

INTERNATIONAL STANDARD

NORME INTERNATIONALE



EMC IC modelling –

Part 2: Models of integrated circuits for EMI behavioural simulation – Conducted emissions modelling (ICEM-CE)

Modèles de circuits intégrés pour la CEM –

Partie 2: Modèles de circuits intégrés pour la simulation du comportement lors de perturbations électromagnétiques – Modélisation des émissions conduites (ICEM-CE)



THIS PUBLICATION IS COPYRIGHT PROTECTED

Copyright © 2017 IEC, Geneva, Switzerland

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either IEC or IEC's member National Committee in the country of the requester. If you have any questions about IEC copyright or have an enquiry about obtaining additional rights to this publication, please contact the address below or your local IEC member National Committee for further information.

Droits de reproduction réservés. Sauf indication contraire, aucune partie de cette publication ne peut être reproduite ni utilisée sous quelque forme que ce soit et par aucun procédé, électronique ou mécanique, y compris la photocopie et les microfilms, sans l'accord écrit de l'IEC ou du Comité national de l'IEC du pays du demandeur. Si vous avez des questions sur le copyright de l'IEC ou si vous désirez obtenir des droits supplémentaires sur cette publication, utilisez les coordonnées ci-après ou contactez le Comité national de l'IEC de votre pays de résidence.

IEC Central Office
3, rue de Varembe
CH-1211 Geneva 20
Switzerland

Tel.: +41 22 919 02 11
Fax: +41 22 919 03 00
info@iec.ch
www.iec.ch

About the IEC

The International Electrotechnical Commission (IEC) is the leading global organization that prepares and publishes International Standards for all electrical, electronic and related technologies.

About IEC publications

The technical content of IEC publications is kept under constant review by the IEC. Please make sure that you have the latest edition, a corrigenda or an amendment might have been published.

IEC Catalogue - webstore.iec.ch/catalogue

The stand-alone application for consulting the entire bibliographical information on IEC International Standards, Technical Specifications, Technical Reports and other documents. Available for PC, Mac OS, Android Tablets and iPad.

IEC publications search - www.iec.ch/searchpub

The advanced search enables to find IEC publications by a variety of criteria (reference number, text, technical committee,...). It also gives information on projects, replaced and withdrawn publications.

IEC Just Published - webstore.iec.ch/justpublished

Stay up to date on all new IEC publications. Just Published details all new publications released. Available online and also once a month by email.

Electropedia - www.electropedia.org

The world's leading online dictionary of electronic and electrical terms containing 20 000 terms and definitions in English and French, with equivalent terms in 16 additional languages. Also known as the International Electrotechnical Vocabulary (IEV) online.

IEC Glossary - std.iec.ch/glossary

65 000 electrotechnical terminology entries in English and French extracted from the Terms and Definitions clause of IEC publications issued since 2002. Some entries have been collected from earlier publications of IEC TC 37, 77, 86 and CISPR.

IEC Customer Service Centre - webstore.iec.ch/csc

If you wish to give us your feedback on this publication or need further assistance, please contact the Customer Service Centre: csc@iec.ch.

A propos de l'IEC

La Commission Electrotechnique Internationale (IEC) est la première organisation mondiale qui élabore et publie des Normes internationales pour tout ce qui a trait à l'électricité, à l'électronique et aux technologies apparentées.

A propos des publications IEC

Le contenu technique des publications IEC est constamment revu. Veuillez vous assurer que vous possédez l'édition la plus récente, un corrigendum ou amendement peut avoir été publié.

Catalogue IEC - webstore.iec.ch/catalogue

Application autonome pour consulter tous les renseignements bibliographiques sur les Normes internationales, Spécifications techniques, Rapports techniques et autres documents de l'IEC. Disponible pour PC, Mac OS, tablettes Android et iPad.

Recherche de publications IEC - www.iec.ch/searchpub

La recherche avancée permet de trouver des publications IEC en utilisant différents critères (numéro de référence, texte, comité d'études,...). Elle donne aussi des informations sur les projets et les publications remplacées ou retirées.

IEC Just Published - webstore.iec.ch/justpublished

Restez informé sur les nouvelles publications IEC. Just Published détaille les nouvelles publications parues. Disponible en ligne et aussi une fois par mois par email.

Electropedia - www.electropedia.org

Le premier dictionnaire en ligne de termes électroniques et électriques. Il contient 20 000 termes et définitions en anglais et en français, ainsi que les termes équivalents dans 16 langues additionnelles. Egalement appelé Vocabulaire Electrotechnique International (IEV) en ligne.

Glossaire IEC - std.iec.ch/glossary

65 000 entrées terminologiques électrotechniques, en anglais et en français, extraites des articles Termes et Définitions des publications IEC parues depuis 2002. Plus certaines entrées antérieures extraites des publications des CE 37, 77, 86 et CISPR de l'IEC.

Service Clients - webstore.iec.ch/csc

Si vous désirez nous donner des commentaires sur cette publication ou si vous avez des questions contactez-nous: csc@iec.ch.

INTERNATIONAL STANDARD

NORME INTERNATIONALE



EMC IC modelling –

Part 2: Models of integrated circuits for EMI behavioural simulation – Conducted emissions modelling (ICEM-CE)

Modèles de circuits intégrés pour la CEM –

Partie 2: Modèles de circuits intégrés pour la simulation du comportement lors de perturbations électromagnétiques – Modélisation des émissions conduites (ICEM-CE)

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

COMMISSION
ELECTROTECHNIQUE
INTERNATIONALE

31.200; 33.100.10

ISBN 978-2-8322-3876-9

Warning! Make sure that you obtained this publication from an authorized distributor.

Attention! Veuillez vous assurer que vous avez obtenu cette publication via un distributeur agréé.

CONTENTS

FOREWORD.....	7
1 Scope.....	9
2 Normative references	9
3 Terms, definitions, abbreviations and conventions	9
3.1 Terms and definitions.....	9
3.2 Abbreviations.....	11
3.3 Conventions.....	11
4 Philosophy.....	11
4.1 General.....	11
4.2 Conducted emission from core activity (digital culprit).....	12
4.3 Conducted emission from I/O activity	12
4.4 Data exchange format.....	12
5 ICEM-CE basic components	13
5.1 General.....	13
5.2 Internal Activity (IA)	13
5.2.1 General	13
5.2.2 Examples of IA	14
5.3 Passive Distribution Network (PDN).....	14
5.3.1 General	14
5.3.2 Examples of PDN	15
6 IC macro-models	16
6.1 Types of IC macro-models	16
6.2 General IC macro-model	16
6.3 Block-based IC macro-model	17
6.3.1 Block component	17
6.3.2 Inter-Block Coupling component (IBC)	18
6.3.3 Block-based IC macro-model structure	19
6.4 Sub-model-based IC macro-model	21
6.4.1 Sub-model component.....	21
6.4.2 Sub-model-based IC macro-model structure	22
7 CEML format	23
7.1 General.....	23
7.2 CEML structure	24
7.3 Global keywords	24
7.4 Header section.....	24
7.5 Lead definitions	25
7.6 SPICE macro-models.....	26
7.7 Validity section	28
7.7.1 General	28
7.7.2 Attribute definitions.....	29
7.8 PDN.....	31
7.8.1 General	31
7.8.2 Attribute definitions.....	32
7.8.3 Description	36
7.9 IBC	40
7.9.1 General	40

7.9.2	Attribute definitions	40
7.10	IA	42
7.10.1	General	42
7.10.2	Attribute definitions	42
7.10.3	Description	46
8	Requirements for parameter extraction	47
8.1	General	47
8.2	Environmental extraction constraints	47
8.3	IA parameter extraction	47
8.4	PDN parameter extraction	47
8.5	IBC parameter extraction	48
Annex A	(normative) Preliminary definitions for XML representation	49
A.1	XML basics	49
A.1.1	XML declaration	49
A.1.2	Basic elements	49
A.1.3	Root element	49
A.1.4	Comments	50
A.1.5	Line terminations	50
A.1.6	Element hierarchy	50
A.1.7	Element attributes	50
A.2	Keyword requirements	50
A.2.1	General	50
A.2.2	Keyword characters	51
A.2.3	Keyword syntax	51
A.2.4	File structure	51
A.2.5	Values	53
Annex B	(normative) CEML valid keywords and usage	56
B.1	Root element keywords	56
B.2	File header keywords	57
B.3	Validity section keywords	58
B.4	Global keywords	59
B.5	Lead Keyword	59
B.6	Lead_definitions section attributes	60
B.7	Macromodels section attributes	60
B.8	Pdn section keywords	61
B.8.1	Lead element keywords	61
B.8.2	Netlist section keywords	63
B.9	Ibc section keywords	63
B.9.1	Lead element keywords	63
B.9.2	Netlist section keywords	65
B.10	Ia section keywords	65
B.10.1	Lead element keywords	65
B.10.2	Voltage section keywords	66
B.10.3	Current section keywords	68
Annex C	(informative) Example of ICEM-CE macro-model in CEML format	70
C.1	General	70
C.2	PDN and IBC sub-model	70
C.3	IA sub-model	71

C.4	Frequency domain ICEM-CE in CEML.....	73
C.5	Time domain ICEM-CE in CEML	75
Annex D	(informative) Conversions between parameter types	77
D.1	General.....	77
D.2	Conversion for one-port PDN	77
D.3	Conversion for two-port PDN	77
Annex E	(informative) Model parameter generation.....	79
E.1	General.....	79
E.2	Default structure and values	79
E.2.1	General	79
E.2.2	IA parameters.....	79
E.2.3	PDN parameters	80
E.3	Model parameter generation from design information	81
E.3.1	General	81
E.3.2	IA parameters.....	81
E.3.3	PDN parameters	85
E.4	Model parameter generation from measurements.....	87
E.4.1	IA parameters.....	87
E.4.2	PDN parameters	90
Annex F	(informative) Decoupling capacitors optimization.....	100
Annex G	(informative) Conducted emission prediction	102
Annex H	(informative) Conducted emission prediction at PCB level	103
Bibliography		105
Figure 1	– Decomposition example of a digital IC for conducted emissions analysis	12
Figure 2	– IA component in the case of a current source.....	13
Figure 3	– Example of IA characteristics in the time domain.....	14
Figure 4	– Example of IA characteristics in the frequency domain	14
Figure 5	– Example of a four-terminal PDN using lumped elements	15
Figure 6	– Example of a seven-terminal PDN using distributed elements	16
Figure 7	– Example of a twelve-terminal PDN using matrix representation	16
Figure 8	– General IC macro-model	17
Figure 9	– Example of block component with a single IA	18
Figure 10	– Example of block components for I/Os	18
Figure 11	– Example of IBC with four internal terminals	19
Figure 12	– Relationship between blocks and IBC.....	19
Figure 13	– Block-based IC macro-model.....	20
Figure 14	– Example of block-based IC macro-model.....	21
Figure 15	– Example of simple sub-model.....	21
Figure 16	– Sub-model-based IC macro-model	22
Figure 17	– CEML inheritance hierarchy	23
Figure 18	– Example of a netlist file defining a sub-circuit.....	28
Figure 19	– PDN represented as <i>S</i> -parameters in Touchstone format.....	38
Figure 20	– Simulated IA waveform with corresponding parameters	45
Figure A.1	– Multiple XML (CEML) files.....	52

Figure A.2 – XML files with data files (*.dat)	52
Figure A.3 – XML files with additional files	53
Figure C.1 – Example pin-out of a microcontroller and the modelled pins	70
Figure C.2 – PDN sub-model topology	71
Figure C.3 – IA sub-model topology	72
Figure C.4 – IA of digital block in frequency domain	72
Figure C.5 – IA of digital block in time domain	73
Figure E.1 – Typical characterization current gate schematic	82
Figure E.2 – Current peak during switching transition	82
Figure E.3 – Example of IA extraction procedure from design	83
Figure E.4 – Technology Influence	83
Figure E.5 – Final current waveform for a program period	84
Figure E.6 – Comparison between measurement and simulation	84
Figure E.7 – Example lumped element model of a package	85
Figure E.8 – Circuit structure of the netlist	87
Figure E.9 – Principle of the IA computation in the frequency domain	88
Figure E.10 – Process involved to model $i_A(t)$	89
Figure E.11 – $i_{Ext}(t)$ measured using IEC 61967-4	89
Figure E.12 – $i_A(t)$ and $i_{Ext}(t)$ profiles	90
Figure E.13 – Conventional one-port S -parameter measurement	90
Figure E.14 – Two-port method for low impedance measurement	91
Figure E.15 – Two-port method for high impedance measurement	91
Figure E.16 – Example of a hardware set-up used to extract the PDN parameters	92
Figure E.17 – Miniature 50 Ω coaxial connectors	93
Figure E.18 – Impedance probe using two miniature coaxial connectors	93
Figure E.19 – Open and short terminations	93
Figure E.20 – Measurement probe model	94
Figure E.21 – De-embedding principle	94
Figure E.22 – Example of a predefined PDN structure	95
Figure E.23 – RL configuration	96
Figure E.24 – RLC configuration	97
Figure E.25 – RLC with magnetic coupling configuration	97
Figure E.26 – Impedance seen from Vcc and Gnd	97
Figure E.27 – Complete PDN component	98
Figure E.28 – Set-up for correlation (left), measurement and prediction model (right)	99
Figure E.29 – Set-up used to measure the internal decoupling capacitor	99
Figure F.1 – Equivalent schematic of the complete electronic system	100
Figure F.2 – Impedance prediction and measurements	101
Figure G.1 – IEC 61967-4 test set-up standard	102
Figure G.2 – Comparison between prediction and measurement	102
Figure H.1 – Prediction of ETVddc noise level at PCB level	103
Figure H.2 – Good agreements on the noise envelope	104

Table 1 – Attributes of Lead keyword in the <i>Lead_definitions</i> section	25
Table 2 – Compatibility between the Mode and Type fields for correct CEML annotation.....	26
Table 3 – <i>Subckt</i> definition.....	26
Table 4 – Definition of the <i>Validity</i> section	28
Table 5 – Definition of the Lead keyword for <i>Pdn</i> section	32
Table 6 – Valid data formats and their default units in the <i>Pdn</i> section.....	35
Table 7 – Valid file extensions in the <i>Pdn</i> section	35
Table 8 – Valid fields of the Lead keyword in the <i>Pdn</i> section	36
Table 9 – <i>Netlist</i> definition	39
Table 10 – Differences between the <i>Pdn</i> and <i>Ibc</i> section fields	41
Table 11 – Valid fields of the <i>Lead</i> keyword for IBC definition	41
Table 12 – Definition of the <i>Lead</i> keyword in the <i>Ia</i> section.....	42
Table 13 – <i>Voltage</i> and <i>Current</i> definition	43
Table 14 – Valid file extensions in the <i>Ia</i> section.....	43
Table 15 – Definition of the <i>Pulse</i> keyword in the <i>Voltage</i> or <i>Current</i> section	44
Table 16 – Base units of the Pulse section's fields.....	44
Table 17 – Valid data formats and their default units for the <i>Voltage</i> and <i>Current</i> elements.....	46
Table A.1 – Valid logarithmic units.....	54
Table D.1 – One-port conversion	77
Table D.2 – Two-port conversion	78
Table E.1 – Typical parameters for CMOS logic technologies	80
Table E.2 – Typical number of logic gates vs. CPU technology	80
Table E.3 – R, L and C parameters for various package types	81
Table E.4 – Measurement configurations and extracted RLC parameters.....	95

