ETSI TR 126 973 V15.2.0 (2019-03)

TECHNICAL REPORT

5G;
Update to fixed-point basic operators
(3GPP TR 26.973 version 15.2.0 Release 15)

Reference
RTR/TSGS-0426973vf20

Keywords
5G

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

iTeh STANDARD PREVIEW
(standards.iteh.ai)
Full standard:
https://standards.iteh.ai/catalog/standards/sist/1c3586eb-8408-4478-9be2-1f7204169070/etsi-tr-126-973-v15.2.0-2019-03

Important notice

The present document can be downloaded from:
http://www.etsi.org/standards-search

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the prevailing version of an ETSI deliverable is the one made publicly available in PDF format at www.etsi.org/deliver.

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at
https://portal.etsi.org/TB/ETSIDeliverableStatus.aspx

If you find errors in the present document, please send your comment to one of the following services:
https://portal.etsi.org/People/CommiteeSupportStaff.aspx

Copyright Notification

No part may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm except as authorized by written permission of ETSI.
The content of the PDF version shall not be modified without the written authorization of ETSI.
The copyright and the foregoing restriction extend to reproduction in all media.

© ETSI 2019.
All rights reserved.

DECTTM, **PLUGTESTS**TM, **UMTS**TM and the ETSI logo are trademarks of ETSI registered for the benefit of its Members.
3GPPTM and **LTE**TM are trademarks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners.
oneM2M™ logo is a trademark of ETSI registered for the benefit of its Members and of the oneM2M Partners.
GSM® and the GSM logo are trademarks registered and owned by the GSM Association.

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (https://ipr.etsi.org/).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

Foreword

This Technical Report (TR) has been produced by ETSI 3rd Generation Partnership Project (3GPP).

The present document may refer to technical specifications or reports using their 3GPP identities, UMTS identities or GSM identities. These should be interpreted as being references to the corresponding ETSI deliverables.

The cross reference between GSM, UMTS, 3GPP and ETSI identities can be found under http://webapp.etsi.org/key/queryform.asp.

Modal verbs terminology

In the present document "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

Contents

Foreword

This Technical Report has been produced by the 3rd Generation Partnership Project (3GPP).

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of the present document, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

- x the first digit:
  - 1 presented to TSG for information;
  - 2 presented to TSG for approval;
  - 3 or greater indicates TSG approved document under change control.
- y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.
- z the third digit is incremented when editorial only changes have been incorporated in the document.

Introduction

The last major update to the ITU-T Basic Operators [6] was in 2005, with a follow on update in 2009. These basic operators serve as a foundation for reference software of codecs specified by 3GPP. During the last several years, processors with wide accumulators, and support for single-instruction-multiple-data (SIMD), and very long instruction word (VLIW) features have become prevalent. The basic operators of 2009 now need to be extended to leverage these capabilities of modern processors so that implementations with lower mega-cycles-per-second (MCPS) and lower-power may be realized.

Enhanced Voice Services (EVS) is one of the recent codecs defined by 3GPP that can leverage these features of modern processors. The existing EVS reference software would have to be appropriately modified to leverage these extended basic operators without changing the underlying algorithm. This is referred to as an alternative EVS implementation using the extended basic operators.

This alternative EVS implementation would have to be evaluated to ensure that inter-operability is maintained in addition to ensuring that voice quality is not impacted.

1 Scope

The present document covers the following topics:

1) Assessment of the gaps between modern processors and the existing set of basic operators (STL2009) [6].
2) Proposal of an extended set of operators addressing modern DSP architectures as an extension to STL2009.
3) Assessment of merits of an alternative EVS implementation using extended STL2009 Basic Operators.
4) Proposal for validation of an alternative EVS implementation using extended STL2009 Basic Operators.

2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document*.

[1] 3GPP TR 21.905: "Vocabulary for 3GPP Specifications".

[2] 3GPP TS 26.442: "Codec for Enhanced Voice Services (EVS); ANSI C code (fixed-point)".

[3] Recommendation ITU-T P.800 (08/1996): "Methods for subjective determination of transmission quality".

[4] Recommendation ITU-T P.863 (09/2014): "Perceptual objective listening quality assessment".

[5] 3GPP TS 26.443: "Codec for Enhanced Voice Services (EVS); ANSI C code (floating-point)".

[6] Recommendation ITU-T G.191 (03/10): "Software tools for speech and audio coding standardization".

[7] 3GPP TR 26.952: "Codec for Enhanced Voice Services (EVS); Performance Characterization (Release 14)".

