Consultative Committee for Space Data Systems

REPORT CONCERNING SPACE DATA SYSTEM STANDARDS

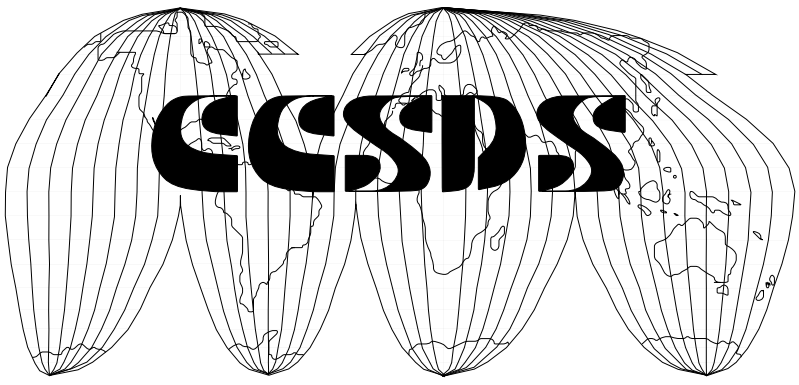
THE DATA DESCRIPTION LANGUAGE EAST—

LIST OF CONVENTIONS

CCSDS 646.0-G-1

GREEN BOOK

May 1997



AUTHORITY

Issue:	Green Book, Issue 1
Date:	May 1997
Location:	São José dos Campos São Paulo, Brazil

This document has been approved for publication by the Management Council of the Consultative Committee for Space Data Systems (CCSDS) and reflects the consensus of technical panel experts from CCSDS Member Agencies. The procedure for review and authorization of CCSDS Reports is detailed in the *Procedures Manual for the Consultative Committee for Space Data Systems*.

This document is published and maintained by:

CCSDS Secretariat
Program Integration Division (Code MG)
National Aeronautics and Space Administration
Washington, DC 20546, USA

FOREWORD

This Report is a companion book to Reference [1] and contains the description of conventions mentioned in Reference [1].

Through the process of normal evolution, it is expected that expansion, deletion, or modification of this document may occur. This Recommendation is therefore subject to CCSDS document management and change control procedures. Current versions of CCSDS documents are maintained at the CCSDS Web site:

http://www.ccsds.org/ccsds/

Questions relating to the contents or status of this document should be addressed to the CCSDS Secretariat at the address indicated on page i.

At time of publication, the active Member and Observer Agencies of the CCSDS were

- Agenzia Spaziale Italiana (ASI)/Italy.
- British National Space Centre (BNSC)/United Kingdom.
- Canadian Space Agency (CSA)/Canada.
- Centre National d'Etudes Spatiales (CNES)/France.
- Deutsche Forschungsanstalt für Luft- und Raumfahrt e.V. (DLR)/Germany.
- European Space Agency (ESA)/Europe.
- Instituto Nacional de Pesquisas Espaciais (INPE)/Brazil.
- National Aeronautics and Space Administration (NASA)/USA.
- National Space Development Agency of Japan (NASDA)/Japan.
- Russian Space Agency (RSA)/Russian Federation.

Observer Agencies

- Austrian Space Agency (ASA)/Austria.
- Central Research Institute of Machine Building (TsNIIMash)/Russian Federation.
- Centro Tecnico Aeroespacial (CTA)/Brazil.
- Chinese Academy of Space Technology (CAST)/China.
- Commonwealth Scientific and Industrial Research Organization (CSIRO)/Australia.
- Communications Research Laboratory (CRL)/Japan.
- Danish Space Research Institute (DSRI)/Denmark.
- European Organization for the Exploitation of Meteorological Satellites (EUMETSAT)/Europe.
- European Telecommunications Satellite Organization (EUTELSAT)/Europe.
- Federal Service of Scientific, Technical & Cultural Affairs (FSST&CA)/Belgium.
- Hellenic National Space Committee (HNSC)/Greece.
- Indian Space Research Organization (ISRO)/India.
- Industry Canada/Communications Research Centre (CRC)/Canada.
- Institute of Space and Astronautical Science (ISAS)/Japan.
- Institute of Space Research (IKI)/Russian Federation.
- KFKI Research Institute for Particle & Nuclear Physics (KFKI)/Hungary.
- MIKOMTEK: CSIR (CSIR)/Republic of South Africa.
- Korea Aerospace Research Institute (KARI)/Korea.
- Ministry of Communications (MOC)/Israel.
- National Oceanic & Atmospheric Administration (NOAA)/USA.
- National Space Program Office (NSPO)/Taipei.
- Swedish Space Corporation (SSC)/Sweden.
- United States Geological Survey (USGS)/USA.