INTERNATIONAL ELECTROTECHNICAL COMMISSION

EMC IC MODELLING –

Part 2: Models of integrated circuits for EMI behavioural simulation – Conducted emissions modelling (ICEM-CE)

FOREWORD

- 1) The International Electrotechnical Commission (IEC) is a worldwide organization for standardization comprising all national electrotechnical committees (IEC National Committees). The object of IEC is to promote international co-operation on all questions concerning standardization in the electrical and electronic fields. To this end and in addition to other activities, IEC publishes International Standards, Technical Specifications, Technical Reports, Publicly Available Specifications (PAS) and Guides (hereafter referred to as “IEC Publication(s)”). Their preparation is entrusted to technical committees; any IEC National Committee interested in the subject dealt with may participate in this preparatory work. International, governmental and non-governmental organizations liaising with the IEC also participate in this preparation. IEC collaborates closely with the International Organization for Standardization (ISO) in accordance with conditions determined by agreement between the two organizations.
- 2) The formal decisions or agreements of IEC on technical matters express, as nearly as possible, an international consensus of opinion on the relevant subjects since each technical committee has representation from all interested IEC National Committees.
- 3) IEC Publications have the form of recommendations for international use and are accepted by IEC National Committees in that sense. While all reasonable efforts are made to ensure that the technical content of IEC Publications is accurate, IEC cannot be held responsible for the way in which they are used or for any misinterpretation by any end user.
- 4) In order to promote international uniformity, IEC National Committees undertake to apply IEC Publications transparently to the maximum extent possible in their national and regional publications. Any divergence between any IEC Publication and the corresponding national or regional publication shall be clearly indicated in the latter.
- 5) IEC itself does not provide any attestation of conformity. Independent certification bodies provide conformity assessment services and, in some areas, access to IEC marks of conformity. IEC is not responsible for any services carried out by independent certification bodies.
- 6) All users should ensure that they have the latest edition of this publication.
- 7) No liability shall attach to IEC or its directors, employees, servants or agents including individual experts and members of its technical committees and IEC National Committees for any personal injury, property damage or other damage of any nature whatsoever, whether direct or indirect, or for costs (including legal fees) and expenses arising out of the publication, use of, or reliance upon, this IEC Publication or any other IEC Publications.
- 8) Attention is drawn to the Normative references cited in this publication. Use of the referenced publications is indispensable for the correct application of this publication.
- 9) Attention is drawn to the possibility that some of the elements of this IEC Publication may be the subject of patent rights. IEC shall not be held responsible for identifying any or all such patent rights.

International Standard IEC 62433-2 has been prepared by subcommittee 47A: Integrated Circuits, of IEC technical committee 47: Semiconductor devices.

This second edition cancels and replaces the first edition published in 2008. This edition constitutes a technical revision.

This edition includes the following significant technical changes with respect to the previous edition:

Incorporation of an XML based exchange format for model representation.

The text of this standard is based on the following documents:

FDIS	Report on voting
47A/999/FDIS	47A/1007/RVD

Full information on the voting for the approval of this standard can be found in the report on voting indicated in the above table.

This publication has been drafted in accordance with the ISO/IEC Directives, Part 2.

A list of all parts in the IEC 62433 series, published under the general title *EMC IC modelling*, can be found on the IEC website.

The committee has decided that the contents of this publication will remain unchanged until the stability date indicated on the IEC website under "<http://webstore.iec.ch>" in the data related to the specific publication. At this date, the publication will be

- reconfirmed,
- withdrawn,
- replaced by a revised edition, or
- amended.

IMPORTANT – The 'colour inside' logo on the cover page of this publication indicates that it contains colours which are considered to be useful for the correct understanding of its contents. Users should therefore print this document using a colour printer.

EMC IC MODELLING –

Part 2: Models of integrated circuits for EMI behavioural simulation – Conducted emissions modelling (ICEM-CE)

1 Scope

This part of IEC 62433 specifies macro-models for an Integrated Circuit (IC) to simulate conducted electromagnetic emissions on a printed circuit board. The model is commonly called Integrated Circuit Emission Model – Conducted Emission (ICEM-CE).

The ICEM-CE macro-model can also be used for modelling an IC-die, a functional block and an Intellectual Property (IP) block.

The ICEM-CE macro-model can be used to model both digital and analogue ICs.

Basically, conducted emissions have two origins:

- conducted emissions through power supply terminals and ground reference structures;
- conducted emissions through input/output (I/O) terminals.

The ICEM-CE macro-model addresses those two types of origins in a single approach.

This standard defines structures and components of the macro-model for EMI simulation taking into account the IC's internal activities.

This part of IEC 62433 has two main parts:

- the first is the electrical description of ICEM-CE macro-model elements along with the specific requirements for information.
- the second part proposes a universal data exchange format called CEML based on XML. This format allows encoding the ICEM-CE in a more useable and generic form for simulating the conducted emissions.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

IEC TS 62433-1:2011, *EMC IC modelling – Part 1: General modelling framework*

CISPR 17, *Methods of measurement of the suppression characteristics of passive EMC filtering devices*

3 Terms, definitions, abbreviations and conventions

3.1 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

3.1.1

external terminal

terminal of an integrated circuit (IC) macro-model which interfaces the model to the external environment of the IC

EXAMPLE Power supply pins and input/output pins.

Note 1 to entry: In this part of IEC 62433, the name of each external terminal starts with "ET".

3.1.2

internal terminal

terminal of an integrated circuit (IC) macro-model's component which interfaces the component to other components of the IC macro-model

Note 1 to entry: In this part of IEC 62433, the name of each internal terminal starts with "IT".

3.1.3

section

XML element placed one level below the root element or within another section and that contains one or more XML elements, but no value

3.1.4

parent

keyword which is one level above another keyword

3.1.5

child

keyword which is one level below another keyword

3.1.6

parser

tool for syntactic analysis of data that is encoded in a specified format

3.1.7

***S*-parameter**

scattering parameter

S_{ij}

element of the *S*-matrix expressing the transmission and reflection coefficients of a device

Note 1 to entry: The most commonly used, each *S*-parameter relates the complex electric field strength of a reflected or transmitted wave to that of an incident wave; the subscripts of a typical *S*-parameter S_{ij} refer to the output and input ports related by the *S*-parameter, which may vary with frequency.

[SOURCE: CISPR 17:2011, 3.1.13]

3.1.8

CEML

Conducted Emissions Markup Language

data exchange format for ICEM-CE macro-model

Note 1 to entry: This note applies to the French language only.

3.1.9

CEMLBase

Conducted Emissions Markup Language Base

abstract type from which all CEML model components are directly or indirectly derived in the ICEM-CE macro-model definition

3.1.10**PDN**

Passive Distribution Network

component of an IC model that represents the characteristics of propagation path of electromagnetic noises such as power distribution network

[SOURCE: IEC TS 62433-1:2011, 3.4]

3.1.11**IA**

Internal Activity

component of an IC model represented by a current or voltage source, which originates in activity of active devices in an IC or in a portion of the IC

[SOURCE: IEC TS 62433-1:2011, 3.3]

3.1.12**IBC**

Inter-Block Coupling

network of passive elements that presents a coupling effect between circuit blocks within an IC

[SOURCE: IEC TS 62433-1:2011, 3.5]

3.1.13**VNA**

Vector Network Analyzer

network analyzer capable of measuring complex values of the S -parameters

[SOURCE: CISPR 16-1-4:2010, 3.1.21, modified — for VNA with two or more ports]

3.2 Abbreviations

CEM	Conducted Emission Model
XML	eXtensible Markup Language
SPICE	Simulation Program with Integrated Circuit Emphasis

3.3 Conventions

For the sake of clarity, but with some exceptions, the writing conventions of XML have been used in text and tables.

The symbol “ μ ” is used in the text part to define micro = $1\text{e-}6$. The symbol “u” is used in the XML parts to define the micro = $1\text{e-}6$.

4 Philosophy**4.1 General**

Integrated circuits will have more and more gates on silicon and technical progress will develop faster. To predict the electromagnetic behaviour of equipment, it is required to model the switching of the input and output interface and the internal activities of an IC effectively.

Figure 1 depicts an example of decomposition of an IC to enable conducted emissions analysis. The internal digital activity (culprit) is a source of electromagnetic noise that originates in switching of active devices. The coupling path propagates the emissions to the

IC's external terminals: pins/pads. The coupling path is the power distribution network or I/O lines inside the IC.

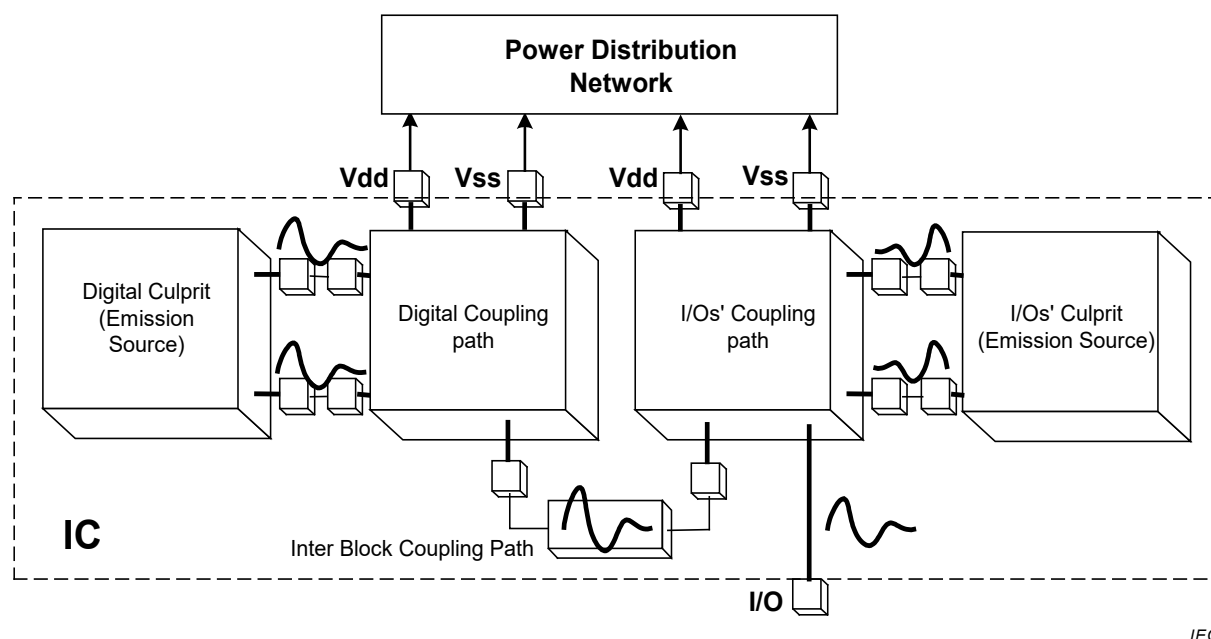


Figure 1 – Decomposition example of a digital IC for conducted emissions analysis

4.2 Conducted emission from core activity (digital culprit)

The current transients are created in the core area on the IC-die. Due to the characteristics of the digital coupling paths, the passive distribution network on printed circuit board (PCB) and the availability of on-chip decoupling, a portion of these current transients will occur at the power supply pins of the IC.

NOTE These off-chip power supply currents can be measured according to the IEC 61967 series [1]¹.

4.3 Conducted emission from I/O activity

I/Os activities may create voltage fluctuations of power and ground levels, and conducted emissions appear at power and ground pins through the I/Os' coupling path. And the output signals at output pins themselves are sources of conducted emissions to the printed circuit boards.

NOTE The measurement set-up is done according to the IEC 61967 series [1].

4.4 Data exchange format

ICEM-CE macro-model data is arranged in a decipherable nested manner using XML format. The objective of this exchange format, called Conducted Emissions Markup Language (CEML), is to create simple and practical universal access to ICEM-CE macro-model. The preliminary definitions for XML representation is given in Annex A. The CEML keywords and their usage rules are detailed in Annex B. An example ICEM-CE macro-model, in CEML format, is given in Annex C.

¹ Numbers in square brackets refer to the Bibliography.

5 ICEM-CE basic components

5.1 General

The basic components are component parts of the IC macro-model or block component or sub-model component. The basic components of the ICEM-CE macro-model are defined in 5.2 and 5.3.

NOTE The block component and the sub-model component are defined in 6.3.1 and 6.4.1 respectively.

5.2 Internal Activity (IA)

5.2.1 General

The Internal Activity (IA) component is the electromagnetic noise source that originates in switching of active devices in the IC or in a portion of the IC. This component is applicable for both analogue and digital circuitry.

The IA is described using an independent current source or an independent voltage source with two internal terminals. Figure 2 illustrates an example of an independent current source.

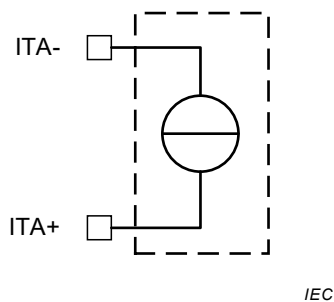


Figure 2 – IA component in the case of a current source

The characteristics of IA component are typically described in the time domain, and the characteristics can also be described in the frequency domain.

The description of an IA component shall contain the following information.

- Name of the IA component
- Names of its internal terminals
- Operational mode or test vector
- Domain (time or frequency)
- Definition of origin of time, and cycle-time for the operational mode (for time domain)
- Definition of origin of phase (for frequency domain)
- Operational conditions and applicable ranges
 - a) Power supply voltage ranges
 - b) Temperature range
 - c) Frequency range
- Characteristics of the IA
 - a) Current or voltage waveform over the whole cycle-time (for time domain)
 - b) Current or voltage amplitude and phase, versus frequency over the whole frequency range (for frequency domain)

5.2.2 Examples of IA

5.2.2.1 Time-domain IA example

Figure 3 shows an example of characteristics of IA in the time domain. The waveform depends on the specific operational mode of function. A simple waveform such as a triangular waveform can be used for the component description.