3 Abbreviations

For the purposes of the present document, the abbreviations given in 3GPP TR 21.905 [1] and the following apply. An abbreviation defined in the present document takes precedence over the definition of the same abbreviation, if any, in 3GPP TR 21.905 [1].

SIMD	Single Instruction Multiple Data
STL	Software tools for speech and audio coding standardization
VLIW	Very Long Instruction Word.

4 Extension to the STL2009 Basic Operators

4.1 Analysis of the gap between current basic operators and modern DSP architectures

State-of-the-art processor architectures, such as the recent ones from Intel, ARM, QUALCOMM, Texas Instruments etc., support wide accumulators, SIMD and VLIW capabilities. The last major update to the ITU-T Basic Operators was in 2005, with a follow on update in 2009 [6]. It appears that these earlier versions of the Basic Operators (2009 and earlier) were influenced by older DSP architectures such as the Texas Instruments TMS320C5x and TMS320C54x processors where the accumulator was 40 bits wide.

However, a survey of the state-of-the-art processor architectures shows that most of them support the following capabilities:

- Wider (64 bit) accumulators and registers.
- Wider accumulators enable additional guard bits which eliminate the need for checking for saturation after every basic operation.
- SIMD (Single Instruction Multiple Data) instructions which can process vector data. For example, a single instruction can process two 32-bit data elements or four 16-bit elements in parallel.
- VLIW (Very Long Instruction Word) enables several operations to be executed in parallel in a single cycle.

Basic operators that are friendlier to compilers, and enable SIMD and VLIW features to be leveraged, can significantly reduce implementation time. Improved compiler technology and software development tools interpret data types and associated basic operators to map them to a processor architecture for better Out-of-box (OOB) performance. Without this computer assisted optimization, an engineer would have to hand-optimize the code which would result in increased engineering effort and longer time to market.

Many recent audio/hybrid codecs make extensive use of 16bit x 32bit MAC (multiply and accumulate) and 32bit x 32bit MAC operations which are realized quite differently between VLIW and SIMD architectures and the current Basic Operators:

- Current STL2009 Basic operators require saturation and truncation after every multiply-accumulate (MAC) operation to maintain bit-exactness.
- The current Basic operator saturation checks prevent use of SIMD parallelism.
- To maintain bit-exactness, cycles are wasted resulting in higher MCPS and power on VLIW and SIMD capable devices.
- Higher precision variables, such as 64bit operands, are partitioned into smaller width operands, processed and then put back to the original width. This results in an overhead and processor cycles are wasted.

Considering the capabilities of modern processor architectures, as well as the characteristics of the latest speech and audio codecs, there is a need for extending STL2009 with additional basic operators & data types to better leverage the capabilities of state-of-the-art processor architectures and characteristics of DSP algorithms.

4.2 Test methodology for validating the extended basic operators

4.2.0 General

This clause describes a test framework that will compare the fixed-point arithmetic accuracy of the extended basic operators against a floating-point implementation of the extended basic operators. Each basic operator will be tested for 4 different data patterns.

In table 1 below, the extended basic operators have been classified into four main classes. The test patterns used for testing and the build options of the test framework are also shown below.

Table 1: Classification of the extended basic operators

Test framework for extended basic operators			
Main class	**Subclass**	**Total basops**	**Covered basops**
64 bit accumulator	64-bit Integer Mac	4	4
	64-bit Mac	7	7
	64-bit Math	12	12
	64-bit scale	7	7
	64-bit move	5	0
Complex	Complex Math	7	7
	Complex Mac	9	9
	Complex Move	10	0
	Complex Scale	9	9
Enhanced 32 bit	32*16 bit Enh MAC	6	6
	32*32 bit Enh MAC	6	6
Control code ops		18	0
Total		100	67

Test data patterns:

- -1.0 to 1.0 float range with configurable interval.
- Random numbers.
- Special values: very low level values (e.g., in the range of 1e-3, 1e-6 etc.), nominal and large values
- Custom mode: users can specify their customized array of size N.

Build options:

MSVC 2017 and MSVC 2013 workspaces are provided, with 2 options:

- MSVC 2017/2013 project.
- Gcc based makefiles.