DOCUMENT CONTROL

Document	Title	Date	Status/Remarks
CCSDS 646.0-G-1	Report Concerning Space Data System Standards: The Data Description Language EAST—List of Conventions, Issue 1	May 1997	Original Issue

CONTENTS

1 INTRODUCTION

1.1 PURPOSE AND SCOPE

The purpose of this document is to establish an evolving list of conventions used in the data generation process to produce real numbers. These conventions are referenced in EAST Data Descriptions (see reference [1]) and could be referenced in other Data Description Records (DDRs), written in other Data Description Languages (DDL).

This document describes the algorithms, associated to the conventions, that must be applied to an n-bit field to retrieve a real value. Every convention is identified by an Authority and Description Identifier (ADID) (see reference [2]).

This document contributes, with the EAST Recommendation, to the definition of a standard language for describing and expressing data in order to interchange data in a more uniform and automated fashion within and among Agencies participating in the Consultative Committee for Space Data Systems (CCSDS).

The list of conventions is just initialized and needs to be completed as necessary.

1.2 DOCUMENT STRUCTURE

This document is intended to provide to potential users of EAST some additional information necessary to fully interpret data that are described by an EAST DDR: this concerns the way of calculating real number values from their bit pattern representation and from the standard convention identification. Through the process of normal evolution of architectures, it is expected that new conventions may occur. This document is therefore subject to expansions.

If there is a need to reference a new convention, any Member Agency Control Authority Office (MACAO) of the CCSDS is allowed to create a new ADID associated to this new convention (see reference [2] for the description of the Control Authority Procedures). The editor of this document should be informed of any proposed new convention.

1.3 DEFINITIONS

The algorithms used to calculate the values are not provided in the EAST descriptions but only referenced by their ADIDs.

The following conventions apply in every provided algorithm:

- E represents the value of the Exponent;
- M the value of the Mantissa;
- S the value of the Sign-bit;
- V the value of the real.

1.4 ACRONYMS

ADID	Authority and Description IDentifier
DDR	Data Description Record
DDL	Data Description Language
MACAO	Member Agency Control Authority Office

1.5 REFERENCES

[1] *The Data Description Language EAST Specification (CCSD0010)*. Recommendation for Space Data System Standards, CCSDS 644.0-B-1. Blue Book. Issue 1. Washington, D.C.: CCSDS, May 1997.

[2] *Standard Formatted Data Units — Control Authority Procedures.* Recommendation for Space Data System Standards, CCSDS 630.0-B-1. Blue Book. Issue 1. Washington, D.C.: CCSDS, June 1993.

[3] *Binary Floating Point Arithmetic*. American National Standard, ANSI/IEEE 754-1985(R1991). New York: ANSI, 1985.

[4] *DEC FORTRAN: User Manual for Open VMS VAX Systems*. Rev. 2. Maynard, Massachusetts, USA: Digital Equipment Corp., 1993.

[5] *Sixteen-Bit Computer Instruction Set Architecture*. Rev. A. Military Standard, MIL-STD-1750A. NPFC, 1980.

[6] *CDC Hardware Reference Manual*: *N.604 69420*. Minneapolis, Minnesota, USA: Control Data Corp.

[7] *CDC Hardware Reference Manual*: *N.604 58890*. Rev. E. Minneapolis, Minnesota, USA: Control Data Corp.

[8] *Programming Guide for CMS and MVS* (sc26-4222-6).