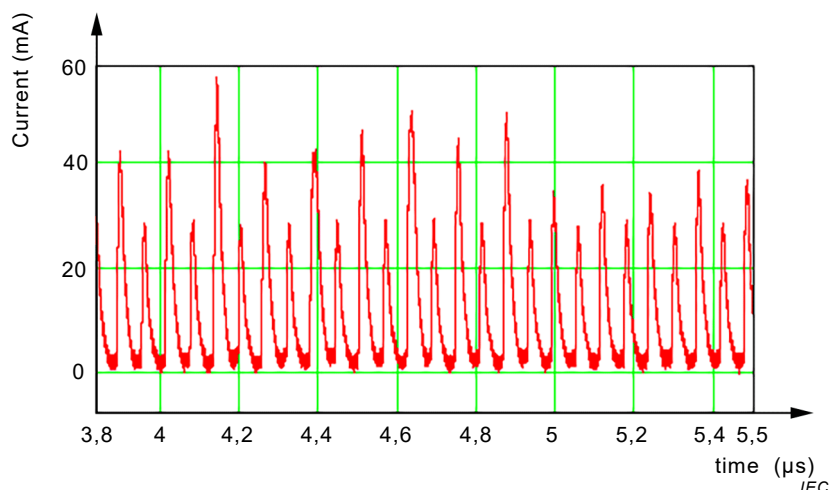


Figure 3 – Example of IA characteristics in the time domain

5.2.2.2 Frequency-domain IA example

Figure 4 shows an example of characteristics of IA in the frequency domain.

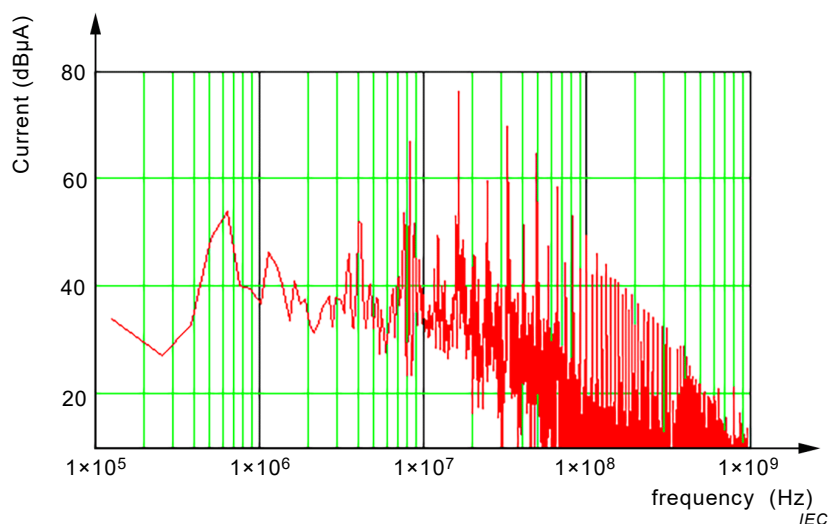


Figure 4 – Example of IA characteristics in the frequency domain

5.3 Passive Distribution Network (PDN)

5.3.1 General

The Passive Distribution Network (PDN) component presents the characteristics of propagation path of electromagnetic noises such as power distribution network (part of the PDN). The PDN can be linear or non-linear.

The PDN consists of passive elements, and is equipped with internal and external terminals.

For cases when the PDN can be assumed to be linear, it can be specified by different network parameters such as:

- $Z(f)$: Impedance, which is the ratio of voltage and current at the disturbance input of the PDN. It represents the electrical schematic of passive input impedance, often consisting of parasitic elements (resistance, inductance and capacitance)
- $Y(f)$: Admittance, which is the inverse of $Z(f)$
- $S(f)$: S -parameter, which is the ratio of the reverse and forward voltage waves at the disturbance input of the PDN. This parameter is typically used to measure the characteristics of Radio Frequency (RF) signal ports

The conversion between the three types of parameters is described in Annex D. The frequency validity range of the PDN is defined by the measurement conditions.

For linear or non-linear PDN, it can be described using a SPICE like netlist [2].

The description of a PDN component shall contain the following information.

- Name of the PDN component
- Names of its internal terminals and external terminals
- Applicable ranges
 - Power supply voltage range
 - Temperature range
 - Applicable load conditions if the PDN is for output
 - Applicable frequency range
- Characteristics of the PDN

5.3.2 Examples of PDN

Figure 5 shows an example of a four-terminal PDN using lumped elements. The ETVdd and ETVss are two external terminals of the PDN. The IT0 and the IT1 are two internal terminals.

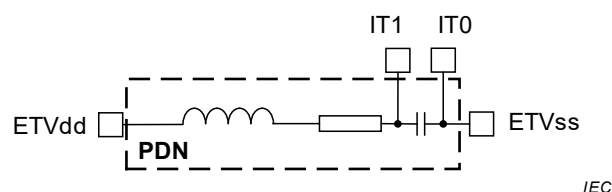


Figure 5 – Example of a four-terminal PDN using lumped elements

Figure 6 depicts the seven-terminal PDN structure using distributed elements such as transmission lines. The ETVxx are the four external terminals, the ITVxx are two internal terminals and the ETGnd is the common ground of the four transmission lines, connected to the PCB ground.

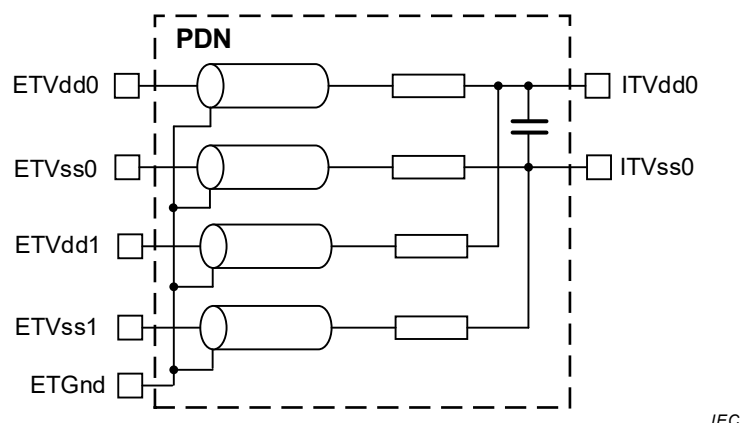


Figure 6 – Example of a seven-terminal PDN using distributed elements

Figure 7 shows an example of a twelve-terminal PDN using scattering parameters in a matrix format (black box). The ET1 to ET6 are external terminals. The IT1 to IT6 are internal terminals. A ground plane below the modelled IC is taken as an ideal reference ground for these terminals.

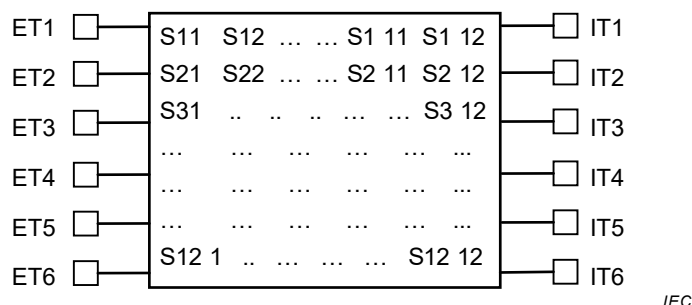


Figure 7 – Example of a twelve-terminal PDN using matrix representation

6 IC macro-models

6.1 Types of IC macro-models

An IC is modelled as an IC macro-model. Three types of IC macro-models are possible: general model, block-based model and sub-model-based model as defined in 6.2, 6.3 and 6.4 respectively.

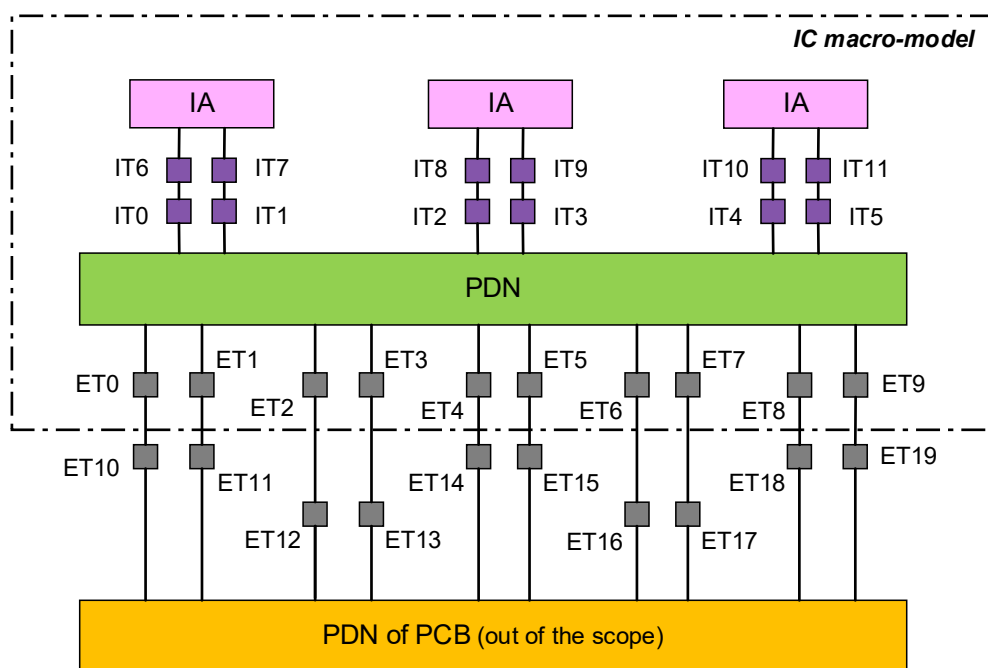
The description of an IC macro-model shall contain the following information for model circulation.

- Name of the IC macro-model
- Type of the model, general model or block-based model or sub-model-based model
- Names of components that are included in the IC macro-model
- Names of its external terminals
- Connections of the internal terminals of its components

6.2 General IC macro-model

The general model consists of a single PDN and one or more IAs as shown in Figure 8. The PDN shall include both the whole PDN on the IC die(s) and the whole PDN of the package. An on-chip decoupling capacitor shall also be included in the PDN if it exists.

NOTE This structure is suitable for the model circulation to IC users because of the least disclosure of proprietary information of the IC vendor.



IEC

Figure 8 – General IC macro-model

6.3 Block-based IC macro-model

6.3.1 Block component

6.3.1.1 General

The block component represents EMC properties of a specific functional block of an IC such as embedded memory.

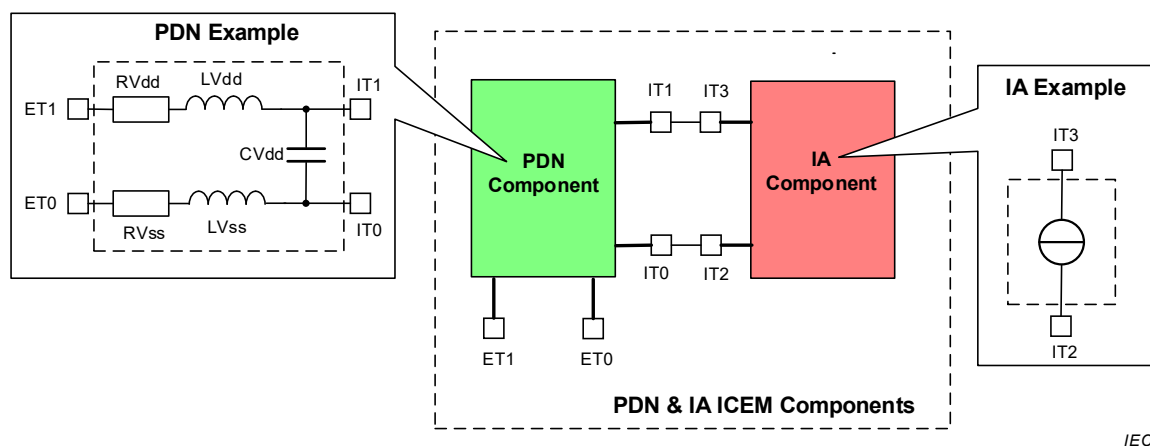
The block component consists of a single PDN and one or more IAs. The PDN includes PDN of the specific functional block, a portion of global power/ground network and a portion of package PDN, which are directly involved in the block's functionality. The on-chip decoupling capacitor is a part of the PDN. The component is equipped with external terminals and internal terminals.

The description of a block component shall include the following information.

- Name of the block component
- Names of the basic components that make up the block component
- Connections of the internal terminals of its basic components

6.3.1.2 Example of block-component with a single PDN and IA

Figure 9 shows an example of block component. The block consists of an IA and a PDN. The internal terminals of the IA are connected to the internal terminals of the PDN.

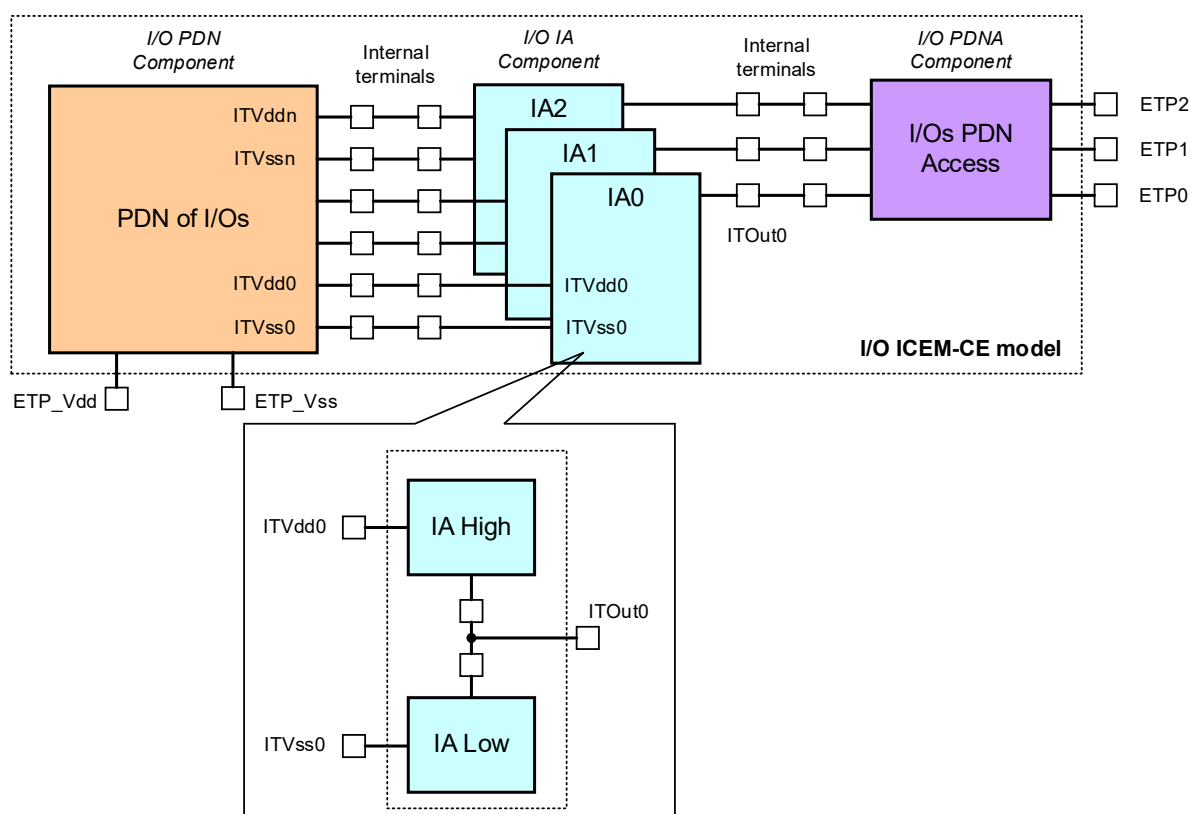


IEC

Figure 9 – Example of block component with a single IA

6.3.1.3 Example of block-component with multiple IAs

Figure 10 depicts a three I/Os model. The I/O PDN component describes how I/Os are powered and the I/O PDNA describes noise transfer characteristics among terminals. The I/O IA components describe the current activity. They are built up using two IA components; one to specify the high state behaviour and the other one to specify the low state. Figure 10 shows the two types of I/O PDN components of the complete I/O model.