4.2.1 Test methodology

In Figure 1 below, a block diagram explains how to validate the extended STL2009 Basic Operators implementation against a reference floating-point implementation. A data generator generates floating-point notation data values that are then converted into fixed-point notation and these are input to the design under test (DUT) implementation of the extended STL2009 Basic Operators implementation. The same fixed-point data is converted into floating-point notation, and then input to a reference floating-point implementation of the extended STL2009 Basic Operators. The fixed-point output of the DUT is converted to floating-point notation, and then compared against the reference floating-point implementation output and an error value is generated and logged.

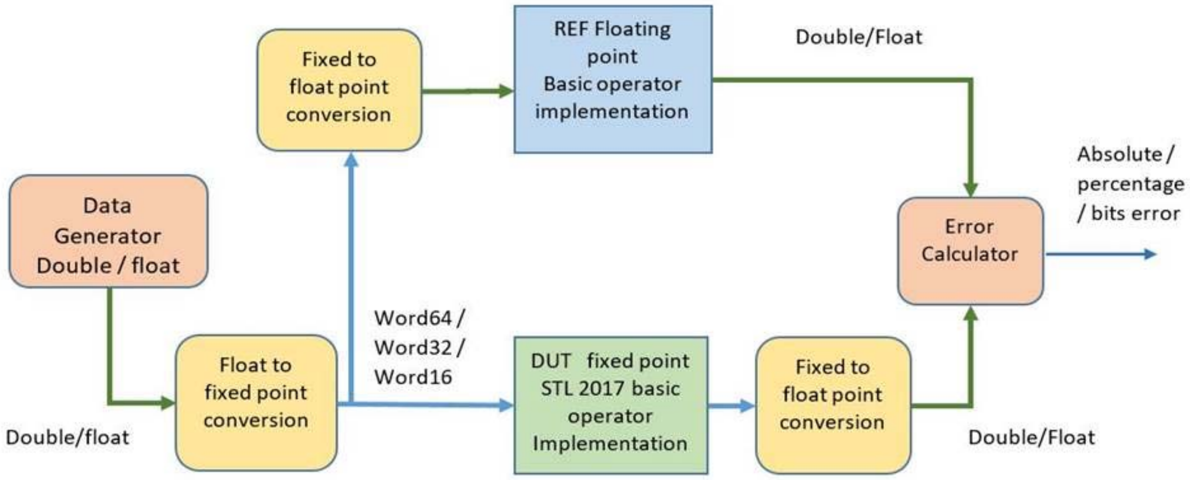

Figure 1: Block diagram illustrating how the fixed-point implementation is validated against a floating-point reference implementation of the extended STL2009 basic operators

In the following clauses, the test results for an example basic operator, Mpy_32_16_1 are reported.

4.2.2 Test results for basic operator Mpy_32_16_1

The setup in figure 1 was used for testing with four different types of data:

1) Random input numbers
2) A sweep from a negative number to a positive number
3) A piecewise sweep from a negative number to a positive number
4) A custom input where a user can specify an array of size N with custom inputs

Figures 2, 3, 4 and 5 illustrate the results of the test for the above four different data types. The error between the fixed-point implementation and floating-point implementation are extremely small thereby validating the fixed-point implementation.