2 CONVENTIONS

2.1 FCSTC000

FCSTC000 is the ADID of the ANSI/IEEE Standard 754 convention (reference [3]). This convention defines the algorithm to be used to retrieve the value of an IEEE 754 real number from the n-bit field according to the binary representation provided in the EAST description.

Algorithm:

if E = (2*Bias + 1) and M /= 0	it is not a number
if E = (2*Bias + 1) and M = 0	$V = (-1)^S \infty$
if 0 < E < (2*Bias + 1)	$V = (-1)^S * (1 + M) * 2^{(E - Bias)}$
if E = 0 and M /= 0	$V = (-1)^S * M * 2^{-(Bias - 1)}$
if E = 0 and M = 0	$V = (-1)^S 0$

where E and M are computed according to the locations of the exponent and the mantissa, respectively, given in the binary representation, and where Bias represents the value of the bias provided in the binary representation (i.e., 127 for a 32-bit real and 1023 for a 64-bit real).

NOTE – The Mantissa is a sign and magnitude number; the MSB of the Mantissa represents 2^{-1}.

Real Size:

The convention provides 32-bit and 64-bit representations.

Illustration:

On a SUN architecture, the binary representation of a 32-bit real is the following:

```
Binrep = constant REAL_PHYSICAL_DESCRIPTION := (
        NUMBER_OF_SUBFIELDS_IN_EXPONENT => 1,
        NUMBER_OF_SUBFIELDS_IN_MANTISSA => 1,
        CONVENTION_USED => FCSTC000,
        SIGN_BIT_NUMBER => 0,
        COMPLEMENT => SIGN_AND_MAGNITUDE,
        EXPONENT_BASE => 2,
        BIAS => 127,
        LOCATION_OF_EXPONENT => (1 => (1,8)),
        LOCATION_OF_MANTISSA => (1 => (9,31)));
```

On a SUN architecture, the binary representation of a 64-bit real is the following:

```
Binrep = constant REAL_PHYSICAL_DESCRIPTION := (
        NUMBER_OF_SUBFIELDS_IN_EXPONENT => 1,
        NUMBER_OF_SUBFIELDS_IN_MANTISSA => 1,
        CONVENTION_USED => FCSTC000,
        SIGN_BIT_NUMBER => 0,
        COMPLEMENT => SIGN_AND_MAGNITUDE,
        EXPONENT_BASE => 2,
        BIAS => 1023,
        LOCATION_OF_EXPONENT => (1 => (1,11)),
        LOCATION_OF_MANTISSA => (1 => (12,63)));
```

On a PC or Mac architecture, the binary representation of a 32-bit real is the following:

```
Binrep = constant REAL_PHYSICAL_DESCRIPTION := (
        NUMBER_OF_SUBFIELDS_IN_EXPONENT => 2,
        NUMBER_OF_SUBFIELDS_IN_MANTISSA => 3,
        CONVENTION_USED => FCSTC000,
        SIGN_BIT_NUMBER => 24,
        COMPLEMENT => SIGN_AND_MAGNITUDE,
        EXPONENT_BASE => 2,
        BIAS => 127,
        LOCATION_OF_EXPONENT => ( 1 => (25,31), 2 => (16,16)),
        LOCATION_OF_MANTISSA => (  1 => (17,23), 2 => (8,15), 3 => (0,7)));
```

On a PC or Mac architecture, the binary representation of a 64-bit real is the following:

```
Binrep = constant REAL_PHYSICAL_DESCRIPTION := (
        NUMBER_OF_SUBFIELDS_IN_EXPONENT => 2,
        NUMBER_OF_SUBFIELDS_IN_MANTISSA => 7,
        CONVENTION_USED => FCSTC000,
        SIGN_BIT_NUMBER => 56,
        COMPLEMENT => SIGN_AND_MAGNITUDE,
        EXPONENT_BASE => 2,
        BIAS => 1023,
        LOCATION_OF_EXPONENT => ( 1 => (57,63), 2 => (48,51)),
        LOCATION_OF_MANTISSA => (  1 => (52,55), 2 => (40,47),
                                   3 => (32,39), 4 => (24,31),
                                   5 => (16,23), 6 => (8,15), 7 => (0,7)));
```

2.2 FCSTC001

FCSTC001 is the ADID of the Digital Equipment Corporation for VAX processors/OpenVMS Systems convention (reference [4]). This convention defines the algorithm to be used to retrieve the value of a DEC real number from the n-bit field according to the binary representation provided in the EAST description.