IEC

Figure 10 – Example of block components for I/Os

6.3.2 Inter-Block Coupling component (IBC)

6.3.2.1 General

The Inter-Block Coupling (IBC) is a network of passive elements that presents a coupling effect between blocks. The IBC is equipped with two or more internal terminals. Such blocks can be used for modelling the coupling phenomena between different IC ground terminals.

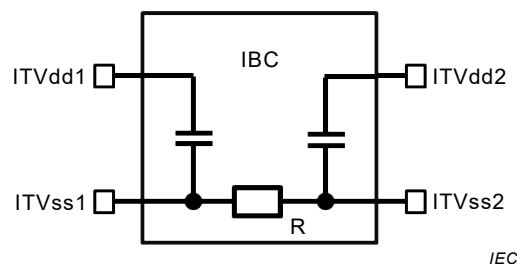
substrate losses, mutual inductances at die-level, insulation between internal ground and power terminals.

The description of an IBC component shall contain the following information.

- a) Name of the IBC
- b) Names of its internal terminals
- c) Applicable ranges
 - 1) Power supply voltage ranges
 - 2) Temperature range
 - 3) Applicable frequency range
- d) Characteristics of the IBC

6.3.2.2 Example of an IBC and its relationship with the PDN

An example of an IBC network is shown in Figure 11. In this example, both capacitors model the dielectric properties and the resistor models the resistive properties of the substrate. Other properties can be modelled using more complex IBC networks.



NOTE ITVdd1, ITVss1, ITVdd2 and ITVss2 are internal terminals.

Figure 11 – Example of IBC with four internal terminals

The relationship between different PDN blocks and the IBC is shown in Figure 12.

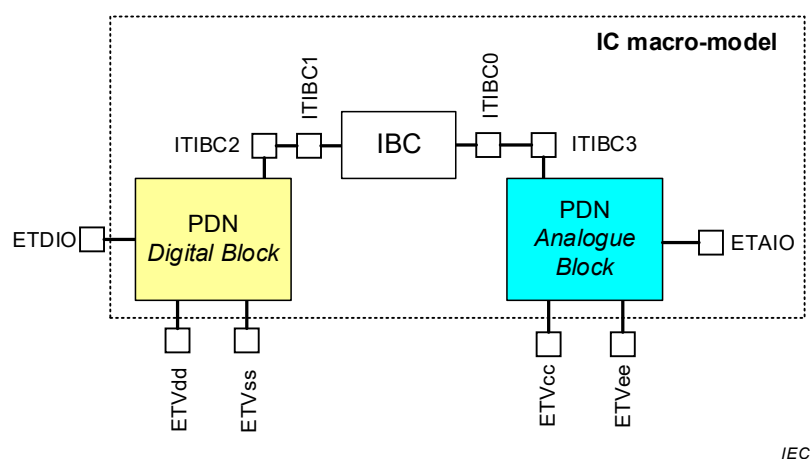


Figure 12 – Relationship between blocks and IBC

6.3.3 Block-based IC macro-model structure

An example of a block-based structure is shown in Figure 13. The model consists of block components and IBC components. The PDN of global wiring on die and the PDN on package are incorporated into the PDNs of blocks.

NOTE 1 This structure is suitable for modelling from measurements.

NOTE 2 By combining PDNs of block and IBCs, this structure can be converted into general structure.

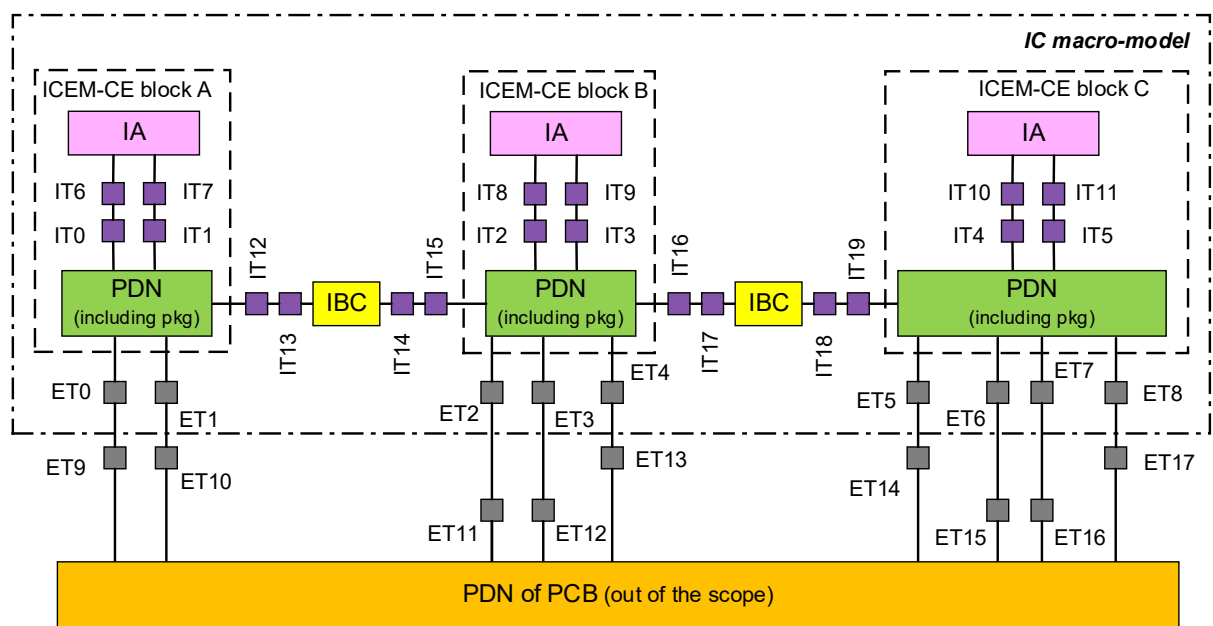


Figure 13 – Block-based IC macro-model

Figure 14 depicts an example of block-based IC macro-model. Each block has two external terminals and three internal terminals. Three IBC blocks interconnect the internal terminals, IT2, IT3, IT4, IT5 and IT6. In this example, IBCs are used to model the substrate coupling caused by the sheet resistance between the three internal ground terminals.

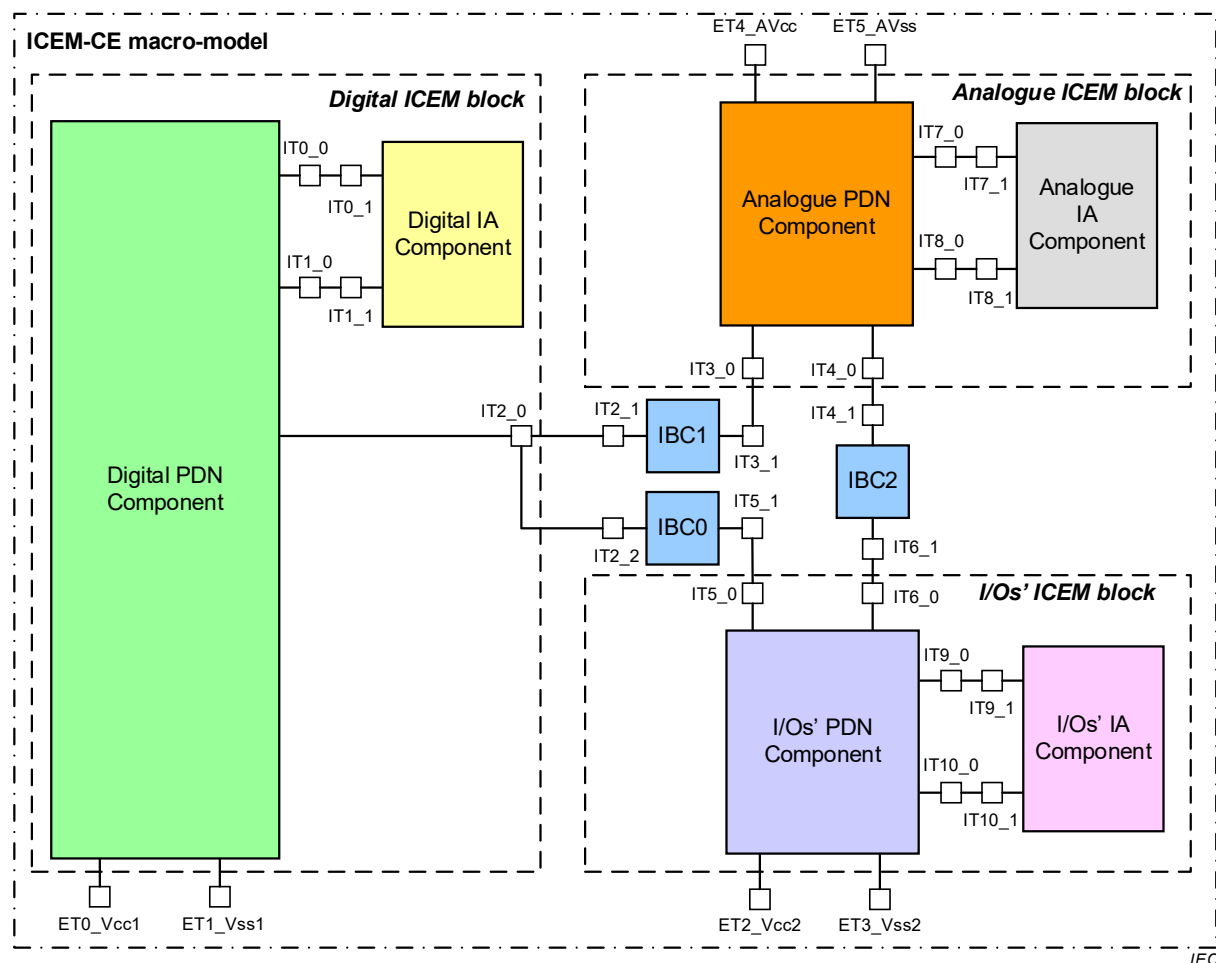


Figure 14 – Example of block-based IC macro-model

6.4 Sub-model-based IC macro-model

6.4.1 Sub-model component

The sub-model in Figure 16 represents the electromagnetic behaviour of specific functional circuits of IC. Some examples include an IP block, functional circuits such as embedded memory, Central Processing Unit (CPU) core, etc. These blocks can be modelled using the sub-model; it can be repeatedly used in the IC and/or other ICs.

The sub-model consists of a single PDN and one or more IAs. This PDN is a PDN of the specific functional circuits. The sub-model is equipped with internal terminals but does not have any external terminals. An example of a simple sub-model is shown in Figure 15.

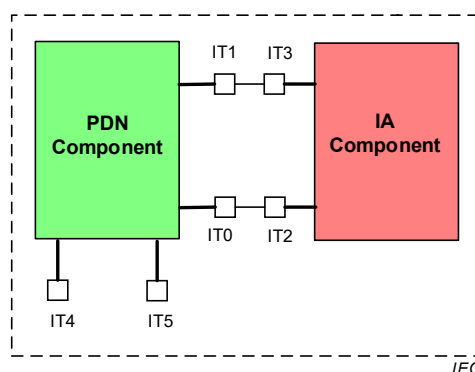


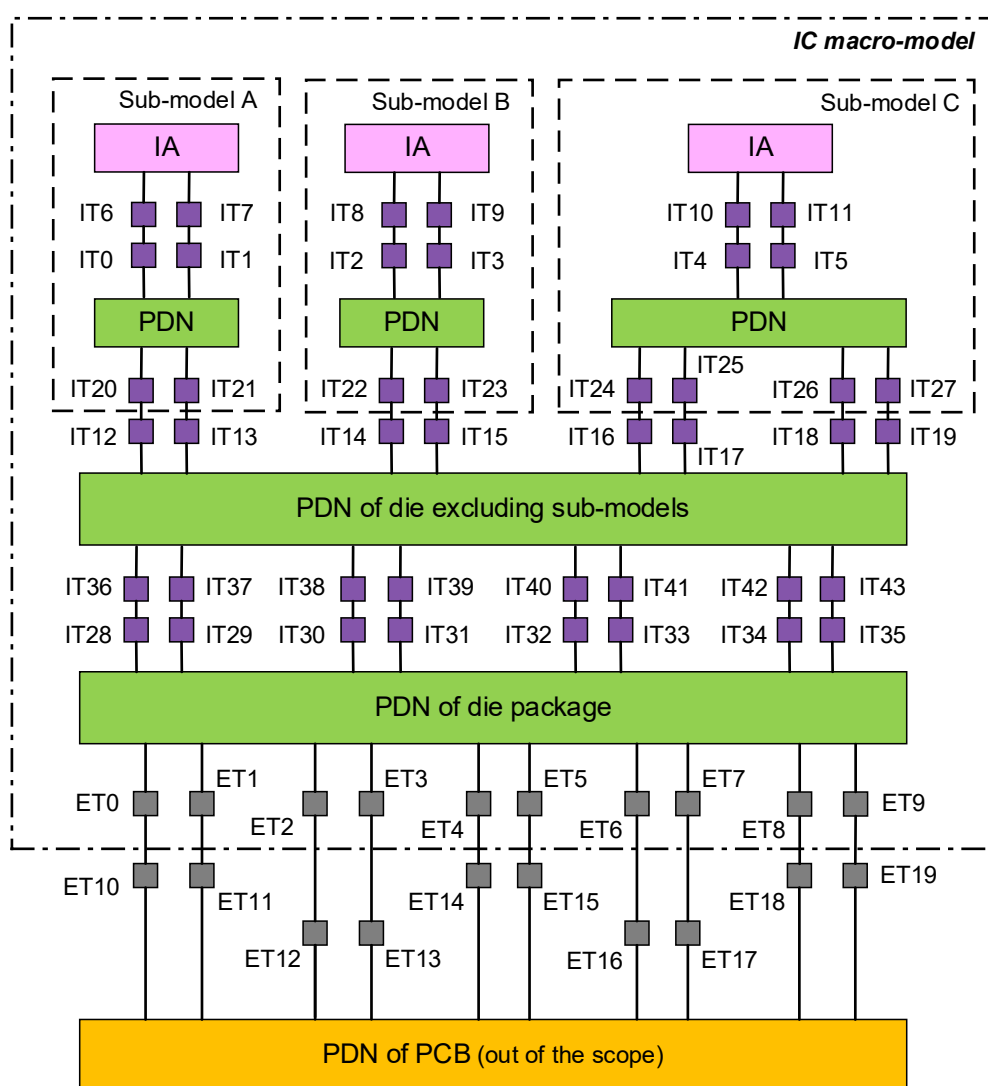
Figure 15 – Example of simple sub-model

The sub-model description shall contain the following information.