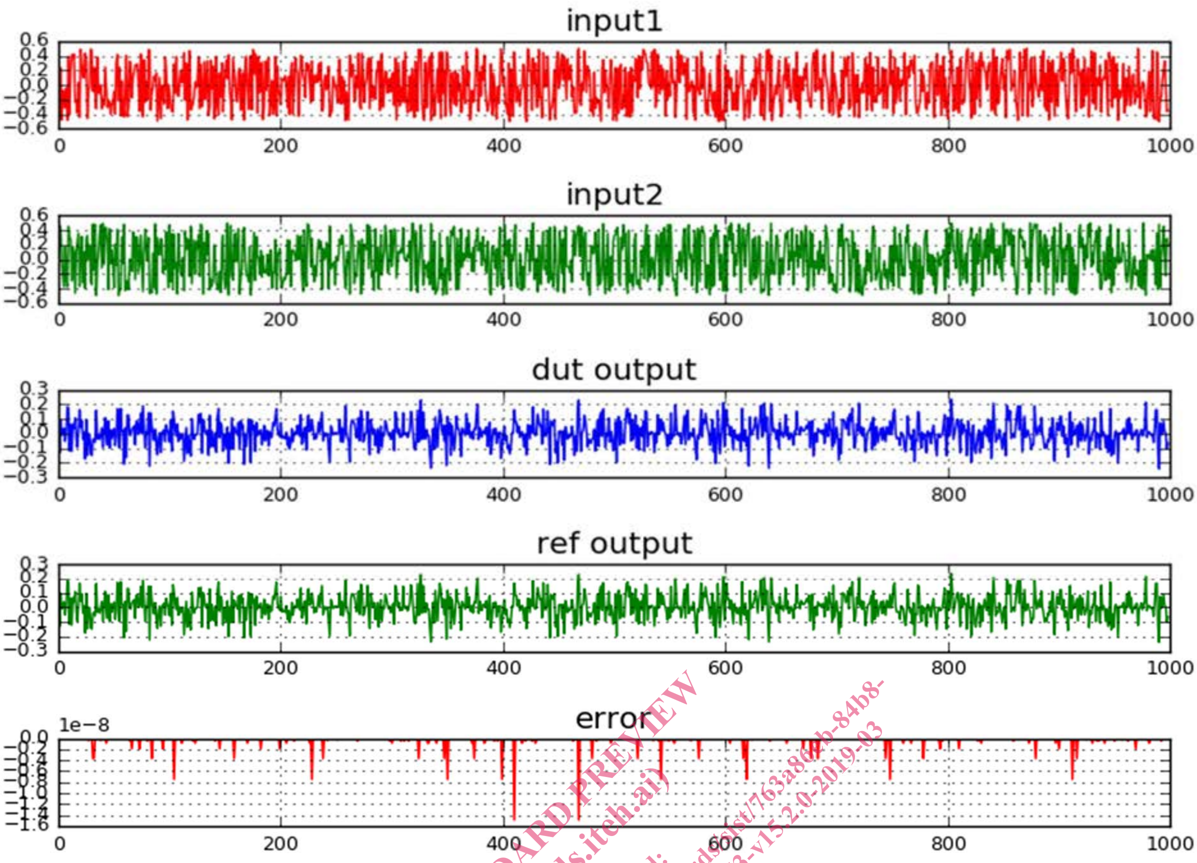

Figure 2: Test results for basic operator Mpy_32_16_1 using random input. The error between the fixed-point output and floating-point output is very small.