Algorithm:

if S = 1 and E = 0	it is a reserved operand *)
if S = 0 and E = 0	V = 0 *)
if E > 0	$V = (-1)^S * (0.5 + M) * 2^{(E - Bias)}$

where E and M are computed according to the locations of the exponent and the mantissa, respectively, given in the binary representation, and where Bias represents the value of the bias provided in the binary representation (i.e., 128 in the F_Floating Representation, 128 in the D_Floating Representation, 1024 in the G_Floating Representation, and 16384 in the H_Floating Representation).

NOTE – The redundant MSB of the Mantissa is not represented; i.e., the first encountered bit of the Mantissa represents 2^{-2}.

Real Size:

The convention provides 32-bit, 64-bit and 128-bit representations.

Illustration:

On a VAX/VMS architecture, the binary representation of a 32-bit real in the F_Floating Representation is the following:

```
Binrep = constant REAL_PHYSICAL_DESCRIPTION := (
        NUMBER_OF_SUBFIELDS_IN_EXPONENT => 2,
        NUMBER_OF_SUBFIELDS_IN_MANTISSA => 3,
        CONVENTION_USED => FCSTC001,
        SIGN_BIT_NUMBER => 8,
        COMPLEMENT => SIGN_AND_MAGNITUDE,
        EXPONENT_BASE => 2,
        BIAS => 128,
        LOCATION_OF_EXPONENT => (1 => (9,15), 2 => (0,0)),
        LOCATION_OF_MANTISSA => (1 => (1,7), 2 => (24, 31), 3 => (16,23)));
```

*) Regardless of the value of the mantissa.

On a VAX/VMS architecture, the binary representation of a 64-bit real in the D_Floating Representation is the following:

```
Binrep = constant REAL_PHYSICAL_DESCRIPTION := (
        number_of_subfields_in_exponent => 2,
        number_of_subfields_in_mantissa => 7,
        CONVENTION_USED => FCSTC001,
        SIGN_BIT_NUMBER => 8,
        COMPLEMENT => SIGN_AND_MAGNITUDE,
        EXPONENT_BASE => 2,
        BIAS => 128,
        LOCATION_OF_EXPONENT => (1 => (9,15), 2 => (0,0)),
        LOCATION_OF_MANTISSA => ( 1 => (1,7), 2 => (24, 31), 3 => (16,23)
                                  4 => (40,47), 5 => (32,39),
                                  6 => (56,63), 7 => (48,55)));
```

On a VAX/VMS architecture, the binary representation of a 64-bit real in the G_Floating Representation is the following:

```
Binrep = constant REAL_PHYSICAL_DESCRIPTION := (
        NUMBER_OF_SUBFIELDS_IN_EXPONENT => 2,
        NUMBER_OF_SUBFIELDS_IN_MANTISSA => 7,
        CONVENTION_USED => FCSTC001,
        SIGN_BIT_NUMBER => 8,
        COMPLEMENT => SIGN_AND_MAGNITUDE,
        EXPONENT_BASE => 2,
        BIAS => 1024,
        LOCATION_OF_EXPONENT => (1 => (9,15), 2 => (0,3)),
        LOCATION_OF_MANTISSA => ( 1 => (4,7), 2 => (24, 31), 3 => (16,23)
                                  4 => (40,47), 5 => (32,39),
                                  6 => (56,63), 7 => (48,55)));
```