- Name of the sub-model
- Names of its internal terminals
- Names of the basic components that are included in the sub-model
- Connections of the internal terminals of basic components

6.4.2 Sub-model-based IC macro-model structure

The sub-model-based structure is shown in Figure 16. It consists of one or more sub-models, a PDN of the die, and a PDN of the package. The PDN of the die includes the global power/ground network of the IC die and the on-chip decoupling capacitor if it exists, but does not include the PDNs that belong to sub-models. Some IAs such as IAs for standard cell circuits can be directly connected to the PDN of the die (not shown in the figure).



IEC

Figure 16 – Sub-model-based IC macro-model

NOTE 1 This structure is suitable for modelling using IC design information.

NOTE 2 By combining PDNs of the whole IC macro-model, a sub-model-based IC macro-model can be converted into a general IC macro-model.

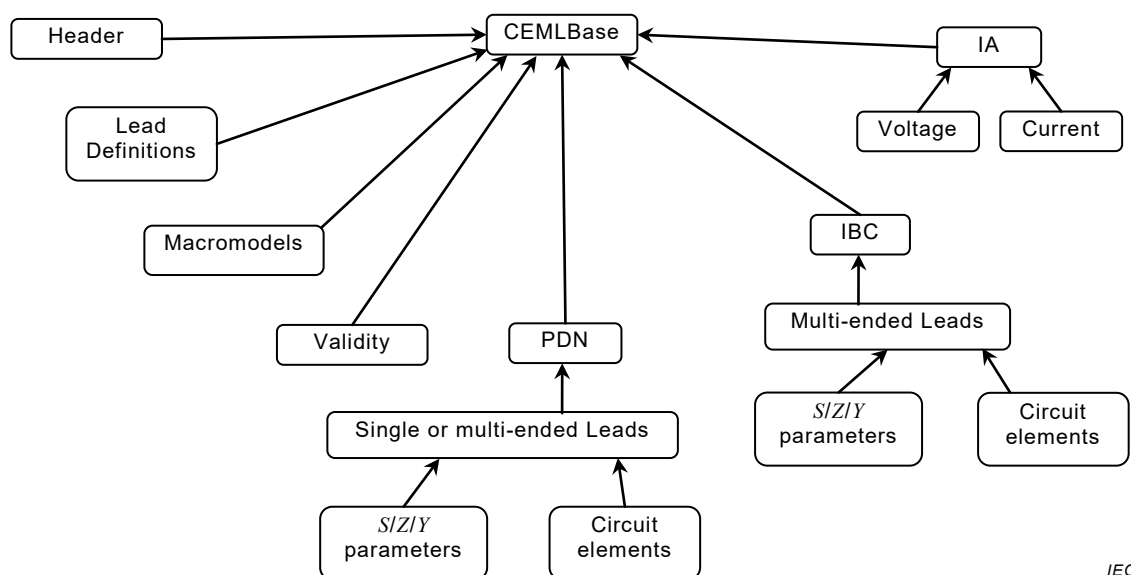
7 CEML format

7.1 General

Data of the ICEM-CE macro-model is arranged in xml format, henceforth called as Conducted Emissions Markup Language (CEML). The model data is separated into seven parts:

- Header containing general information
- Description of IC's Lead or Node definitions
- Description of macro-models for use within the PDN and IBC data
- Description of the model validity conditions
- Description of the PDN
- Description of the IBC
- Description of the IA

The inheritance hierarchy is depicted in Figure 17.



IEC

Figure 17 – CEML inheritance hierarchy

The top level of a CEML model definition simply consists of these components:

```

beginning of model definition
  model header definition
  DUT lead definitions
  SPICE macro-models
  model validity conditions
  model PDN
  model IBC
  model IA
end of model definition
  
```

This exchange format uses eXtensible Markup Language (XML) 1.0 (Fourth Edition) to structure the information. XML is derived from the Standard Generalized Markup Language (SGML) [3].

The rules below ensure that the file can be correctly parsed by a Conducted Emissions Model (CEM) parser. An example CEML file is shown in Annex C.

7.2 CEML structure

Considering all the rules defined in Annex A, and being consistent with model description requirements in IEC TS 62433-1, a typical ICEM-CE macro-model file is represented as shown below:

```
<?xml version="1.0" encoding="UTF-8"?>
<CEmodel>
<!-- HEADER -->
  <Header>
    ...
  </Header>
<!-- DUT LEAD DEFINITIONS-->
  <Lead_definitions>
    ...
  </Lead_definitions>
<!-- SPICE MACROMODEL DEFINITIONS -->
  <Macromodels>
    ...
  </Macromodels>
<!-- MODEL VALIDITY CONDITIONS -->
  <Validity>
    ...
  </Validity>
<!-- MODEL PDN DATA -->
  <Pdn>
    ...
  </Pdn>
<!-- MODEL IBC DATA -->
  <Ibc>
    ...
  </Ibc>
<!-- MODEL IA DATA -->
  <Ia>
    ...
  </Ia>
</CEmodel>
```

The *Header*, *Lead_definitions*, *Validity*, *Pdn*, and *Ia* sections are minimum and mandatory information of the ICEM-CE macro-model. The *Macromodels* and *Ibc* sections are optional. Each section of the file is presented below.

7.3 Global keywords

Documentation, *Notes*, *Unit* sections are global keywords and can be placed anywhere in the file except within an element containing a value. See B.4 for more information on usage.

7.4 Header section

Readers may wonder about the motivations for an independent Header section. A simpler approach to creating the header information would be to place them all directly at the top level under <CEmodel> ... </CEmodel>. It is instead chosen to group them within the XML element <Header>, because such an approach would help to organize the components in a better format and make visual reading of model definitions easier. It is thus proposed to define header information within the *Header* tag. The minimum details are the model version number, filename and the file version number. Other header contents are freely dimensioned as per IEC TS 62433-1, giving information such as:

- DUT reference
- Authors' name

- Date
- Measurement method
- Copyright
- Disclaimer
- Documentation

EXAMPLE An example *Header* section may be defined as shown

```
<Header>
  <Cem_ver>1.0</Cem_ver>
  <Filename>ExampleICEMCE_file.CEML</Filename>
  <File_ver>1.0</File_ver>
  <Author>Valeo 1</Author>
  <Dut>MCYY/Dut>
  <Date>April 3, 2013</Date>
  <Meas_method> PDN using Netlist, IA in time-domain </Meas_method>
</Header>
```

A detailed list of valid keywords under the *Header* section is available in Annex B.

7.5 Lead definitions

This section describes the various leads or terminals of the IC under test. Each lead in the *Lead_definitions* section is made using the *Lead* tag, whose definition is shown in Table 1. Several *Lead* structures are listed one after another to form the *Lead_definitions* structure.

Table 1 – Attributes of Lead keyword in the *Lead_definitions* section

<i>Lead</i>
Id: pin identity as a valid string (required) Name: Name of the pin as designated in the datasheet (optional). Default = "None" Mode: Mode in which the pin is used for ICEM-CE ("GND") (optional). Default="None" Type: Type in which the pin is used for ICEM-CE ("internal", "external") (optional). Default="external"

Every *Lead* structure in the *Lead_definitions* section has one required field, *Id*, representing the identity of the lead. The lead may be additionally defined with its *Name*, *Mode* and *Type* fields. If the *Name* field is undefined or absent, it defaults to "None". The *Mode* field is considered as "None" under the following conditions:

- If *Mode* for a particular pin is absent
- If explicitly set as "None" since the respective pin is not GND (reference return path).

The *Type* field is optional and can be "external" or "internal". See 3.1.1 and 3.1.2 for the definition of these terms.

Leads with *Type*="external" are used to interface the model to the external environment of the IC, such as power supply pins and I/O pins. The "internal" terminals are used for interfacing different blocks of the block-based or sub-model-based ICEM-CE macro-model (as explained in 6.3 and 6.4). The "internal" terminals shall be defined with *Mode*="None". These leads are dedicated for interfacing the different blocks of the block-based or sub-model-based ICEM-CE components.

Leads that are defined with *Mode*="GND" are considered as "external" terminals.

NOTE Internal ground terminals can be specified as any other internal node. i.e., *Type*="internal" with *Mode*="None".

Table 2 lists the compatibility between the *Mode* and *Type* fields of the *Lead* structure for correct CEML annotation by the CEM parser.

Table 2 – Compatibility between the Mode and Type fields for correct CEML annotation

Mode	Type	
	external	internal
GND	Yes	No
None	Yes	Yes

An example of the *Lead_definitions* section, representing Figure 5, is shown below.

```
<Lead_definitions>
  <Lead Id="1" Name="ETVdd" Type="external"/>
  <Lead Id="2" Name="IT1" Type="internal" Mode="None"/>
  <Lead Id="3" Name="IT0" Type="internal"/>
  <Lead Id="4" Name="ETVss" Mode="GND"/>
</Lead_definitions>
```

7.6 SPICE macro-models

This section described the various SPICE (Simulation Program with Integrated Circuit Emphasis) macro-models in netlist format [2].

These sub-circuits are references within the *Pdn* tag under the Netlist section as explained in 7.8.3.4 and 7.8.3.5. Each SPICE macro-model in the *Macromodels* section is defined using the *Subckt* tag, whose definition is shown in Table 3. This section is optional for the ICEM-CE macro-model.

Table 3 – Subckt definition

<i>Subckt</i>
Name: Name of the macro-model (required)
Nodes: External Nodes connecting to the main circuit (required)
Kind: SPICE netlist format (optional) default: "SPICE3"
Data_files: Macro-model defined in an external file (optional)

The *Subckt* keyword has two required fields: *Name* and *Nodes*. The *Name* field consists of letters and numbers defined in A.2.5.4. Since these definitions share the same namespace within the CEML format, every sub-circuit shall carry a unique name

The *Nodes* field defines the external nodes that connect outside the sub-circuit, i.e., the nodes through which the sub-circuit element connects to the main circuit. The different external nodes are defined in sequence separated by a comma (",") and these nodes are strictly local to the *Subckt* definition. These *Nodes* can be identified with either numbers and/or characters [2].

The optional attribute *Kind* tells the CEM parser that the defined sub-circuit follows a specific syntax. CEML version 1 supports industry-standard SPICE-like netlist syntaxes [2]. The *Kind* field defaults to generic "SPICE3" if absent.

An example *Subckt* element is shown in Example 1.

EXAMPLE 1 An example of a SPICE-like *Subckt* element.

```
<Subckt Name="PDN_pin1" Nodes="Node1,Node2" Kind="SPICE3">
  L1 Node1 int1 2.2e-9
  R1 int1 int2 10e-3
  C1 int2 int3 800e-12
  L1 int3 Node2 2.2e-9
</Subckt>
```

For compatibility reasons between several SPICE kinds, "0" is not allowed as an external node. The different sub-circuit elements are listed one after another to form the *Macromodels* section (see Example 2).

EXAMPLE 2 An example of a *Macromodels* section with several *Subckt* elements.

```
<Macromodels>
  </Subckt Name="PDN_example1" Nodes="Node1,Node2">
    ...
  </Subckt>
  </Subckt Name="PDN_example2" Nodes="Node1,Node2,Node3">
    ...
  </Subckt>
  ...
</Macromodels>
```

Once defined, the various sub-circuits are usable anywhere within the *Pdn* and *Ibc* tags using an identifier starting with the character "X". The call shall be made under the *Netlist* section and is referenced using the *Name* field (see 7.8.3.4 and 7.8.3.5). The number of nodes on the call line shall match the number listed in the *Nodes* attribute line of the specific *Subckt*. For example the above defined "PDN_pin1" (in Example 1) can be called:

```
<Pdn>
  ...
  <Lead Id="1" Type="Ckt">
    ...
    <Netlist>
      Xpin1 1 4 PDN_pin1
    </Netlist>
  </Lead>
  ...
</Pdn>
```

If one or more sub-circuit models is (are) defined in an external library file, then the file(s) is (are) referenced using the *Data_files* tag as shown in Example 3.

EXAMPLE 3 An example of a *Macromodels* section with several *Subckt* elements defined in external data files.

```
<Macromodels>
  <Subckt Kind="SPICE3">
    <Data_files>
      subckt_example1.lib
      subckt_example2.lib
      ...
    </Data_files>
  </Subckt>
</Macromodels>
```

A typical sub-circuit definition is shown in Figure 18.

```

*Netlist file created on Fri 27 Nov 2012
*Time: 12:01:27
*PDN macro-model model of PIN1 with respect to GND1
.Subckt Pin1_PDN N1 N2
C1 N1 int1 20e-9
L1 int1 int2 9e-9
R1 int2 N2 230e-3
.ends

```

IEC

Figure 18 – Example of a netlist file defining a sub-circuit

While using external data files containing the SPICE model libraries, to avoid ambiguity, the CEM parser parses only the data statements defined within the SPICE keywords: ".SUBCKT" and ".ENDS". These keywords are not case sensitive [2]. If multiple ".SUBCKT" sections are found, then they are parsed as independent sub-circuit elements.

7.7 Validity section

7.7.1 General

The *Validity* section shall be used to represent the conditions in which the ICEM-CE macro-model is defined as per IEC TS 62433-1. This section is strictly informative to the user and independent of all other sections in the CEML file.

Table 4 lists the various recognized keywords in the *Validity* section.

Table 4 – Definition of the *Validity* section

<i>Validity</i>
Power_supply: Power supply range as a string (required)
Temperature_range: Temperature range in which the model is extracted with units (required)
Frequency_range: Frequency range as a string with units (required for frequency-domain model data)
Phase_origin: Origin of phase for frequency-domain IA (required for frequency-domain model data)
Time_range: Time range as a string with units (required for time-domain model data)
Time_origin: Origin of time for time-domain IA (required for time-domain model data)

This definition is not exhaustive and is open for progress and improvement.

Documentation pertaining to the IC such as datasheets, test-reports and ICEM-CE extraction report shall be specified with the global keyword *Documentation* in the form of a valid string that specifies the path to the corresponding documentation. Many file paths' can be listed one below another. See B.4 for more information on the usage of the *Documentation* keyword.

Any other specific details pertaining to the complete ICEM-CE macro-model definition, that may be needed to understand the model, shall be defined with the *Notes* section in the form of a valid string. See B.4 for more information on the usage of the *Notes* keyword.

This section is mandatory and is defined directly under the *CEmodel* root element as shown in Example 1 and Example 2.

EXAMPLE 1 An example of the *Validity* section of a frequency-domain ICEM-CE macro-model is shown below.

```

<CEmodel>
...
<Validity>

```

```

    <Power_supply>5V</Power_supply>
    <Temperature_range>[20Celsius-40Celsius]</Temperature_range>
    <Frequency_range>[1MHz - 1GHz]</Frequency_range>
    <Phase_origin>10degrees</Phase_origin>
    <Notes>Only PLL is activated</Notes>
    <Documentation>
        test_report.pdf
        ICdatasheet.pdf
    </Documentation>
</Validity>
...
</CEmodel>

```

EXAMPLE 2 An example of the *Validity* section of a time-domain ICEM-CE macro-model is shown below:

```

<CEmodel>
...
<Validity>
    <Power_supply>5V</Power_supply>
    <Temperature_range>[20Celsius-40Celsius]</Temperature_range>
    <Time_range>from 1us to 101us</Time_range>
    <Time_origin>1us</Time_origin>
    <Notes>Only PLL is activated</Notes>
    <Documentation>
        test_report.pdf
        ICdatasheet.pdf
    </Documentation>
</Validity>
...
</CEmodel>

```

7.7.2 Attribute definitions

7.7.2.1 Power_supply

The *Power_supply* attribute is used define the supply conditions for which ICEM-CE is valid. This attribute informs the user that the ICEM-CE macro-model is extracted in the specified supply range. This is a required field.