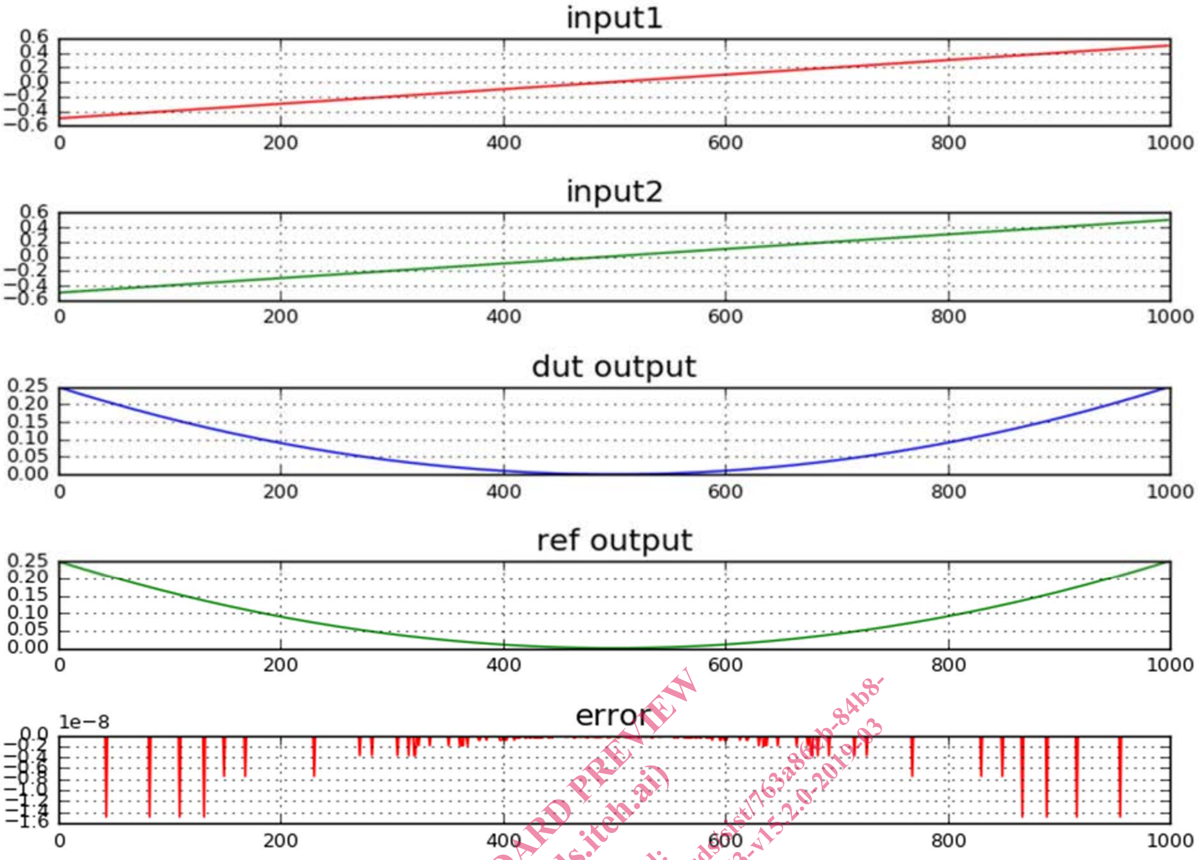

Figure 3: Test results for basic operator Mpy_32_16_1 using a sweep input. The error between the fixed-point output and floating-point output is very small.

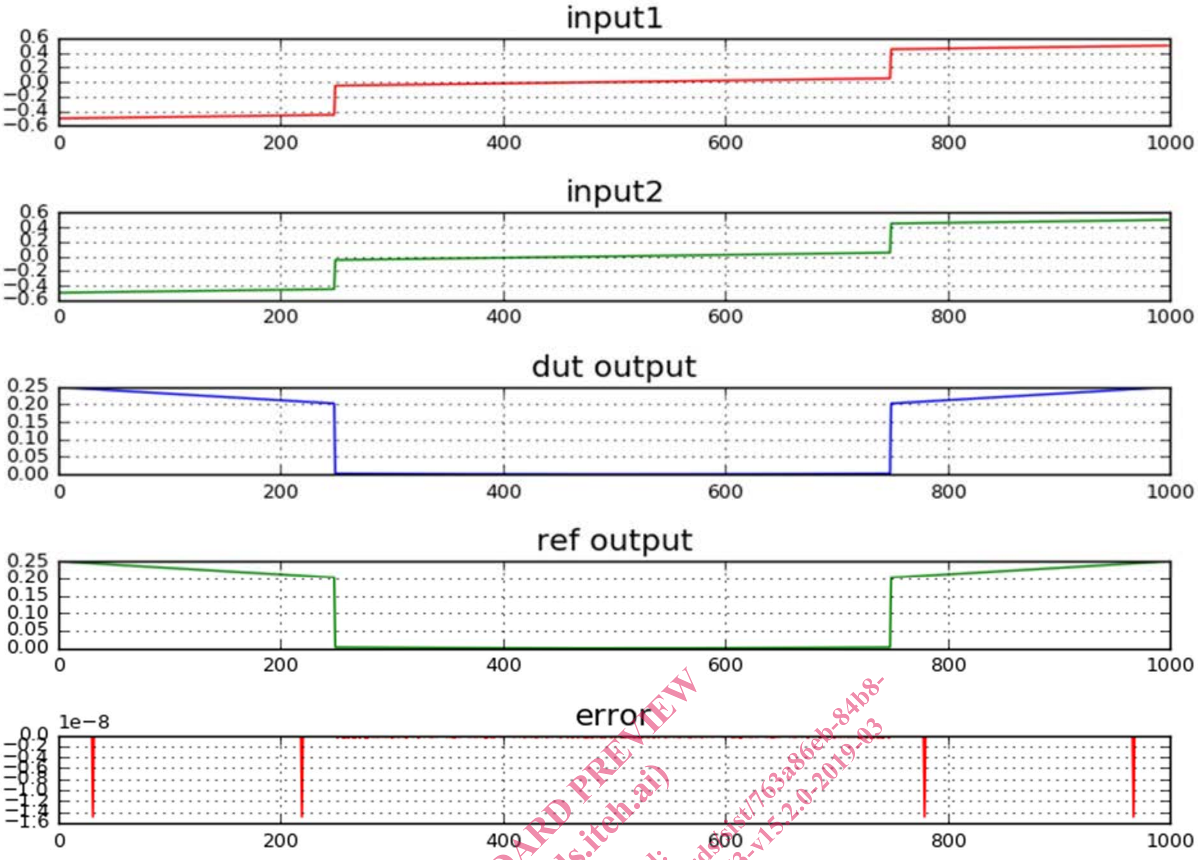

Figure 4: Test results for basic operator Mpy_32_16_1 using a piecewise sweep input. The error between the fixed-point output and floating-point output is very small.