On a VAX/VMS architecture, the binary representation of a 128-bit real in the H_Floating Representation is the following:

```
Binrep = constant REAL_PHYSICAL_DESCRIPTION := (
        NUMBER_OF_SUBFIELDS_IN_EXPONENT => 2,
        NUMBER_OF_SUBFIELDS_IN_MANTISSA => 14,
        CONVENTION_USED => FCSTC001,
        SIGN_BIT_NUMBER => 8,
        COMPLEMENT => SIGN_AND_MAGNITUDE,
        EXPONENT_BASE => 2,
        BIAS => 16 384,
        LOCATION_OF_EXPONENT => (1 => (9,15), 2 => (0,7)),
        LOCATION_OF_MANTISSA => ( 1 => (24, 31), 2 => (16,23)
                                  3 => (40,47), 4 => (32,39),
                                  5 => (56,63), 6 => (48,55),
                                  7 => (72,79), 8 => (64,71),
                                  9 => (88,95), 10=> (80,87),
                                  11 => (104,111), 12 => (96,103),
                                  13 => (120,127), 14 => (112,119)));
```

2.3 FCSTC002

FCSTC002 is the ADID of the MIL-STD-1750A convention (reference [5]). This convention defines the algorithm to be used to retrieve the value of a 1750A real number from the n-bit field according to the binary representation provided in the EAST description.

Algorithm:

$$V = M * 2^{E}$$

where E and M are computed according to the locations of the exponent and the mantissa, respectively, given in the binary representation.

NOTE – The Mantissa is a two's complement number and the exponent is also a two's complement number; the first encountered bit of the Mantissa represents the sign bit and the second encountered bit represents 2^{-1}.

Real Size:

The convention provides 32-bit and 48-bit representations.

Illustration:

On a 1750A architecture, the binary representation of a 32-bit real is the following:

```
Binrep = constant REAL_PHYSICAL_DESCRIPTION := (
         NUMBER_OF_SUBFIELDS_IN_EXPONENT => 1,
         NUMBER_OF_SUBFIELDS_IN_MANTISSA => 1,
         CONVENTION_USED => FCSTC002 ,
         SIGN_BIT_NUMBER => 0,
         COMPLEMENT => TWOS_COMPLEMENT,
         EXPONENT_BASE => 2,
         BIAS => 0,
         LOCATION_OF_EXPONENT => (1 => (24,31)),
         LOCATION_OF_MANTISSA => (1 => (0,23)));
```

On a 1750A architecture, the binary representation of a 48-bit real is the following:

```
Binrep = constant REAL_PHYSICAL_DESCRIPTION := (
        NUMBER_OF_SUBFIELDS_IN_EXPONENT => 1,
        NUMBER_OF_SUBFIELDS_IN_MANTISSA => 2,
        CONVENTION_USED => FCSTC002 ,
        SIGN_BIT_NUMBER => 0,
        COMPLEMENT => TWOS_COMPLEMENT,
        EXPONENT_BASE => 2,
        BIAS => 0,
        LOCATION_OF_EXPONENT => (1 => (24,31)),
        LOCATION_OF_MANTISSA => (1 => (0,23), 2 => (32,47)));
```

2.4 FCSTC003

FCSTC003 is the ADID of the Control Data Corporation/NOS-VE convention (reference [7]). This convention defines the algorithm to be used to retrieve the value of a CDC real number from the n-bit field according to the binary representation provided in the EAST description.

Algorithm:

if C = 7	V = Indefinite Number
if 5≤C≤6	V = Infinity
if 3≤C≤4	$V = (-1)^S * M * 2^{(E - Bias)}$
else	V = 0

where E and M are computed according to the locations of the exponent and the mantissa, respectively, given in the binary representation, and where Bias represents the value of the bias provided in the binary representation (i.e., 16384 for a 64-bit real or a 128-bit real). C is a positive value computed with bits 1 through 3.

NOTE – The Mantissa is a sign and magnitude number; the MSB of the Mantissa represents 2^{-1}.

Real Size:

The convention provides 64-bit and 128-bit representations.