There is no particular format for defining the value for this attribute; it shall be easily comprehensible for proper model usage. The value shall be a data string containing valid text string and/or numerical values with units. See A.2.5.5 for valid units.

A few examples are shown in Examples 1 to 3.

EXAMPLE 1 The following syntax specifies that the model is defined for a supply voltage of 5 V.

```
<Power_supply>5V</Power_supply>
```

EXAMPLE 2 The following syntax specifies that the digital blocks of the model data are defined with a supply voltage of 5 V and the analogue blocks are defined for a supply voltage of 12 V.

```
<Power_supply>5V for digital blocks, 12V for analogue blocks</Power_supply>
```

EXAMPLE 3 The following syntaxes specify that the model is defined for supply voltages between 2,5 V and 18 V.

```
<Power_supply>[2.5V-18V]</Power_supply>
```

or

```
<Power_supply>between 2.5V and 18V</Power_supply>
```

7.7.2.2 Temperature_range

The *Temperature_range* attribute is used define the temperature range in which ICEM-CE is extracted. This is a required field.

There is no particular format for defining the value for this attribute; it shall be easily comprehensible for proper model usage. The value shall be a data string containing valid text string and/or numerical values with units. See A.2.5.5 for valid units.

Two examples are shown in Example 1 and Example 2.

EXAMPLE 1 The following syntaxes specify that the model is valid in the temperature range between 20 °C and 40 °C.

```
<Temperature_range>[20Celsius-40Celsius]</Temperature_range>
or
<Temperature_range>from 20Celsius to 40Celsius</Temperature_range>
```

EXAMPLE 2 The following syntax specifies that model is defined only at 298,15 K.

```
<Temperature_range>298.15K</Temperature_range>
```

The temperature range of validity of the ICEM-CE macro-model shall be the common temperature range of the PDN and IA components.

7.7.2.3 Frequency_range

The *Frequency_range* attribute is used define the frequency range in which ICEM-CE is valid. This attribute informs the user that the ICEM-CE macro-model is extracted in the frequency domain, is specified in the given frequency range and is usable in the same range. This is a required field for frequency-domain ICEM-CE macro-model.

There is no particular format for defining the value for this attribute; it shall be easily comprehensible for proper model usage. The value shall be a data string containing valid text string and/or numerical values with units (see Example 1).

EXAMPLE 1 The following syntaxes specify that the model is valid in the frequency range from 1 MHz to 1 GHz.

```
<Frequency_range>[1MHz-1GHz]</Frequency_range>
or
<Frequency_range>from 1MHz to 1GHz</Frequency_range>
```

The frequency range of validity of the ICEM-CE macro-model shall be the common frequency range of the PDN and IA components.

7.7.2.4 Phase_origin

The *Phase_origin* attribute is used define the origin of phase in which ICEM-CE is valid. This information is specific to the IA sub-model. This is a required field for frequency-domain ICEM-CE macro-model.

There is no particular format for defining the value for this attribute; it shall be easily comprehensible for proper model usage. The value of this attribute is a data string containing valid text string and/or numerical values with units (see Examples 1 and 2).

EXAMPLE 1 The following syntax specifies that the phase origin of the model data (IA) is 10 degrees.

```
<Phase_origin>10degrees</Phase_origin>
```

EXAMPLE 2 The following syntax specifies that the phase origin of the IA data of the digital block is 5 degrees and that of the analogue block is 22 degrees.

```
<Phase_origin>5degrees for digital block, 22 degrees for analogue block</Phase_origin>
```

7.7.2.5 Time_range

The *Time_range* attribute is used to define the time range in which ICEM-CE is valid. This attribute informs the user that the ICEM-CE macro-model is extracted in the time-domain, is specified in the given time range and is usable in the same range. This is a required field for time-domain ICEM-CE macro-model.

There is no particular format for defining the value for this attribute; it shall be easily comprehensible for proper model usage. The value of this attribute is a data string containing valid text string and/or numerical values with units (see Example 1).

EXAMPLE 1 The following syntax specifies that the model is valid in the time range from 100 ps to 1 ms.

```
<Time_range>[100ps-1ms]</Time_range>
or
<Time_range>from 100ps to 1ms</Time_range>
```

The time range of validity of the ICEM-CE macro-model shall be the common time range of the PDN and IA components.

7.7.2.6 Time_origin

The *Time_origin* attribute is used to define the origin of time and cycle-time for the operational mode in which ICEM-CE is valid. This information is specific to the IA sub-model. This is a required field for time-domain ICEM-CE macro-model.

There is no particular format for defining the value for this attribute; it shall be easily comprehensible for proper model usage. The value of this attribute is a data string containing valid text string and/or numerical values with units (see Example 1).

EXAMPLE 1 The following syntax specifies that the time origin of the IA data is 100 ns.

```
<Time_origin>100ns</Time_origin>
```

EXAMPLE 2 The following syntax specifies that the time origin of the IA data of the digital block is 100 ps and that of the analogue block is 1 µs.

```
<Time_origin>100ps for digital block, 1us for analogue block</Time_origin>
```

7.8 PDN

7.8.1 General

The *Pdn* section of the ICEM-CE macro-model contains the PDN data that describes the model. The data shall be defined within the *Pdn* keyword as follows:

```
<Pdn>
    PDN data
</Pdn>
```

A PDN data is defined for a particular IC and thus the definition shall be done within a *Lead* tag. See Figure 17 for the structural hierarchy. Table 5 lists the various recognized fields of the *Lead* keyword:

Table 5 – Definition of the Lead keyword for *Pdn* section

<i>Lead</i>
Id : pin identity as a valid string (required)
Ground_id : return pin identity as a valid string (required if Type="S", "Z", "Y", else optional)
Blockname : PDN block name as a valid string (optional)
Type : PDN source parameter ("S", "Z", "Y", "Ckt")
Param_order : Order in which PDN parameters are defined
Format : Data format ("RI", "MA", "DB")
Meas_type : Method implemented for performing PDN measurements
Reference_impedance : Reference impedance used in performing PDN measurements
Use : Parameter that is to be specifically used
Netlist : PDN definition using standard netlist format (required if Type="Ckt")
Unit_freq : Unit definition of the frequency terms
Unit_param : Unit of the PDN parameters
Power_level : Measurement power level during PDN extraction
Data_files : PDN source parameter defined in an external file (required if not List)
List : PDN parameter list (required if not Data_files)

The PDN data can be defined as a single or multiple PDN blocks (interconnected or disconnected). As explained in 5.3.1, the PDN (or PDN blocks) can be represented using single or multi-port S -, Y - or Z - parameters (network parameters) or using SPICE type netlists [2]. In both the cases, data can be defined inline or using external data files.

The frequency range or time range of the PDN information shall be specified in the *Validity* section under the *Frequency_range* tag or the *Time_range* tag respectively.

NOTE The frequency range (or time range) of validity of the ICEM-CE macro-model is the common frequency range (or time range) of the PDN and IA components.

As per IEC TS 62433-1, any other relevant information required for correct understanding (and usage) of the PDN shall be optionally defined within the *Notes* and *Documentation* tags. Details such as IC's operating mode, decoupling capacitors on supply lines, power-supply ranges, activated functions, load conditions, grounding details, datasheets and test-reports can be defined. See B.4 for more information on the usage of the *Documentation* and *Notes* keywords.

7.8.2 Attribute definitions

7.8.2.1 Id

The lead identities (Ids) used in this PDN definition should have been previously defined in the *Lead_definitions* tag as described in 7.5. Both "internal" and "external" leads can be defined.

This is a required field.

7.8.2.2 Ground_id

The return lead identity (Ground_id) used in this PDN definition should have been previously defined in the *Lead_definitions* tag as described in 7.5. This lead defines the return signal path for defining the PDN.

This attribute is defined only if network parameters are used for defining the PDN (Type="S" or "Z" or "Y"). *Ground_id* attribute is optional if Type="Ckt", that is, if a netlist is used for representing the PDN. See 7.8.2.4 for more information on the *Type* attribute.

When used, only one unique *Ground_id* is permitted per *Lead* definition. The ground terminal shall be the ideal reference ground used for obtaining the network parameters as stated in 5.3.2.

This is an optional field and is required if PDN is represented using network parameters.

7.8.2.3 Blockname

The *Blockname* field is used to define the name of the PDN for block-based or sub-model-based ICEM-CE modelling. If both the PDN and the IA share the same name, then they are regrouped together by the CEM parser as a sub-model.

This field is optional.

7.8.2.4 Type

The *Type* attribute is used to represent the type of the PDN data. Valid types are:

- "S": *S*-parameter data
- "Z": *Z*-parameter data. These parameters are not normalized to the reference impedance.
- "Y": *Y*-parameter data. These parameters are not normalized to the reference impedance.
- "Ckt": Circuit description using netlists

This field is optional. When absent, the default value is "S".

7.8.2.5 Param_order

The *Param_order* attribute tells the CEM parser how the PDN data is represented. The following strings are dedicated for specifying the parameter order:

- "Freq" and "Frequency": Frequency quantity used for parameters' definition
- "Time": Time quantity used for parameters' definition
- "Sij": *S*-parameters, *i* and *j* are integers representing measurement ports/pins (example: "S11", "S21", "S31", etc.)
- "Zij": *Z*-parameters, *i* and *j* are integers representing measurement ports/pins (example: "Z11", "Z21", "Z31", etc.)
- "Yij": *Y*-parameters, *i* and *j* are integers representing measurement ports/pins (example: "Y11", "Y21", "Y31", etc.)
- All: All defined terms with respect to the number of lead Id's are considered.

The different terms of *Param_order* shall be separated by a comma character (",").

This field is optional. It is not defined if the PDN is represented as a Circuit model as discussed in 7.8.3.4 and 7.8.3.5. If absent, the default value is "Freq,All".

7.8.2.6 Format

The *Format* attribute decides the data format. It is not defined if the PDN is represented as a Circuit model as discussed in 7.8.3.4 and 7.8.3.5. Valid data formats are:

- "RI": real/imaginary format (frequency-domain PDN)
- "DB": Magnitude in decibel scale with phase angle in degrees (frequency-domain PDN)

- "MA": Magnitude in linear scale with phase angle in degrees (frequency-domain PDN)
- "DBMAG": magnitude in decibel scale without phase data (time-domain PDN)
- "MAG": Magnitude data in linear scale without phase data (time-domain PDN)

This field is optional. If absent, the default value is "RI".

7.8.2.7 Meas_type

The *Meas_type* attribute is used by the CEM parser in order to compute the impedance of the specific lead *Ids* when *S*-parameters (Type= "S") are used. Depending on the type of the PDN (single-ended, differential or multi-ended), the *Meas_type* attribute takes multiple definitions. CEM version 1 supports the following *Meas_type* values for single-ended PDN with *S*-parameters:

- "0" for standard or conventional measurement configuration (default)
- "1" when the DUT is in parallel to the measurement ports (shunt connection)
- "2" when the DUT is in series to the measurement ports (series connection)

For multi-ended PDN, *Meas_type* is always "0".

The different measurement configurations and set-up for single-ended PDN extraction shall conform to the CISPR 17 specifications. Some basic aspects are discussed in Annex E.

This field is defined only when Type="S" and is optional. When absent, it defaults to "0".

7.8.2.8 Reference_impedance

The *Reference_impedance* attribute is used by the CEM parser in order to compute the impedance of the specific lead *Id* when network parameters are used for PDN definition.

This field is defined only when Type="S" and is optional. Unless defined with a numerical value, the default is "50ohm" as per CISPR 17.

7.8.2.9 Use

The *Use* attribute tells the CEM parser to use one of the values in the *Param_order* attribute. When explicitly defined, the corresponding parameter is used as PDN. If undefined, the first "Sij" or "Zij" or "Yij" term is used by default.

This field is optional.

7.8.2.10 Netlist

The Netlist keyword is used to define the PDN as a SPICE netlist that represents the electrical connectivity of the PDN elements. The electrical connectivity flow shall be as per SPICE specification [2]. A detailed description of the circuit model is presented in 7.8.3.4 and 7.8.3.5.

This field is required if Type="Ckt".

7.8.2.11 Unit_freq and Unit_param

The parameter units are defined under the *Unit_freq* and *Unit_param* tags for frequency and data respectively. If absent, *Unit_freq* defaults to basic units, "Hz". Table 6 below shows the default value of the *Unit_param* attribute as a function of data type and format when undefined:

Table 6 – Valid data formats and their default units in the *Pdn* section

Data type	Data format	Default Unit_param values
S	RI	1
	MA or MAG	1
	DB or DBMAG	dB
Z	RI	ohm
	MA or MAG	ohm
	DB or DBMAG	dBohm
Y	RI	S
	MA or MAG	S
	DB or DBMAG	dBS

See A.2.5.5 for valid units.

These fields are optional.

7.8.2.12 Power_level

The measurement power level used for PDN extraction is specified in the *Power_level* field. For simplicity reasons, the power level along with units should be defined together as discussed in A.2.5.3.

This is an optional field. If absent, it defaults to "0dBm".

7.8.2.13 Data_files and List

The *Data_files* tag is used to differentiate between inline and external data. If the PDN data is defined in an external file, the link to the file shall be specified within the *Data_files* tag. Inline data is directly defined under the top level *Lead* tag within a *List* tag if needed. Files with extensions listed in Table 7 are valid.

Table 7 – Valid file extensions in the *Pdn* section

File extension	Common name
dat or DAT	Data file
csv or CSV	Comma separated values
txt or TXT	Text file
snp or SnP	Touchstone, n is an integer (1,2,...)
cir or CIR	Circuit file (Netlist)
lib or LIB	Library file (Netlist)
net or NET	Netlist file

List is not defined if circuit model is used for the PDN model. *Data_files* is defined for network parameters (*S/Z/Y*) and *Netlist* definitions.

This is a required field; a unique *List* or *Data_files* element shall be used to define the PDN data.

7.8.3 Description

7.8.3.1 General

As stated in 6.1, the PDN of the ICEM-CE macro-model can either be a single block (generic IC macro model) or consist of multiple blocks (and sub-models) for block-based ICEM-CE modelling. Depending on the simulation tool used, the PDN could be represented with different parameters, typically S -, Z -, Y - parameters, or as a circuit/netlist.

Table 8 lists the valid fields of a Lead structure PDN definition and their usage.

