G</b>	<b>Transmission systems and media, digital systems and networks</b>
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	Cable networks and transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Environment and ICTs, climate change, e-waste, energy efficiency; construction, installation and protection of cables and other elements of outside plant
Series M	Telecommunication management, including TMN and network maintenance
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	Switching and signalling, and associated measurements and tests
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	Data networks, open system communications and security
Series Y	Global information infrastructure, Internet protocol aspects, next-generation networks, Internet of Things and smart cities
Series Z	Languages and general software aspects for telecommunication systems