Illustration:

On a CDC/NOS-VE architecture, the binary representation of a 64-bit real is the following:

```
Binrep = constant REAL_PHYSICAL_DESCRIPTION := (
        NUMBER_OF_SUBFIELDS_IN_EXPONENT => 1,
        NUMBER_OF_SUBFIELDS_IN_MANTISSA => 1,
        CONVENTION_USED => FCSTC003 ,
        SIGN_BIT_NUMBER => 0,
        COMPLEMENT => SIGN_AND_MAGNITUDE,
        EXPONENT_BASE => 2,
        BIAS => 16384,
        LOCATION_OF_EXPONENT => (1 => (1,15)),
        LOCATION_OF_MANTISSA => (1 => (16,63)));
```

On a CDC/NOS-VE architecture, the binary representation of a 128-bit real is the following:

```
Binrep = constant REAL_PHYSICAL_DESCRIPTION := (
         NUMBER_OF_SUBFIELDS_IN_EXPONENT => 1,
         NUMBER_OF_SUBFIELDS_IN_MANTISSA => 2,
         CONVENTION_USED => FCSTC003 ,
         SIGN_BIT_NUMBER => 0,
         COMPLEMENT => SIGN_AND_MAGNITUDE,
         EXPONENT_BASE => 2,
         BIAS => 16384,
         LOCATION_OF_EXPONENT => (1 => (1,15)),
         LOCATION_OF_MANTISSA => (1 => (16,63), 2 => (80,127)));
```

NOTE – The bits 64 through 79 are set equal to bits 0 through 15.

2.5 FCSTC004

FCSTC004 is the ADID of the Control Data Corporation/NOS-BE convention (reference [6]). This convention defines the algorithm to be used to retrieve the value of a CDC real number from the n-bit field according to the binary representation provided in the EAST description.

Algorithm:

$$V = (-1)^S * M * 2^{(E - 1024)}$$

where E and M are computed according to the locations of the exponent and the mantissa, respectively, given in the binary representation.

NOTES

1 The Mantissa is a sign and magnitude integer number and the Exponent is one's complement signed number; the LSB of the Mantissa represents 2^0.

2 Two opposite numbers have one's complement representation.

Real Size:

The convention provides a 60-bit representation.

Illustration:

On a CDC/NOS-BE architecture, the binary representation of a 60-bit real is the following:

```
Binrep = constant REAL_PHYSICAL_DESCRIPTION := (
        NUMBER_OF_SUBFIELDS_IN_EXPONENT => 1,
        NUMBER_OF_SUBFIELDS_IN_MANTISSA => 1,
        CONVENTION_USED => FCSTC004 ,
        SIGN_BIT_NUMBER => 0,
        COMPLEMENT => SIGN_AND_MAGNITUDE,
        EXPONENT_BASE => 2,
        BIAS => 1024,
        LOCATION_OF_EXPONENT => (1 => (1,11)),
        LOCATION_OF_MANTISSA => (1 => (12,59)));
```

2.6 FCSTC005

FCSTC005 is the ADID of the CMS and MVS IBM/Mainframe convention (reference [8]). This convention defines the algorithm to be used to retrieve the value of an IBM real number from the n-bit field according to the binary representation provided in the EAST description.

Algorithm:

if M = 0	$V = (-1)^S 0$
if M /= 0	$V = (-1)^S * M * 16^{(E - Bias)}$

where E and M are computed according to the locations of the exponent and the mantissa, respectively, given in the binary representation, and where Bias represents the value of the bias provided in the binary representation (i.e., 64 for a 32-bit real).

NOTE – The Mantissa is a sign and magnitude number; the MSB of the Mantissa represents 2^{-1}.

Real Size:

The convention provides a 32-bit representation.

Illustration:

On an IBM/Mainframe architecture, the binary representation of a 32-bit real is the following:

```
Binrep =constant REAL_PHYSICAL_DESCRIPTION := (
        NUMBER_OF_SUBFIELDS_IN_EXPONENT => 1,
        NUMBER_OF_SUBFIELDS_IN_MANTISSA => 1,
        CONVENTION_USED => FCSTC005 ,
        SIGN_BIT_NUMBER => 0,
        COMPLEMENT => SIGN_AND_MAGNITUDE,
        EXPONENT_BASE => 16,
        BIAS => 64,
        LOCATION_OF_EXPONENT => (1 => (1,7)),
        LOCATION_OF_MANTISSA => (1 => (8,31)));
```