Table 8 – Valid fields of the Lead keyword in the *Pdn* section

Field	Description	Usage	Rules
Id	Identifier or pin number	Required	One or more Id's, external or internal nodes previously defined in the Lead-definitions section
Ground_id	Identifier of the ground pin	Required if Type="S" or "Z" or "Y"	One unique Id previously defined in the Lead-definitions section
Blockname	Name of the PDN block	Optional	Valid string (see 7.8.1)
Type	PDN source parameter (S/Z/Y/Circuit)	Optional	Valid String: "S" (default) or "Z" or "Y" or "Ckt"
Param_order	Order in which parameters are defined	Optional	Valid string (see 7.8.1)
Format	Data format	Optional	"RI" (default) or "DB" or "MA"
Meas_type	Method implemented for performing PDN measurements	Optional	"0" or "1" or "2"
Reference_impedance	Reference impedance used in PDN measurements	Optional	Valid numerical value with units: "50ohm"
Use	Parameter that is to be specifically used	Optional	One of the values in Param_order other than the Frequency term. Default: First S or Z or Y term
Netlist	SPICE type netlist	Optional	See 7.8.3.4 and 7.8.3.5
Unit_freq	Frequency units	Optional	See A.2.5.5
Unit_param	Parameter units	Optional	See A.2.5.5
Power_level	Measurement power level	Optional	See A.2.5.3
Data_files	PDN source parameter defined in an external file	Required if not List	Valid for external netlist file or external network parameter file.
List	PDN data entries in the form of a list defined inline	Required if not Data_files	Valid numerical values with or without units in the specified order

The compositions of *Id*, *Type*, *Param_order*, *Meas_type*, *Reference_impedance* and *Format* fields have been previously discussed in 7.8.1. For single-ended PDN, *Id* is a unique string and for multi-ended PDN, *Id* contains multiple pin Id's referenced together and separated by a comma (",") character. A single-ended PDN is shown in Example 1 and a multi-ended PDN definition is illustrated in Example 2.

EXAMPLE 1 A single-ended PDN of lead "1" with respect to the ground lead "5", defined using S -parameters, is represented as:

```
<Lead Id="1" Ground_id="5" Type="S".../>
```

EXAMPLE 2 A multi-ended PDN of leads "1", "2", "3", "4" etc. with respect to the ground lead "5", defined using *S*-parameters, is represented as:

```
<Lead Id="1,2,3,4,..." Ground_id="5" Type="S" ... />
```

For block-based or sub-model-based IC macro-modelling, the name of the PDN sub-block is specified using the *Blockname* field. This field is intended for informational purposes only. The CEM parser does not interpret this field. The PDN, IBC and the IA sub-models share the same namespace. Therefore, if both a PDN (and IBC) and an IA hold the same value (name string) in their respective *Blockname* element, they are regrouped together as a single ICEM-CE sub-model component. An example PDN in a block-based ICEM-CE macro-model is illustrated in Example 3.

EXAMPLE 3 A single-ended PDN representing the "Digital_block_A" is represented as

```
<Lead Id="1" Blockname="Digital_block_A".../>
```

Data can either be defined directly within the CEML file or can be defined using external data files for both network parameter and circuit (netlist) definition.

7.8.3.2 Inline data – network parameter

In common practice, network parameters such as *S*-, *Z*- and *Y*- parameters are extracted and defined in the frequency domain. If they are defined for a single frequency, then they can be defined within the *Lead* keyword directly. If the PDN is defined for multiple frequencies (more than one frequency), then data shall be written within the *List* keyword. It is also possible to represent *S*-parameters in the time-domain.

The different leads (internal and external), whose PDN is to be defined, shall be defined using the *Id* attribute and the ground lead (lead with Mode="GND" in *Lead_definitions* section) used as return signal path (ideal reference ground) shall be specified using the *Ground_id* attribute.

The frequency units shall be defined within the *Unit_freq* keyword and data units shall be specified using *Unit_param* keyword. The power level used for measurement shall be defined using *Power_level* tag and the reference impedance using the *Reference_impedance* tag. See 7.8.2 for more information on these attributes and their default values.

NOTE The lead Ids' value need not be coherent with the port numbers used for measurements. The values are matched with the port numbers in sequence, from left to right, in the touchstone format.

Example 1 illustrates a multi-ended PDN of leads "1","2","3","4" with respect to the ground lead "5". *S*-parameter data (frequency-domain), extracted with power of "10dBm", is defined in Real/Imaginary format and the *Param_order* field is defined as "Freq,All" which signifies that all the *S*-parameters are defined in sequence (touchstone format): Frequency, Re{S11}, Im{S11}, Re{S12}, Im{S12}, Re{S13}, Im{S13}, Re{S14}, Im{S14}, and so on...

EXAMPLE 1 Example of a frequency-domain multi-ended PDN.

```
<Pdn>
  <!-- S Parameters inline data, multi-frequencies -->
  <Lead Id="1,2,3,4";Ground_id="5" Type="S"; Param_order="Freq,All"; Format="RI">
    <Unit_freq>kHz</Unit_freq>
    <Unit_param>1</Unit_param>
    <Power_level>10dBm</Power_level>
    <List>
100 0.004 0.002 0.0018 0.0023 -0.0013 -0.0021 0.99 -0.0037 0.0019 0.002 0.0042 0.0024
0.998 -0.0038 -0.0011 -0.002 -0.0013 -0.0021 0.998 -0.0037 0.0041 0.0027 0.0013
0.0026 0.998 -0.0036 -0.001 -0.0019 0.0013 0.0027 0.0022 0.0027
...
    </List>
  </Lead>
</Pdn>
```

7.8.3.3 External file data – network parameter

For directing the CEM parser to use the necessary external data file in ASCII format, definition shall be made using *Data_files* keyword. If the external file has a touchstone extension (*.snp or *.SnP where n is an integer 1, 2, ...), *Unit_param*, *Unit_freq*, *Param_order*, *Reference_impedance* and *Type* definitions are not required (ignored if explicitly defined). For all other file extensions (*.dat, *.csv, *.txt) the defined values are used. If absent, default values are automatically considered. See Table 8 for default properties. Similar to inline definition seen in 7.8.3.2, the *Ground_id* field is required and the *Meas_type* field is optional.

Measurement power level should always be a part of the external file as a comment. Comment lines in the PDN file shall start with an exclamation symbol ("!") as shown below in Example 1.

EXAMPLE 1 !This is a comment line

Figure 19 shows an example of the four-port *S*-parameter file in Touchstone format, in the frequency-domain, with *data* in dB format. This file is used to define the PDN of leads "1","2","3","4" with respect to ground lead "5" (see Example 2). The model is block based with a name defined.

NOTE The measurement power level is included as a comment in the PDN file.

```
!Agilent Technologies,E5071C,MY46100659,A.07.02
!Date: Wed Jan 23 09:49:50 2013
!Data & Calibration Information:
!Freq
!Power_level: 10dBm
# Hz S RI R 50
100000 0.004274546 0.00241179 0.001886209 0.002311387 -0.001362228 -0.002127369 0.9983465 -0.003734921
0.00190445 0.00233215 0.004235411 0.002405719 0.998264 -0.003837676 -0.001106818 -0.002012462
-0.001369703 -0.002143684 0.9981545 -0.003736369 0.004101523 0.002755493 0.001347644 0.002668388
0.9983041 -0.003624817 -0.001178987 -0.001959733 0.001348257 0.002710212 0.002291472 0.002748889
100679 0.00422729 0.002403554 0.001871881 0.002334189 -0.001349262 -0.002154236 0.998417 -0.003759051
0.001902756 0.002356657 0.004170153 0.002522967 0.9982947 -0.003870883 -0.001127652 -0.002029074
-0.001356177 -0.002142675 0.9980684 -0.003715199 0.00410353 0.002777282 0.001371339 0.002686834
0.9982006 -0.003591272 -0.001147017 -0.001990634 0.001357759 0.002703257 0.002186219 0.002865791
101359 0.004252984 0.002479831 0.001877719 0.002357767 -0.001379005 -0.002139817 0.9983823 -0.00378314
0.001916786 0.002348517 0.004238612 0.002526162 0.9983067 -0.003854939 -0.001140214 -0.002014498
-0.00136173 -0.002137753 0.9980044 -0.003766839 0.004121085 0.002838985 0.001346626 0.00268385
0.9981698 -0.003693599 -0.001171574 -0.00198756 0.001361229 0.00271512 0.002269425 0.002912863
```

IEC

Figure 19 – PDN represented as *S*-parameters in Touchstone format

EXAMPLE 2 Describing a PDN using an external data file

```
<Pdn>
  <Lead Id="1,2,3,4;Ground_id="5">
    <Power_level>0dBm</Power_level>
    <Blockname>memory_block</Blockname>
    <Data_files>
      Pin1234_PDN.s4p
    </Data_files>
  </Lead>
</Pdn>
```

7.8.3.4 Inline circuit definition

Another way to define the PDN is in the form of a schematic (circuit). CEM parser understands that the PDN has a circuit definition when the *Type* attribute of a particular lead or port is set to "Ckt". The circuit definition shall be directly declared within the *Netlist* tag under the *Lead* section. The *Netlist* section shall contain the following definitions:

- *Power_level* definition
- SPICE like data statements defining the electrical connectivity of components [2]

All SPICE recognised elements can be defined in this section and the representation shall follow SPICE syntax. The data statements have a free format and consist of fields separated by a blank. If one wants to continue a statement to the next line, one uses a "+" sign (continuation sign) at the beginning of the next line. Numbers can be integers, or floating points [2].

EXAMPLE 1 A resistor R1 with a value of 1 kΩ connecting nodes "In" and "Out" can be represented as:

```
R1 In Out 1e3
```

All nodes used in the data statements share the same namespace with the lead Ids defined in the *Lead_definition* section. The *Ground_id* field is optional (ignored by the CEM parser if defined) when defining the PDN using a netlist. The *Id* attribute of the *Lead* keyword shall carry the Id information of the internal, external and ground pins (as defined in the *Lead_definitions* section).

Within the several data statements used to define the netlist, at least one node shall correspond to the lead defined as internal or external and at least one node shall be GND in the *Lead_definition* section. All other interior nodes used for circuit definition shall not carry the lead Id's defined in the *Lead_definition*. For compatibility reasons between several SPICE kinds, "0" is not allowed as node number.

The *Netlist* keyword definition is shown in Table 9.

Table 9 – Netlist definition

<i>Netlist</i>
Kind: SPICE netlist format (optional) default: "SPICE3"
Data_files: SPICE netlist in external file (optional)

The optional attribute *Kind* tells the CEM parser that the defined sub-circuit follows a specific syntax. CEM version 1 supports industry-standard SPICE like netlist syntaxes.

The *Kind* field defaults to generic "SPICE3" if absent.

Example 1 illustrates an example PDN using a SPICE netlist. The same can also be expressed as a SPICE macro-model as illustrated in Example 2. In Example 2, it is assumed that "PDN_pin12345" is defined in the *Macromodels* section (see 7.6).

EXAMPLE 1 An example of PDN using inline netlist definition. In this case, lead with Id="5" is used as ground lead.

```
<Pdn>
  <!-- General circuit model -->
  <Lead Id="1,2,3,4,5" Type="Ckt">
    <Power_level>10dBm</Power_level>
    <Netlist Kind="SPICE3">
      L1 1 n00 1.820e-09
      R1 n01 n02 2.408e-01
      C1 n00 5 1e-12
      L2 2 n10 1.828e-09
      R2 n11 n12 2.434e-01
      C2 n10 5 1e-12
      ...
    </Netlist>
  </Lead>
</Pdn>
```

EXAMPLE 2 An example of a PDN using SPICE macro-models. In this case, lead with Id="5" is used as ground lead. XPDN1 is the sub-circuit's identifier in the main circuit.

```

<Pdn>
  <!-- General circuit model using macro-model-->
  <Lead Id="1,2,3,4,5" Type="Ckt">
    <Power_level>10dBm</Power_level>
    <Netlist>
      XPDN1 1 2 3 4 5 PDN_pin12345
    </Netlist>
  </Lead>
</Pdn>

```

7.8.3.5 External file containing the netlist

Instead of the inline netlist definition (see 7.8.3.4), it is possible to define the PDN in an external netlist file. The definition is done with the *Data_files* tag under the *Netlist* section. All rules with respect to the node numbers used in the netlist definition are the same as defined in 7.8.3.4.

In some cases, a netlist file could contain the main circuit definition and sub-circuit definitions together. To avoid ambiguity, the CEM parses only the data statements defined outside the SPICE keywords: ".SUBCKT" and ".ENDS". These keywords are not case sensitive. For parsing the sub-circuit element, it is mandatory to define the same file within the *Macromodels* section (see 7.6).

Example 1 illustrates the use of an external netlist file and the PDN is defined as a block.

EXAMPLE 1 An example of a PDN using external file containing the netlist definition. In this case, lead with Id="5" is used as ground lead.

```

<Pdn>
  <!-- General circuit model using netlist -->
  <Lead Id="1,2,3,4,5" Type="Ckt">
    <Power_level>0dBm</Power_level>
    <Blockname>block2</Blockname>
    <Netlist>
      <Data_files>Pin12345_PDN_Ckt.lib</Data_files>
    </Netlist>
  </Lead>
</Pdn>

```

7.9 IBC

7.9.1 General

Similar to the PDN, different IBC complexity levels can be considered. The simplest configuration level is modelling the IBC with two internal terminals linking two internal terminals of different PDN blocks. IBC configuration interfacing many PDN blocks and many internal terminals can become very complex.

The *Ibc* section of the ICEM-CE macro-model contains the IBC data that describes the ICEM-CE PDN. The data shall be defined within the *Ibc* keyword as follows:

```

<Ibc>
  IBC data
</Ibc>

```

An IBC data is defined for interfacing different PDN blocks and thus the definition shall be done within a *Lead* tag (similar to the PDN). See Figure 17 for the structural hierarchy. Many *Lead* elements can be listed one below another within the *Ibc* section.

7.9.2 Attribute definitions

The general structure of the *Lead* keyword is coherent with that listed in Table 5. However, some attribute specifications are different for the IBC network. These differences are tabulated in Table 10. All other fields in Table 5 remain unchanged.

Table 10 – Differences between the *Pdn* and *Ibc* section fields

Field	Description	Usage		Rules	
		PDN	IBC	PDN	IBC
Id	Identifier or pin number	Required	Required	<u>One or more Ids</u> previously defined in the Lead-definitions section as <u>external or internal terminals</u>	<u>Two or more Id's</u> previously defined in the Lead-definitions section as <u>internal terminals</u>
Ground_id	Identifier or pin number of the return signal pin	Required for Type="S" or "Z" or "Y"	NA	One unique Id previously defined as GND in the Lead-definitions section	NA

Table 11 lists the valid fields of a *Lead* structure for an IBC definition and their usage.

Table 11 – Valid fields of the *Lead* keyword for IBC definition

Field	Description	Usage	Rules
Id	Identifier or pin number	Required	Two or more Id's previously defined in the Lead-definitions section as internal terminals
Blockname	Name of the IBC component	Optional	Valid string (see 7.8.1)
Type	IBC source parameter (S/Z/Y/Circuit)	Optional	Valid String: "S" (default) or "Z" or "Y" or "Ckt"
Param_order	Order in which IBC parameters are defined	Optional	Valid string (see 7.8.1)
Format	Data format	Optional	"RI" (default) or "DB" or "MA"
Meas_type	Method implemented for performing IBC measurements	Optional"0"	Valid numerical value: "0" or "1" or "2" Default: "0"
Reference_impedance	Reference impedance used in IBC measurements	Optional	Valid numerical value with units: "50ohm"
Use	Parameter that is to be specifically used	Optional	One of the values in <i>Param_order</i> other than the Frequency term. Default: First S or Z or Y term
Netlist	SPICE type netlist	Optional	See 7.8.3.4 and 7.8.3.5
Unit_freq	Frequency units	Optional	See A.2.5.5
Unit_param	Parameter units	Optional	See A.2.5.5
Power_level	Measurement power level	Optional	See A.2.5.3
Data_files	IBC source parameter defined in an external file	Required if not List	Valid for external netlist file or external network parameter file.
List	IBC data entries in the form of a list defined inline	Required if not Data_files	Valid numerical values with or without units in the specified order

Any relevant information required for correct understanding (and usage) of the IBC shall be optionally defined within the *Notes* and *Documentation* tags.

7.10 IA

7.10.1 General

The *Ia* section of the ICEM-CE macro-model contains the IA data that describes the model. The data shall be defined within the *Ia* keyword as follows:

```
<Ia>
    IA data
</Ia>
```

Similar to the PDN and IBC definition, IA data is defined for IC pin or terminal and thus the definition shall be done within a *Lead* tag (see Figure 17 for the structural hierarchy). Table 12 lists the various recognized fields of the *Lead* keyword in the IA section:

Table 12 – Definition of the *Lead* keyword in the *Ia* section

<i>Lead</i>
Id: pin identity as a valid string (required)
Type: IA source parameter -"time" for time-domain data and "freq" for frequency-domain data (required)
Blockname: IA block name as a valid string (optional)
Voltage: IA data defined as a voltage source (required if not Current)
Current: IA data defined as a current source (required if not Voltage)

The conditions of validity (frequency or time range, power-supply conditions, time or phase origin, etc.) of the IA sub-model shall be specified in the *Validity* section (see 7.7).

NOTE The frequency range (or time range) of validity of the ICEM-CE macro-model is the common frequency (or time) range of the PDN and IA.

As per IEC TS 62433-1, any other relevant information required for correct understanding (and usage) of the IA shall be optionally defined within the *Notes* and *Documentation* tags. Details such as IC's operating mode, decoupling capacitors on supply lines, activated functions, grounding details, datasheets and test-reports can be defined.

7.10.2 Attribute definitions

7.10.2.1 Id

The identity, *Id*, of the leads used in the IA definition should have been previously defined as internal pins in the *Lead_definitions* tag as described in 7.5. Only two *Ids* are permitted per *Lead* definition since the IA source is always connected between two terminals. The first *Id* terminal is assumed to be the positive node and the second *Id* terminal is the negative node. Current flows from the positive node to the negative node.

This is a required field.

7.10.2.2 Type

The *Type* attribute is used to represent the type of the IA data. CEML version 1 supports two types: "time" for time-domain IA data and "freq" for IA data in the frequency domain.

This is a required field.

7.10.2.3 Blockname

The *Blockname* field is used to define the name of the IA block. This field is required for the IA. If both the PDN and the IA share the same block name, then they are regrouped together by the CEM parser as a sub-block.

This is an optional field.

7.10.2.4 Voltage and Current

The main IA source is defined under the *Voltage* and/or *Current* sections.

At least one definition (*Voltage* field or *Current* field) is required in the IA section. Voltage data is defined within the *Voltage* keyword and the current data is defined within the *Current* keyword.

The common definition of the *Voltage* and *Current* sections is shown in Table 13.

Table 13 – Voltage and Current definition

<i>Voltage / Current</i>
List: IA data list (required if not <i>Data_files</i> or <i>Pulse</i>)
Data_files: IA source parameter defined in an external file (required if not <i>List</i> or <i>Pulse</i>)
Pulse: IA data as a simulated waveform (required if not <i>List</i> or <i>Data_files</i> , valid only if <i>Type</i> ="time")
Param_order: Order in which IA parameters are defined (optional)
Format: IA Data format (optional)
Unit_voltage: Unit of the voltage terms (optional)
Unit_current: Unit of the current terms (optional)
Unit_freq: Unit of the frequency terms (optional)
Unit_time: Unit of the time terms (optional)

These sections have one required child element: *List* or *Data_files* or *Pulse*; either one shall be used to define the IA data. All other attributes and elements in Table 13 are optional. When absent, default values and units are used.

7.10.2.5 Data_files and List

The definitions of the *Data_files* and *List* keywords are similar to that seen with the PDN definition (see 7.8.2.13). The *Data_files* tag is used to differentiate between inline and external data. If the PDN data is defined in an external file, the link to the file(s) shall be specified within the *Data_files* tag. Inline data is directly defined under the top level *Voltage* or *Current* tag within a *List* tag if needed. Files with extensions listed in Table 14 are allowed for defining the IA.

Table 14 – Valid file extensions in the *Ia* section

File extension	Common name
dat or DAT	Data file
csv or CSV	Comma separated values
txt or TXT	Text file

7.10.2.6 Pulse

This keyword is dedicated for representing the IA sub-model using a simulated waveform, such as triangular or pulse waveform; it is permitted to do so only for defining the IA in the time-domain, i.e. Type="time" (see 7.10.2.2). The *Pulse* section is directly defined under the top level *Voltage* or *Current* tag.

This section is required if no *List* or *Data_files* element has been used to define the voltage or current data defining the IA. Table 15 lists the various recognized fields of the *Pulse* keyword in the *Voltage* or *Current* section.

Table 15 – Definition of the *Pulse* keyword in the *Voltage* or *Current* section

<i>Pulse</i>
Initial_value: Initial or minimum amplitude (required)
Final_value: Final or maximum amplitude (required)
Period: Time period of the signal (required if not Frequency)
Frequency: Frequency of the signal (required if not Period)
Time_delay: Time delay before the leading edge of the signal (optional) default:"0"
Rise_time: Time taken by the signal to rise from the initial value to reach the final value (optional) default:"0"
Fall_time: Time taken by the signal to fall from final value to the initial value (optional) default:"0"
On_time: Time duration for which the signal maintains its final value (optional) default: half of period
Ncycles: Number of repetition cycles (optional) default: infinite

A *Pulse* section has three required fields: *Initial_value*, *Final_value* and *Period* or *Frequency* (inverse of period). All other attributes in Table 15 are optional. The value of all the fields is a numerical value with or without units. See A.2.5.5 for valid units.

The base units of these fields are described in Table 16; see A.2.5.5 for valid scaling factors. Parameter units may also be set using the keywords defined in 7.10.2.9.

Table 16 – Base units of the *Pulse* section's fields

Attribute	Base units	
	Voltage section	Current section
Initial_value	V	A
Final_value	V	A
Period	s	
Frequency	Hz	
Time_delay	s	
Rise_time	s	
Fall_time	s	
On_time	s	
Ncycles	1	

A typical waveform defined by the *Pulse* keyword, with the corresponding fields, is shown in Figure 20.

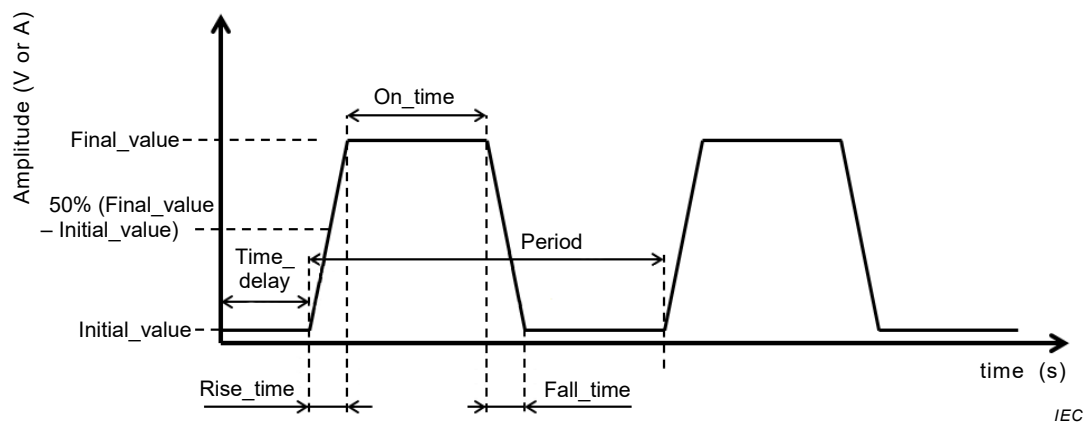


Figure 20 – Simulated IA waveform with corresponding parameters

An example of IA data using a *Pulse* tag is given in the Example below.

EXAMPLE An example *Pulse* section may be defined as shown.

```
<Ia>
  <Lead Id="1,3" Type="time">
    <Current>
      <Pulse>
        <Initial_value>0</Initial_value>
        <Final_value>50e-3</Final_value>
        <Period>125n</Period>
        <Rise_time>2n</Rise_time>
        <Fall_time>4n</Fall_time>
        <On_time>0.1n</On_time>
      </Pulse>
    </Current>
  </Lead>
</Ia>
```

7.10.2.7 Param_order

The *Param_order* attribute tells the CEM parser on how the IA data is represented. The following strings are dedicated for specifying the parameter order:

- "Freq" and "Frequency": Frequency quantity used for parameters' definition
- "Time": Time quantity used for parameters' definition
- "Volt" and "Voltage": Voltage used for parameters' definition
- "Curr" and "Current": Current used for parameters' definition

If absent, it defaults to:

- "Freq,Volt" in the *Voltage* section and "Freq,Curr" in the *Current* section if Type="freq"
- "Time,Volt" in the *Voltage* section and "Time,Curr" in the *Current* section if Type="time"

7.10.2.8 Format

The *Format* attribute decides the data format. Valid data formats for IA are:

- "RI": real/imaginary format. This is valid for frequency-domain IA.
- "DB": Magnitude in decibel scale with phase angle in degrees. This is valid for frequency-domain IA.
- "MA": Magnitude in linear scale with phase angle in degrees. This is valid for frequency-domain IA.
- "DBMAG": Magnitude in decibels without phase information. This is valid for frequency-domain and time-domain IA.

- "MAG": Magnitude in linear scale without phase information. This is valid for frequency-domain and time-domain IA.

This field is optional. If absent, the default value is "MAG".

7.10.2.9 Unit_freq, Unit_time, Unit_voltage and Unit_current

The parameter units are defined under the *Unit_freq*, *Unit_time*, *Unit_voltage* and *Unit_current* tags for frequency, time, voltage and current quantities respectively. Depending on the parent element, only the corresponding units are considered, i.e., *Unit_voltage* is allowed under the *Voltage* section and so on. Only *Unit_freq* and *Unit_time* attributes are defined for multiple sections.

If absent, *Unit_freq* defaults to "Hz" and *Unit_time* defaults to "s". See Table 17 for default values of *Unit_voltage* and *Unit_current* tags as a function of data format.

Table 17 – Valid data formats and their default units for the *Voltage* and *Current* elements

Parameter	Data format	Default Unit_voltage values	Default Unit_current values
Voltage	RI	V	-
	MA or MAG	V	-
	DB or DBMAG	dBV	-
Current	RI	-	A
	MA or MAG	-	A
	DB or DBMAG	-	dB A

If complex values are used to represent the IA (Format="MA" or "DB"), the default units of phase quantities is "degrees". See A.2.5.5 for valid units.

7.10.3 Description

7.10.3.1 Inline data

If the IA data (voltage and current) are defined for a single frequency, then they can be defined within the corresponding section (*Voltage* and *Current*) keyword directly. For multi-frequency IA, the data shall be written within the *List* keyword under the corresponding *Current* or *Voltage* section. The frequency units shall be defined within the *Unit_freq* keyword, time units within the *Unit_time* keyword, voltage units shall be specified using *Unit_voltage* keyword and that of current using *Unit_current* keyword. When absent, default values are used (see 7.10.2.9).

Example 1 illustrates the IA data of a frequency-domain ICEM-CE macro-model using inline data definition. A current quantity is used to define the IA in decibel format ("DB"). The current is specified in "dBuA" units and the frequency quantities are defined in "kHz".

EXAMPLE 1 An example of an IA section in CEML is shown.

```

<Ia>
  <Lead Id="10,11" Type="freq">
    <Current Param_order ="Freq,Current" Format="DB">
      <Unit_freq>kHz</Unit_freq>
      <Unit_current>dBuA</Unit_current>
      <List>
        7.700018e+03      -2.207281e+01
        7.800018e+03      -1.900327e+01
        7.900019e+03      -1.330357e+01
        8.000019e+03      6.099079e+01
        8.100019e+03      -1.338519e+01
        8.200019e+03      -1.912021e+01
        8.300020e+03      -2.216923e+01
        ...
      </List>
    </Current>
  </Lead>
</Ia>

```

7.10.3.2 External file data

For directing the CEM parser to use the necessary external IA data file in ASCII format, definition shall be made using *Data_files* keyword under the *Voltage* or *Current* section. All other rules are the same as that seen with inline data definition in 7.10.3. Note that all comment lines, if present, are indicated by "!" symbol in the beginning, but they not parsed by the CEM parser.

8 Requirements for parameter extraction

8.1 General

ICEM-CE model parameters can be extracted from either design information or measurements. Detailed methodology for model parameter extractions are not the purpose of this standard. This clause gives basic requirements for model parameter extractions from measurements.

NOTE Annex A gives examples of parameter extractions from design information and from measurements.

8.2 Environmental extraction constraints

The ICEM macro-model parameter extractions have to be performed under normal room temperature conditions: $23\text{ °C} \pm 5\text{ °C}$. There are no additional requirements on air pressure and humidity.

When open silicon is used, the lights shall be dimmed or switched off in order not to generate photonic effects.

NOTE Some substrate materials are hygroscopic which might affect the permittivity of the material and its loss tangent.

8.3 IA parameter extraction

IA can be derived by design data or by calculating by measurements under the following conditions:

- Nominal power supply voltages and typical loadings shall be applied to the device under test.
- Input signals such as specific test vector which correspond to the specific operational mode shall be applied.

8.4 PDN parameter extraction

The PDN parameters shall be derived by analyzing impedances between the terminals under the nominal power supplies.

The derived parameters are only suited and re-useable for conducted emission simulations when the PVT (process, voltage and temperature) conditions are the same.

NOTE 1 PDN is static, but due to the N-well and gate-oxide capacitances that are involved, the power supply voltage will affect it.

NOTE 2 Most of the impedance parameters can be derived from measurements using an impedance analyzer or a VNA (Vector Network Analyzer).

8.5 IBC parameter extraction

The IBC impedance can be derived from the Vdd to Vss impedance by subtracting the impedance part that belongs to the PDN. A detailed method should be elaborated in the future.